UISIL

PROYECTO No. 1: Data Mining o Minería de Datos

Marlon Jiménez Madrigal

Samuel Saldaña

Especialización Ciencia de Datos

PROYECTO No. 1:

Data Mining o Minería de Datos

I. EMPLEAR LA FASE CONCEPTUAL DEL MODELO CRISP-DM DE DATA MINING, E INDICAR QUÉ ÁREA DEL DATA MINING HA DE EMPLEAR EN SU PROYECTO FINAL.

1. Comprensión del negocio (organización, tipo de actividad)

   Tipo de organización: Agencia gubernamental, organización de investigación socioeconómica, empresa de análisis de datos, o entidad educativa.

   Objetivo principal: Analizar los factores socioeconómicos que influyen en los ingresos de las personas y entender las disparidades económicas entre diferentes grupos demográficos.

   ***Actividad principal*:** Investigación y análisis de datos socioeconómicos y demográficos para generar informes, políticas, y estrategias que aborden las desigualdades económicas y laborales.

   Objetivos

- Determinar qué características individuales (como edad, educación, ocupación, etc.) están más fuertemente asociadas con los ingresos.
- Analizar cómo factores demográficos (como raza, sexo, y país de origen) afectan los ingresos.
- Categorizar a los individuos en diferentes grupos de ingresos para identificar patrones y tendencias socioeconómicas.

2. ***Comprensión de los datos (identificar y explorar los datos).***

Esta base de datos tiene un total de 32,561 filas y 15 columnas. Los datos que se muestran inicialmente son puros, lo que significa que estos datos se encuentran en un estado sin procesar, por lo tanto, mantiene datos con valores nulos y la posibilidad de registros duplicados que aún no han sido identificados ni gestionados.

Esta base de datos o archivo .csv hace referencia a un censo realizado en los Estados Unidos para confirmar lo ingresos y el estatus de las personas encuestadas. La base de datos alberga información diversa y heterogénea de personas pertenecientes a diferentes grupos étnicos, estratos socioeconómicos, niveles educativos y estados civiles. Esta variedad de datos refleja una amplia gama de características demográficas y socioeconómicas que serán objeto de análisis en etapas posteriores del proyecto. Las columnas presentes en la base de datos y que fueron cuestionadas son: la edad, el tipo de ocupación, el nivel de educación alcanzado, el estado civil, la ocupación, la relación familiar, la raza, el sexo, las ganancias de capital y las pérdidas de capital. Cada una de ellas aporta una dimensión valiosa al análisis, permitiendo una comprensión integral de las características y comportamientos de los individuos en el conjunto de datos.

	age	workclass	fnlwgt	education	education.num	marital.status	occupation	relationship	race	sex	capital.gain	capital.loss	hours.per.week	native.country	income
0	90	?	77053	HS-grad	9	Widowed	?	Not-in-family	White	Female	0	4356	40	United-States	<=50K
1	82	Private	132870	HS-grad	9	Widowed	Exec-managerial	Not-in-family	White	Female	0	4356	18	United-States	<=50K
2	66	?	186061	Some-college	10	Widowed	?	Unmarried	Black	Female	0	4356	40	United-States	<=50K
3	54	Private	140359	7th-8th	4	Divorced	Machine-op-inspct	Unmarried	White	Female	0	3900	40	United-States	<=50K
4	41	Private	264663	Some-college	10	Separated	Prof-specialty	Own-child	White	Female	0	3900	40	United-States	<=50K

Debido a que estos datos no han sido sometidos a un análisis o limpieza de datos, es fundamental llevar a cabo una serie de pasos y análisis para verificar la integridad y calidad de la información. Algunos de

estos pasos que vamos a ejecutar esta, la identificación y eliminación de valores nulos, así como la detección y manejo de posibles duplicados, para garantizar que el análisis se base en un conjunto de datos confiable y representativo. Como bien sabemos la limpieza de datos es importante para asegurar que los datos con los que se vaya a trabajar sean precisos y no haya errores o sean los mínimos. En caso de haber datos faltantes estaríamos reemplazándolos con el 1 o la palabra unknown en caso de ser texto. Analizando de antemano la base de datos podemos confirmar que ya se encuentra normalizada, sin embargo, puede haber datos incorrectos a la hora de que se ingresaron los datos.

De igual manera, se debe realizar un análisis exploratorio de los datos para identificar patrones, tendencias y posibles anomalías. Este análisis preliminar ayudará a formular hipótesis y determinar las técnicas de modelado más adecuadas para el estudio. Las técnicas de minería de datos, como la minería de reglas de asociación, la clasificación, el agrupamiento en clústeres y el análisis de secuencias y trayectorias, se aplicarán posteriormente para extraer conocimientos significativos y apoyar la toma de decisiones basada en datos.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32561 entries, 0 to 32560
Data columns (total 15 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   age             32561 non-null  int64
 1   workclass       32561 non-null  object
 2   fnlwgt          32561 non-null  int64
 3   education       32561 non-null  object
 4   education.num   32561 non-null  int64
 5   marital.status  32561 non-null  object
 6   occupation      32561 non-null  object
 7   relationship    32561 non-null  object
 8   race            32561 non-null  object
 9   sex             32561 non-null  object
 10  capital.gain    32561 non-null  int64
 11  capital.loss    32561 non-null  int64
 12  hours.per.week  32561 non-null  int64
 13  native.country  32561 non-null  object
 14  income          32561 non-null  object
dtypes: int64(6), object(9)
```

Data info

Como se puede ver en esta imagen los datos tenemos unas listas que reflejan más o menos el estado actual de la base de datos. Podemos ver que hay unos datos que reflejan el símbolo de pregunta "?" lo cual puede traer problemas a la hora de procesar y obtener la información necesaria para generar análisis y reportes.

Las columnas la podemos obtener mediante el código data.columns.values y nos da como resultado: *age, workclass, fnlwgt, education, education.num, marital.status, occupation, relationship, race, sex, capital.gain, capital.loss, hours.per.week, native.country, income], dtype=object)*

Por lo que podemos cada uno hace referencia a un espacio específico:

a) **Age:** edad
b) **Workclass:** clase de trabajo.
c) **Fnlwgt:** se traduce como final weight, es el número de personas en el censo y su número en la posición.
d) **Education:** el grado de educación que tiene cada entrevistado.
e) **Education.num:** es el número de valor.
f) **Marital status**: estado civil.
g) **Occupation:** ocupación.
h) **Relationship:** relación. Esto nos indica si vive en familia, solo o no.
i) **Race:** raza.
j) **Sex:** sexo.
k) **Capital.gain:** capital ganado.
l) **Capital.loss:** capital perdido.
m) **Hour.per.week:** horas trabajadas por semana.
n) **Native.country:** país nativo.

o) **Income:** ingreso.

Ahora vamos a realizar un análisis de los datos que podemos encontrar en nuestra base de datos para identificar cuáles son los más relevantes y útiles para nuestro proyecto. Este análisis es fundamental para asegurarnos de que estamos utilizando la información correcta y aquellos que se ajustan a lo que queremos entregar para su futuro análisis e interpretación de los departamentos encargados. Al ser un censo sabemos que esta información puede ser útil para alguna entidad bancaria, gubernamental, entre otras. Nos enfocaremos en diversas variables demográficas y socioeconómicas que son cruciales para el entendimiento de los patrones y relaciones dentro de nuestra base de datos.

La selección de las variables adecuadas es esencial para obtener resultados significativos y precisos. Entre las vamos a considerar están: edad, clase trabajadora, educación, estado civil, relación, raza, sexo, horas trabajadas, país.

Estas variables proporcionan una visión completa y multifacética de la población estudiada, permitiendo una comprensión profunda de las diferentes dimensiones que afectan a las personas en diversos contextos socioeconómicos.

Con respecto a la edad, podemos obtener información vital sobre en qué etapa de su vida se encuentra, y esto también nos puede ayudar a confirmar si la persona se encuentra en su clímax laboral, recién empieza o ya ha terminado su etapa como persona trabajadora y asalariada. Al obtener información sobre su posición laboral, se puede saber también sus ocupaciones y de ahí también si la persona cuenta con estudios o no los finalizo. A pesar de que muchas veces, altos ingresos signifiquen alta escolaridad, no siempre es así y aquí vamos a identificar esto.

El estado civil y las relaciones familiares nos permiten explorar las dinámicas personales y familiares que pueden influir en el bienestar económico y social de los individuos.

La raza y el género son variables importantes para identificar y analizar las posibles disparidades y desigualdades dentro de la población. Como bien se ha dicho, las personas son discriminadas según su género y su raza como por ejemplo al ser mujer latina o afrodescendiente. Las horas trabajadas proporcionan información sobre el esfuerzo laboral y su relación con otros factores como la remuneración y el tipo de empleo. A más horas laboradas más ingresos, sin embargo, no todas las posiciones permiten que sus empleados puedan hacer horas extras o incluso, trabajan horas de más sin recibir pago extra por sus funciones al considerarse empleados de confianza o similar. También tenemos el país de origen nos ayuda a entender la diversidad cultural y geográfica, y cómo ésta puede impactar en las condiciones socioeconómicas.

Este análisis inicial nos permitirá no sólo identificar las características más importantes de nuestra base de datos, sino también establecer las bases para un proceso de limpieza y preparación de datos riguroso. Es vital asegurarse de que los datos estén libres de inconsistencias, valores nulos y duplicados, lo que garantizará la fiabilidad de nuestros resultados.

Al tener una comprensión clara de las variables más importantes, podremos aplicar técnicas de minería de datos de manera más efectiva. Esto incluye la minería de reglas de asociación para descubrir patrones ocultos, técnicas de clasificación para predecir

categorías específicas, agrupamiento en clústeres para segmentar la población en grupos homogéneos, y análisis de secuencias y trayectorias para entender comportamientos a lo largo del tiempo.

Entre los códigos que usamos para obtener estos datos están:

Para obtener la edad y ordernarlos.

```
val_unique = data["age"].unique()

val_unique.sort()  # Ordenarlos

val_unique = val_unique[::-1]

print(val_unique)
```

Y a nivel general para obtener la información de estas columnas fueron:

```
data["column_name"].describe()

data["column_name "].unique()

data["column_name "].mode()

print("El más común es", data["column_name "].mode())

unm = data["column_name "].value_counts().max

print(unm)
```

a) ***Age:*** tenemos que la edad mínima es de 17 años y el de mayor edad es de 90 años y el que más se repite es la de 36 años.
   Las edades que aparecen en la base de datos son: [*90, 88, 87, 86, 85, 84, 83, 82, 81, 80, 79, 78, 77, 76, 75, 74, 73, 72, 71, 70, 69, 68, 67, 66, 65, 64, 63, 62, 61, 60, 59, 58, 57, 56, 55, 54, 53, 52, 51, 50, 49, 48, 47, 46, 45, 44, 43, 42, 41, 40, 39, 38, 37, 36, 35, 34, 33, 32, 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17*]

b) ***Workclass:*** tenemos que hay diferentes tipos de valores. Habiendo hasta 9 diferentes. La más común es 'Private'. Además tenemos también: *State-gov, Federal-gov, Self-emp-not-inc, Self-emp-inc, Local-gov, Without-pay, Never-worked.*

c) ***Education:*** en este Podemos ver que el grado académico más común es 'HS-grad' el cual es el equivalente al bachillerato de colegio. También podemos encontrar *'Some-college, 7th-8th, 10th, Doctorate, Prof-school, Bachelors, Masters, 11th, Assoc-acdm, Assoc-voc, 1st-4th, 5th-6th, 12th, 9th, Preschool*

Es interesante observar la gran cantidad de diferencia educativa presente en nuestra base de datos, que va desde personas que apenas han asistido al kínder hasta aquellos que han alcanzado el nivel de doctorado. Esta amplia gama de niveles educativos podría significar mucho a la hora de reflejar sus posibles impactos en la vida laboral y personal.

La inclusión de personas con educación mínima, como aquellos que solo han asistido al kínder, permite analizar cómo las oportunidades y desafíos se distribuyen a lo largo de diferentes niveles educativos. Por otro lado, la presencia de individuos con títulos avanzados, como doctorados, nos brinda la oportunidad de investigar las ventajas y beneficios asociados con niveles superiores de educación. Esta diversidad es esencial para entender las dinámicas educativas y socioeconómicas que influyen en el mercado laboral, los ingresos y la calidad de vida.

d) ***Marital.status:*** en el estado civil tenemos como el más común: Married-civ-spouse. También tenemos: *Widowed, Divorced,*

Separated, Never-married, Married-civ-spouse, Married-spouse-absent, Married-AF-spouse.

e) ***Relationship:*** respect a la relación podemos observar que el más común es Husband. Pero también tenemos: *Not-in-family, Unmarried, Own-child, Other-relative, Husband, Wife.*

f) ***Race:*** dentro de la raza la más común es: "White". Sin embargo, también podemos encontrar: *White, Black, Asian-Pac-Islander, Other, Amer-Indian-Eskimo.*

g) ***Sex:*** en esta columna la más común es: "Male". Aquí solo tenemos dos opciones que son masculino y femenino.

h) ***Work.per.week:*** las horas trabajadas por semana más común es la de 40 horas. Pero a gran variedad de horas laboradas en la semana: [*99, 98, 97, 96, 95, 94, 92, 91, 90, 89, 88, 87, 86, 85, 84, 82, 81, 80, 78, 77, 76, 75, 74, 73, 72, 70, 68, 67, 66, 65, 64, 63, 62, 61, 60, 59, 58, 57, 56, 55, 54, 53, 52, 51, 50, 49, 48, 47, 46, 45, 44, 43, 42, 41, 40, 39, 38, 37, 36, 35, 34, 33, 32, 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1*]

i) **Native.country:** el país más común es United States pero también tenemos: *United-States, ?, Mexico, Greece, Vietnam, China, Taiwan, India, Philippines, Trinadad&Tobago, Canada, South, Holand-Netherlands, Puerto-Rico, Poland, Iran, England, Germany, Italy, Japan, Hong, Honduras, Cuba, Ireland, Cambodia, Peru, Nicaragua, Dominican-Republic, Haiti, El-Salvador, Hungary, Columbia, Guatemala, Jamaica, Ecuador, France, Yugoslavia, Scotland, Portugal, Laos, Thailand, Outlying-US(Guam-USVI-etc).*

j) ***Income*****:** el ingreso más común es: <=50K

Ya vimos cuáles son los valores que están en cada una de las columnas y cuáles son las más comunes. Ahora vamos a verificar cuántas veces sale cada uno de los elementos en cada columna.

Workclass:

Private	22696
Self-emp-not-inc	2541
Local-gov	2093
?	1836
State-gov	1298
Self-emp-inc	1116
Federal-gov	960
Without-pay	14
Never-worked	7

Education:

HS-grad	10501
Some-college	7291
Bachelors	5355
Masters	1723
Assoc-voc	1382
11th	1175
Assoc-acdm	1067
10th	933
7th-8th	646
Prof-school	576
9th	514
12th	433

Doctorate	413
5th-6th	333
1st-4th	168
Preschool	51

Marital.status:

Married-civ-spouse	14976
Never-married	10683
Divorced	4443
Separated	1025
Widowed	993
Married-spouse-absent	418
Married-AF-spouse	23

Sex:

Male	21790
Female	10771

Relationship:

Husband	13193
Not-in-family	8305
Own-child	5068
Unmarried	3446
Wife	1568
Other-relative	981

Race:

White	27816
Black	3124
Asian-Pac-Islander	1039
Amer-Indian-Eskimo	311
Other	271

Native.country:

United-States	29170
Mexico	643
?	583
Philippines	198
Germany	137
Canada	121
Puerto-Rico	114
El-Salvador	106
India	100
Cuba	95
England	90
Jamaica	81
South	80
China	75
Italy	73
Dominican-Republic	70
Vietnam	67
Guatemala	64
Japan	62
Poland	60
Columbia	59
Taiwan	51
Haiti	44

Iran	43
Portugal	37
Nicaragua	34
Peru	31
Greece	29
France	29
Ecuador	28
Ireland	24
Hong	20
Cambodia	19
Trinadad&Tobago	19
Laos	18
Thailand	18
Yugoslavia	16
Outlying-US(Guam-USVI-etc)	14
Hungary	13
Honduras	13
Scotland	12
Holand-Netherlands	1

Income:

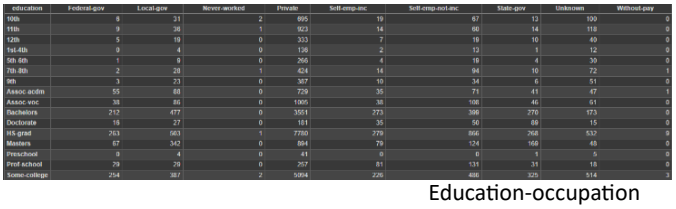
<=50K	24720
>50K	7841

En nuestra base de datos, observamos que varias columnas contienen exclusivamente datos numéricos, ya sea de valores enteros o flotantes. Estas columnas cuantitativas son fundamentales para el análisis, ya que permiten realizar cálculos estadísticos, evaluaciones y modelado de datos con precisión. En total, la base de datos consta de 32,561 filas y 6 columnas numéricas. Esta configuración permite una gran variedad de análisis, desde simples estadísticas descriptivas hasta complejos modelos predictivos. La presencia de una gran cantidad de filas asegura que el análisis sea robusto y representativo, capturando una amplia gama de variabilidad en los datos.

3. Preparación de los datos (organizar, clasificar, agrupar, etc.).

Estamos preparados para proceder al análisis comparativo entre columnas. En este caso, vamos a centrarnos en analizar la relación entre las columnas education y workclass.

El objetivo de este análisis es entender cómo se distribuyen los niveles educativos entre las diferentes categorías de clases trabajadoras y detectar patrones o tendencias que puedan surgir. Este análisis nos permitirá obtener mayor información sobre la interrelación entre la educación y el tipo de ocupación que tienen las personas, lo cual es crucial para entender la dinámica del mercado laboral y las oportunidades que diferentes niveles educativos ofrecen a los individuos.



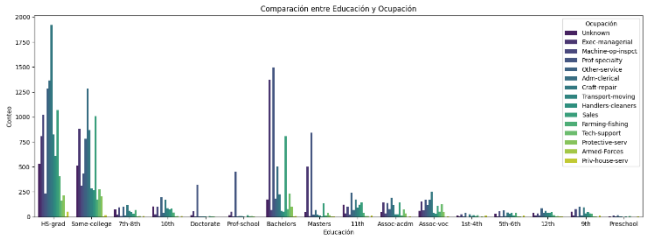
Education	Federal-gov	Local-gov	Never-worked	Private	Self-emp-inc	Self-emp-not-inc	State-gov	Unknown	Without-pay	
10th	9	31	2	895	19	67	13	199	0	
11th	9	36	1	923	14	60	14	118	0	
12th	5	19	0	333	7	19	10	49	0	
1st-4th	3	4	0	138	2	13	1	12	0	
5th-6th	1	8	0	266	4	18	4	39	0	
7th-8th	2	20	1	424	14	94	10	72	1	
9th	3	23	0	387	10	34	6	51	0	
Assoc-acdm	55	68	0	729	35	71	41	47	1	
Assoc-voc	38	86	0	1005	38	108	46	61	0	
Bachelors	212	477	0	3551	273	396	270	173	0	
Doctorate	18	27	0	181	35	50	88	15	0	
HS-grad	263	663	1	7760	279	566	268	532	0	
Masters	67	342	0	884	79	124	160	48	0	
Preschool	0	4	0	41	0	0	1	5	0	
Prof-school	29	28	0	257	81	131	31	18	0	
Some-college	254	367	2	5094	226	486	329	514	1	

Education-occupation

Por ejemplo, podemos notar que la mayoría de graduados del colegio trabajan en la parte privada, y como bien, no es de impresionarse, ninguna persona que llego hasta el kínder trabaja en alguna entidad federal, sin embargo, sí hay personas con solamente en el kínder que se encuentran trabajado para

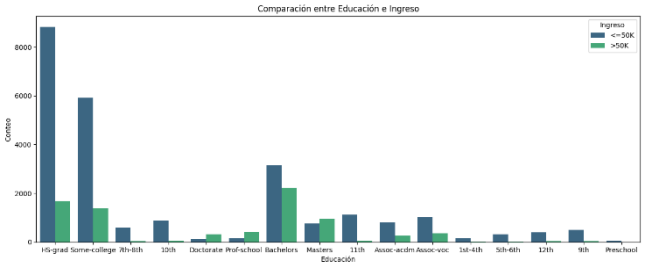
gobiernos locales. De igual forma, es notable que pocas personas con un grado de doctorado trabajan para instituciones del gobierno al contrario de las personas con un bachillerato universitario, que gran parte trabajan para entidades gubernamentales.

En este apartado podemos ver la comparación education e income. El análisis revela que el grupo con mayor frecuencia de individuos es el de aquellos que



Education – workclass Gráfico

han completado la educación secundaria (HS-grad). Dentro de este grupo, observamos que una proporción significativa de individuos gana más de $50,000 al año.



Education-income Gráfico

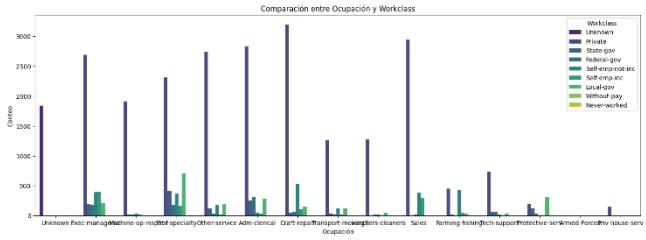
education	income <=50K	>50K
10th	871	62
11th	1115	60
12th	400	33
1st-4th	162	6
5th-6th	317	16
7th-8th	606	40
9th	487	27
Assoc-acdm	802	265
Assoc-voc	1021	361
Bachelors	3134	2221
Doctorate	107	306
HS-grad	8826	1675
Masters	764	959
Preschool	51	0
Prof-school	153	423
Some-college	5904	1387

Education-income

En esta comparación de datos vamos a ver occupation y workclass.

occupation	workclass Federal-gov	Local-gov	Never-worked	Private	Self-emp-inc	Self-emp-not-inc	state-gov	Unknown	Without-pay
?	0	0	7	0	0	0	0	1836	0
Adm-clerical	317	283	0	2833	31	50	253	0	3
Armed-Forces	9	0	0	0	0	0	0	0	0
Craft-repair	64	146	0	3195	106	531	56	0	1
Exec-managerial	180	214	0	2691	400	392	189	0	0
Farming-fishing	8	29	0	455	51	430	15	0	6
Handlers-cleaners	23	47	0	1273	2	15	9	0	1
Machine-op-inspct	14	12	0	1913	13	36	13	0	1
Other-service	35	193	0	2740	27	175	124	0	1
Priv-house-serv	0	0	0	149	0	0	0	0	0
Prof-specialty	175	705	0	2313	160	373	414	0	0
Protective-serv	28	304	0	190	5	6	116	0	0
Sales	14	7	0	2942	291	385	11	0	0
Tech-support	68	38	0	736	3	26	57	0	0
Transport-moving	25	115	0	1266	27	122	41	0	1

Occupation-workclass



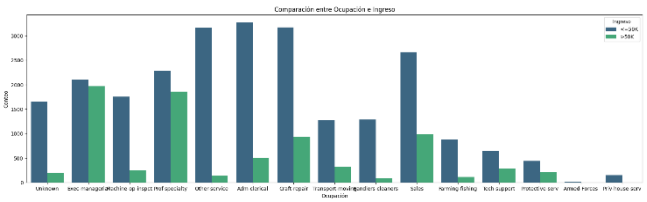
Occupation-Workclass Gráfico

Notamos que gran parte de los encuestados trabajan en ventas para el sector privado, otra gran parte de ellos trabaja en la parte manual, o en reparación de igual forma para el sector privado.

income	<=50K	>50K
occupation		
Adm-clerical	3263	507
Armed-Forces	8	1
Craft-repair	3170	929
Exec-managerial	2098	1968
Farming-fishing	879	115
Handlers-cleaners	1284	86
Machine-op-inspct	1752	250
Other-service	3158	137
Priv-house-serv	148	1
Prof-specialty	2281	1859
Protective-serv	438	211
Sales	2667	983
Tech-support	645	283
Transport-moving	1277	320
Unknown	1652	191

Income-occupation

Respecto a income y occupation vemos que mayor parte de las personas encuestadas reportan ingresos menores a $50,000 anuales. Hay ciertas cosas que resultan interesantes. Por ejemplo, entre los trabajan en las fuerzas armadas, solo una persona gana más de $50,000 al año. Esto me parece curioso debido a que es la única de los encuestados con estos ingresos, probablemente se debe a factores como el rango militar alcanzado o la cantidad de años de servicio. Comprender estos aspectos podría proporcionar una visión más completa de las dinámicas salariales dentro de la población.



Occupation-income Gráfico

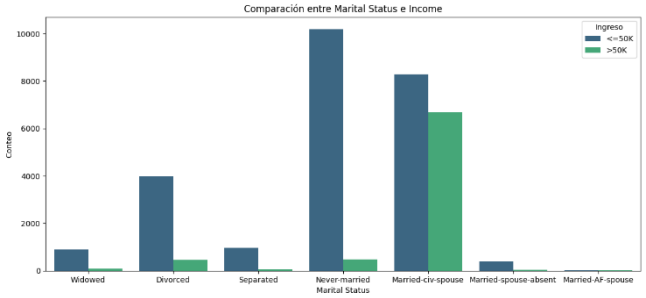
Es importante notar que una proporción considerable de las personas que se dedican a labores manuales o de reparación también reportan ingresos superiores a $50,000 anuales. Este hallazgo desafía la percepción común de que los trabajos manuales son generalmente menos remunerados. Es probable que la experiencia y especialización en ciertas áreas técnicas aumenten la capacidad de estos trabajadores para negociar salarios más altos.

Respecto a la comparación entre el estado civil y los ingresos, podemos observar que el mayor porcentaje de personas que ganan más de $50,000 al año están casadas y viven con su cónyuge, con un total de 6,692 individuos en esta categoría, la cifra más alta de todas.

income	<=50K	>50K
marital.status		
Divorced	3980	463
Married-AF-spouse	13	10
Married-civ-spouse	8284	6692
Married-spouse-absent	384	34
Never-married	10192	491
Separated	959	66
Widowed	908	85

Marital.status-income

Esto sugiere que las personas casadas con cónyuge presente pueden tener mayores ingresos, posiblemente debido a factores como la estabilidad financiera, el apoyo mutuo en el hogar, y la posible acumulación de bienes y recursos a lo largo del tiempo.



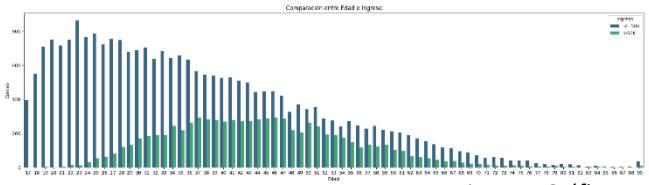
Marital status-income Gráfico

También se evidencia que la mayoría de los entrevistados que nunca se han casado ganan menos de $50,000 al año. Este grupo, predominantemente compuesto por individuos solteros, probablemente se encuentra en las primeras etapas de su vida o carrera profesional, lo cual podría explicar sus ingresos más bajos. La falta de experiencia laboral extensa y la posible continua inversión en educación y formación profesional pueden ser factores contribuyentes a esta situación.

income	<=50K	>50K
age		
17	395	0
18	550	0
19	710	2
20	753	0
21	717	3
...	...	...
85	3	0
86	1	0
87	1	0
88	3	0
90	35	8

Income-age

Este análisis nos lleva a una consideración adicional importante: la relación entre la edad y los ingresos. Al investigar cómo la edad influye en los ingresos, podríamos descubrir patrones que revelen cómo las diferentes etapas de la vida y del desarrollo profesional impactan la capacidad de generar ingresos.



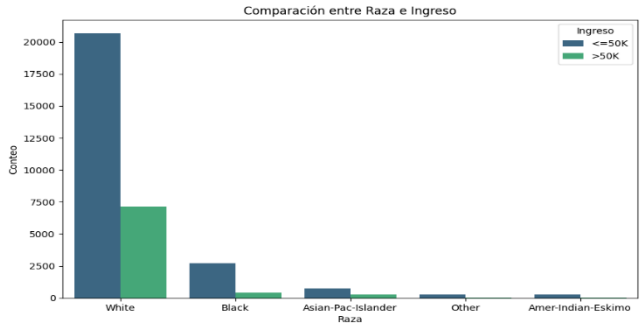
Age-income Gráfico

Como se muestra en la imagen, la mayor cantidad de personas jóvenes ganan menos de 50.000 al año. Entre las personas menores de 21 años se encuentra ganando más de 50k. Al contrario, podemos que las personas de 90 años entre ellos 8 ganan más de 50k.

income	<=50K	>50K
race		
Amer-Indian-Eskimo	275	36
Asian-Pac-Islander	763	276
Black	2737	387
Other	246	25
White	20699	7117

Income-race

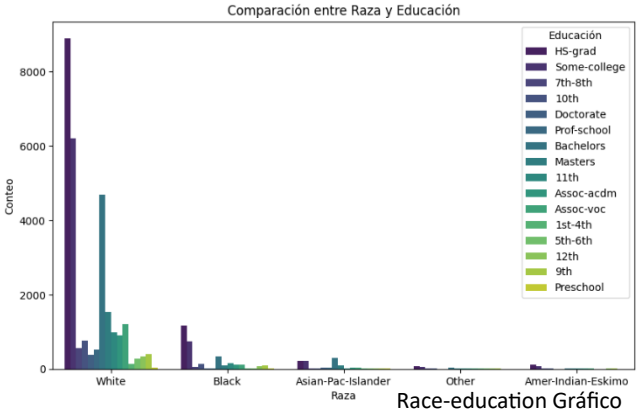
En este apartado podemos observar la comparación entre la raza y los ingresos, lo que revela que la mayoría de los encuestados son blancos, mientras que las minorías pertenecen a otras razas no especificadas. Es interesante notar que, dentro del grupo de individuos identificados como negros, hay



Income-race gráfico

una proporción significativa que tiene ingresos superiores a $50,000 al año.

Esta observación es notable porque desafía algunas expectativas y estereotipos comunes acerca de la distribución de ingresos entre diferentes grupos raciales. La presencia de un considerable número de personas negras con ingresos altos sugiere que hay factores complejos y diversos que influyen en el poder adquisitivo, incluyendo, pero no limitándose a, la educación, la ocupación, la experiencia laboral y las oportunidades de crecimiento profesional.



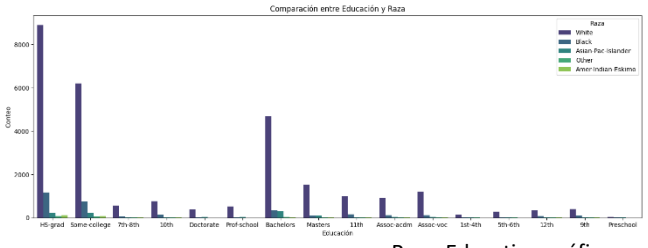
Race-education Gráfico

Es interesante observar que el grupo con la mayor cantidad de personas que solo alcanzaron el nivel de preescolar pertenece a la raza blanca. En contraste, las personas pertenecientes al grupo americano-indígena-eskimo muestran un nivel de escolaridad notablemente alto, con muchos individuos que han alcanzado grados avanzados como doctorados y maestrías. Esta disparidad en los niveles educativos entre los diferentes grupos raciales pone de manifiesto las variadas trayectorias educativas y socioeconómicas presentes en la población encuestada. La tendencia observada sugiere que, a pesar de las diferencias culturales y de origen, algunos grupos minoritarios han logrado alcanzar altos niveles de educación formal, lo cual podría estar relacionado con factores como el acceso a

race	Amer-Indian-Eskimo	Asian-Pac-Islander	Black	Other	White
education					
10th	16	13	133	9	762
11th	14	21	153	10	977
12th	5	9	70	14	335
1st-4th	4	5	16	9	134
5th-6th	2	18	21	13	279
7th-8th	9	11	56	17	553
9th	5	9	89	8	403
Assoc-acdm	8	29	107	8	915
Assoc-voc	19	38	112	6	1207
Bachelors	21	289	330	33	4682
Doctorate	3	28	11	2	369
HS-grad	119	226	1174	78	8904
Masters	5	88	86	7	1537
Preschool	0	6	5	2	38
Prof-school	2	41	15	4	514
Some-college	79	208	746	51	6207

Race-education

recursos educativos, el apoyo comunitario y las políticas de inclusión educativa.



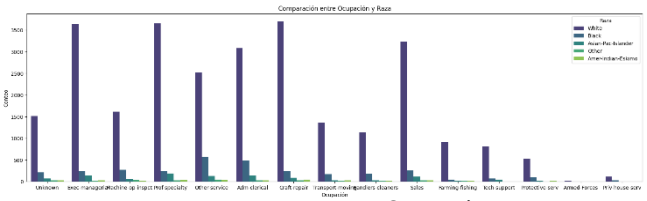
Race-Education gráfico

race	Amer-Indian-Eskimo	Asian-Pac-Islander	Black	Other	White
occupation					
Adm-clerical	31	139	490	26	3084
Armed-Forces	1	0	1	0	7
Craft-repair	44	89	244	28	3694
Exec-managerial	30	135	244	11	3646
Farming-fishing	10	16	42	11	915
Handlers-cleaners	22	23	179	12	1134
Machine-op-inspct	19	59	274	39	1611
Other-service	33	128	571	40	2523
Priv-house-serv	0	4	28	3	114
Prof-specialty	33	186	239	31	3651
Protective-serv	8	15	102	5	519
Sales	26	108	254	25	3237
Tech-support	4	44	71	3	806
Transport-moving	25	28	170	14	1360
Unknown	25	65	215	23	1515

Race-occupation

En esta imagen, se observa que en las fuerzas armadas predominan las personas de raza blanca,

acompañadas por una persona de raza negra y otra de origen amerindio-eskimo. Además, llama la atención la presencia de un número considerable de individuos cuyo oficio es desconocido. Este dato subraya la importancia de mejorar la precisión en la recopilación de información ocupacional para entender mejor las dinámicas y características del mercado laboral.



Occupation-Race
Gráfico

4. **Modelado de datos (consolidar datos resultantes).**

De igual forma, podemos comprobar que hay datos faltantes o que se muestran con el símbolo “?”, el cual podría traer problemas y debe ser procesado de diferentes formas según la columna que corresponda. La presencia de estos valores faltantes puede afectar significativamente la calidad del análisis, ya que pueden introducir sesgos y distorsionar los resultados. Por ello, es crucial abordar estos valores de manera adecuada.

Una vez identificados los datos dummies en nuestra base de datos, el siguiente paso consistirá en determinar en qué columnas específicas se encuentran estos valores. Este análisis nos permitirá comprender mejor cómo afectan o influencian nuestras variables de interés.

Este proceso implica examinar cuidadosamente cada columna para determinar dónde están presentes los datos dummies y cómo podrían estar distorsionando o afectando la integridad de nuestros resultados. Este tipo de análisis nos ayuda a tomar mejores decisiones sobre cómo manejar estos valores, ya sea eliminándolos, imputándolos con valores adecuados o considerando su impacto en el contexto del análisis que estamos llevando a cabo.

Para verificar estos datos, vamos a ejecutar el siguiente código:

```
def buscar_dummy_columns(data, dummy_value='?'):
    dummy_columns = []
    for col in data.columns:
        if data[data[col] == dummy_value].shape[0] > 0:
            dummy_columns.append(col)
    return dummy_columns

dummy_columns = buscar_dummy_columns(data)
print("Columnas con dummies:")
for col in dummy_columns:
    print(f"Columna '{col}' tiene dummies.")
```

```
Número de veces que '?' aparece en cada columna:
age                0
workclass       1836
fnlwgt             0
education          0
education.num      0
marital.status     0
occupation      1843
relationship       0
race               0
sex                0
capital.gain       0
capital.loss       0
hours.per.week     0
native.country   583
income             0
```
buscar_vacios

```
Columnas con dummies:
Columna 'workclass' tiene dummies.
Columna 'occupation' tiene dummies.
Columna 'native.country' tiene dummies.
```

Buscar dummies

Para mejorar la integridad y la consistencia de los datos, es necesario realizar la sustitución del símbolo "?" en las columnas que lo contengan. Al hacerlo, se evita que estos valores faltantes alteren los análisis posteriores. Dado que en este caso los valores corresponden a texto, se ha decidido sustituirlos por la palabra "Unknown". Esta sustitución no solo mantiene la estructura del conjunto de datos, sino que también facilita el tratamiento de estos valores durante el análisis.

Por ejemplo, en la columna occupation, el símbolo "?" aparece 1843 veces y en la columna native.country aparece 583 veces. Al reemplazar estos valores con "Unknown", se asegura que no se pierda información valiosa y que el análisis de las demás variables se mantenga consistente. Este proceso de limpieza de datos es un paso fundamental para preparar el conjunto de datos para cualquier tipo de análisis o modelado posterior, garantizando así resultados más precisos y fiables. Para poder verificar estos datos vamos a usar el código:

```
bool_df = data.applymap(lambda x: x == '?')

buscar_vacios = bool_df.sum()

print("\nNúmero de veces que '?' aparece en cada columna:")

print(buscar_vacios)
```

Una vez que hemos identificado cuántos datos dummies tenemos en nuestra base de datos, es importante proceder con una consulta detallada para entender qué tipo de problemas pueden estar generando estos valores en nuestro análisis. Esta fase implica revisar específicamente cuáles columnas y filas están afectadas por los datos dummies, y evaluar cómo podrían estar distorsionando los resultados o introduciendo sesgos en nuestras conclusiones. Esta evaluación nos permitirá determinar si es necesario limpiar los datos eliminando o imputando estos valores, o si podemos gestionarlos de manera que no comprometan la precisión y validez de nuestro análisis. Para poder ejecutar este cambio y sustituir los valores, vamos a hacerlo con este código:

```
def show_dummy_values(data, column, dummy_value='?'):
    dummy_rows = data[data[column] == dummy_value]
    dummy_count = dummy_rows.shape[0]
    if dummy_count > 0:
        print(f"\nColumna '{column}' tiene {dummy_count} valor(es) dummy '{dummy_value}'.")
        print(dummy_rows)
    else:
        print(f"\nNo se encontraron valores dummy '{dummy_value}' en la columna '{column}'.")

for col in data.columns:
```

show_dummy_values(data, col)

```
Columna 'workclass' tiene 1836 valor(es) dummy '?'.
       age workclass  fnlwgt     education  education.num      marital.status  \
0       90         ?   77053       HS-grad              9             Widowed
2       66         ?  186061  Some-college             10             Widowed
14      51         ?  172175     Doctorate             16       Never-married
24      61         ?  135285       HS-grad              9  Married-civ-spouse
44      71         ?  100820       HS-grad              9  Married-civ-spouse
...    ...       ...     ...           ...            ...                 ...
32533   35         ?  320084     Bachelors             13  Married-civ-spouse
32534   30         ?   33811     Bachelors             13       Never-married
32541   71         ?  287372     Doctorate             16  Married-civ-spouse
32543   41         ?  202822       HS-grad              9           Separated
32544   72         ?  129912       HS-grad              9  Married-civ-spouse
```

Dummy Workclass

En la columna '***workclass***', se identificaron 1836 valores que corresponden al valor dummy. Estos valores se encuentran distribuidos desde la primera fila hasta la fila 32544 del conjunto de datos. Es importante conocer estos valores dummy de manera adecuada para garantizar que nuestro análisis sea confiable.

```
Columna 'occupation' tiene 1843 valor(es) dummy '?'.
       age workclass  fnlwgt     education  education.num      marital.status
0       90         ?   77053       HS-grad              9             Widowed
2       66         ?  186061  Some-college             10             Widowed
14      51         ?  172175     Doctorate             16       Never-married
24      61         ?  135285       HS-grad              9  Married-civ-spouse
44      71         ?  100820       HS-grad              9  Married-civ-spouse
...    ...       ...     ...           ...            ...                 ...
32533   35         ?  320084     Bachelors             13  Married-civ-spouse
32534   30         ?   33811     Bachelors             13       Never-married
32541   71         ?  287372     Doctorate             16  Married-civ-spouse
32543   41         ?  202822       HS-grad              9           Separated
32544   72         ?  129912       HS-grad              9  Married-civ-spouse
```

Dummy Occupation

En relación con la columna " ", se identificaron 1843 filas que contienen valores dummies. Estos valores representan casos donde la ocupación de la persona encuestada no fue registrada o no se proporcionó la información correspondiente. La presencia de estos datos dummies puede afectar la fiabilidad de cualquier análisis realizado con respecto a la ocupación de los encuestados. Es esencial abordar y corregir estos datos para asegurar que los resultados derivados de este conjunto de datos sean precisos y confiables, evitando así posibles sesgos o interpretaciones erróneas en el análisis final.

```
Columna 'native.country' tiene 583 valor(es) dummy '?'.
       age         workclass  fnlwgt     education  education.num  \
9       41           Private   70037  Some-college             10
18      22           Private  119592    Assoc-acdm             12
65      60     Self-emp-inc  226355     Assoc-voc             11
86      39  Self-emp-not-inc  218490   Prof-school             15
87      43      Federal-gov  156996   Prof-school             15
...    ...               ...     ...           ...            ...
32459   44     Self-emp-inc   71556       Masters             14
32476   58     Self-emp-inc  181974     Doctorate             16
32498   42  Self-emp-not-inc  217597       HS-grad              9
32515   39           Private  107302       HS-grad              9
32528   81                 ?  120478     Assoc-voc             11
```

Dummy Native.country

En la columna "**native.country**", se detectaron 583 datos tipo dummy, lo cual representa una proporción significativa dentro del conjunto de datos. Estos valores podrían indicar que hubo problemas al recopilar o registrar la información sobre el país de origen de los encuestados. La presencia de estos datos dummies puede afectar la calidad y la precisión de cualquier análisis que busque relacionar características demográficas con el país de origen. Además, la falta de información en esta variable podría introducir sesgos o limitaciones en la interpretación de los resultados.

```
      native.country income
9                  ?   >50K
18                 ?   >50K
65                 ?   >50K
86                 ?   >50K
87                 ?   >50K
...              ...    ...
32459              ?   >50K
32476              ?  <=50K
32498              ?  <=50K
32515              ?   >50K
32528              ?  <=50K
```

Dummy native.country

Además, podemos confirmar que la columna "relationship" proporciona información sobre el estado de las relaciones de los encuestados, como si están casados, solteros, entre otros. En esta columna no presentamos datos dummies, por lo tanto, nos

asegura que podemos confiar en las interpretaciones obtenidas.

En la columna "race", la integridad de los datos nos permite hacer análisis demográficos sin preocupaciones sobre la precisión de la información racial de los encuestados. Esto es crucial para estudios que exploran desigualdades o disparidades raciales en diversos aspectos socioeconómicos.

La columna "sex", también nos muestra que está de datos dummies, nos permite llevar a cabo análisis de género con plena confianza en la exactitud de los datos. Esto es especialmente importante para investigaciones centradas en la igualdad de género y las diferencias de ingresos entre hombres y mujeres.

Las columnas "capital.gain" y "capital.loss" registran las ganancias y pérdidas de capital de los encuestados, respectivamente. La limpieza de estos datos asegura que cualquier análisis financiero que realicemos será preciso y no estará sesgado por valores incorrectos o faltantes.

Finalmente, la columna "hours.per.week" nos proporciona datos sobre el número de horas trabajadas por semana, una variable esencial para estudios sobre la carga laboral y su impacto en los ingresos y la calidad de vida. La ausencia de datos dummies en esta columna garantiza que podemos evaluar de manera precisa las tendencias laborales y sus correlaciones con otras variables socioeconómicas.

Vamos a verificar el porcentaje que abarca cada columna respecto a los valores dummies. Esta verificación nos permitirá identificar la proporción de datos faltantes o no válidos en cada columna de nuestro conjunto de datos. Es crucial realizar esta tarea, ya que la presencia de valores dummies puede afectar la calidad y precisión de nuestro análisis.

Al calcular estos porcentajes, podremos determinar qué columnas están más afectadas por la falta de datos y priorizar su limpieza o tratamiento. Por ejemplo, si encontramos que una columna tiene un porcentaje muy alto de valores dummies, podríamos decidir eliminarla del análisis o buscar formas de imputar estos valores de manera efectiva.

Este análisis nos dará una idea clara de la integridad de nuestros datos y nos ayudará a tomar decisiones informadas sobre cómo manejar los valores dummies para mantener la confiabilidad y validez de nuestros resultados. Para confirmar esto, vamos a usar el código:

```
valor_dummy = '?'

valores_dummy = (data == valor_dummy).sum()

valores_dummy = pd.DataFrame(valores_dummy, columns=['count'])

sum_total = len(data)

valores_dummy['percentage'] = (valores_dummy['count'] / sum_total) * 100

valores_dummy['percentage'] = round(valores_dummy['percentage'], 3)

valores_dummy = valores_dummy.sort_values('percentage', ascending=False)

print(valores_dummy)
```

	count	percentage
occupation	1843	5.660
workclass	1836	5.639
native.country	583	1.790
age	0	0.000
fnlwgt	0	0.000
education	0	0.000
education.num	0	0.000
marital.status	0	0.000
relationship	0	0.000
race	0	0.000
sex	0	0.000
capital.gain	0	0.000
capital.loss	0	0.000
hours.per.week	0	0.000
income	0	0.000

Porcentaje dummy

Como podemos observar, la columna “occupation” contiene 1843 datos dummies, lo que la convierte en la columna con la mayor cantidad de valores dummies en nuestro conjunto de datos. Estos 1843 valores representan un porcentaje total de 5,660%. La alta proporción de valores faltantes en esta columna sugiere que la información sobre la ocupación de los encuestados no siempre fue proporcionada adecuadamente. Por lo que nos podría afectar significativamente en nuestros análisis, especialmente si la ocupación es una variable clave en nuestro estudio.

Mientras tanto, la columna “workclass” presenta un total de 1836 datos dummies, lo que corresponde a un porcentaje del 5,639%. Aunque la cantidad de valores dummies en esta columna es ligeramente menor que en la columna “occupation”, sigue siendo importante. La clase laboral de los encuestados es una variable importante que puede influir en muchos aspectos de nuestro análisis, y el hecho de que una proporción considerable de estos datos esté faltante requiere que prestemos especial atención a cómo manejamos estos valores durante el proceso de limpieza de datos.

Por último, la columna “native.country” tiene un total de 583 datos dummies, lo que representa un porcentaje de 1,790%. Aunque el número de valores faltantes en esta columna es significativamente menor en comparación con las columnas de “occupation” y “workclass”, todavía es notable. La información sobre el país de origen de los encuestados puede ser relevante en ciertos análisis.

También es posible observar que para las demás columnas el valor de datos dummies corresponde a 0%, lo que significa que no hay valores faltantes en esas columnas que puedan afectar nuestro análisis. La ausencia de datos dummies en estas columnas asegura que la información contenida en ellas es completa y confiable, permitiéndonos realizar análisis precisos sin la necesidad de un tratamiento adicional de valores faltantes.

De igual manera, podemos visualizar esta información gráficamente para obtener una comprensión más clara y rápida de cómo se distribuyen los valores dummies en nuestro conjunto de datos. En el siguiente gráfico de barras, podemos observar cómo se distribuyen los valores dummies en las columnas occupation, workclass y native.country.

Este gráfico muestra la cantidad de valores dummies en cada columna, permitiéndonos ver cuál de ellas tiene la mayor cantidad de datos que necesitan ser tratados para mejorar la calidad del análisis. Por ejemplo, la columna occupation tiene el mayor porcentaje de valores dummies, seguida de cerca por workclass, mientras que native.country tiene una cantidad significativamente menor de valores

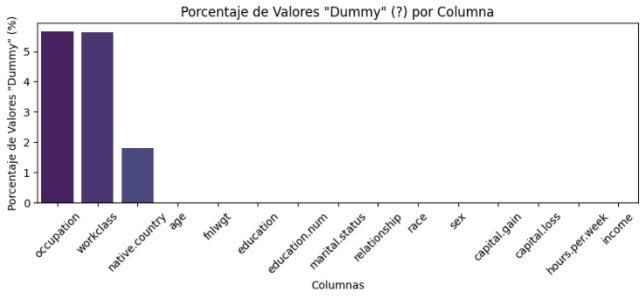


Gráfico Porcentajes Dummies

dummies. Esta información es crucial para priorizar los esfuerzos de limpieza y preparación de datos.

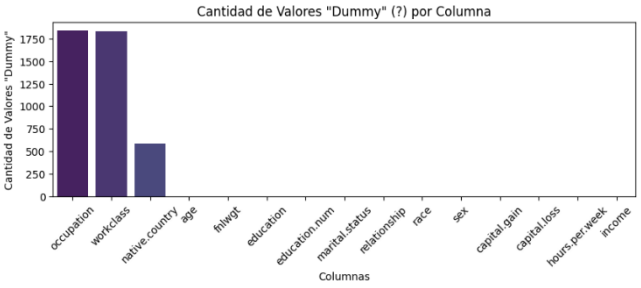


Gráfico Cantidad Dummies #1

Este gráfico lo obtuvimos mediante el siguiente código:

plt.figure(figsize=(10, 3))

sns.barplot(x=valores_dummy.index, y=valores_dummy['count'], palette='viridis')

plt.title('Cantidad de Valores "Dummy" (?) por Columna')

plt.xlabel('Columnas')

plt.ylabel('Cantidad de Valores "Dummy"')

plt.xticks(rotation=45)

plt.show()

El siguiente gráfico presenta los porcentajes de valores dummies en las columnas occupation, workclass y native.country. En este gráfico, podemos observar que la columna occupation tiene el mayor porcentaje de valores dummies, seguida de cerca por workclass, mientras que native.country tiene una cantidad significativamente menor de valores dummies. Esta información es fundamental para determinar dónde concentrar nuestras acciones correctivas.

Para obtener este gráfico, realizamos el siguiente código:

plt.figure(figsize=(10, 3))

sns.barplot(x=valores_dummy.index, y=valores_dummy['percentage'], palette='viridis')

plt.title('Porcentaje de Valores "Dummy" (?) por Columna')

plt.xlabel('Columnas')

plt.ylabel('Porcentaje de Valores "Dummy" (%)')

plt.xticks(rotation=45)

plt.show()

Una vez confirmados los datos dummies, conocemos su porcentaje y también ya sabemos que nos pueden afectar para continuar con nuestro análisis. Es crucial abordar estos valores dummies de manera adecuada para garantizar la precisión y confiabilidad de nuestros resultados. Los valores dummies pueden

sesgar nuestros análisis y llevar a conclusiones incorrectas, por lo que es necesario tratarlos antes de proceder con cualquier análisis avanzado.

La presencia de datos dummies en las columnas occupation, workclass y native.country nos ha revelado que hay una cantidad significativa de datos faltantes o no válidos que podrían afectar nuestros resultados.

Para evitar estos problemas y mantener la integridad de nuestro análisis, es necesario reemplazar estos valores dummies con un valor significativo que no afecte nuestros resultados. En este caso, hemos decidido reemplazar los valores dummies con la palabra "unknown". Esto permitirá mantener la coherencia en el conjunto de datos y evitará cualquier interferencia que estos valores podrían causar en el análisis posterior.

A través del siguiente código, procederemos a realizar el reemplazo de los valores dummies por la palabra "unknown", asegurando así que nuestros datos estén en una condición óptima para un análisis preciso y fiable:

data['workclass']= data['workclass'].replace('?', 'Unknown')

data['native.country']=data['native.country'].replace('?', 'Unknown')

	age	workclass	fnlwgt	education	education.num	marital.status	occupation	relationship	race	sex	capital.gain	capital.loss	hours.per.week	native.country	income
0	90	Unknown	77053	HS-grad	9	Widowed	?	Not-in-family	White	Female	0	4356	40	United-States	<=50K
1	82	Private	132870	HS-grad	9	Widowed	Exec-managerial	Not-in-family	White	Female	0	4356	18	United-States	<=50K
2	66	Unknown	186061	Some-college	10	Widowed	?	Unmarried	Black	Female	0	4356	40	United-States	<=50K
3	54	Private	140359	7th-8th	4	Divorced	Machine-op-inspct	Unmarried	White	Female	0	3900	40	United-States	<=50K
4	41	Private	264663	Some-college	10	Separated	Prof-specialty	Own-child	White	Female	0	3900	40	United-States	<=50K

Replace ?

Una vez realizada esta modificación podemos confirmar que no haya valores nulos en nuestra base de datos por medio del comando:

data.isnull().sum()

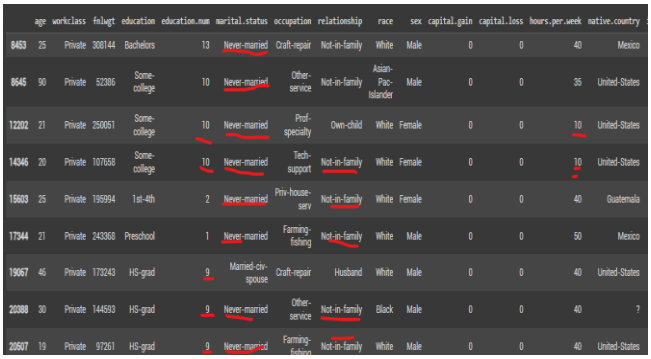
```
age               0
workclass         0
fnlwgt            0
education         0
education.num     0
marital.status    0
occupation        0
relationship      0
race              0
sex               0
capital.gain      0
capital.loss      0
hours.per.week    0
native.country    0
income            0
dtype: int64
```

data.isnull().sum()

Vamos a confirmar a ver si hay valores duplicados por medio del comando: data.duplicated().sum() esto nos muestra que hay 24 valores duplicados.

El resultado nos muestra que hay 24 valores que se repiten. Sin embargo, vamos a realizar un análisis más profundo para confirmar los datos duplicados.

Mediante el comando: data.duplicated() nos muestra estos valores.

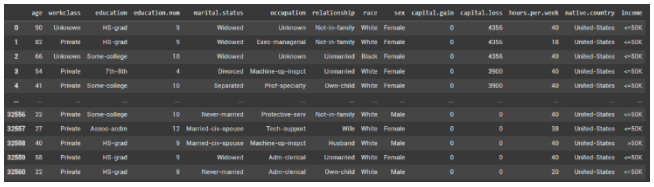


	age	workclass	fnlwgt	education	education.num	marital.status	occupation	relationship	race	sex	capital.gain	capital.loss	hours.per.week	native.country
8453	25	Private	308144	Bachelors	13	Never-married	Craft-repair	Not-in-family	White	Male	0	0	40	Mexico
8645	90	Private	52386	Some-college	10	Never-married	Other-service	Not-in-family	Asian-Pac-Islander	Male	0	0	35	United-States
12202	21	Private	250051	Some-college	10	Never-married	Prof-specialty	Own-child	White	Female	0	0	10	United-States
14346	20	Private	107658	Some-college	10	Never-married	Tech-support	Not-in-family	White	Female	0	0	10	United-States
15603	25	Private	195994	1st-4th	2	Never-married	Priv-house-serv	Not-in-family	White	Female	0	0	40	Guatemala
17344	21	Private	243368	Preschool	1	Never-married	Farming-fishing	Not-in-family	White	Male	0	0	50	Mexico
19067	46	Private	173243	HS-grad	9	Married-civ-spouse	Craft-repair	Husband	White	Male	0	0	40	United-States
20388	30	Private	144593	HS-grad	9	Never-married	Other-service	Not-in-family	Black	Male	0	0	40	?
20507	19	Private	97261	HS-grad	9	Never-married	Farming-fishing	Not-in-family	White	Male	0	0	40	United-States

data[data.duplicated()]

Más a fondo podemos ver confirmar que los valores que se repiten no necesariamente se deben a que sean todos completamente iguales, al contrario, pertenece a una persona diferente.

De igual manera vamos a eliminar la columna fnlwgt, ya que no nos está mostrando un dato muy importante e incluso puede ser confuso para interpretarlo. Para eliminarlo usaremos el código *data.drop(columns=['fnlwgt'])*



	age	workclass	education	education.num	marital.status	occupation	relationship	race	sex	capital.gain	capital.loss	hours.per.week	native.country	income
0	90	Unknown	HS-grad	9	Widowed	Unknown	Not-in-family	White	Female	0	4356	40	United-States	<=50K
1	82	Private	HS-grad	9	Widowed	Exec-managerial	Not-in-family	White	Female	0	4356	18	United-States	<=50K
2	66	Unknown	Some-college	10	Widowed	Unknown	Unmarried	Black	Female	0	4356	40	United-States	<=50K
3	54	Private	7th-8th	4	Divorced	Machine-op-inspct	Unmarried	White	Female	0	3900	40	United-States	<=50K
4	41	Private	Some-college	10	Separated	Prof-specialty	Own-child	White	Female	0	3900	40	United-States	<=50K
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
32556	22	Private	Some-college	10	Never-married	Protective-serv	Not-in-family	White	Male	0	0	40	United-States	<=50K
32557	27	Private	Assoc-acdm	12	Married-civ-spouse	Tech-support	Wife	White	Female	0	0	38	United-States	<=50K
32558	40	Private	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband	White	Male	0	0	40	United-States	>50K
32559	58	Private	HS-grad	9	Widowed	Adm-clerical	Unmarried	White	Female	0	0	40	United-States	<=50K
32560	22	Private	HS-grad	9	Never-married	Adm-clerical	Own-child	White	Male	0	0	20	United-States	<=50K

data.drop(columns=['fnlwgt'])

II. Aplicar técnicas de Data Mining en su tema de proyecto:

1. ***Minería de reglas de asociación:*** La minería de reglas de asociación es una técnica utilizada para descubrir patrones frecuentes, asociaciones y relaciones entre variables en grandes conjuntos de datos. Esta técnica es especialmente útil para identificar conjuntos de elementos que suelen ocurrir juntos, como productos comprados en conjunto en un supermercado o síntomas que se presentan simultáneamente en pacientes con ciertas enfermedades. En el contexto de nuestro análisis, la minería de reglas de asociación puede aplicarse para explorar y descubrir relaciones interesantes entre diferentes variables del conjunto de datos. Por ejemplo, se pueden identificar asociaciones entre el nivel educativo y la clase trabajadora, la educación y los ingresos, la ocupación y los ingresos, así como entre la ocupación y la clase trabajadora. Además, se pueden encontrar patrones entre el estado civil y los ingresos, los ingresos y la edad, así como entre la raza y el nivel educativo, la raza y los ingresos, y la raza y la clase trabajadora. Estas asociaciones pueden proporcionar valiosas ideas sobre cómo diferentes factores demográficos y socioeconómicos interactúan y afectan los resultados económicos, permitiendo una comprensión más profunda de las dinámicas presentes en los datos y apoyando la toma de decisiones basada en datos.
2. ***Clasificación.:*** La clasificación es una técnica de aprendizaje supervisado que se utiliza para asignar una categoría o etiqueta a nuevos datos basándose en la información proporcionada por un conjunto de datos de entrenamiento. Es especialmente útil cuando se cuenta con un conjunto de datos que incluye variables predictoras y una variable de respuesta categórica. En este contexto, la clasificación puede emplearse para predecir la clase de ingresos, determinando si una persona gana más o menos de $50,000 al año. Esto se logra analizando diversas características como la educación, la ocupación, la edad, el estado civil, entre otras. Al entrenar un modelo de clasificación con estos datos, es posible identificar patrones y relaciones significativas que permiten hacer

predicciones precisas sobre la categoría de ingresos de nuevos individuos. Este enfoque no solo facilita la segmentación de la población en diferentes grupos de ingresos, sino que también proporciona una herramienta valiosa para la toma de decisiones estratégicas y la implementación de políticas socioeconómicas efectivas.

3. ***Agrupación en clústeres***: La agrupación en clústeres es una técnica de aprendizaje no supervisado que se utiliza para dividir un conjunto de datos en grupos o clústeres basados en la similitud de las observaciones. Es útil cuando no se tienen etiquetas predefinidas en los datos y se desea explorar la estructura subyacente de los datos. En tu trabajo, podrías aplicar agrupación en clústeres para identificar grupos de personas con características similares, por ejemplo, podrías agrupar a las personas en función de su edad, nivel educativo y ocupación para identificar patrones demográficos.

   K-Means: Agrupar las observaciones en K clústeres donde K es un número predefinido. Esto podría ayudarte a identificar grupos de individuos con características similares.

   Jerárquico: Crear una jerarquía de clústeres utilizando el enfoque aglomerativo o divisivo, lo que puede ser útil para entender las relaciones entre diferentes grupos.

4. **Análisis de secuencias y trayectorias:** El análisis de secuencias y trayectorias es una técnica que se utiliza para analizar y descubrir patrones en secuencias de eventos o acciones a lo largo del tiempo. Es útil cuando se tiene un conjunto de datos que registra la secuencia de eventos o acciones realizadas por individuos en un período de tiempo determinado. En tu trabajo, podrías aplicar análisis de secuencias y trayectorias para estudiar cómo cambian las características de las personas (como la educación, la ocupación, etc.) a lo largo del tiempo, o para identificar patrones de comportamiento en función de las interacciones con diferentes variables.

III. Aplicar Data Mining según la empresa elegida (dataset de su proyecto) y según el tipo de minería (mining):

1. ***Minería predictiva:*** La minería predictiva se utiliza para predecir eventos futuros o tendencias basadas en datos históricos. En el contexto de este proyecto, vamos a usar minería predictiva para predecir el comportamiento del mercado, la demanda del producto, tipo de persona y educación. Por ejemplo, se podría utilizar técnicas de aprendizaje automático para predecir la demanda de ciertos productos en función de variables como el precio, la temporada del año y las tendencias del mercado, lo que ayudaría a la empresa a planificar su inventario y sus estrategias de marketing de manera más eficaz.

IV. **Realizar u obtener una estadística descriptiva o sumario del dataset del proyecto**:

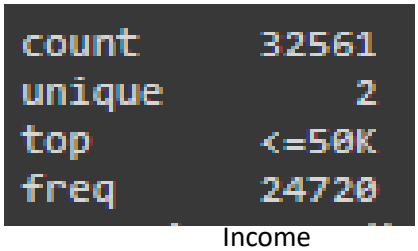
1. **Realizar un análisis de cada técnica de estadística descriptiva (count, mean, std, etc.) que ofrezca o aporte un valor a la toma de decisiones de ejecutivos, alta dirección, gerente, etc.**

Dado que no todas las columnas contienen datos que puedan ser fácilmente interpretados mediante valores descriptivos, vamos a enfocarnos en aquellas que sí permiten una interpretación clara. La información que podemos obtener de ellas va a ser más valiosa y nos va a ayudar para facilitar la toma de decisiones importantes. De estas columnas,

podemos extraer patrones y tendencias que ayudan a comprender mejor la distribución de características clave dentro del conjunto de datos. Esto nos permitirá identificar correlaciones significativas y anomalías, así como obtener detalles sobre los factores que pueden influir en los ingresos y otras variables de interés. Con datos más claros vamos a poder llevar a cabo mejores estrategias.

Las columnas que vamos a aplicarle el método describe() van a ser: income, age, education, workclass, marital.status, race, relationship.

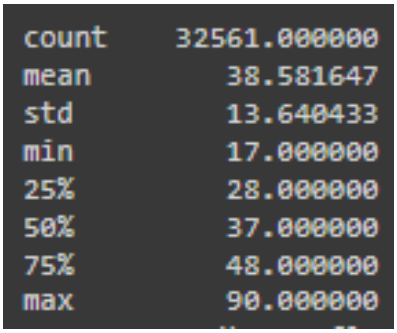
Para iniciar vamos a verificar income:



Income

Solo tenemos dos posibles categorías de ingresos con la información de ingresos: los que ganan más de 50k y aquellos que ganan menos de 50k. Sin embargo, se puede notar claramente que la mayoría de los encuestados tienen ingresos por debajo del promedio. Esto conlleva a que solo el 24.2% de los encuestados tienen ingresos mayores a 50k, lo cual representa un total de 7841 personas. Este dato es importante tenerlo en cuenta, ya que muestra los niveles de ingresos y sugiere que la mayoría de la población encuestada podría enfrentar limitaciones económicas.

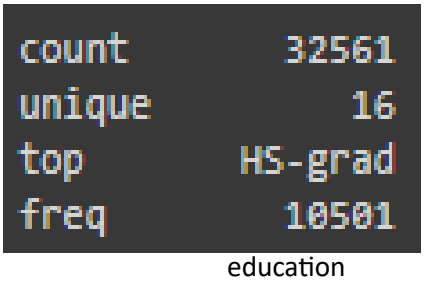
Ahora vamos a analizar age:



age

Se observa que la persona más joven entrevistada tiene 17 años, mientras que la edad promedio de los encuestados es de aproximadamente 38 años. La persona de mayor edad tiene 90 años. Es evidente que la mayoría de las personas se encuentran en un rango de edad entre los 28 y los 48 años. Esto indica que la población encuestada es en gran medida madura y establecida, probablemente con una carrera profesional ya desarrollada. Este rango de edad sugiere que la mayoría de los encuestados han acumulado una experiencia considerable en sus respectivos campos laborales, lo cual podría influir en sus ingresos y estabilidad económica.

Ahora vamos a realizar el análisis de la educación:



education

Es importante observar que una gran cantidad de las personas entrevistadas poseen un nivel académico elevado. Como se ha señalado anteriormente, los grados académicos varían considerablemente, abarcando desde la educación preescolar hasta títulos de doctorado. Sin embargo, la mayoría de los

encuestados ha completado al menos la educación secundaria. Este hecho destaca que la población en estudio es, en su mayoría, bien educada. La presencia de altos niveles de escolaridad entre los encuestados sugiere que estamos tratando con un grupo que, en general, ha tenido acceso a oportunidades educativas significativas. Esta educación puede tener un impacto positivo en sus perspectivas laborales y en su capacidad para generar ingresos, lo cual es crucial para interpretar y contextualizar los resultados del análisis de datos. La diversidad en los niveles educativos también ofrece una rica base para explorar cómo diferentes grados de formación académica pueden influir en diversas variables socioeconómicas y de ingresos.

Vamos a analizar workclass:

count	32561
unique	9
top	Private
freq	22696

workclass

Como hemos observado, una gran parte de los encuestados tiene un nivel educativo intermedio y se encuentra en una etapa madura de su vida. Además, la mayoría de estos individuos están empleados en el sector privado. Esta combinación de educación y experiencia sugiere que muchos de los encuestados tienen una formación suficiente para desempeñarse en roles que requieren cierta cualificación, y su participación en el sector privado indica que están integrados en un entorno laboral dinámico y competitivo.

Vamos a analizar marital status:

count	32561
unique	7
top	Married-civ-spouse
freq	14976

Marital status

Al igual que en la información que hemos venido revisando, podemos observar que la mayoría de los encuestados está casada por civil y vive con su familia o esposa. Este es un punto importante por considerar, ya que influye significativamente en la distribución de los ingresos disponibles. A pesar de que estas personas puedan tener mayores ingresos anuales en comparación con alguien más joven, como una persona de 17 años, el dinero disponible para gastos personales podría ser menor. Esto se debe a que tener una familia conlleva responsabilidades adicionales, como alimentar y mantener a otros miembros del hogar. Los ingresos se deben repartir entre las necesidades de todos los integrantes de la familia, lo cual incluye gastos en alimentación, vivienda, educación, y otros costos asociados con el cuidado y bienestar de los dependientes. Por lo tanto, el impacto económico de los ingresos debe ser analizado en el contexto de las responsabilidades familiares, ya que estas pueden reducir la cantidad de dinero disponible para el gasto individual y afectar la calidad de vida y las decisiones financieras de los encuestados.

Race:

count	32561
unique	5
top	White
freq	27816

race

En relación con la raza, podemos ver que el grupo predominante es la blanca con un poco más de la mitad. Sin embargo, una gran parte pertenece a la gente de raza negra.

Relationship:

count	32561
unique	6
top	Husband
freq	13193

relationship

En este análisis podemos observar que la mayoría de los entrevistados son hombres y están casados, lo que sugiere que muchos de ellos son jefes de hogar. Esta situación implica que probablemente la responsabilidad financiera principal recae sobre sus hombros. Al ser los principales proveedores, estos individuos suelen tener una mayor carga económica, encargándose de cubrir los gastos esenciales del hogar, como la alimentación, la vivienda, la educación de los hijos y otros costos asociados al bienestar familiar. Este rol de proveedor no solo influye en la distribución de sus ingresos, sino también en sus decisiones laborales y financieras, ya que deben asegurar la estabilidad y el sustento de su familia.

Hours.per.week:

count	32561.000000
mean	40.437456
std	12.347429
min	1.000000
25%	40.000000
50%	40.000000
75%	45.000000
max	99.000000

Hour per week

En relación con las horas trabajadas semanalmente, podemos observar que una significativa proporción de la población encuestada tiene empleos que requieren un compromiso de 40 horas por semana. Esto sugiere que muchas personas están empleadas en trabajos de tiempo completo que siguen la norma estándar de horas laborales en muchos sectores. Sin embargo, también es notable que existe un grupo de personas que reporta trabajar hasta 99 horas a la semana, lo cual es un número excepcionalmente alto y sugiere una carga de trabajo extremadamente pesada. Este nivel de compromiso laboral puede tener diversas implicaciones, tanto positivas como negativas. Por un lado, podría indicar una dedicación intensa y una necesidad de maximizar ingresos, posiblemente debido a altas responsabilidades financieras o la naturaleza de ciertos trabajos que demandan más tiempo. Por otro lado, trabajar tantas horas podría tener efectos adversos sobre la salud, el bienestar y el equilibrio entre la vida laboral y personal de estos individuos.

2. Ofrecer tres conclusiones y dos recomendaciones técnicas del análisis obtenido.

a) La base de datos revela una diversidad significativa en términos de edad, educación, ocupación y raza. La mayoría de los encuestados se encuentran en un rango de edad madura, tienen un nivel educativo intermedio y están empleados en el sector privado. Esta diversidad proporciona una visión amplia y rica de las diferentes realidades socioeconómicas de la población.

b) La mayoría de los encuestados gana menos de 50,000 dólares al año, con solo una minoría superando este ingreso. Esta disparidad de ingresos se manifiesta en diferentes grupos demográficos, con factores como el estado civil, la raza y la ocupación desempeñando un papel significativo en la distribución de los ingresos. Por ejemplo, la mayoría de los casados ganan más de 50,000 dólares, mientras que los solteros tienden a ganar menos.

c) La cantidad de horas trabajadas semanalmente varía considerablemente, con muchos encuestados trabajando 40 horas por semana, pero algunos alcanzan hasta 99 horas. Además, los datos sugieren que la responsabilidad financiera principal recae sobre los hombres casados, lo que podría influir en sus decisiones laborales y financieras, así como en su bienestar general.

Recomendaciones técnicas:

a) ***Aplicación de Clustering para Segmentación de Grupos:*** La aplicación de técnicas de clustering para la segmentación de grupos es un enfoque poderoso para analizar y comprender mejor los datos. En particular, técnicas como K-means o DBSCAN son herramientas efectivas para agrupar encuestados basándose en características similares. Este proceso de agrupación permite identificar patrones ocultos en los datos que no son evidentes a simple vista.

Al utilizar K-means, se asignan los datos a un número predefinido de clusters, donde cada dato pertenece al cluster con el centroide más cercano. Este método es especialmente útil cuando se tiene una idea aproximada de cuántos grupos diferentes pueden existir en los datos. Por ejemplo, se podría utilizar K-means para segmentar a los encuestados en grupos basados en características como la edad, el nivel educativo, los ingresos y la ocupación. Estos grupos pueden revelar diferencias significativas en el comportamiento económico y social, lo que permite una comprensión más profunda de la población.

Por otro lado, DBSCAN (Density-Based Spatial Clustering of Applications with Noise) es una técnica de clustering que no requiere especificar el número de clusters de antemano. En su lugar, DBSCAN identifica clusters basándose en la densidad de puntos en el espacio de características. Esto es particularmente útil para identificar grupos de formas arbitrarias y para detectar puntos de

ruido que no pertenecen a ningún cluster. DBSCAN puede ser utilizado para descubrir subgrupos de encuestados que comparten características inusuales o que representan casos atípicos, proporcionando así una visión más detallada y matizada de la población.

La segmentación de grupos mediante clustering tiene múltiples aplicaciones prácticas. En el contexto de políticas públicas y estrategias económicas, puede ayudar a los responsables de la toma de decisiones a entender mejor las necesidades y comportamientos de diferentes segmentos de la población. Por ejemplo, al identificar grupos de personas con niveles educativos similares pero con diferentes niveles de ingresos, se pueden diseñar programas de capacitación y educación específicos que atiendan a las necesidades de aquellos que están en desventaja económica.

Además, la segmentación puede ser útil para las empresas que desean adaptar sus productos y servicios a diferentes segmentos del mercado. Al comprender las características demográficas y socioeconómicas de sus clientes, las empresas pueden personalizar sus ofertas y estrategias de marketing para satisfacer mejor las necesidades de cada grupo. Por ejemplo, un banco podría utilizar clustering para identificar grupos de clientes que podrían beneficiarse de diferentes tipos de productos financieros, como cuentas de ahorro, préstamos personales o inversiones.

b) ***Reglas de Asociación:*** Implementar técnicas de minería de reglas de asociación es una estrategia poderosa para descubrir relaciones significativas entre diferentes variables dentro de un conjunto de datos amplio y complejo. Esta técnica permite identificar patrones y asociaciones que no son evidentes a simple vista, proporcionando una visión detallada de las interacciones entre diferentes atributos. Por ejemplo, en un análisis de datos de encuestas, la minería de reglas de asociación puede revelar cómo ciertas combinaciones de educación y ocupación están asociadas con ingresos superiores a $50,000 al año. Esto puede ser extremadamente valioso para comprender los factores que influyen en los ingresos y para identificar oportunidades de mejora en diferentes áreas.

La minería de reglas de asociación funciona explorando todas las posibles combinaciones de atributos y determinando qué combinaciones ocurren con mayor frecuencia y con un patrón consistente. En el caso de un análisis de ingresos, podríamos descubrir que individuos con un nivel educativo de maestría y que trabajan en sectores tecnológicos tienen una alta probabilidad de ganar más de $50,000 anuales. Esta información puede ser utilizada por empresas y organizaciones para desarrollar estrategias de contratación y formación, dirigidas a maximizar el potencial de ingresos de sus empleados.

Además, estas reglas de asociación pueden revelar interacciones entre variables que no se habrían considerado previamente. Por ejemplo, podríamos encontrar que una combinación de ciertos niveles de educación y una ocupación específica se correlacionan con ingresos altos solo en determinados grupos de edad

o géneros. Esta capacidad para descubrir patrones complejos y multidimensionales es una de las fortalezas clave de la minería de reglas de asociación.

La implementación de esta técnica generalmente sigue varios pasos. Primero, es crucial preparar y limpiar los datos para asegurarse de que estén en un formato adecuado para el análisis. Esto incluye manejar los valores faltantes, eliminar duplicados y codificar las variables categóricas. Luego, se generan transacciones a partir de los datos, donde cada transacción representa una combinación de atributos de interés.

PARTE II

ANÁLISIS COMPONENTE PRINCIPAL (PCA)

En esta sección vamos a poner en ejecución lo que vimos anteriormente en relación con los análisis de machine learning, aplicando técnicas avanzadas para explorar y entender mejor nuestros datos. En particular, vamos a utilizar el Análisis de Componentes Principales (PCA), K-Means y DBSCAN para llevar a cabo un análisis más profundo y detallado.

Utilizaremos el Análisis de Componentes Principales (PCA) para reducir la dimensionalidad de nuestro conjunto de datos. PCA es una técnica estadística que permite transformar un conjunto de variables posiblemente correlacionadas en un conjunto de valores de variables no correlacionadas denominadas componentes principales. Al hacerlo, podemos visualizar y analizar nuestros datos en un espacio de menor dimensión, lo que facilita la identificación de patrones y tendencias subyacentes.

Una vez que hayamos reducido la dimensionalidad de nuestros datos con PCA, aplicaremos el algoritmo de clustering K-Means. Este es uno de los métodos de agrupamiento más populares y se utiliza para dividir un conjunto de datos en un número específico de clusters (grupos) basados en características similares. K-Means trabaja iterativamente para asignar cada punto de datos a uno de los clusters basándose en la distancia euclidiana entre los puntos y los centroides de los clusters. Esta técnica nos permitirá identificar grupos naturales en nuestros datos y entender mejor cómo se relacionan entre sí las diferentes observaciones.

Además de K-Means, también utilizaremos DBSCAN (Density-Based Spatial Clustering of Applications with Noise), otro algoritmo de clustering, pero con un enfoque diferente. A diferencia de K-Means, DBSCAN no requiere especificar el número de clusters de antemano. En su lugar, agrupa puntos basándose en la densidad de puntos en una región, lo que lo hace especialmente útil para detectar clusters de forma arbitraria y para manejar el ruido en los datos. DBSCAN identifica regiones de alta densidad de puntos que están separadas por regiones de baja densidad, permitiendo una detección más natural de clusters y outliers (datos atípicos).

Para iniciar con estos análisis, vamos a empezar con el PCA, como bien sabemos el PCA es un método estadístico cuya utilidad radica en la reducción de la dimensionalidad de la base de datos (BDD) con la que estamos trabajando. Esta técnica se utiliza cuando queremos simplificar la base de datos, ya sea para elegir un menor número de predictores para pronosticar una variable objetivo, o para comprender una BDD de una forma más simple.

El método de Análisis de Componentes Principales (PCA) permite condensar la información aportada por múltiples variables en solo unas pocas componentes.

Esto lo convierte en una herramienta muy útil antes de aplicar otras técnicas estadísticas, como regresión o clustering. Sin embargo, es importante recordar que aún se necesita disponer de los valores de las variables originales para calcular estas componentes. PCA es una técnica de reducción de dimensionalidad no supervisada, que agrupa puntos de datos similares basándose en la correlación de características entre ellos sin necesidad de supervisión (o etiquetas).

Los componentes principales son fundamentales en el PCA, ya que representan la esencia de los datos. En términos sencillos, al proyectar los datos en un espacio de menor dimensión (por ejemplo, de tres dimensiones) desde un espacio de mayor dimensión, estas nuevas dimensiones son los componentes principales. Estos componentes capturan la mayor parte de la varianza o información de los datos originales. Es decir, los componentes principales son nuevas variables creadas a partir de combinaciones lineales de las variables originales, y contienen la información más significativa, permitiendo una mejor interpretación y visualización de los datos en un espacio reducido.

Matemáticamente, es necesario calcular los vectores propios (eigenvectors) y los valores propios (eigenvalues) de la matriz de correlaciones o de la matriz de varianzas-covarianzas de las variables. Los eigenvectors de una matriz son aquellos vectores que, al ser multiplicados por la matriz, resultan en un vector que es un múltiplo escalar del vector original. Los eigenvalues son los factores escalares resultantes de esta multiplicación. Este proceso permite identificar las direcciones en las que varía más la información y, por lo tanto, los componentes principales que mejor representan la estructura interna de los datos.

PCA facilita la comprensión y el análisis de grandes conjuntos de datos al reducir su complejidad mientras se conserva la mayor parte de la información relevante. Esto lo hace especialmente valioso en la preparación de datos para modelos predictivos y técnicas de agrupación, mejorando la eficiencia y la precisión de estos análisis.

Para comenzar a obtener los valores y resultados deseados, es fundamental estructurar y ejecutar el código de manera meticulosa. Primero, asegurémonos de que todos los datos estén preparados adecuadamente, lo que incluye la limpieza de datos, el manejo de valores faltantes y la normalización de las variables. Este proceso inicial es crucial para asegurar la precisión y eficacia de cualquier análisis posterior. Y que bien, ya se realizó anteriormente.

Una vez que los datos estén listos, procedemos a la implementación del código, que implicará varias etapas. En primer lugar, aplicaremos técnicas de preprocesamiento de datos para garantizar que todas las variables estén en un formato adecuado para el análisis.

Luego, iniciaremos el análisis principal. Si estamos trabajando con técnicas de reducción de dimensionalidad como el Análisis de Componentes Principales (PCA), comenzaremos calculando la matriz de covarianza de los datos. A partir de esta matriz, extraeremos los vectores propios (eigenvectors) y los valores propios (eigenvalues), que nos ayudarán a identificar las componentes principales que capturan la mayor parte de la varianza en los datos. Este paso es esencial para reducir la dimensionalidad del conjunto de datos, facilitando su visualización y análisis sin perder información significativa.

El código a usar sería inicialmente:

```
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
from sklearn.pipeline import make_pipeline
from sklearn.preprocessing import scale
import matplotlib.pyplot as plt
import seaborn as sns

import warnings
warnings.filterwarnings('ignore')
```

Este conjunto de importaciones y configuraciones prepara el entorno para realizar un Análisis de Componentes Principales (PCA) con los datos. Las bibliotecas importadas proporcionan herramientas para estandarizar los datos StandardScaler, es utilizada para estandarizar las características eliminando la media y escalando a la varianza unitaria. Transforma los datos de manera que tengan una media de 0 y una desviación estándar de 1.

En la línea de PCA: se utiliza para realizar el Análisis de Componentes Principales (PCA), y para visualizar los resultados usaremos las librerías matplotlib.pyplot, seaborn. Por último tenemos la línea de warnings.filterwarnings('ignore')) asegura que las advertencias no interfieran con la salida visual y los resultados del análisis.

El Análisis de Componentes Principales (PCA) no puede aplicarse directamente a datos no numéricos. Por lo que se requiere que los datos sean numéricos porque se basa en cálculos algebraicos como la media, la varianza y la matriz de covarianza, los cuales no se pueden aplicar a datos categóricos o textuales.

A continuación, tenemos este código:

```
data_cov = pd.get_dummies(data)

data_cov = data_cov.cov()
print(data_cov)
```

La función pd.get_dummies(data) transforma variables categóricas en variables dummies (también conocidas como variables ficticias o indicadores).

Esta función convierte cada categoría en una nueva columna. Cada columna representa una categoría y contiene 1 si el valor de la variable es esa categoría, o 0 en caso contrario.

La función .cov() calcula la matriz de covarianza de los datos transformados. La covarianza mide cómo varían juntas dos variables. Una matriz de covarianza muestra la covarianza entre cada par de variables en el DataFrame.

La covarianza entre dos variables es positiva si ambas variables tienden a aumentar o disminuir juntas. Es negativa si una tiende a aumentar mientras la otra disminuye.

La matriz de covarianza es una matriz cuadrada donde cada entrada en la fila i y columna j representa la covarianza entre las variables i y j.

La matriz de covarianza para estas dos variables puede verse así:

Matriz de Covarianza

$$\begin{pmatrix} Var(x) & Cov(x,y) \\ Cov(x,y) & Var(y) \end{pmatrix}$$

Donde Var(X) y Var(Y) son las varianzas de X y Y, y Cov(X,Y) es la covarianza entre X y Y.

Y gracias a estas variables tenemos estos datos:

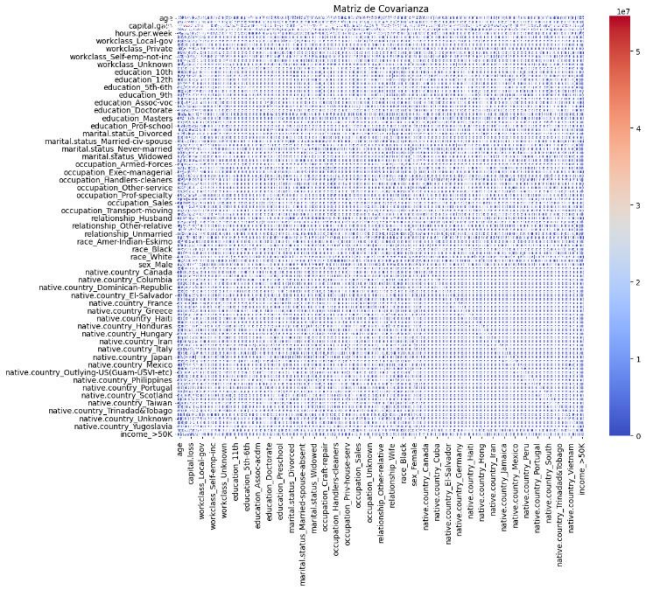
	age	education.num	capital.gain
age	186.061400	1.281849	7.824819e+03
education.num	1.281849	6.618890	2.330008e+03
capital.gain	7824.818537	2330.007877	5.454254e+07
capital.loss	317.560742	82.856445	-9.408576e+04
hours.per.week	11.580130	4.705338	7.150032e+03
...	...	...	...
native.country_Unknown	0.002577	0.009274	1.305070e+01
native.country_Vietnam	-0.009305	-0.000872	-8.581840e-01
native.country_Yugoslavia	0.000113	-0.000070	-3.589184e-01
income_<=50K	-1.364997	-0.368685	-7.052309e+02
income_>50K	1.364997	0.368685	7.052309e+02

data_cov.cov()

	capital.loss	hours.per.week
age	317.560742	11.580130
education.num	82.856445	4.705338
capital.gain	-94085.760688	7150.032029
capital.loss	162376.937814	269.953755
hours.per.week	269.953755	152.458995
...	...	...
native.country_Unknown	0.554388	0.019256
native.country_Vietnam	-0.000011	-0.006367
native.country_Yugoslavia	-0.042901	0.002027
income_<=50K	-25.935432	-1.212651
income_>50K	25.935432	1.212651

Data_cov.cov() 2

Para visualizarlo gráficamente podemos observar



Las variables age, education.num y hours.per.week tienen varianzas moderadas, lo que indica una variabilidad notable pero no extrema en estos atributos dentro del conjunto de datos. La varianza en age (186.061400) sugiere que existe una amplia gama de edades entre los individuos, reflejando una distribución diversa de edades en la población estudiada.

En hours.per.week (152.458995) indica una variabilidad significativa en el número de horas trabajadas por semana. Esto puede reflejar diferentes tipos de empleos, desde trabajos a tiempo parcial hasta empleos a tiempo completo, así como la flexibilidad laboral de los individuos.

Podemos observar que capital.gain y capital.loss presentan una covarianza alta y negativa de -94,085.76. Esta relación sugiere que cuando las ganancias de capital son altas, las pérdidas de capital tienden a ser bajas, y viceversa. Esta relación negativa es lógica, ya que una alta ganancia de capital podría estar asociada con una menor necesidad de realizar deducciones por pérdidas de capital, reflejando una gestión financiera eficaz. Esto también podría indicar que los individuos que experimentan altas ganancias de capital suelen ser aquellos que minimizan sus pérdidas, optimizando sus estrategias de inversión y ahorro.

La covarianza moderada entre age y capital.gain (≈7,824.82) indica que hay una cierta relación entre la edad y las ganancias de capital, aunque no es extremadamente fuerte. Esto sugiere que a medida que las personas envejecen, tienden a tener mayores ganancias de capital, posiblemente debido a una mayor experiencia en inversiones o una acumulación de activos a lo largo del tiempo. Sin embargo, esta relación no es lo suficientemente fuerte como para afirmar una correlación directa y significativa.

Para poder visualizar este código hicimos el siguiente código:

plt.figure(figsize=(12, 10))

sns.heatmap(data_cov, annot=True, cmap='coolwarm', fmt=".2f")

plt.title('Matriz de Covarianza')

plt.show()

A continuación, vamos a visualizar el siguiente código:

ss = StandardScaler()

x_cols = ['age', 'education.num', 'capital.gain', 'capital.loss', 'hours.per.week']

X_subset = data[x_cols]

data[x_cols] = ss.fit_transform(X_subset)

print(data)

Para entender mejor el código que estamos usando en: En x_cols vamos a agregar las columnas con las que hemos venido trabajando, específicamente aquellas que queremos normalizar para asegurar que tengan una distribución comparable.

ss.fit_transform(X_subset) hace dos cosas:

fit: En esta etapa, se calculan la media y la desviación estándar de cada columna en X_subset. Estos cálculos son esenciales para comprender la distribución de los datos antes de transformarlos. Por ejemplo, supongamos que obtenemos los siguientes resultados:

Por ejemplo:

age: 25, media=43.33, std=17.56

education.num: 13, media=9.67, std=3.06

capital.gain: 0, media=1666.67, std=2357.02

transform: Utilizando las medias y desviaciones estándar calculadas en la etapa de "fit", se transforman los datos de X_subset. El objetivo es que cada columna tenga una media de 0 y una desviación estándar de 1, lo que facilita la comparación y el análisis de los datos, especialmente en técnicas de machine learning. La fórmula para la estandarización es:

$$\frac{Valor\ original - media}{Desviación\ estándar}$$

Por lo que nos daría como resultado según los valores anteriores:

age:

- Valor original: 25
- Media: 43.33
- Desviación estándar: 17.56
- Resultado: $\frac{25-43.33}{17,56} = -1,04$

education.num:

- Valor original: 13
- Media: 9.67
- Desviación estándar: 3.06
- Resultado: $\frac{13-9,67}{3,06} = 1,09$

capital.gain:

- Valor original: 0
- Media: 1666.67
- Desviación estándar: 2357.02
- Cálculo: $\frac{0-1666,67}{2357,02} = -0,71$

El resultado de estas transformaciones es que ahora tenemos datos estandarizados, lo que significa que todos los valores están centrados en torno a 0 con una desviación estándar de 1. Esto es crucial para muchos algoritmos de machine learning.

Al transformar los datos de esta manera, eliminamos las diferencias de escala entre las variables, permitiendo

que se puedan comparar de manera más directa y se puedan identificar patrones y relaciones subyacentes más fácilmente.

Por lo que se obtiene el siguiente análisis:

	age	education.num	capital.gain	capital.loss	hours.per.week
0	3.769612	-0.420060	-0.14592	10.593507	-0.035429
1	3.183112	-0.420060	-0.14592	10.593507	-1.817204
2	2.010110	-0.031360	-0.14592	10.593507	-0.035429
3	1.130359	-2.363558	-0.14592	9.461864	-0.035429
4	0.177296	-0.031360	-0.14592	9.461864	-0.035429
...	...	...	...	...	...
32556	-1.215643	-0.031360	-0.14592	-0.216660	-0.035429
32557	-0.849080	0.746039	-0.14592	-0.216660	-0.197409
32558	0.103983	-0.420060	-0.14592	-0.216660	-0.035429
32559	1.423610	-0.420060	-0.14592	-0.216660	-0.035429
32560	-1.215643	-0.420060	-0.14592	-0.216660	-1.655225

Como se pudo observar anteriormente, este es el resultado generado al aplicar las funciones descritas. La transformación de los datos ha sido exitosa, lo que se refleja en los valores estandarizados obtenidos. Este proceso garantiza que todas las columnas de nuestro subconjunto de datos, X_subset, tengan una media centrada en 0 y una desviación estándar de 1. Podemos confirmar que la estandarización se ha llevado a cabo correctamente.

Es importante mantener la desviación estándar entre 0 y 1 porque al estandarizar las variables, aseguramos que todas ellas operen en la misma escala. Esto es fundamental para evitar que las diferencias de escala entre las variables originales distorsionen los resultados de los análisis y modelos posteriores. Además, tener los datos en una escala estándar facilita la comparación entre diferentes características y observaciones. Esto permite identificar patrones de manera más clara y directa.

Para continuar con el proceso vamos a usar el siguiente código:

```
numeric_cols = data.select_dtypes(include=['int64', 'float64'])
pca = PCA(n_components=2)  # Elegir 2 componentes principales para visualizar en 2D
pca_result = pca.fit_transform(scaled_data)
pca_data = pd.DataFrame(data=pca_result, columns=['PC1', 'PC2'])
# Verificar si 'income' existe en el DataFrame original y renombrarlo si es necesario
if 'income' in data.columns:
    pca_data['income'] = data['income']
elif 'target' in data.columns: # Check if the column is named 'target' instead
    pca_data['income'] = data['target']
else:
    print("Warning: No 'income' or 'target' column found. The scatterplot will not be colored by class.")
plt.figure(figsize=(10, 6))
if 'income' in pca_data.columns:
    sns.scatterplot(x='PC1', y='PC2', hue='income', data=pca_data, palette='bright')
else:
    sns.scatterplot(x='PC1', y='PC2', data=pca_data, palette='bright') # Plot without hue if 'income' is missing
plt.title('PCA de Datos')
plt.xlabel('Primer Componente Principal (PC1)')
plt.ylabel('Segundo Componente Principal (PC2)')
plt.show()
```

 # Mostrar la varianza explicada por cada componente principal

*explained_variance = pca.explained_variance_ratio_*

print(f'Varianza explicada por el PC1: {explained_variance[0]:.2f}')

print(f'Varianza explicada por el PC2: {explained_variance[1]:.2f}')

Para iniciar, se seleccionan solo las columnas que contienen datos numéricos (tipos int64 y float64) del DataFrame. Esta selección es esencial ya que solo estos tipos de datos pueden ser estandarizados y utilizados para el análisis de componentes principales (PCA). La estandarización es el siguiente paso clave, donde se transforman los datos para tener una media de 0 y una desviación estándar de 1, lo que es crucial para que el PCA funcione correctamente.

A continuación, mediante PCA(n_components=2), se aplica el análisis de componentes principales (PCA) con el objetivo de reducir la dimensionalidad de los datos a dos componentes principales. Esta reducción de dimensionalidad no solo simplifica los datos, sino que también facilita su visualización en un espacio bidimensional (2D), permitiendo una mejor comprensión de la estructura interna de los datos.

Luego, se ajusta el modelo PCA a los datos estandarizados (scaled_data) utilizando el método fit_transform(), lo que da como resultado pca_result. Este resultado contiene los datos transformados en términos de los dos componentes principales seleccionados. Este paso esencialmente proyecta los datos originales en un nuevo espacio con dos dimensiones, que capturan la mayor parte de la variabilidad en los datos originales.

Con los resultados del PCA, se crea un nuevo DataFrame pca_data que contiene las dos nuevas columnas denominadas PC1 (Primer Componente Principal) y PC2 (Segundo Componente Principal). Estas columnas representan las nuevas dimensiones en las que se han proyectado los datos originales.

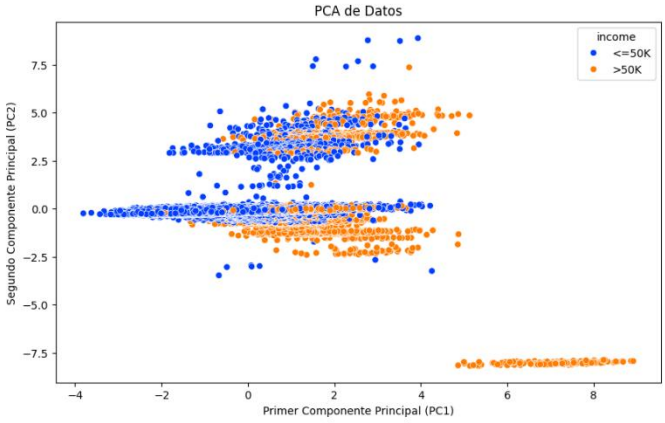


Gráfico de PCA

De igual manera. se verifica si la columna income (o la que se esté analizando) existe en el DataFrame original data. Si está presente, se agrega esta columna a pca_data para permitir la colorización de los puntos en el gráfico según la clase income. Si la columna no existe, se genera una advertencia para informar que el gráfico no se podrá colorizar por clases. Una vez con esta información se realiza un gráfico de dispersión para visualizar los datos transformados. Si la columna income está presente, se utiliza para diferenciar los puntos en el gráfico mediante colores, lo que ayuda a identificar patrones o agrupaciones según las clases.

Al final, se obtiene la varianza explicada por cada componente principal a través del atributo explained_variance_ratio_ del PCA. Este atributo indica la proporción de la varianza total del conjunto de datos que es capturada por cada componente principal. Los porcentajes de varianza explicada por los dos primeros componentes se imprimen para proporcionar una medida de cuánta información de los datos

originales se retiene en los dos componentes principales. Este paso es crucial para entender la efectividad del PCA en la reducción de dimensionalidad y en la preservación de la información original de los datos.

Para continuar con nuestro análisis vamos a usar este código:

```
def correlation_matrix(data):
    # Seleccionar solo columnas numéricas para calcular la correlación
    numeric_data = data.select_dtypes(include=['int64', 'float64'])
    fig = plt.figure(figsize=(16, 12))
    ax1 = fig.add_subplot(111)
    cmap = cm.get_cmap("jet", 30)

    # Calcular correlación en datos numéricos solamente
    cax = ax1.imshow(numeric_data.corr(), interpolation="nearest", cmap=cmap)
    ax1.grid(True)
    plt.title("Correlación de características del conjunto de datos", fontsize=15)
    labels = numeric_data.columns  # Usar etiquetas de datos numéricos
    ax1.set_xticks(np.arange(len(labels)))
    ax1.set_yticks(np.arange(len(labels)))
    ax1.set_xticklabels(labels, fontsize=9)
    ax1.set_yticklabels(labels, fontsize=9)
    # Añadir barra de color
    fig.colorbar(cax, ticks=[0.1 * i for i in range(-11, 11)])
    plt.show()
```

En este código, trabajaremos con los datos numéricos y crearemos un gráfico para visualizar las relaciones entre ellos. El proceso se lleva a cabo de la siguiente manera:

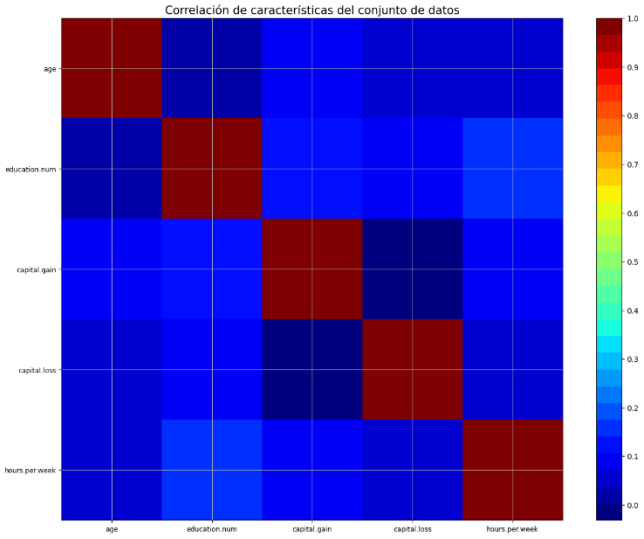
Primero, seleccionamos las columnas que contienen datos numéricos del DataFrame original. Esto se hace utilizando la función select_dtypes(include=['int64', 'float64']), que filtra las columnas según sus tipos de datos. Este paso es crucial porque solo las columnas numéricas pueden participar en el cálculo de la correlación.

A continuación, configuramos el gráfico para la visualización. Utilizamos plt.figure(figsize=(16, 12)) para crear una figura de tamaño 16x12 pulgadas, proporcionando un amplio espacio para el gráfico y asegurando que las etiquetas de los ejes sean legibles. Luego, añadimos un subplot a la figura con fig.add_subplot(111), que indica que tendremos un único subplot en la figura.

Para la paleta de colores, usamos cm.get_cmap("jet", 30). La función cm.get_cmap del módulo matplotlib.cm obtiene una paleta de colores llamada "jet" con 30 colores diferentes. Esta paleta se aplicará a la matriz de correlación para visualizar los diferentes niveles de correlación entre las variables.

El siguiente paso es calcular la matriz de correlación utilizando numeric_data.corr(). Este método calcula la correlación entre todas las columnas numéricas del DataFrame. La correlación es una medida estadística

que describe la relación lineal entre dos variables. Los valores de correlación oscilan entre -1 y 1.



La barra de color a la derecha del gráfico muestra que los valores de correlación varían de 0.0 (sin correlación) a 1.0 (correlación perfecta positiva).

En este caso, parece que no hay correlaciones negativas fuertes ya que la escala mostrada solo va de 0.0 a 1.0.

Podemos ver que las horas trabajadas no están muy relacionadas con la edad.

Vamos a analizar también la varianza explicada por cada componente principal, por medio del siguiente código:

plt.figure(figsize=(10, 6))

plt.bar(range(1, len(explained_variance) + 1), explained_variance, color='skyblue')

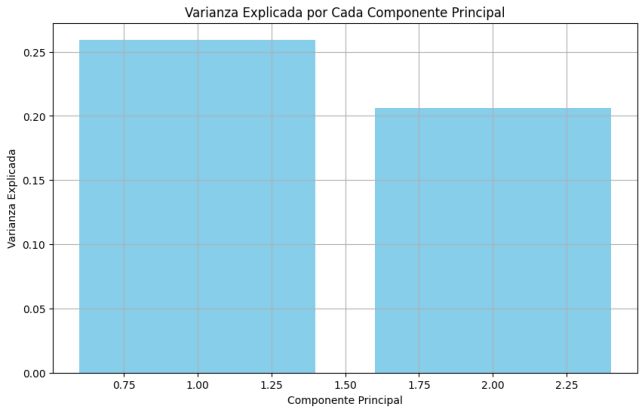
plt.title('Varianza Explicada por Cada Componente Principal')

plt.xlabel('Componente Principal')

plt.ylabel('Varianza Explicada')

plt.grid(True)

plt.show()



Varianza explicada

Podemos notar que el Componente Principal 1 (PC1) explica aproximadamente el 25% de la varianza total en los datos. Esto significa que una cuarta parte de toda la información presente en el conjunto de datos original puede ser representada por este primer componente, lo que lo convierte en el más significativo en términos de captura de varianza.

Por otro lado, el Componente Principal 2 (PC2) explica alrededor del 20% de la varianza total en los datos. Aunque menos influyente que el PC1, el PC2 todavía retiene una parte sustancial de la información, aproximadamente una quinta parte de la varianza total.

Juntos, estos dos componentes principales (PC1 y PC2) explican un total del 45% de la varianza en los datos. Este resultado indica que casi la mitad de toda la información en el conjunto de datos puede ser capturada mediante estos dos componentes, lo cual es significativo y muestra que el PCA ha sido efectivo en reducir la dimensionalidad mientras retiene una cantidad considerable de información.

Los resultados del PCA son extremadamente útiles para la reducción de dimensionalidad, una técnica que simplifica los datos al reducir el número de variables bajo consideración, facilitando el análisis y la

visualización sin perder una cantidad significativa de información. Aunque solo dos componentes explican una cantidad razonable de la varianza, puede ser necesario considerar más componentes para capturar la mayor parte de la información en el conjunto de datos. Esto es particularmente importante en conjuntos de datos con una estructura compleja, donde la varianza se distribuye a lo largo de muchos componentes.

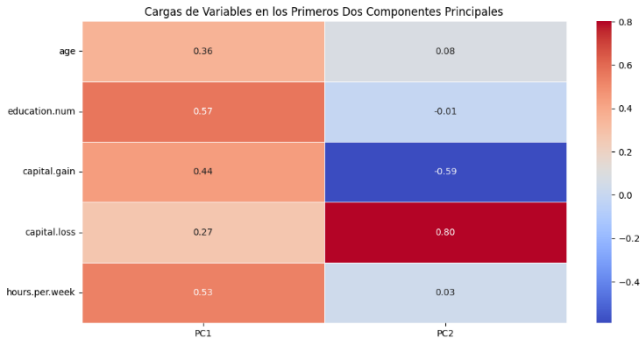
Mediante el código:

```
pca_loadings = pd.DataFrame(pca.components_.T, columns=[f'PC{i+1}' for i in range(pca.n_components_)], index=x_cols)

print(pca_loadings)

# Gráfico de cargas de variables en los primeros dos componentes principales

plt.figure(figsize=(12, 6))

sns.heatmap(pca_loadings[['PC1', 'PC2']], annot=True, cmap='coolwarm', fmt=".2f", linewidths=0.5)

plt.title('Cargas de Variables en los Primeros Dos Componentes Principales')

plt.show()
```

pca.components_: Este atributo del objeto PCA contiene los componentes principales obtenidos después de ajustar el PCA. Cada fila en pca.components_ representa un componente principal y cada columna representa una carga (coeficiente) de una variable original. Para facilitar el análisis y la visualización, transponemos esta matriz usando .T, convirtiendo las filas en columnas y viceversa. Esto se hace porque queremos que las variables originales sean las filas y los componentes principales sean las columnas en nuestro DataFrame.

Una vez realizado esto, creamos un DataFrame llamado pca_loadings que contiene las cargas de cada variable original en cada componente principal. Este DataFrame tiene las variables originales como filas y los componentes principales como columnas, lo que facilita la interpretación de los resultados del PCA.

Finalmente, para visualizar estas cargas de manera efectiva, utilizamos un mapa de calor (heatmap) que muestra gráficamente las contribuciones de cada variable a los primeros dos componentes principales.



Podemos ver que un mapa de calor (heatmap) que representa las cargas de variables originales en los dos primeros componentes principales (PC1 y PC2) obtenidos. Los colores van del azul (cargas negativas) al rojo (cargas positivas), indicando la dirección y magnitud de la contribución de cada variable a los componentes principales.

Para la variable age (edad), la carga en el primer componente principal (PC1) es 0.36, lo que indica una contribución moderada y positiva. Esto sugiere que, en el espacio de PC1, la edad tiene una influencia moderada, contribuyendo de manera significativa a la variabilidad capturada por este componente. En cambio, en el segundo componente principal (PC2), la carga es 0.08, lo que indica una contribución muy baja

y positiva, sugiriendo que la edad tiene una influencia prácticamente insignificante en PC2.

Las variables education.num (nivel educativo) y hours.per.week (horas trabajadas por semana) son las que más contribuyen positivamente a PC1, con cargas de 0.57 y 0.53 respectivamente. Esto implica que PC1 captura principalmente la variabilidad asociada con el nivel educativo y las horas trabajadas. La variable capital.gain (ganancias de capital) también contribuye de manera significativa a PC1, con una carga de 0.44, seguida de age con una carga de 0.36.

Aunque capital.loss (pérdidas de capital) tiene una carga de 0.27 en PC1, indicando una contribución positiva, su influencia es menor en comparación con las otras variables mencionadas anteriormente.

Sin embargo, para PC2, la variable capital.loss tiene una influencia muy fuerte y positiva, con una carga de 0.80. Esto sugiere que PC2 captura principalmente la variabilidad relacionada con las pérdidas de capital. En contraste, capital.gain tiene una fuerte influencia negativa en PC2, con una carga de -0.59, indicando que las ganancias de capital también son una fuente importante de variabilidad en este componente, pero en la dirección opuesta.

Por otro lado, las variables age, education.num y hours.per.week tienen influencias muy bajas en PC2, con cargas de 0.08, -0.01, y 0.03 respectivamente. Esto indica que estas variables no contribuyen significativamente a la variabilidad capturada por PC2.

A continuación, vamos a realizar un análisis por categorías para entender mejor la distribución de diferentes variables en nuestro conjunto de datos. Comenzaremos con la distribución de la edad en la columna age.

Para ello, utilizaremos el mismo código de visualización, pero adaptándolo para cada categoría específica. En este caso, cambiaremos el dato en pca_data["columna"] para reflejar la columna que queremos analizar. El código es: *numeric_cols = data.select_dtypes(include=['int64', 'float64'])*

scaler = StandardScaler()

scaled_data = scaler.fit_transform(numeric_cols)

pca = PCA(n_components=2) # Elegir 2 componentes principales para visualizar en 2D

pca_result = pca.fit_transform(scaled_data)

pca_data = pd.DataFrame(data=pca_result, columns=['PC1', 'PC2'])

 # Añadir las etiquetas de clase (income) para colorizar el gráfico

 # Asegurarse de que 'native.country' está presente en el DataFrame original 'data'

pca_data['age'] = data['age'] # Add this line to include 'native.country'

plt.figure(figsize=(10, 6))

sns.scatterplot(x='PC1', y='PC2', hue='age', data=pca_data, palette='bright')

plt.title('PCA de Datos')

plt.xlabel('Primer Componente Principal (PC1)')

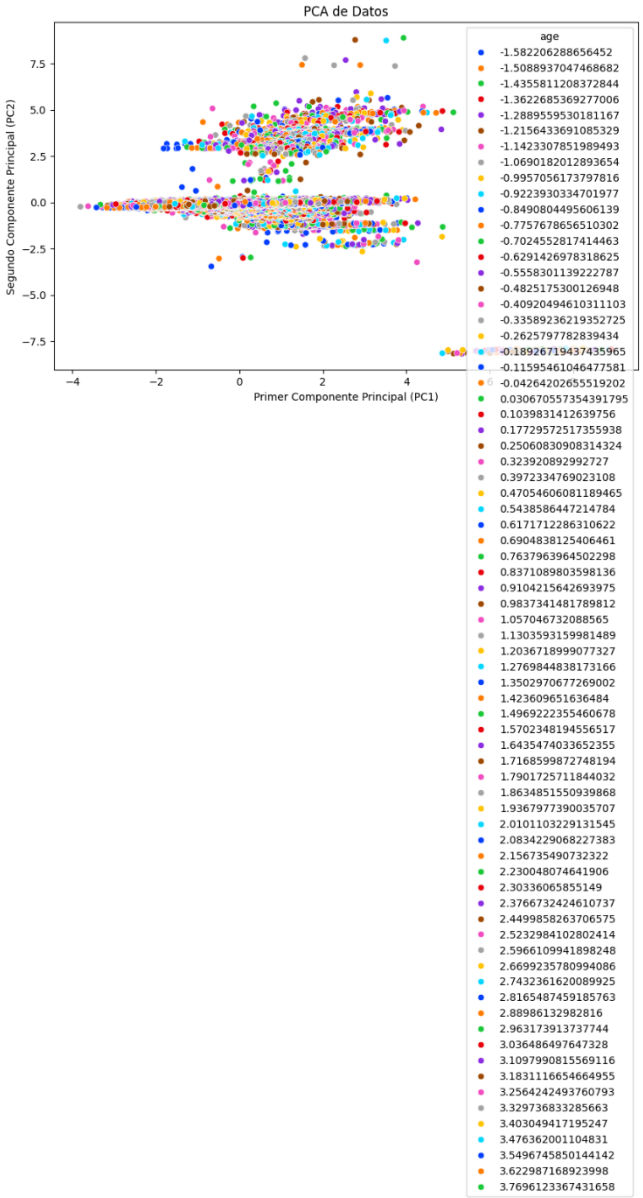
plt.ylabel('Segundo Componente Principal (PC2)')

plt.show()

*explained_variance = pca.explained_variance_ratio_*

print(f'Varianza explicada por el PC1: {explained_variance[0]:.2f}')

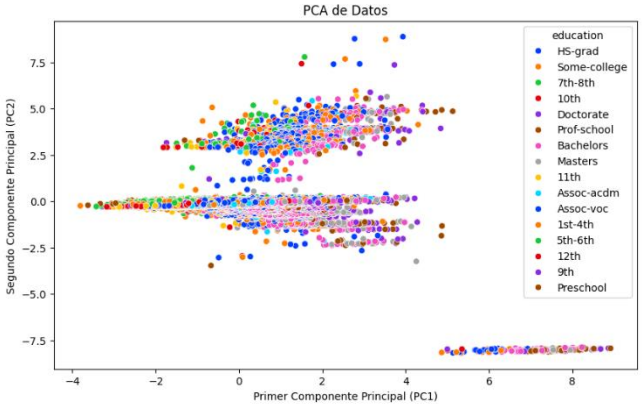
print(f'Varianza explicada por el PC2: {explained_variance[1]:.2f}')



PCA age

La distribución de los puntos en función del color sugiere que los colores más cálidos (naranja, rojo) representan edades más altas, mientras que los colores más fríos (azul, verde) representan edades más bajas. Están más dispersas hacia los extremos, especialmente en PC1.

Esto indica que la edad podría tener una relación no lineal con las componentes principales, donde ciertas edades están más centralizadas mientras que otras están más dispersas.

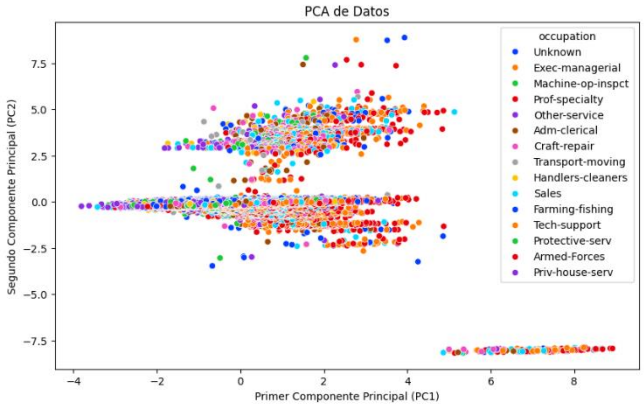


PCA education

Ahora vamos a analizar la columna education. Según el gráfico obtenido, podemos concluir que la mayoría de los puntos se agrupan en franjas horizontales a lo largo del eje X (PC1), con una densidad mayor alrededor de la media de PC1 y PC2. Esto indica que la mayor parte de la variabilidad en los datos se encuentra cerca del origen de estos componentes principales.

Los niveles educativos representados en el gráfico van desde Preschool (preescolar) hasta Doctorate (doctorado), pasando por diferentes grados de educación secundaria y superior. Los puntos están coloreados según el nivel educativo, lo que permite visualizar cómo se distribuyen estos niveles en el espacio de los componentes principales.

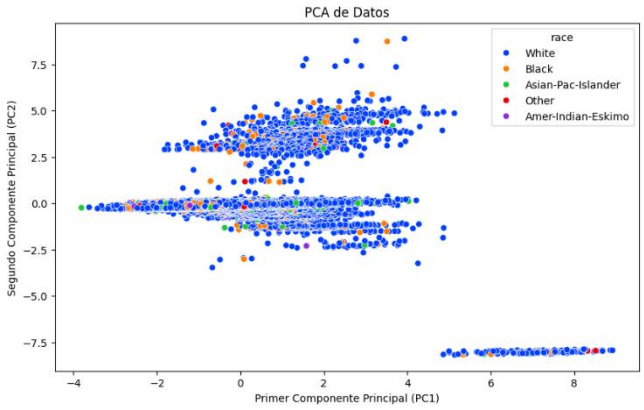
Los niveles educativos como HS-grad (Graduado de secundaria) y Some-college (Algo de universidad) están bien representados y distribuidos a lo largo del gráfico. Esto sugiere que son comunes en el conjunto de datos y no están asociados con variabilidades extremas en los componentes principales. Por otro lado, los niveles educativos superiores como Doctorate (Doctorado) y Prof-school (Escuela profesional), aunque menos frecuentes, están distribuidos de manera similar a otros niveles. Esto indica que estos individuos no se agrupan en áreas específicas de los componentes principales.



PCA occupation

Como podemos ver en el siguiente análisis correspondiente a la columna occupation, las ocupaciones incluyen una variedad de categorías como Exec-managerial, Machine-op-inspct, Prof-specialty, Adm-clerical, entre otras, incluyendo un valor Unknown para ocupaciones no especificadas. Las ocupaciones como Exec-managerial (Ejecutivo-gerencial) y Prof-specialty (Especialista profesional) están bien representadas y distribuidas a lo largo del gráfico. Esto sugiere que son comunes en el conjunto de datos y no están asociadas con variabilidades extremas en los componentes principales.

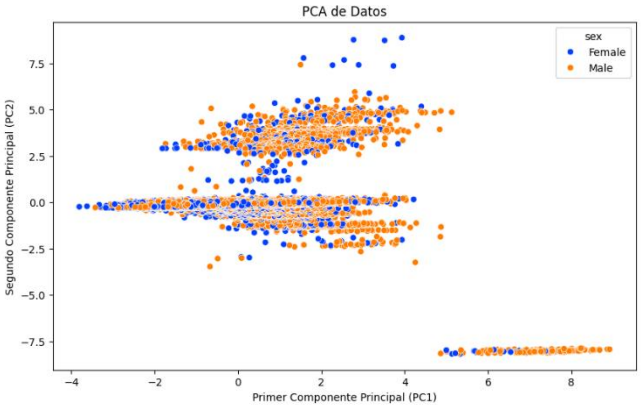
Las ocupaciones menos frecuentes, como Armed-Forces (Fuerzas armadas) y Priv-house-serv (Servicio doméstico privado), muestran una distribución similar a otras ocupaciones, indicando que estos individuos no se agrupan en áreas específicas de los componentes principales. Los puntos etiquetados como Unknown están distribuidos de manera uniforme, lo que indica que las ocupaciones no especificadas no presentan una variabilidad distinta en comparación con otras ocupaciones.



PCA race

Las razas incluyen categorías como White, Black, Asian-Pac-Islander, Amer-Indian-Eskimo, y Other. En el gráfico, se observa que la mayoría de los puntos son de color azul, lo que indica que la raza blanca es la más representada en el conjunto de datos. Aunque menos frecuentes que la raza blanca, las razas negra y asiático-pacífica están distribuidas de manera uniforme a lo largo del gráfico. Las razas indio americano-eskimo y otras, aunque son las menos frecuentes, muestran una

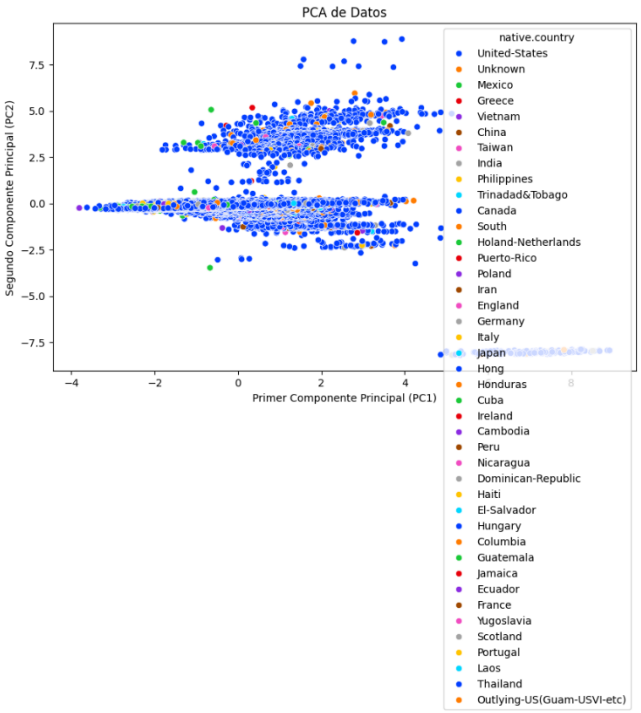
distribución similar a las demás razas. Esto indica que estas razas no presentan una variabilidad distinta en comparación con otras razas.



PCA sex

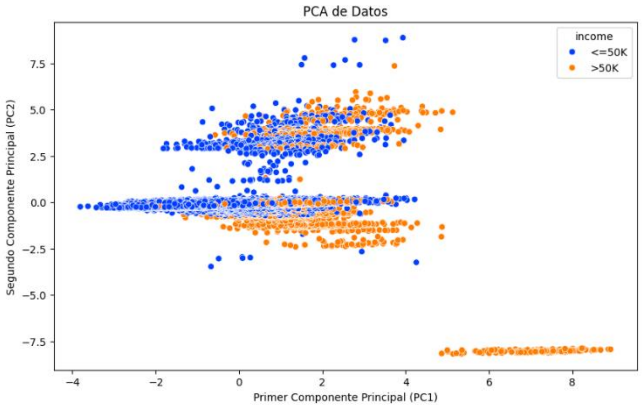
En general, la distribución de los colores sugiere que no hay una fuerte segregación de las razas a lo largo de los componentes principales. Los puntos se agrupan en franjas horizontales a lo largo del eje X (PC1), con una densidad mayor alrededor de la media de PC1 y PC2, lo que indica que la mayor parte de la variabilidad en los datos se encuentra cerca del origen de estos componentes principales. La representación predominante de la raza blanca en el gráfico y la distribución uniforme de las demás razas sugieren que las características raciales no están asociadas con variabilidades extremas en los componentes principales.

En el gráfico, los puntos azules representan a las mujeres y están distribuidos de manera uniforme a lo largo del gráfico. Esto indica que las mujeres están bien representadas en el conjunto de datos y no están asociadas con variabilidades extremas en los componentes principales. De manera similar, los puntos naranjas que representan a los hombres también están bien distribuidos. Esto muestra que los hombres son igualmente representados y que su distribución no presenta una variabilidad distinta en comparación con las mujeres. La representación equilibrada de ambos sexos en el gráfico sugiere que tanto hombres como mujeres contribuyen de manera similar a la variabilidad capturada por los componentes principales, sin que uno de los sexos esté asociado con variabilidades extremas o características inusuales en comparación con el otro.



PCA native country

Estados Unidos es la categoría más representada, seguida de otras como Unknown (Desconocido), México, Grecia, Vietnam, China, entre otros. La mayoría de los puntos son de color azul, indicando que Estados Unidos es el país más representado en el conjunto de datos. Aunque menos frecuentes que Estados Unidos, los países Unknown (Desconocido) y México están distribuidos de manera uniforme a lo largo del gráfico. Otros países como Grecia, Vietnam, China, y demás, están representados en menor cantidad y muestran una distribución similar a otras categorías, lo que indica que no presentan una variabilidad distinta en comparación con otros países.



PCA income

Respecto a los ingresos, la mayor densidad de puntos azules en el centro del gráfico sugiere que la mayoría de los individuos con ingresos menores o iguales a 50K tienen características similares que son capturadas por los componentes principales. En contraste, la dispersión de puntos naranjas a lo largo del primer componente principal (PC1) sugiere que los individuos con ingresos mayores a 50K tienen una mayor variabilidad en sus características. Este componente parece ser un buen discriminante para los ingresos, ya que los individuos con mayores ingresos (>50K) están desplazados hacia la derecha en el gráfico. Esto indica que las características capturadas por PC1 están asociadas con niveles de ingresos, proporcionando una clara diferenciación entre los distintos niveles de ingresos en el conjunto de datos.

ANÁLISIS K-MEANS

El algoritmo k-means es un método de agrupamiento que divide un conjunto de datos en k grupos o clusters. Los datos se agrupan de tal manera que los puntos en el mismo clúster sean más similares entre sí que los puntos en otros clusters. Este método es ampliamente utilizado en análisis de datos y aprendizaje automático debido a su simplicidad y eficiencia.

La razón por la que existe este método es porque hoy en día la cantidad total de datos creados, capturados, copiados y consumidos globalmente es de aproximadamente 100 Zettabytes y seguirá creciendo. Con el algoritmo k-means es posible recopilar grandes cantidades de información similar en un mismo lugar, hecho que ayuda a encontrar patrones y hacer predicciones en grandes conjuntos de datos. Este proceso de agrupamiento permite a las empresas y científicos de datos comprender mejor sus datos, identificar tendencias y comportamientos, segmentar mercados y clientes, y mejorar la toma de decisiones basada en datos.

Para utilizar el algoritmo k-means, primero se especifica el número de clusters deseados (k). Por ejemplo, al establecer «k» igual a 2, su conjunto de datos se agrupará en 2 grupos, mientras que si establece «k» igual a 4, agrupará los datos en 4 grupos. La elección del valor de k puede ser un desafío y a menudo se determina mediante el método del codo, que implica ejecutar el algoritmo con varios valores de k y observar la disminución de la inercia (una medida de la variabilidad dentro de los clusters) para identificar el punto donde la disminución se estabiliza.

Cada grupo está representado por su centro o centroide, que corresponde a la media aritmética de los puntos de datos asignados al grupo. De esta manera, el algoritmo funciona a través de un proceso iterativo hasta que cada punto de datos está más cerca del centroide de su propio grupo que de los centroides de otros grupos, minimizando la distancia dentro del grupo en cada paso. Este proceso iterativo se denomina ajuste de k-means y generalmente converge rápidamente, lo que hace que k-means sea adecuado para grandes conjuntos de datos.

El algoritmo k-means comienza con la selección inicial de k centroides, que pueden ser seleccionados de manera aleatoria o utilizando métodos como k-means++. Luego, cada punto de datos se asigna al clúster cuyo centroide esté más cercano. Una vez que todos los puntos de datos han sido asignados, los centroides se recalculan como la media de los puntos de datos en cada clúster. Este proceso de asignación y recalculación de centroides se repite hasta que los centroides no cambian significativamente entre iteraciones, indicando que el algoritmo ha convergido y los clústeres son estables.

El algoritmo k-means, aunque poderoso, tiene algunas limitaciones. Una de ellas es que requiere que se especifique el número de clusters de antemano, lo cual puede no ser siempre evidente. Además, k-means asume que los clusters tienen formas esféricas y aproximadamente el mismo tamaño, lo que puede no ser el caso en todos los conjuntos de datos. Pese a estas limitaciones, k-means sigue siendo una herramienta valiosa en el análisis de datos, especialmente cuando se combinan con otros métodos y técnicas de validación para mejorar su rendimiento y precisión.

Para empezar con nuestro análisis vamos primero a empezar a codificar:

```
data = data.replace('Unknown', np.nan)

data = data.dropna()

data['income'] = data['income'].apply(lambda x: 1 if x == '>50K' else 0)

data_for_clustering = data[['age', 'income']]

scaler = StandardScaler()

data_scaled = scaler.fit_transform(data_for_clustering)
```

El código primero convierte la columna income de un dataset en valores numéricos, asignando 1 a los ingresos mayores a 50K y 0 a los ingresos menores o iguales a 50K. Esto se realiza con la función apply y una función lambda que evalúa cada valor de la columna income y lo reemplaza con 1 o 0 según corresponda.

Luego, se seleccionan las columnas age e income del dataset original para crear un nuevo DataFrame llamado data_for_clustering. Este nuevo DataFrame contendrá únicamente estas dos columnas, que son las características que se usarán para el análisis posterior.

Para asegurar que estas características tengan una escala comparable, se estandarizan utilizando la clase StandardScaler de sklearn. El escalador se ajusta a los datos y luego transforma los valores, centrando y escalando cada característica a una media de 0 y una desviación estándar de 1. El resultado de esta transformación es un array data_scaled, que contiene los datos estandarizados listos para ser usados en algoritmos de aprendizaje automático o análisis posterior.

Seguida a esto temenos que usar:

```
modelo_kmeans = KMeans(n_clusters=4, n_init=25, random_state=123)

modelo_kmeans.fit(X=data_scaled)

# Predecir clusters

y_predict = modelo_kmeans.predict(data_scaled)
```

El código crea un modelo de clustering KMeans configurado para identificar cuatro clusters en los datos. Se especifica que el algoritmo debe inicializarse 25 veces con diferentes centroides iniciales para asegurar una mejor solución final. También se

establece una semilla aleatoria para garantizar que los resultados sean reproducibles en futuras ejecuciones del mismo código.

Después de configurar el modelo KMeans, se entrena con los datos estandarizados. Este proceso implica que el algoritmo ajusta sus centroides iterativamente hasta que la variación dentro de cada cluster se minimiza, agrupando los datos de manera que los puntos dentro del mismo cluster sean más similares entre sí que los puntos en otros clusters.

Una vez entrenado el modelo, se utilizan los datos estandarizados para predecir a qué cluster pertenece cada punto de datos. Las predicciones se almacenan en y_predict, un array que contiene las etiquetas de cluster para cada punto de datos en el conjunto original. Esto permite identificar cómo se agrupan los datos según las características especificadas, proporcionando una base para análisis posteriores o visualizaciones de los clusters formados.

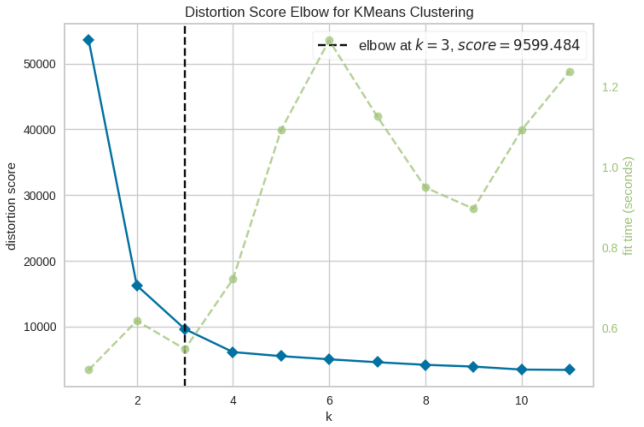
Para luego continuar usando este otro código:

```
import yellowbrick


# Handle non-numerical values and select relevant features
data_for_clustering = data[['age', 'income']]
data_for_clustering = data_for_clustering.replace('Unknown', np.nan).dropna()  # Handle missing/unknown values
data_for_clustering['income'] = data_for_clustering['income'].apply(lambda x: 1 if x == '>50K' else 0)

# Standardize the data
scaler = StandardScaler()
data_scaled = scaler.fit_transform(data_for_clustering)


# Apply KElbowVisualizer to the *scaled* numerical data
model = KMeans()
visualizer = yellowbrick.cluster.elbow.KElbowVisualizer(model, k=(1, 12))
visualizer.fit(data_scaled)  # Fit to the scaled numerical data
visualizer.show()
```

Importando la librería yellowbrick, que es una herramienta de visualización para el aprendizaje automático. A continuación, se define un modelo de KMeans sin especificar el número de clusters, permitiendo que el número óptimo sea determinado posteriormente.

El visualizador de codo, KElbowVisualizer, es creado a partir del modelo KMeans. Este visualizador está configurado para probar diferentes números de clusters, en este caso, variando de 1 a 12. El método de codo es una técnica comúnmente utilizada para determinar el número óptimo de clusters al observar la suma de las distancias cuadradas dentro de los clusters.

El visualizador es entrenado con los datos proporcionados. Durante este proceso, el algoritmo KMeans se ajusta a los datos varias veces, probando diferentes números de clusters, y calculando la suma de las distancias cuadradas para cada configuración. El objetivo es identificar el punto en el gráfico donde la disminución de la inercia comienza a estabilizarse, formando un "codo".



Elbow code

En este gráfico, el codo se identifica alrededor de k=3. Este punto es donde la tasa de disminución de la puntuación de distorsión se estabiliza significativamente, lo que indica que elegir un valor de k en el codo es óptimo. Este equilibrio proporciona una buena combinación entre la minimización de la distorsión y la simplicidad del modelo, manteniendo un número manejable de clusters. Dividir los datos en tres grupos resulta en una baja distorsión y un modelo manejable. Optar por un valor de k mucho mayor no proporcionará una mejora significativa en la reducción de la distorsión, pero incrementará la complejidad del modelo innecesariamente.

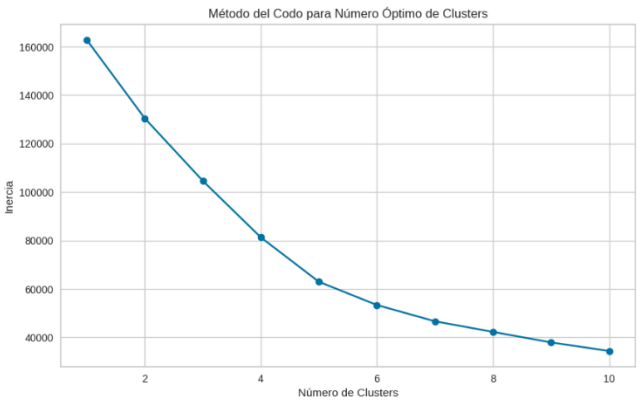
Una vez que hemos obtenido el codo y la cantidad ideal de clusters que vamos a usar generamos este código:

inertia = []

for n in range(1, 11):

kmeans = KMeans(n_clusters=n, random_state=0)

kmeans.fit(data_scaled)



Cluster ideal

inertia.append(kmeans.inertia_)

 # Graficar el método del codo

plt.figure(figsize=(10, 6))

plt.plot(range(1, 11), inertia, marker='o')

plt.title('Método del Codo para Número Óptimo de Clusters')

plt.xlabel('Número de Clusters')

plt.ylabel('Inercia')

plt.grid(True)

plt.show()

El código realiza un análisis para determinar el número óptimo de clusters en un conjunto de datos utilizando el algoritmo KMeans. Primero, define un rango de posibles números de clusters, desde 1 hasta 14. Luego, para cada valor en este rango, se crea un modelo KMeans con 20 inicializaciones diferentes para garantizar una solución robusta y se entrena con los datos estandarizados. Después de ajustar el modelo, se registra la inercia, que mide la suma de las

distancias cuadradas de los puntos a sus centroides dentro de cada cluster.

A continuación, se crea una figura para visualizar cómo cambia la inercia en función del número de clusters. La gráfica resultante muestra la evolución de la varianza intra-cluster total, permitiendo identificar visualmente el punto de "codo", donde añadir más clusters deja de proporcionar una reducción significativa en la inercia. Este punto ayuda a determinar el número óptimo de clusters para el modelo, equilibrando la precisión y la simplicidad. Finalmente, se muestran los resultados en un gráfico, facilitando la interpretación del análisis.

El gráfico muestra la evolución de la inercia, o varianza intra-cluster total, en función del número de clusters en un análisis de KMeans. La inercia es una medida de la suma de las distancias cuadradas de los puntos a sus centroides dentro de cada cluster. En este gráfico, el eje vertical representa la inercia, mientras que el eje horizontal muestra el número de clusters (k).

Observando el gráfico, podemos notar que a medida que el número de clusters aumenta, la inercia disminuye. Esto se debe a que más clusters permiten que los puntos estén más cerca de sus centroides, reduciendo la varianza dentro de los clusters. Inicialmente, la disminución de la inercia es bastante pronunciada, lo que indica que agregar más clusters mejora significativamente la cohesión dentro de los grupos.

Con esta información ya confirmada vamos a usar este código para continuar con nuestro análisis:

```
optimal_clusters = 3

kmeans = KMeans(n_clusters=optimal_clusters, random_state=0)

clusters = kmeans.fit_predict(data_scaled)


# Agregar los clusters a los datos originales

data['Cluster'] = clusters
```

El código realiza un análisis de clustering utilizando el algoritmo KMeans, configurado con un número óptimo de clusters. Primero, se define el número de clusters óptimos como 3. Luego, se crea una instancia del modelo KMeans, configurada para identificar tres clusters. Para asegurar la reproducibilidad de los resultados, se establece una semilla aleatoria (random_state).

El modelo se entrena con los datos estandarizados (data_scaled) y, una vez ajustado, predice los clusters a los que pertenece cada punto de datos. Las etiquetas de los clusters resultantes se almacenan en la variable clusters. Finalmente, estas etiquetas de clusters se añaden al DataFrame original (data) en una nueva columna llamada 'Cluster'. Esto permite identificar y analizar a qué cluster pertenece cada punto de datos dentro del contexto de los datos originales, facilitando tanto el análisis como la visualización de los resultados del clustering.

Vamos a visualizar este análisis para ir viendo los resultados de nuestro modelo de clustering mediante el siguiente código:

```
!pip install plotly

import plotly.express as px # Import the plotly.express library
```

```
data['income'] = data['income'].apply(lambda x: 1 if x == '>50K' else 0)

# Selección de características relevantes

data_for_clustering = data[['age', 'income', 'capital.gain', 'hours.per.week']]


# Normalización de los datos

data_normalized = pd.DataFrame(normalize(data_for_clustering), columns=data_for_clustering.columns)


kmedia_4 = KMeans(n_clusters=4, init="k-means++", n_init=10, max_iter=300,
                  tol=0.0001, random_state=111, algorithm="elkan")


# Entrenamiento del modelo KMeans

kmedia_4.fit(data_normalized)

labels_4k = kmedia_4.labels_

centroides = kmedia_4.cluster_centers_


# Agregar etiquetas de clusters al DataFrame original

data["clusters_4k"] = labels_4k

fig = px.scatter_3d(data, x="age", y="capital.gain", z="hours.per.week", color="clusters_4k", width=800, height=800)

fig.show()
```

El código comienza instalando la librería plotly y luego importando plotly.express para la visualización. A continuación, convierte la columna income en valores numéricos, asignando 1 para ingresos mayores a 50K y 0 para ingresos menores o iguales a 50K. Se seleccionan las características relevantes para el análisis, que incluyen age, income, capital.gain y hours.per.week. Estos datos se normalizan para asegurar que todas las características tengan una escala comparable.

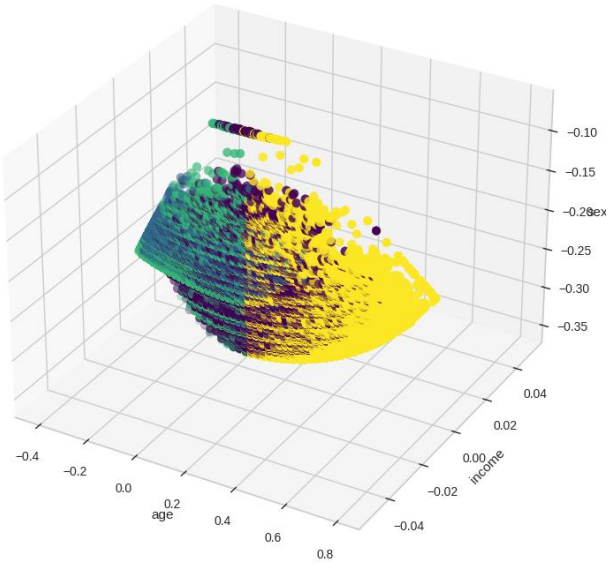
Luego, se configura y entrena un modelo KMeans para identificar cuatro clusters en los datos normalizados. El modelo se inicializa con el método "k-means++" para mejorar la convergencia, se especifica que debe realizar 10 inicializaciones para asegurar robustez y se establece una semilla aleatoria para garantizar reproducibilidad. Una vez entrenado, el modelo asigna etiquetas de cluster a cada punto de datos, que se almacenan en la variable labels_4k. Los centroides de los clusters también se calculan.

Estas etiquetas de clusters se agregan al DataFrame original como una nueva columna llamada clusters_4k. Finalmente, se crea una visualización 3D utilizando plotly.express para mostrar los clusters en función de age, capital.gain y hours.per.week, con los puntos coloreados según el cluster al que pertenecen. La figura se configura con un tamaño específico y se muestra en una ventana interactiva, permitiendo explorar visualmente cómo se han agrupado los datos.

Para tener una visualización estática de los resultados de nuestro análisis de clustering, vamos a utilizar el siguiente código. Este enfoque nos permitirá observar la distribución de los clusters de manera clara y detallada, sin la necesidad de interacciones adicionales.

fig = plt.figure(figsize=(9,9)) # Crear la figura

ax = fig.add_subplot(111, projection="3d") # Añadir un subplot 3D a la figura



x = data["age"]

y = data["income"]

 # Replace 'sex' with a numerical column

z = data["hours.per.week"]

ax.set_xlabel("age")

ax.set_ylabel("income")

ax.set_zlabel("hours.per.week") # Update z-axis label

 # Usar las etiquetas de los clusters para colorear los puntos

ax.scatter(x, y, z, c=labels_4k, s=50, cmap="viridis")

plt.show()

El código crea una visualización 3D estática de los datos utilizando la librería matplotlib. Primero, se configura una figura de tamaño 9x9 pulgadas y se añade un subplot con proyección 3D. Se asignan las variables x, y y z a las columnas age, income y sex del DataFrame, respectivamente.

Se etiquetan los ejes del gráfico 3D como age, income y sex. Luego, se crea un gráfico de dispersión 3D donde cada punto representa un dato en el espacio tridimensional definido por age, income y sex. Los puntos se colorean según las etiquetas de clusters almacenadas en labels_4k, utilizando el mapa de colores "viridis" y se establece un tamaño de 50 para los puntos.

Los ejes del gráfico representan las variables age, income y sex, que han sido normalizadas. Esto se puede inferir por los valores que oscilan en un rango cercano a cero. Los puntos en el gráfico están coloreados según las etiquetas de clusters generadas por el modelo KMeans, lo que permite visualizar cómo se han agrupado los datos en función de estas variables.

Esta visualización es muy útil para identificar posibles mejoras en el modelo de clustering. Por ejemplo, puede sugerir la necesidad de ajustar el número de clusters o modificar el preprocesamiento de los datos. La claridad de los clusters en el espacio 3D indica que el modelo ha capturado patrones significativos en los datos, lo que es positivo. Sin embargo, la superposición de algunos puntos sugiere que puede haber áreas donde la clasificación no es perfecta y podría mejorarse.

El gráfico proporciona una representación visual clara y detallada de cómo se han agrupado los datos en clusters, lo que facilita la interpretación de los resultados del análisis de clustering. Además, destaca áreas donde el modelo funciona bien y donde puede requerir ajustes. Esta información es valiosa para

entender la estructura subyacente de los datos y mejorar el modelo para obtener resultados más precisos.

CONCLUSIONES

- **Reducción Efectiva de Dimensionalidad con PCA:**
  La aplicación de PCA permitió reducir la dimensionalidad del conjunto de datos original a dos componentes principales, PC1 y PC2, que juntos explican aproximadamente el 45% de la varianza total en los datos. Esto indica que los componentes principales capturan de manera eficiente una gran parte de la información contenida en las variables originales, facilitando el análisis y la visualización de los datos sin perder mucha información.
- **Formación de Clusters Coherentes con K-Means:**
  El análisis de clustering utilizando K-Means en el espacio reducido de las componentes principales resultó en la formación de clusters bien definidos. La visualización de los clusters en el espacio de PC1 y PC2 mostró una clara separación entre los diferentes grupos, indicando que el algoritmo K-Means fue capaz de identificar patrones y agrupaciones significativas en los datos.
- **Diferencias Demográficas y Económicas**:
  Los análisis PCA y K-Means destacaron diferencias significativas en términos demográficos y económicos. Por ejemplo, se observó que individuos con niveles educativos más altos y mayores ganancias de capital tienden a agruparse en diferentes clusters comparados con aquellos con menores niveles educativos y pérdidas de capital. Estas diferencias subrayan la importancia de factores educativos y económicos en la caracterización de la población y pueden ser utilizados para desarrollar estrategias específicas en políticas públicas y económicas.

RECOMENDACIONES

1. **Optimización de Horas de Trabajo:** Las horas trabajadas por semana (hours.per.week) también son una variable importante. Las políticas laborales podrían enfocarse en optimizar el número de horas de trabajo para mejorar el bienestar de los empleados y la productividad general.
2. **Desarrollo de Políticas Inclusivas:** La presencia de múltiples clusters indica la existencia de subgrupos dentro de la población con características distintas. Se recomienda desarrollar políticas inclusivas que consideren las necesidades específicas de estos subgrupos, especialmente en términos de educación y oportunidades económicas.
3. **Fomentar la Diversidad Económica:** La variabilidad en ingresos y ocupaciones sugiere la necesidad de fomentar la diversidad económica. Se recomienda apoyar una amplia gama de industrias y sectores para crear un entorno económico más resiliente y dinámico.

4. **Análisis Continuo de Datos:**
   La aplicación de PCA y K-Means ha demostrado ser útil para identificar patrones significativos en los datos. Se recomienda realizar análisis continuos de datos utilizando estas y otras técnicas de Machine Learning para adaptar las políticas y estrategias en tiempo real.
5. **Mejorar la Recolección de Datos:** La precisión de los análisis depende de la calidad de los datos. Se sugiere mejorar los métodos de recolección de datos para obtener información más precisa y completa, lo que permitirá realizar análisis más detallados y tomar decisiones mejor fundamentadas.