

Informe del análisis de datos para el dataset “Adult Income”

Haessler Joan Ortiz Moncada

Universidad Distrital Francisco José de Caldas

Facultad de Ingeniería

Curso: Big Data

3 de septiembre de 2025

1. Introducción

Para este informe se utilizó el `Adult Income Dataset`, también conocido como `Census Income Dataset`, este Dataset proviene del censo de Estados Unidos de 1994, y es ampliamente usado en ejercicios de clasificación. El objetivo de este informe es predecir si una persona gana más de 50,000.00 dólares anuales a partir de características demográficas y socioeconómicas, utilizando diferentes modelos de clasificación, diversas métricas de evaluación y variando los parámetros de los modelos implementados, además de documentar y discutir los resultados obtenidos, así como familiarizar al estudiante con la plataforma Colab y el lenguaje de Python para el análisis de datos.

2. Metodología

Este informe se desarrolla conforme a lo establecido en el `Assignment #2`, aunque con algunas variaciones en la presentación. En aquel caso se solicitaba que todo el trabajo se realizara en un único documento o archivo de Colab; en cambio, aquí se optó por efectuar la codificación en dicho entorno y presentar los hallazgos y el compendio de resultados mediante un informe escrito. Se emplearon principalmente Google Colab y Python, haciendo uso de los paquetes `os`, `pandas`, `scikit-learn`, `matplotlib` y `seaborn`.

Básicamente, lo que se hizo fue ir desarrollando en Colab el código correspondiente y de manera paralela documentar los hallazgos y resultados en este informe. Por último, el código fuente implementado en Colab, los datos utilizados y el archivo de este informe se encuentran en la carpeta `taller_3` del repositorio de GitHub [bigdata_repositorio](#).

3. Descripción del dataset

El conjunto de datos contiene 48,842.00 registros, distribuidos en dos archivos: `adult.data` (con 32,561.00 instancias) y `adult.test` (con 16,281.00 instancias). Incluye 14 características y una variable objetivo denominada `income`. El dataset se descargó de [Adult Income Dataset](#). La Tabla 1 presenta los campos y su tipo.

3.1. Atributos del dataset

| Atributo | Tipo |
|----------------|-----------------------|
| age | Numérico (int64) |
| workclass | Categórico |
| fnlwgt | Numérico (int64) |
| education | Categórico |
| education-num | Numérico (int64) |
| marital-status | Categórico |
| occupation | Categórico |
| relationship | Categórico |
| race | Categórico |
| sex | Categórico |
| capital-gain | Numérico (int64) |
| capital-loss | Numérico (int64) |
| hours-per-week | Numérico (int64) |
| native-country | Categórico |
| income | Categórico (objetivo) |

Tabla 1: Atributos y tipo de dato

Para este informe se unificaron ambos archivos en un único *DataFrame*. La división original probablemente busca facilitar la replicación de resultados entre entrenamiento y prueba. El repositorio incluye, además, el dominio de cada característica y documentación de tipos. El dataset se proyectó en un archivo Excel.

4. Resultados y análisis

A continuación se sigue la estructura del *Assignment #2* del curso de Big Data.

1. Exploración y análisis de datos

- Estadísticas descriptivas para variables numéricas:

| Estadístico | age | fnlwgt | education-num | capital-gain | capital-loss | hours-per-week |
|-------------|-----------|--------------|---------------|--------------|--------------|----------------|
| count | 48,842.00 | 48,842.00 | 48,842.00 | 48,842.00 | 48,842.00 | 48,842.00 |
| mean | 38.64 | 189,664.13 | 10.08 | 1,079.07 | 87.50 | 40.42 |
| std | 13.71 | 105,604.03 | 2.57 | 7,452.02 | 403.00 | 12.39 |
| min | 17.00 | 12,285.00 | 1.00 | 0.00 | 0.00 | 1.00 |
| 25 % | 28.00 | 117,550.50 | 9.00 | 0.00 | 0.00 | 40.00 |
| 50 % | 37.00 | 178,144.50 | 10.00 | 0.00 | 0.00 | 40.00 |
| 75 % | 48.00 | 237,642.00 | 12.00 | 0.00 | 0.00 | 45.00 |
| max | 90.00 | 1,490,400.00 | 16.00 | 99,999.00 | 4,356.00 | 99.00 |

Tabla 2: Estadísticas descriptivas de variables numéricas

El conjunto presenta seis variables numéricas con comportamientos diversos. En cuanto a tendencia central y dispersión, `age` tiene una media de 38.64 y mediana de

37, lo que indica un leve sesgo a la derecha. La variable `hours-per-week` muestra una mediana de 40 y el tercer cuartil en 45, consistente con jornadas laborales estándar y cierta variabilidad. `education-num` es discreta, centrada en 10 años (Q3 en 12). Por otro lado, `fnlwgt` exhibe una varianza muy alta y una cola derecha pronunciada. Tanto `capital-gain` como `capital-loss` presentan valores extremadamente bajos (Q1=Q2=Q3=0), lo que implica que al menos el 75 % de los registros no reportan ganancias ni pérdidas de capital.

Antes de graficar, se detectó que la variable `income` tenía cuatro etiquetas distintas por un error de tipeo: algunas instancias presentaban un punto al final (`<=50K.`, `>50K.`). Este problema se corrigió unificando las etiquetas en solo dos categorías válidas: `<=50K` y `>50K`. A continuación se muestran histogramas y diagramas de caja (*boxplots*) para explorar la distribución de las variables numéricas y la frecuencia de la variable objetivo `income`.

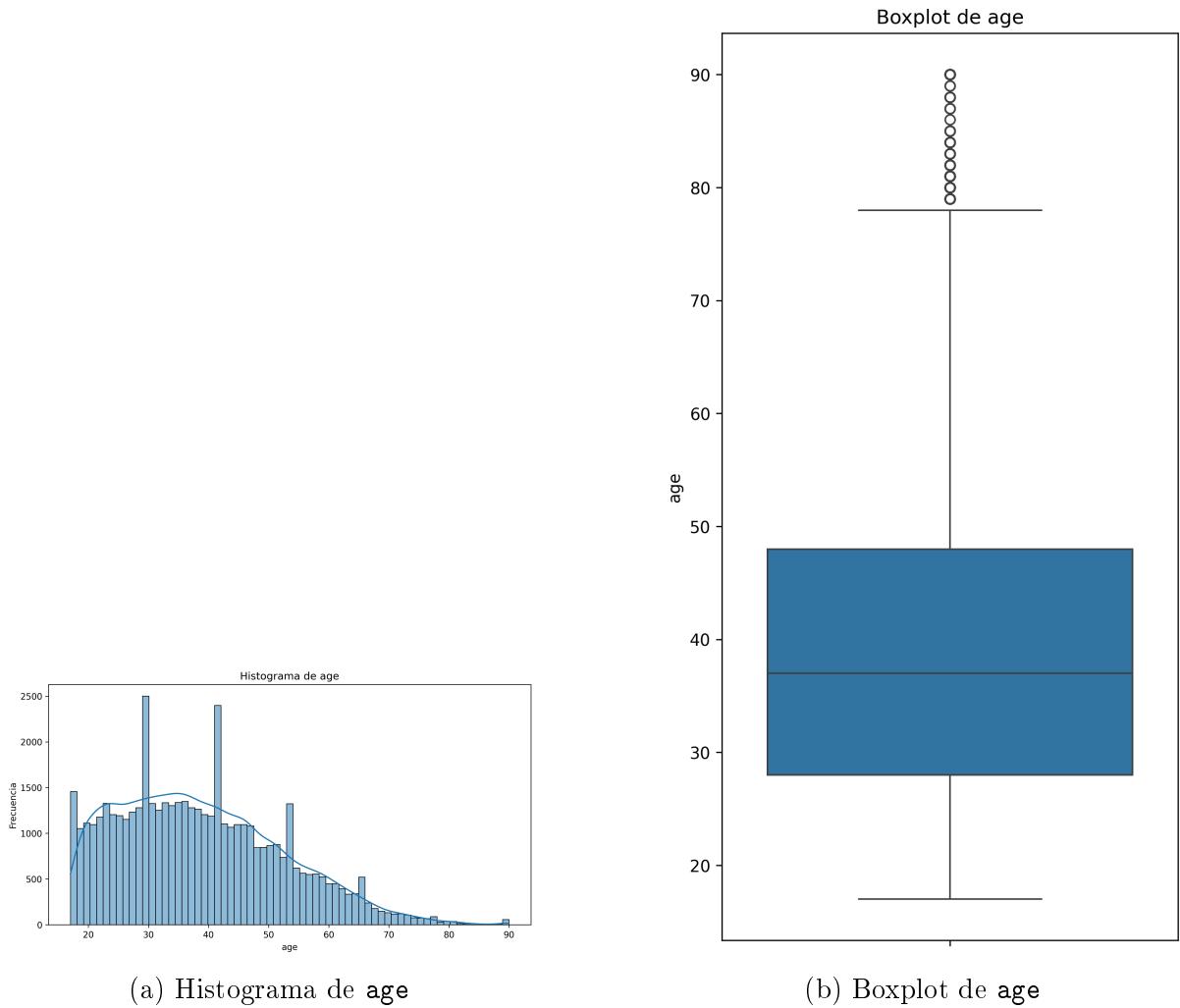


Figura 1: Distribución de `age`: histograma y boxplot.

La distribución de `age` se concentra entre los 28 y 48 años (rango intercuartílico), con mínimos en 17 y máximos en 90. El histograma muestra una mayor densidad entre los 30–50 años y el boxplot revela algunos valores altos atípicos, esperables en poblaciones laborales amplias.

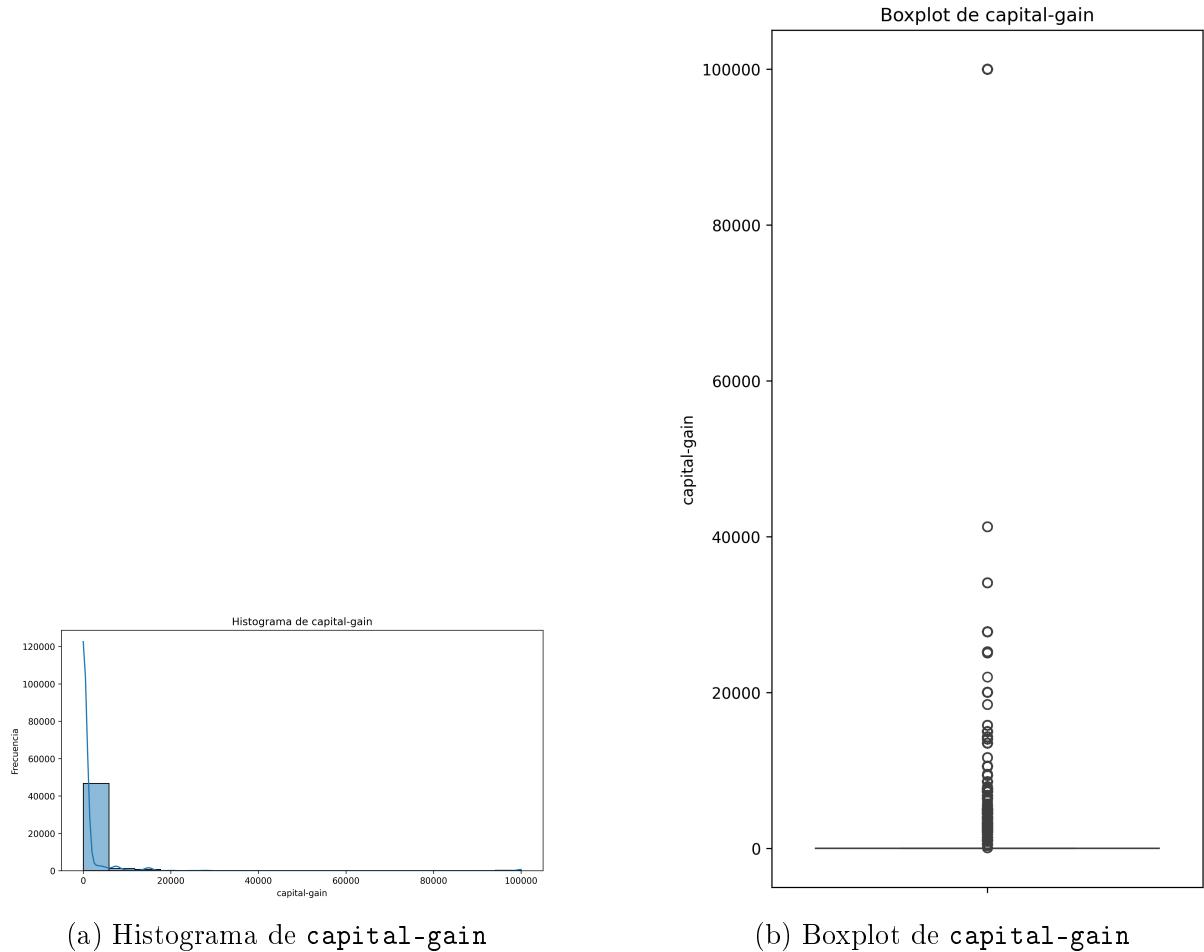


Figura 2: Distribución de `capital-gain`: histograma y boxplot.

capital-gain está fuertemente sesgada a la derecha: tres cuartiles en cero y una larga cola hasta 99,999. El boxplot evidencia numerosos ceros y pocos valores extremadamente altos; el histograma muestra acumulación en cero con muy pocas observaciones en los rangos altos. Esto indica que la mayoría de personas no reportan ganancias de capital, y solo un pequeño grupo tiene valores elevados. Podría convenir aplicar una transformación logarítmica para reducir la asimetría y estabilizar la varianza.

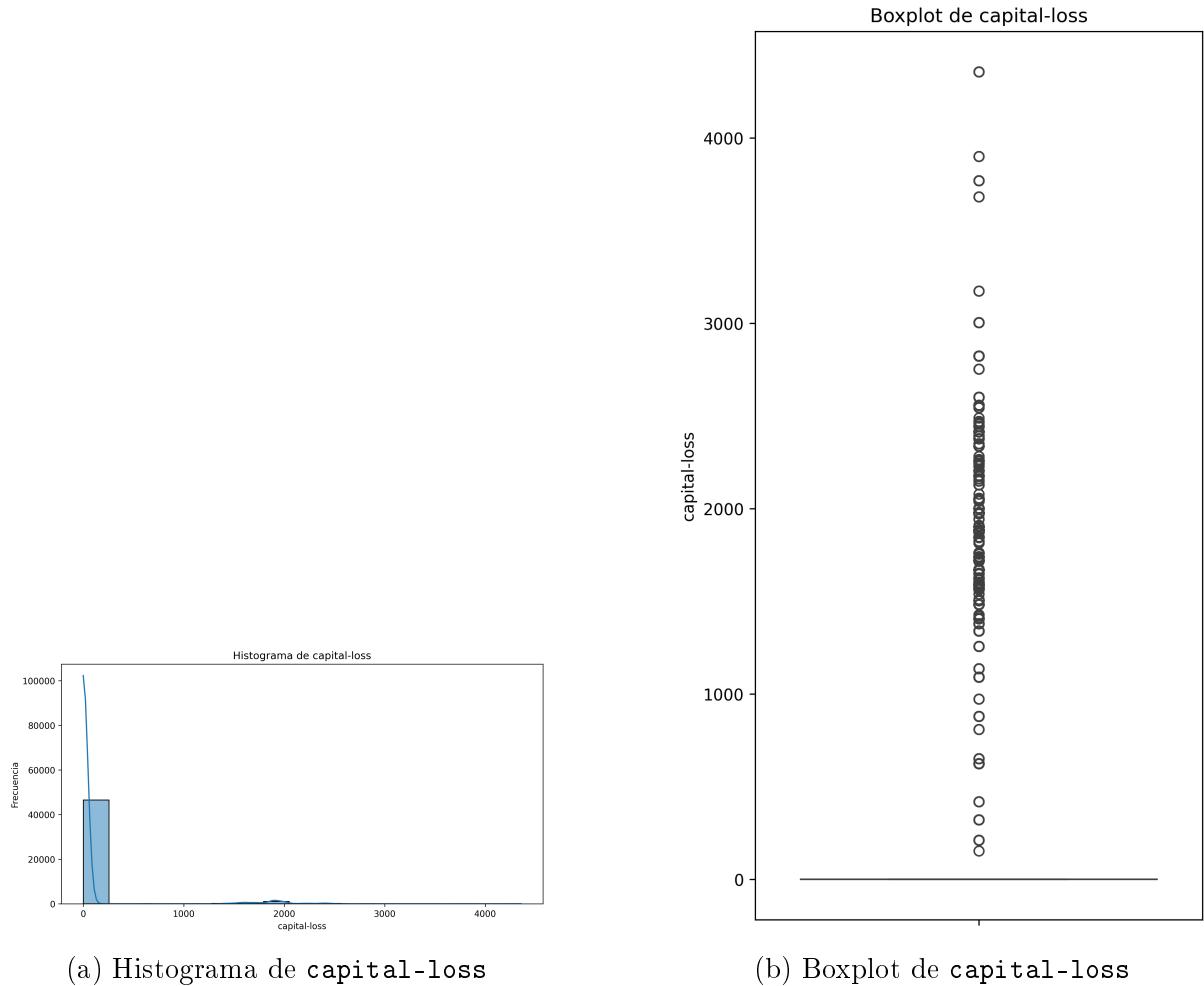


Figura 3: Distribución de **capital-loss**: histograma y boxplot.

capital-loss muestra un patrón similar al de *capital-gain*: el 75 % de los registros tienen valor cero y solo unos pocos presentan pérdidas elevadas. El boxplot evidencia que la mayor variabilidad proviene de estos pocos casos extremos. Para facilitar el análisis, podría ser útil transformar esta variable en binaria (pérdida sí/no) o aplicar técnicas que reduzcan el impacto de los valores atípicos si se mantiene como variable continua.

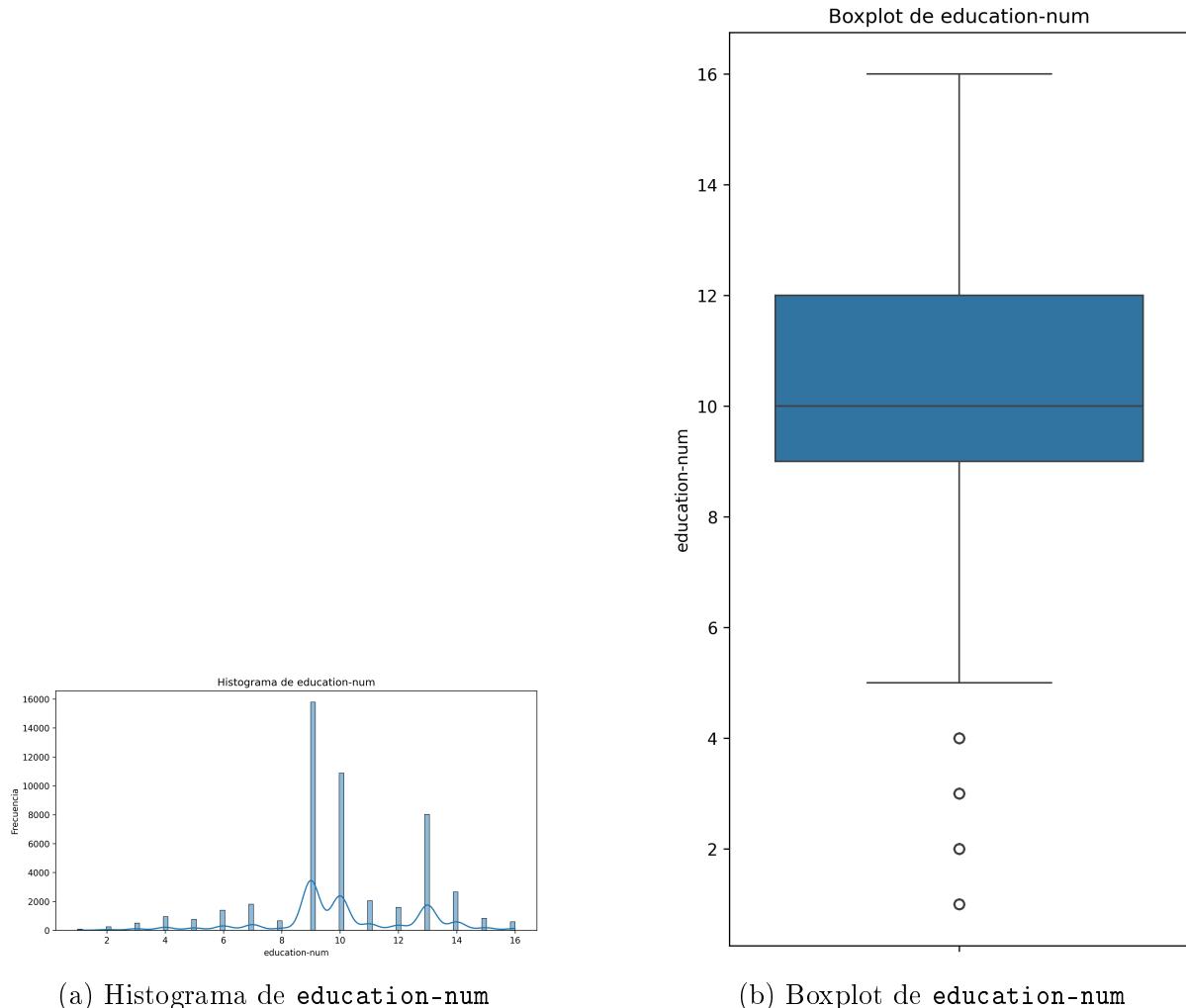


Figura 4: Distribución de **education-num**: histograma y boxplot.

education-num es una variable discreta entre 1 y 16, con mediana 10 y Q3=12. El histograma refleja picos en niveles educativos frecuentes (por ejemplo, secundaria y algunos estudios superiores), y el boxplot confirma baja presencia de atípicos. Su naturaleza ordinal sugiere que modelos lineales simples captan bien su efecto, aunque interacciones con *hours-per-week* o *age* podrían aportar capacidad explicativa.

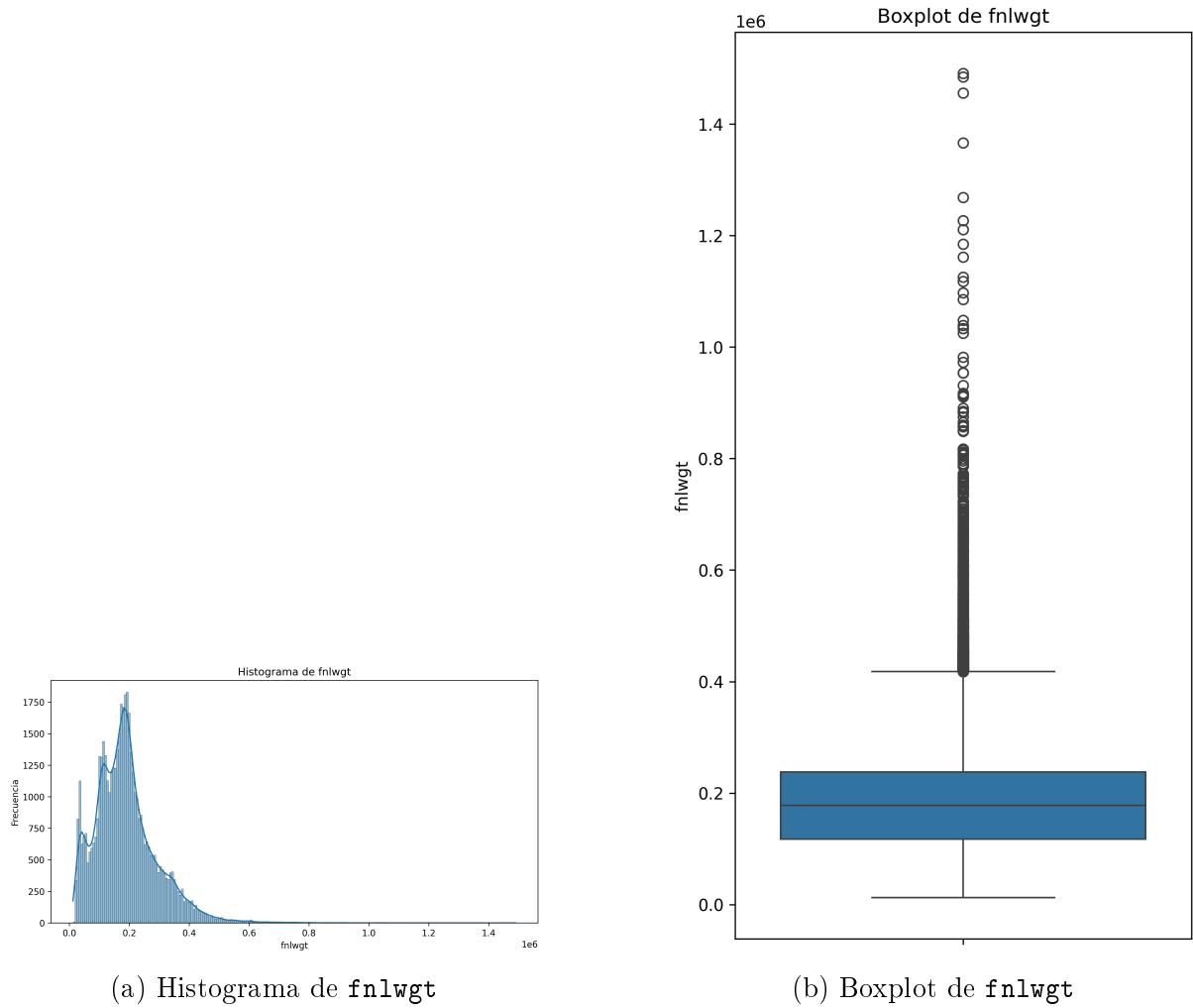
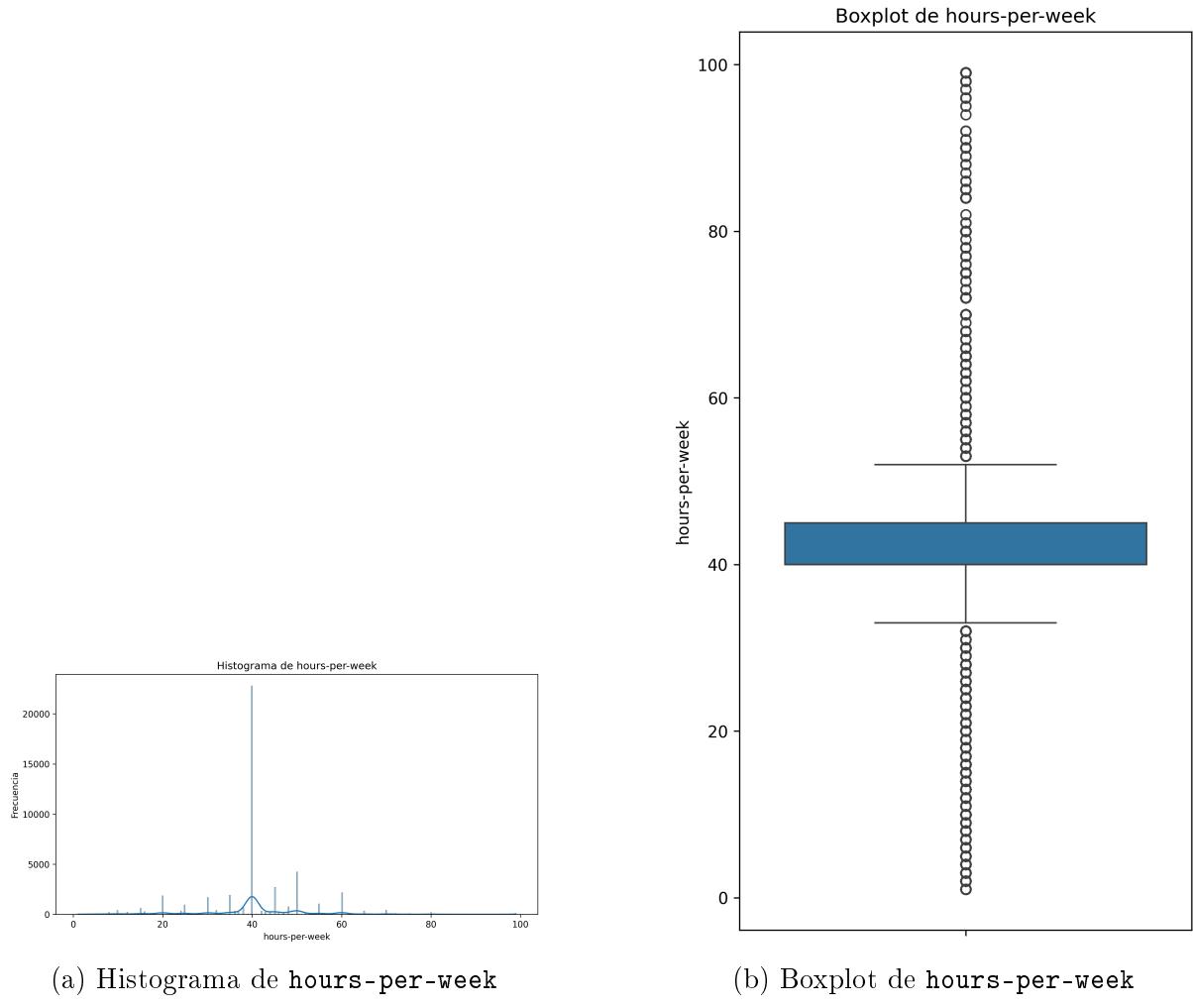


Figura 5: Distribución de *fnlwgt*: histograma y boxplot.

fnlwgt exhibe gran dispersión. El histograma muestra una cola derecha muy extensa y el boxplot múltiples valores atípicos altos. Si se utiliza como predictor, convendría probar transformaciones logarítmicas o normalizaciones robustas.



(a) Histograma de **hours-per-week**

(b) Boxplot de **hours-per-week**

Figura 6: Distribución de **hours-per-week**: histograma y boxplot.

La jornada semanal se concentra en 40 horas (mediana=40), con un incremento hacia 45 horas en Q3 y una cola derecha moderada hasta 99. El histograma sugiere picos en 40 y 45, lo que es esperado si se supone que las personas trabajan 8 horas durante 5 días. El boxplot indica pocos valores atípicos altos.

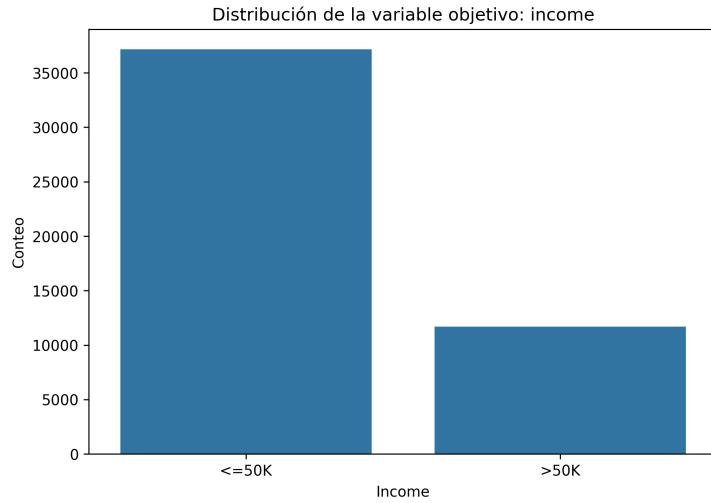


Figura 7: Histograma de la variable objetivo `income`.

La Figura 7 muestra un claro desequilibrio a favor de la categoría `<=50K`, lo cual puede afectar el desempeño de los modelos de clasificación.

2. Limpieza y preprocesamiento de datos

- **Valores nulos.** No se encontraron valores faltantes.

| Columna | Nulos (conteo) | Nulos (%) |
|----------------|----------------|-----------|
| age | 0 | 0.0 |
| workclass | 0 | 0.0 |
| fnlwgt | 0 | 0.0 |
| education | 0 | 0.0 |
| education-num | 0 | 0.0 |
| marital-status | 0 | 0.0 |
| occupation | 0 | 0.0 |
| relationship | 0 | 0.0 |
| race | 0 | 0.0 |
| sex | 0 | 0.0 |
| capital-gain | 0 | 0.0 |
| capital-loss | 0 | 0.0 |
| hours-per-week | 0 | 0.0 |
| native-country | 0 | 0.0 |
| income | 0 | 0.0 |

Tabla 3: Conteo y porcentaje de valores nulos por columna.

- **Duplicados.** Se identificaron 29 filas duplicadas y se eliminaron. No obstante, podrían corresponder a personas distintas con la misma combinación de atributos.
- **Tratamiento de faltantes.** Al no haber valores nulos, no fue necesaria la imputación ni eliminación adicional.

3. División del dataset

- **Split 80/20.** Se dividió en entrenamiento (80 %) y prueba (20 %) preservando la proporción de `income`.
- **Balance por clase.** La Tabla 4 muestra que la proporción por clase es prácticamente idéntica en ambos conjuntos.

| Categoría (<code>income</code>) | Entrenamiento | Prueba |
|-----------------------------------|---------------|----------|
| <=50K | 0.760581 | 0.760607 |
| >50K | 0.239419 | 0.239393 |

Tabla 4: Proporción de `income` en entrenamiento y prueba.

- **Tamaños de los conjuntos.**

| Conjunto | Forma | Descripción |
|----------|-------------|----------------------------------|
| X_train | (39032, 14) | Características de entrenamiento |
| X_test | (9758, 14) | Características de prueba |
| y_train | (39032,) | Etiquetas de entrenamiento |
| y_test | (9758,) | Etiquetas de prueba |

Tabla 5: Tamaños de los conjuntos de entrenamiento y prueba.

4. Correlaciones y relaciones

- **Matriz de correlación.**

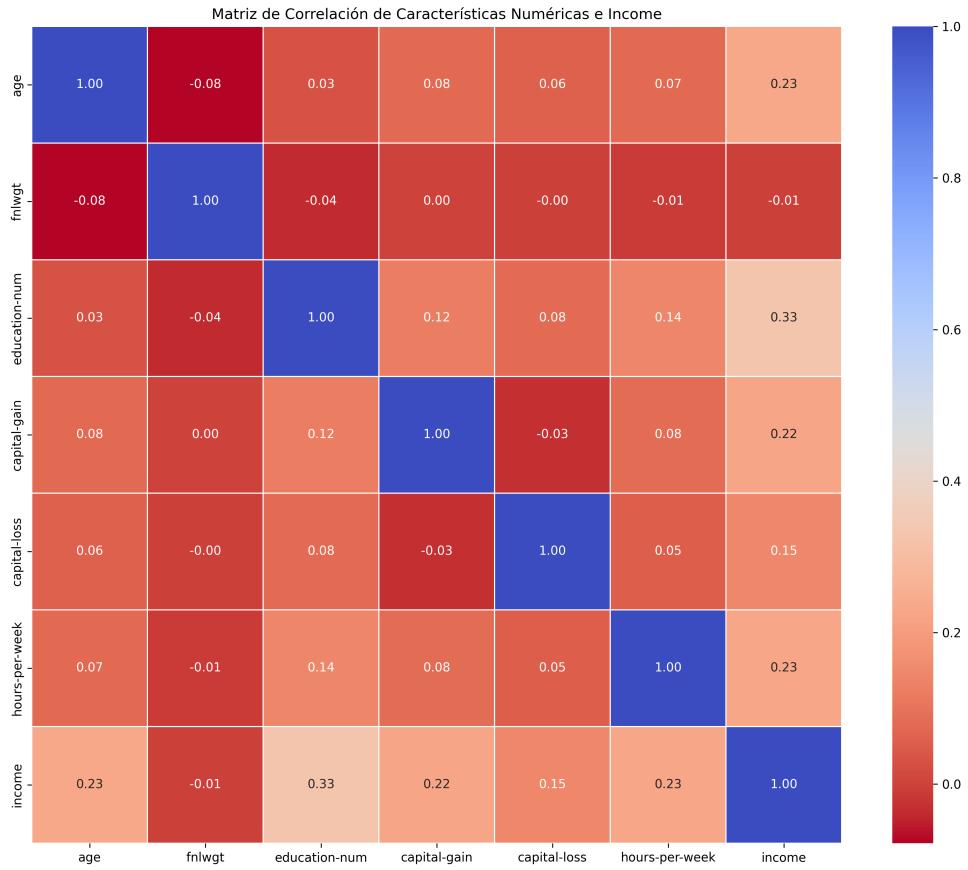


Figura 8: Matriz de correlación entre variables numéricas y *income*.

La matriz de correlación entre variables numéricas y *income*, codificada como binaria, muestra asociaciones bajas a moderadas: *education-num*, *hours-per-week* y *age* presentan señales útiles, mientras que *capital-gain* y *capital-loss* aportan información localizada en sus colas. El mapa de calor confirma que no hay colinealidades severas entre predictores numéricos, lo que favorece modelos lineales y árboles sin necesidad de selección por multicolinealidad.

- **Diagramas de dispersión.**

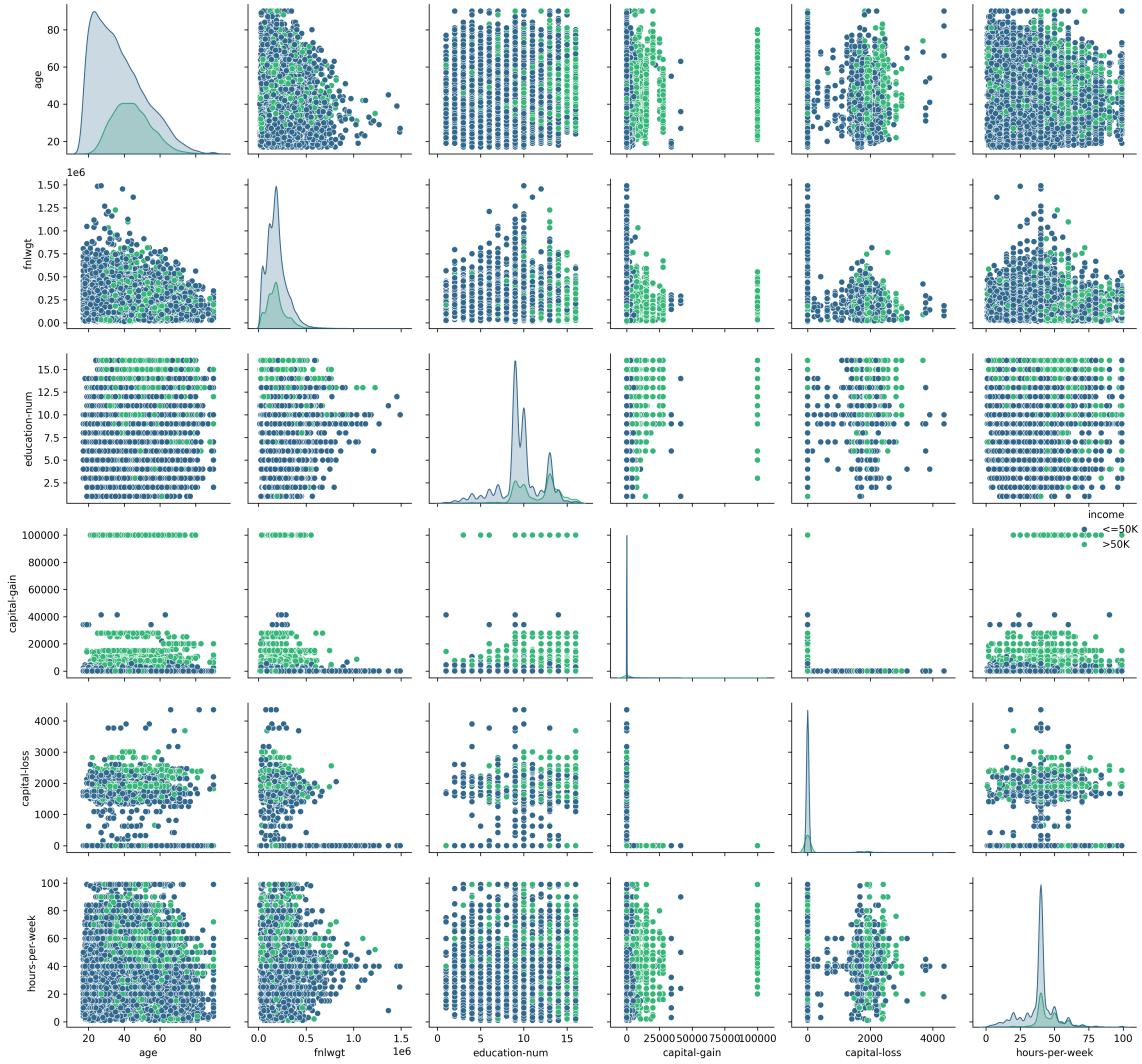


Figura 9: Diagramas de dispersión entre variables numéricas y `income`.

En los diagramas bivariados se observa un solapamiento importante entre clases para buena parte de las combinaciones, con nubes de puntos densas y sin fronteras lineales claras; sin embargo, emergen patrones útiles: valores positivos de `capital-gain` tienden a asociarse más con `>50K`, y combinaciones como `education-num-hours-per-week` o `age-hours-per-week` sugieren gradientes donde mayores niveles educativos o más horas trabajadas elevan la probabilidad de altos ingresos. Estas relaciones recomiendan explorar interacciones y transformaciones no lineales de los datos.

- **Regresión logística como clasificador.**

Con base en los gráficos y estadísticas, se identificaron cuatro variables con mayor capacidad predictiva para la variable objetivo: `age`, `education-num`, `hours-per-week` y `capital-gain`. Se utilizó regresión logística para clasificar la variable objetivo, probando distintas combinaciones de pares de estas variables para evaluar el desempeño y rendimiento del modelo. Se obtuvieron los siguientes resultados:

| Pareja | Accuracy | Macro P | Macro R | Macro F1 | Weighted F1 |
|---------------------------------|---------------|-------------|-------------|-------------|-------------|
| (age, education-num) | 0.7860 | 0.71 | 0.61 | 0.63 | 0.75 |
| (age, hours-per-week) | 0.7550 | 0.60 | 0.52 | 0.50 | 0.69 |
| (age, capital-gain) | 0.7934 | 0.78 | 0.59 | 0.60 | 0.74 |
| (education-num, hours-per-week) | 0.7863 | 0.73 | 0.59 | 0.60 | 0.74 |
| (education-num, capital-gain) | 0.8081 | 0.81 | 0.62 | 0.64 | 0.77 |
| (hours-per-week, capital-gain) | 0.7952 | 0.78 | 0.60 | 0.61 | 0.75 |

Tabla 6: Resumen global por pareja

| Pareja | Precision (>50K) | Recall (>50K) | F1 (>50K) |
|---------------------------------|------------------|---------------|-------------|
| (age, education-num) | 0.62 | 0.27 | 0.38 |
| (age, hours-per-week) | 0.44 | 0.08 | 0.14 |
| (age, capital-gain) | 0.76 | 0.20 | 0.32 |
| (education-num, hours-per-week) | 0.66 | 0.22 | 0.33 |
| (education-num, capital-gain) | 0.81 | 0.26 | 0.39 |
| (hours-per-week, capital-gain) | 0.76 | 0.21 | 0.33 |

Tabla 7: Desempeño en la clase > 50K

Entre las parejas evaluadas, (*education-num, capital-gain*) alcanza el mejor desempeño global con una *accuracy* de 0.8081 y el mayor *macro F1* (0.64), además del mejor *weighted F1* (0.77). Esto significa que juntar el nivel educativo con las ganancias de capital permite distinguir mejor entre quienes ganan más y menos de 50K. En segundo lugar quedan (*hours-per-week, capital-gain*) y (*age, capital-gain*), ambas alrededor de 0.79–0.80 de *accuracy* y *macro F1* en el rango 0.60–0.61, lo que sugiere que *capital-gain* aporta gran parte de la separación y que el segundo predictor (edad u horas) añade una ganancia moderada pero real. La pareja (*age, hours-per-week*) es la menos efectiva, con *accuracy* 0.7550 y un *macro F1* de 0.50, mostrando además el peor recobrado de la clase minoritaria.

Todas las parejas muestran una clara desigualdad: para $\leq 50K$ el *recall* ronda 0.97–0.98, mientras que para $> 50K$ casi nunca pasa de 0.27. La mejor recuperación de $> 50K$ aparece con (*age, education-num*) (*recall* 0.27), seguida de (*education-num, capital-gain*) (0.26). Aun así, (*education-num, capital-gain*) logra a la vez *precision* alta para $> 50K$ (0.81) y el mejor equilibrio general; en cambio, (*age, education-num*) sube el *recall* pero con *precision* menor (0.62) y peor *macro F1*. Esto sugiere que el límite de decisión del modelo favorece a la clase mayoritaria y que la señal de *capital-gain* es “limpia” (pocos falsos positivos) pero poco común (bajo *recall*). Probar con ajustar ese límite, dar más peso a $> 50K$ o crear variables simples (p. ej., un indicador *capital-gain>0* y una versión logarítmica) podría aumentar el *recall* de $> 50K$ sin perder mucha *precision*.

(*education-num, capital-gain*) ofrece el mejor equilibrio: mayor *accuracy* (exactitud) y *macro F1*, y detecciones de $> 50K$ de “buena calidad” (alta *precision*, 0.81), aunque con *recall* moderado (0.26). (*hours-per-week, capital-gain*) y (*age, capital-gain*) confirman que *capital-gain* es la variable que más

separa las clases: mantienen *precision* alta para >50K (0.76 en ambos casos) pero *recall* bajo (0.21–0.20). (*age*, *education-num*) logra el mayor *recall* en >50K (0.27), a costa de *precision* más baja (0.62) y menor *macro F1*; sigue siendo útil si la prioridad es “no perder” tantos casos positivos. (*education-num*, *hours-per-week*) queda en un punto intermedio y estable, mientras que (*age*, *hours-per-week*) se rezaga por su *recall* de 0.08 en >50K.

Si se busca buen desempeño general y pocas falsas alarmas en >50K, (*education-num*, *capital-gain*) es la mejor base. Si se quiere aumentar el *recall* de >50K, podría moverse el umbral de decisión y/o dar más peso a esa clase, sobre todo en parejas con *capital-gain*. Por último, añadir una tercera variable (p. ej., *age* o *hours-per-week*) probablemente ayudaría a mejorar el *recall* de >50K, siempre que también ajustes el umbral o el peso entre clases.

5. Árbol de decisión como clasificador.

| Clase | Precisión | Recall | F1 | Soporte |
|---------------------------|---------------|--------|------|---------|
| ≤50K | 0.89 | 0.88 | 0.88 | 7422 |
| >50K | 0.62 | 0.64 | 0.63 | 2336 |
| Accuracy | 0.8216 | | | |
| Macro promedio | 0.75 | 0.76 | 0.76 | 9758 |
| Promedio ponderado | 0.82 | 0.82 | 0.82 | 9758 |

Tabla 8: Desempeño del clasificador en el conjunto de prueba

El modelo alcanza una precisión general del 82.16 %, lo que indica un buen desempeño global. En la clase menor a 50K, los aciertos son muy altos y constantes (alrededor de 0.88 en todas las métricas), señal de que el modelo identifica bien a quienes ganan hasta 50K. La clase >50K que suele ser la más difícil de predecir muestra un equilibrio sano: cuando el modelo dice que alguien gana más de 50K acierta 62 % de las veces (precisión), y además encuentra 64 % de las personas que realmente están en ese grupo (recuperación). El promedio entre clases (macro) ronda 0.75–0.76, lo que confirma que el rendimiento no depende únicamente de la clase grande; hay un balance razonable entre ambas.

■ Experimentando con diferentes hiperparámetros.

| Clase | Precisión | Recall | F1 | Soporte |
|---------------------------|---------------|--------|------|---------|
| ≤50K | 0.89 | 0.94 | 0.91 | 7422 |
| >50K | 0.78 | 0.61 | 0.68 | 2336 |
| Accuracy | 0.8651 | | | |
| Promedio macro | 0.83 | 0.78 | 0.80 | 9758 |
| Promedio ponderado | 0.86 | 0.87 | 0.86 | 9758 |

Tabla 9: Desempeño del Árbol de Decisión optimizando hiperparámetros

El clasificador optimizado alcanzó una precisión global de 86.51 % en el conjunto de prueba, lo que indica un desempeño sólido en términos generales. La

clase $\leq 50K$ presenta métricas altas (precisión 0.89, recall 0.94 y F1 0.91), evi-
denciando una identificación consistente de esta categoría. Para la clase $> 50K$
se observa un equilibrio favorable entre aciertos y cobertura (precisión 0.78 y
recall 0.61; F1 0.68), lo que sugiere que el modelo es selectivo al predecir ingre-
sos altos y, al mismo tiempo, recupera una proporción significativa de los casos
verdaderamente positivos. Los promedios macro (F1 = 0.80) y ponderado (F1
= 0.86) refuerzan la idea de un rendimiento equilibrado entre ambas clases,
sin depender exclusivamente de la mayoritaria. La búsqueda de hiperparáme-
tros exploró combinaciones de `max_depth` y `min_samples_split` con vali-
dación cruzada, favoreciendo una configuración que controla la complejidad del
árbol y mejora la generalización. La mejor configuración de hiperparámetros
encontrada fue: `max_depth` = 10, `min_samples_split` = 20.

Se intentó generar el arbol de decisión con todas las reglas evaluadas, sin em-
bargo, es visualmente incomodo, porque unas reglas se solapan con otras difu-
cultando su legibilidad.

6. Clasificador K- Vecinos más cercanos.

| Clase | Precision | Recall | F1-score | Soporte |
|---------------------|-----------|--------|----------|---------|
| $\leq 50K$ | 0.82 | 0.91 | 0.86 | 7422 |
| $> 50K$ | 0.56 | 0.34 | 0.43 | 2336 |
| Accuracy | 0.78 | | | |
| Macro avg | 0.69 | 0.63 | 0.64 | 9758 |
| Weighted avg | 0.75 | 0.78 | 0.76 | 9758 |

Tabla 10: Resultados del modelo KNN

El modelo KNN alcanzó una *accuracy* de 0.78, mostrando un buen desempeño para
la clase mayoritaria ($\leq 50K$, con *recall* 0.91 y *precision* 0.82), pero limitaciones
claras en la clase minoritaria ($> 50K$, con *recall* 0.34 y *precision* 0.56). En general,
el modelo tiende a favorecer la detección de la clase más común, lo cual se refleja
en un *macro F1* relativamente bajo (0.64) frente a un *weighted F1* más alto (0.76).
Esto indica que, si bien KNN es confiable para predecir ingresos menores o iguales
a 50K, requiere ajustes en hiperparámetros, balanceo de clases o el uso de variables
 adicionales para mejorar la cobertura en la clase $> 50K$. Para este ejercicio se usó un
número de vecinos (k) de 5.

- **Probando diferentes K.**

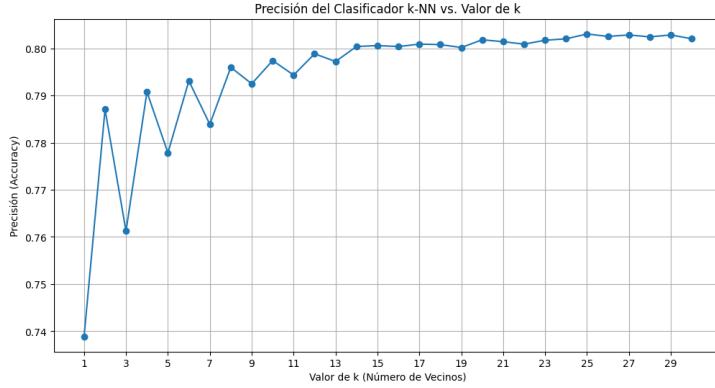


Figura 10: Desempeño del clasificador con diferentes K

La gráfica evidencia que el clasificador KNN con $k = 1$ presenta la menor *accuracy* (0.74), reflejando tal vez un ajuste excesivo a los datos de entrenamiento. A medida que el número de vecinos aumenta, la precisión mejora de forma rápida y alcanza valores cercanos a 0.79 en torno a $k = 5-k = 7$. A partir de $k = 15$, la *accuracy* se estabiliza alrededor de 0.80–0.81, mostrando un comportamiento más consistente y robusto. En general, los valores intermedios y altos de k proporcionan el mejor desempeño y un modelo más confiable. El impacto de k en el clasificador es claro: con valores bajos, el modelo es muy sensible y tiende a sobreajustar; con valores altos, se logra mayor estabilidad pero también se suavizan las fronteras de decisión. Por ello, un rango intermedio (entre 15 y 25 vecinos) resulta el más adecuado, pues equilibra la variabilidad del modelo con su capacidad de generalización.

Los resultados mostrados en la Figura 10 se realizaron sobre el dataset de prueba, previamente habiendo entrenado el modelo con el dataset de entrenamiento.

7. Análisis comparativo.

■ Comparación de rendimiento de los modelos.

En la Tabla 11 se presentan los resultados globales de los tres modelos implementados: clasificador basado en reglas, árbol de decisión y KNN. Se reportan las métricas de *accuracy*, F1 macro y F1 ponderado, así como las principales observaciones respecto a la clase minoritaria ($>50K$).

| Modelo | Accuracy | Macro F1 | Weighted F1 | Recall (>50K) | Observaciones |
|--------------------------------------|----------|----------|-------------|---------------|---|
| Reglas (education-num, capital-gain) | 0.81 | 0.64 | 0.77 | 0.26 | Alta precisión (0.81) en >50K, bajo recall. |
| Árbol de decisión (optimizado) | 0.87 | 0.80 | 0.86 | 0.61 | Mejor balance global, desempeño robusto. |
| KNN (k=5) | 0.78 | 0.64 | 0.76 | 0.34 | Favorece la clase mayoritaria, pobre en >50K. |

Tabla 11: Comparación del rendimiento de los modelos.

En términos generales, el **árbol de decisión optimizado** es el clasificador con mejor rendimiento y balance entre clases, alcanzando una exactitud de 86.5 % y un F1 macro de 0.80. El **clasificador basado en reglas** muestra un desempeño aceptable (accuracy de 81 %), pero con bajo recall para la clase minoritaria. Finalmente, el **KNN** (evaluado en $k = 5$) alcanza 78 % de exactitud, aunque presenta un sesgo importante hacia la clase mayoritaria.

- **Ventajas y desventajas en el contexto del dataset. Clasificador basado en reglas**

- **Ventajas:** fácil de interpretar; reglas simples con sentido socioeconómico (educación y capital-gain).
- **Desventajas:** baja capacidad para capturar relaciones complejas; muy bajo recall en la clase >50K.

Árbol de decisión

- **Ventajas:** mejor balance entre precisión y recall; interpreta relaciones no lineales y variables categóricas sin gran preprocesamiento; relativamente interpretable.
- **Desventajas:** riesgo de sobreajuste si no se regulan los hiperparámetros; árboles grandes son difíciles de interpretar.

KNN

- **Ventajas:** implementación simple, no requiere supuestos estadísticos; buen desempeño con valores intermedios de k .
- **Desventajas:** sensible a la escala y al desbalance de clases; bajo recall en la clase >50K; elevado costo computacional en predicción.

En conclusión, para este conjunto de datos el **árbol de decisión optimizado** resulta la mejor alternativa, ya que logra un desempeño sólido y balanceado entre las dos clases.

Clasificador de mayor votación. Los tres clasificadores con mejor rendimiento identificados son:

- a) Árbol de Decisión con hiperparámetros optimizados (`max_depth=10, min_samples_split=20`)
- b) k-NN con k óptimo ($k=25$, basado en la exploración)
- c) El mejor clasificador basado en reglas (regresión logística) de los pares de variables (`educationnum, capitalgain`)

Al ejecutar el modelo de mayor votación se tienen los siguientes resultados:

| Clase | Precisión | Recall | F1 | Soporte |
|---------------------------|-----------|---------------|------|---------|
| $\leq 50K$ | 0.81 | 0.99 | 0.89 | 7422 |
| $> 50K$ | 0.90 | 0.28 | 0.43 | 2336 |
| Accuracy | | 0.8215 | | |
| Promedio macro | 0.86 | 0.64 | 0.66 | 9758 |
| Promedio ponderado | 0.84 | 0.82 | 0.78 | 9758 |

Tabla 12: Desempeño del clasificador de mayor votación

El clasificador alcanza una precisión global de 82.15 % en el conjunto de prueba, lo que indica un desempeño correcto en términos generales. En la clase $\leq 50K$ se observa una recuperación prácticamente completa ($recall = 0.99$) con una precisión de 0.81 y F1 de 0.89; esto sugiere que el modelo identifica casi todos los casos de

ingresos no superiores a 50K, aun cuando comete algunos falsos positivos hacia esta clase. Para la clase $> 50K$ se presenta el patrón complementario: una precisión muy alta (0.90) y un recall moderado-bajo (0.28), con F1 de 0.43. En términos prácticos, cuando el modelo predice $> 50K$ suele acertar, pero deja sin detectar una proporción importante de casos verdaderamente positivos. Los promedios macro ($F1 = 0.66$) y ponderado ($F1 = 0.78$) reflejan el compromiso entre ambas clases: buen rendimiento general con fuerte sesgo hacia $\leq 50K$ en la recuperación. Para mejorar la cobertura de $> 50K$ puede considerarse un ajuste del umbral de decisión y/o la incorporación de ponderación de clases, asumiendo un incremento esperado en falsos positivos de la clase alta.

8. Comparación con los modelos.

En conjunto, el ensamble por votación no supera al mejor componente (árbol optimizado) y, además, hereda el sesgo de “alta precisión y bajo *recall*” del modelo basado en reglas (regresión logística). Dado que el árbol optimizado ofrece la mejor cobertura de $> 50K$ con el mejor desempeño global, una alternativa más efectiva sería usar *soft voting* con pesos mayores para el árbol, o bien ajustar el umbral del ensamble para incrementar el *recall* de $> 50K$ (asumiendo el costo de más falsos positivos).

9. Variando el tamaño del dataset de prueba.

De acuerdo con lo indicado por el **Assignment #2**, se procedió a realizar el mismo ejercicio de clasificación que se ha realizado hasta el momento con cada uno de los modelos propuestos (Regresión logística, Árbol de Decisión, knn y mayor Votación), pero variando el tamaño (0.6, 0.4, 0.2, 0.1) del dataset de prueba y por tanto el de entrenamiento. Bueno aunque cabe resaltar que los ejercicios de clasificación anteriores ya han trabajado con la proporcion 80/20.

| Modelo | 0.1 | 0.2 | 0.4 | 0.6 |
|---------------------|----------|----------|----------|----------|
| Decision Tree | 0.832343 | 0.821582 | 0.816868 | 0.812837 |
| k-NN | 0.802828 | 0.803033 | 0.798985 | 0.799378 |
| Logistic Regression | 0.812667 | 0.808055 | 0.804212 | 0.804912 |
| Voting Classifier | 0.823529 | 0.819123 | 0.813640 | 0.813008 |

Tabla 13: Desempeño por umbral

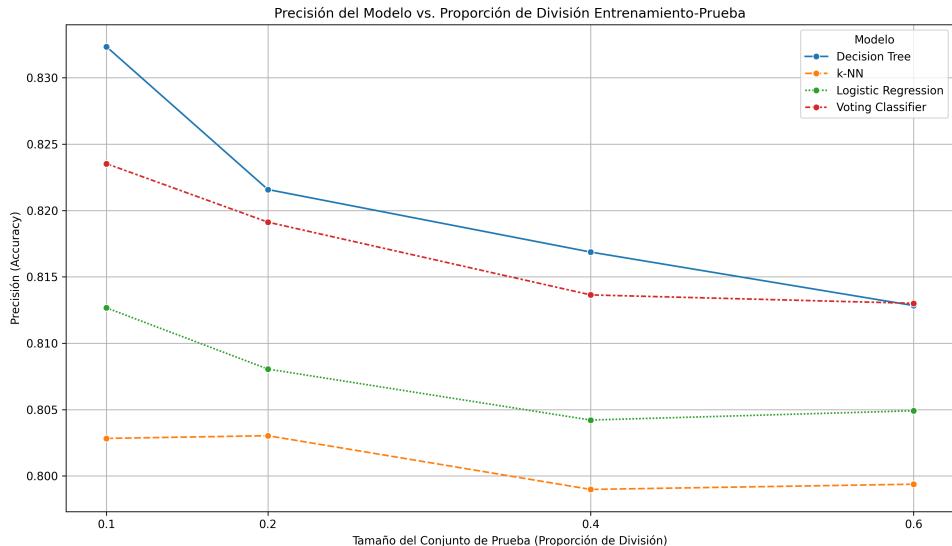


Figura 11: Desempeño modelos variando tamaño del dataset de prueba

En general, al aumentar el tamaño del conjunto de prueba, la precisión de los modelos tendió a disminuir. El *árbol de decisión* obtuvo los mejores resultados tanto en la partición 90/10 como en la 40/60, mientras que *k*-NN fue el de menor desempeño en esos mismos escenarios. La *regresión logística* se mantuvo estable y poco sensible a los cambios en la proporción de entrenamiento y prueba. El modelo de *votación mayoritaria* siguió el mismo patrón que los modelos individuales, con una ligera caída de rendimiento al crecer el conjunto de prueba. Como lectura práctica, los árboles de decisión parecen beneficiarse más de contar con más datos para entrenar y son más sensibles al reparto de los datos que la regresión logística. El ensamble por mayoría no compensó de forma notable la reducción del conjunto de entrenamiento.

5. Conclusiones

1. El conjunto presenta fuerte desbalance ($\leq 50\text{K}$ mayoritario) y variables con asimetrías extremas (p.ej., *capital-gain* y *capital-loss* con $Q_1 = Q_2 = Q_3 = 0$), lo que afecta el *recall* de la clase $> 50\text{K}$ y sugiere transformar estas variables.
2. Entre las reglas basadas en regresión logística con dos predictores, la pareja (*education-num*, *capitalgain*) logró el mejor equilibrio global pero con el patrón de “alta precisión y bajo *recall*” para $> 50\text{K}$.
3. El Árbol de Decisión optimizado fue el mejor clasificador en términos globales, con buen balance entre clases, y usando `max_depth = 10` y `min_samples_split = 20` como valores de estos hiperparámetros.
4. El modelo por mayor votación no superó al mejor componente (árbol optimizado) y heredó el sesgo de alta precisión y bajo *recall* en $> 50\text{K}$. Para mejorar, podría convenir usar *soft voting* con mayor peso al árbol.
5. Al variar la proporción entrenamiento/prueba, la *accuracy* tendió a bajar cuando creció el conjunto de prueba; el árbol se benefició más de contar con más datos de

entrenamiento, mientras que la regresión logística se mantuvo relativamente estable. El modelo por mayor votación no compensó la reducción del tamaño de entrenamiento, reforzando la recomendación de priorizar el árbol de decisión optimizado.