Informe del TP1

Grupo 2 - Célestine Raveneau, Florian Escaffre, Juan Gomez, Luis Condori

Ejercicio 1

Análisis Exploratorio:

en las tablas se describen viajes en la ciudad de nueva york a lo largo de 3 meses

a) VendorID: Un código que indica el proveedor de TPEP que proporcionó el registro. 1= Creative Mobile Technologies, LLC; 2= VeriFone Inc,**tipo:** int 32

b) tpep_pickup_datetime: La fecha y hora en que se activó el medidor, **tipo:** date.

c) tpep_dropoff_datetime: La fecha y hora en que se desconectó el medidor, **tipo:** date.

d) Passenger_count: El número de pasajeros en el vehículo. Este es un valor ingresado por el conductor, **tipo**: inicial float64, pero luego transformado a int8.

e) Trip_distance: La distancia recorrida del viaje en millas reportada por el taxímetro, **tipo:** .float64

f) PULocationID: Zona de Taxi TLC en la que estaba activado el taxímetro, **tipo**: int32.

g) DOLocationID: Zona de Taxi TLC en la que se desactivó el taxímetro, **tipo:** int32.

h) RateCodeID: El código de tarifa final vigente al final del viaje (pasar a entero). 1= Standard rate 2=JFK 3=Newark 4=Nassau or Westchester 5=Negotiated fare 6=Group ride, **tipo**: inicial float64, pero luego transformado a int8.

i) Store_and_fwd_flag: Este indicador indica si el registro de viaje se mantuvo en la memoria del vehículo antes de enviarlo al proveedor, también conocido como "almacenar y reenviar". porque el vehículo no tenía conexión con el servidor. Y = viaje de ida y vuelta N = no es un viaje de ida y vuelta, **Tipo:** inicialmente objet luego pasado a int8 (codificado como 0 y 1)

j) Payment_type: Un código numérico que indica cómo el pasajero pagó el viaje. 1= Credit card 2= Cash 3= No charge 4= Dispute 5= Unknown 6= Voided trip, **tipo:**int64

k) Fare_amount: La tarifa por tiempo y distancia calculada por el taxímetro, **tipo** float64.

l) Extra: Extras y recargos varios. Actualmente, esto sólo incluye los cargos de 0,50y1,00 por hora pico y por noche, **tipo** float64.

m) MTA_tax: Impuesto MTA de $0.50 que se activa automáticamente según la tarifa medida en uso, **tipo:**float64.

ñ) Improvement_surcharge: Recargo por mejora de $0.30 en viajes evaluados al bajar la bandera. El recargo por mejora comenzó a cobrarse en 2015 (pasar a entero), **tipo:** float64. .

o) Tip_amount: Monto de la propina: este campo se completa automáticamente para las propinas de tarjetas de crédito. Las propinas en efectivo no están incluidas, **tipo:** float64 (aca hay varios en 0.0 pero no seria un outsider ya que el cliente no le dejo propina al conductor o se lo dio en efectivo).

p) Tolls_amount: Importe total de todos los peajes pagados en el viaje, **tipo:** float64.

q) Total_amount: El importe total cobrado a los pasajeros. No incluye propinas en efectivo, **tipo:** float64.

r) Congestion_Surcharge: Monto total cobrado en el viaje por el recargo por congestión del Estado de Nueva York, **tipo:** float64.

s) Airport_fee: $1.25 para recogida únicamente en los aeropuertos LaGuardia y John F. Kennedy, **tipo:** float64

Comentar los features más destacables: tipo de dato, qué representa y por qué se

destacan. Listar hipótesis o supuestos que tomaron.

Preprocesamiento de Datos

Detallar las tareas más importantes que realizaron sobre el dataset, les dejamos

algunas preguntas cómo guía:

1. ¿Se eliminaron columnas (Nombre de la columna y motivo de eliminación?

no

2. ¿Detectaron correlaciones interesantes (entre qué variables y qué

coeficiente)?

negativa con menor a 0,5

- vendorID inversamente relacionada con la variable extra

con coeficiente mayor que 0,8

- **tip_amount** relacionada con **fare_amount**
- **trip_distance** relacionada con **fare_amount** y **total_amount**
- **fare_amount** relacionada con **total_amount**
- **mta_tax** relacionada con **improvement_surcharge**
- **pickup_hour_range** relacionada con **dropoff_hour_range**
- **pickup_date_range** relacionada con **dropoff_date_range**

tabla de Abril

index	VendorID	passenger_count	trip_distance	RatecodeID	store_and_fwd_flag	PULocationID	DOLocationID	payment_type	fare_amount	extra	mta_tax	tip_amount	tolls_amount	improvement_surcharge	total_amount	congestion_surcharge	Airport_fee	pickup_hour_range	dropoff_hour_range	pickup_date_range	dropoff_date_range
VendorID	1,000	0,062	0,026	-0,119	-0,084	-0,002	0,002	0,019	0,024	-0,570	-0,053	0,040	0,015	-0,056	0,028	-0,015	0,022	0,016	0,012	0,000	
passenger_count	0,062	1,000	0,046	-0,030	-0,005	-0,013	-0,010	0,026	0,048	-0,033	-0,010	0,015	0,034	0,002	0,045	0,007	0,025	0,023	0,022	-0,024	
trip_distance	0,026	0,046	1,000	0,071	-0,008	-0,135	-0,097	-0,008	0,868	0,170	-0,051	0,572	0,641	0,012	0,871	-0,246	0,643	-0,004	-0,011	0,004	
RatecodeID	-0,119	-0,030	0,071	1,000	-0,005	-0,045	-0,037	-0,031	0,084	-0,066	-0,008	-0,046	0,067	0,005	0,055	-0,223	-0,007	-0,038	-0,032	-0,004	
store_and_fwd_flag	-0,084	-0,005	-0,008	-0,005	1,000	0,002	0,002	-0,001	-0,003	0,045	0,005	-0,007	-0,006	0,006	-0,005	0,003	-0,004	0,014	0,013	0,028	
PULocationID	-0,002	-0,013	-0,135	-0,045	0,002	1,000	0,091	-0,022	-0,117	-0,038	0,011	-0,069	-0,085	0,008	-0,117	0,110	-0,149	0,014	0,020	0,006	
DOLocationID	0,002	-0,010	-0,097	-0,037	0,002	0,091	1,000	-0,026	-0,088	-0,007	0,030	-0,046	-0,065	0,007	-0,083	0,107	-0,047	0,027	0,029	0,007	
payment_type	0,019	0,026	-0,008	-0,031	-0,001	-0,022	-0,026	1,000	-0,081	-0,079	-0,366	-0,383	-0,031	-0,399	-0,153	-0,289	0,002	-0,020	-0,019	-0,013	
fare_amount	0,024	0,048	0,868	0,084	-0,003	-0,117	-0,088	-0,081	1,000	0,159	0,053	0,580	0,626	0,198	0,980	-0,166	0,582	-0,001	-0,002	0,020	
extra	-0,570	-0,033	0,170	-0,066	0,045	-0,038	-0,007	-0,079	0,159	1,000	0,149	0,186	0,219	0,139	0,238	0,054	0,327	0,177	0,173	0,012	
mta_tax	-0,053	-0,010	-0,051	-0,008	0,005	0,011	0,030	-0,366	0,053	0,149	1,000	0,007	-0,097	0,896	0,074	0,567	0,058	0,015	0,014	0,001	
tip_amount	0,040	0,015	0,572	-0,046	-0,007	-0,069	-0,046	-0,383	0,580	0,186	0,007	1,000	0,464	0,083	0,702	-0,046	0,394	0,029	0,026	0,022	
tolls_amount	0,015	0,034	0,641	0,067	-0,006	-0,085	-0,065	-0,031	0,626	0,219	-0,097	0,464	1,000	0,048	0,696	-0,116	0,450	-0,009	-0,005	0,006	
improvement_surcharge	-0,056	0,002	0,012	0,005	0,006	0,008	0,007	-0,399	0,198	0,139	0,896	0,083	0,048	1,000	0,217	0,528	0,070	0,005	0,006	0,002	
total_amount	0,028	0,045	0,871	0,055	-0,005	-0,117	-0,083	-0,153	0,980	0,238	0,074	0,702	0,696	0,217	1,000	-0,120	0,611	0,019	0,017	0,022	
congestion_surcharge	-0,015	0,007	-0,246	-0,223	0,003	0,110	0,107	-0,289	-0,166	0,054	0,567	-0,046	-0,116	0,528	-0,120	1,000	-0,318	0,019	0,025	0,001	
Airport_fee	0,022	0,025	0,643	-0,007	-0,004	-0,149	-0,047	0,002	0,582	0,327	0,058	0,394	0,450	0,070	1,000	-0,318	1,000	0,030	0,019	0,019	
pickup_hour_range	0,016	0,023	-0,004	-0,038	0,014	0,014	0,027	-0,020	-0,001	0,177	0,015	0,029	-0,009	0,005	0,019	0,019	0,030	1,000	0,923	0,004	
dropoff_hour_range	0,012	0,022	-0,011	-0,032	0,013	0,020	0,029	-0,019	-0,002	0,173	0,014	0,026	-0,005	0,006	0,017	0,025	0,019	0,923	1,000	0,004	
pickup_date_range	0,000	-0,024	0,004	-0,004	0,028	0,006	0,007	-0,013	0,020	0,012	0,001	0,022	0,006	0,002	0,022	0,001	0,019	0,004	0,004	1,000	
dropoff_date_range	0,000	-0,024	0,004	-0,003	0,028	0,006	0,007	-0,013	0,020	0,012	0,001	0,022	0,006	0,002	0,022	0,001	0,018	0,004	0,003	0,998	

Tabla de Mayo

index	VendorID	passenger_count	trip_distance	RatecodeID	store_and_fwd_flag	PULocationID	DOLocationID	payment_type	fare_amount	extra	mta_tax	tip_amount	tolls_amount	improvement_surcharge	total_amount	congestion_surcharge	Airport_fee	pickup_hour_range	dropoff_hour_range	pickup_date_range	dropoff_date_range
VendorID	1,000	0,068	0,026	-0,119	-0,117	-0,003	0,001	0,017	0,024	-0,553	-0,053	0,040	0,013	-0,056	0,028	-0,015	0,024	0,022	0,018	0,004	0,004
passenger_count	0,068	1,000	0,033	-0,029	-0,008	-0,014	-0,010	0,018	0,035	-0,037	-0,006	0,009	0,025	0,004	0,033	0,010	0,014	0,021	0,020	0,024	0,024
trip_distance	0,026	0,033	1,000	0,072	-0,008	-0,136	-0,105	-0,010	0,855	0,177	-0,058	0,569	0,641	0,012	0,860	-0,245	0,610	-0,008	-0,011	0,012	0,012
RatecodeID	-0,119	-0,029	0,072	1,000	-0,005	-0,044	-0,036	-0,030	0,082	-0,066	-0,009	-0,045	0,070	0,004	0,054	-0,220	-0,008	-0,040	-0,033	-0,004	-0,004
store_and_fwd_flag	-0,117	-0,008	-0,008	-0,005	1,000	0,001	0,003	0,005	-0,004	0,064	0,006	-0,010	-0,005	0,006	-0,006	0,004	-0,005	-0,003	-0,001	-0,014	-0,014
PULocationID	-0,003	-0,014	-0,136	-0,044	0,001	1,000	0,093	-0,024	-0,116	-0,041	0,011	-0,068	-0,085	0,007	-0,116	0,111	-0,147	0,008	0,013	-0,010	-0,010
DOLocationID	0,001	-0,010	-0,105	-0,036	0,003	0,093	1,000	-0,026	-0,093	-0,010	0,031	-0,049	-0,071	0,007	-0,088	0,110	-0,049	0,027	0,029	-0,009	-0,009
payment_type	0,017	0,018	-0,010	-0,030	0,005	-0,024	-0,026	1,000	-0,084	-0,083	-0,370	-0,378	-0,032	-0,405	-0,155	-0,294	-0,002	-0,021	-0,019	0,013	0,013
fare_amount	0,024	0,035	0,855	0,082	-0,004	-0,116	-0,093	-0,084	1,000	0,170	0,042	0,577	0,621	0,198	0,980	-0,163	0,547	-0,006	-0,003	0,004	0,004
extra	-0,553	-0,037	0,177	-0,066	0,064	-0,041	-0,010	-0,083	0,170	1,000	0,150	0,199	0,232	0,142	0,252	0,052	0,357	0,184	0,180	-0,019	-0,019
mta_tax	-0,053	-0,006	-0,058	-0,009	0,006	0,011	0,031	-0,370	0,042	0,150	1,000	0,001	-0,112	0,890	0,064	0,572	0,053	0,014	0,013	-0,004	-0,004
tip_amount	0,040	0,009	0,569	-0,045	-0,010	-0,068	-0,049	-0,378	0,577	0,199	0,001	1,000	0,465	0,083	0,701	-0,042	0,382	0,025	0,026	-0,007	-0,007
tolls_amount	0,013	0,025	0,641	0,070	-0,005	-0,085	-0,071	-0,032	0,621	0,232	-0,112	0,465	1,000	0,049	0,693	-0,127	0,446	-0,017	-0,008	0,003	0,003
improvement_surcharge	-0,056	0,004	0,012	0,004	0,006	0,007	0,007	-0,405	0,198	0,142	0,890	0,083	0,049	1,000	0,217	0,526	0,066	0,004	0,005	-0,003	-0,003
total_amount	0,028	0,033	0,860	0,054	-0,006	-0,116	-0,088	-0,155	0,980	0,252	0,064	0,701	0,693	0,217	1,000	-0,118	0,583	0,014	0,017	0,001	0,001
congestion_surcharge	-0,015	0,010	-0,245	-0,220	0,004	0,111	0,110	-0,294	-0,163	0,052	0,572	-0,042	-0,127	0,526	-0,118	1,000	-0,295	0,017	0,022	-0,009	-0,009
Airport_fee	0,024	0,014	0,610	-0,008	-0,005	-0,147	-0,049	-0,002	0,547	0,357	0,053	0,382	0,446	0,066	0,583	-0,295	1,000	0,029	0,021	0,010	0,010
pickup_hour_range	0,022	0,021	-0,008	-0,040	-0,003	0,008	0,027	-0,021	-0,006	0,184	0,014	0,025	-0,017	0,004	0,014	0,017	0,029	1,000	0,920	-0,005	-0,005
dropoff_hour_range	0,018	0,020	-0,011	-0,033	-0,001	0,013	0,029	-0,019	-0,003	0,180	0,013	0,026	-0,008	0,005	0,017	0,022	0,021	0,920	1,000	-0,005	-0,005
pickup_date_range	0,004	0,024	0,012	-0,004	-0,014	-0,010	-0,009	0,013	0,004	-0,019	-0,004	-0,007	0,003	-0,003	0,001	-0,009	0,010	-0,005	-0,005	1,000	0,998
dropoff_date_range	0,004	0,024	0,012	-0,004	-0,014	-0,010	-0,009	0,013	0,005	-0,019	-0,004	-0,006	0,004	-0,003	0,001	-0,009	0,011	-0,005	-0,006	0,998	1,000

Tabla de Junio

index	VendorID	passenger_count	trip_distance	RatecodeID	store_and_fwd_flag	PULocationID	DOLocationID	payment_type	fare_amount	extra	mta_tax	tip_amount	tolls_amount	improvement_surcharge	total_amount	congestion_surcharge	Airport_fee	pickup_hour_range	dropoff_hour_range	pickup_date_range	dropoff_date_range
VendorID	1,000	0,068	0,005	-0,114	-0,104	-0,003	0,001	0,013	0,001	-0,553	-0,055	0,041	0,014	-0,059	0,002	-0,019	0,026	0,019	0,015	-0,006	-0,006
passenger_count	0,068	1,000	0,004	-0,028	-0,006	-0,015	-0,010	0,020	0,005	-0,034	-0,009	0,011	0,029	0,002	0,005	0,007	0,019	0,018	0,017	0,002	0,002
trip_distance	0,005	0,004	1,000	0,008	0,000	-0,016	-0,013	-0,001	0,010	0,020	-0,008	0,066	0,075	0,000	0,012	-0,032	0,073	-0,001	-0,002	0,001	0,001
RatecodeID	-0,114	-0,028	0,008	1,000	-0,004	-0,042	-0,033	-0,028	0,007	-0,064	-0,011	-0,040	0,065	0,004	0,006	-0,210	-0,006	-0,040	-0,033	-0,002	-0,002
store_and_fwd_flag	-0,104	-0,006	0,000	-0,004	1,000	-0,001	0,002	0,012	0,000	0,059	0,004	-0,010	0,000	0,005	0,000	-0,002	0,001	-0,001	-0,001	0,013	0,013
PULocationID	-0,003	-0,015	-0,016	-0,042	-0,001	1,000	0,091	-0,022	-0,011	-0,047	0,010	-0,069	-0,082	0,006	-0,014	0,109	-0,154	0,008	0,014	-0,012	-0,012
DOLocationID	0,001	-0,010	-0,013	-0,033	0,002	0,091	1,000	-0,026	-0,009	-0,012	0,028	-0,047	-0,065	0,007	-0,010	0,106	-0,052	0,026	0,029	-0,008	-0,008
payment_type	0,013	0,020	-0,001	-0,028	0,012	-0,022	-0,026	1,000	-0,007	-0,081	-0,371	-0,370	-0,032	-0,409	-0,017	-0,294	-0,004	-0,022	-0,020	0,010	0,010
fare_amount	0,001	0,005	0,010	0,007	0,000	-0,011	-0,009	-0,007	1,000	0,016	0,004	0,052	0,055	0,018	1,000	-0,016	0,052	-0,001	-0,001	0,001	0,001
extra	-0,553	-0,034	0,020	-0,064	0,059	-0,047	-0,012	-0,081	0,016	1,000	0,151	0,191	0,232	0,144	0,028	0,048	0,347	0,178	0,175	0,018	0,018
mta_tax	-0,055	-0,009	-0,008	-0,011	0,004	0,010	0,028	-0,371	0,004	0,151	1,000	-0,002	-0,108	0,889	0,007	0,575	0,057	0,014	0,014	-0,006	-0,006
tip_amount	0,041	0,011	0,066	-0,040	-0,010	-0,069	-0,047	-0,370	0,052	0,191	-0,002	1,000	0,453	0,082	0,078	-0,056	0,392	0,025	0,022	0,004	0,004
tolls_amount	0,014	0,029	0,075	0,065	0,000	-0,082	-0,065	-0,032	0,055	0,232	-0,108	0,453	1,000	0,050	0,077	-0,121	0,447	-0,016	-0,011	0,009	0,009
improvement_surcharge	-0,059	0,002	0,000	0,004	0,005	0,006	0,007	-0,409	0,018	0,144	0,889	0,082	0,050	1,000	0,025	0,529	0,075	0,004	0,005	-0,002	-0,002
total_amount	0,002	0,005	0,012	0,006	0,000	-0,014	-0,010	-0,017	1,000	0,028	0,007	0,078	0,077	0,025	1,000	-0,015	0,068	0,001	0,001	0,001	0,001
congestion_surcharge	-0,019	0,007	-0,032	-0,210	-0,002	0,109	0,106	-0,294	-0,016	0,048	0,575	-0,056	-0,121	0,529	-0,015	1,000	-0,322	0,018	0,024	-0,011	-0,011
Airport_fee	0,026	0,019	0,073	-0,006	0,001	-0,154	-0,052	-0,004	0,052	0,347	0,057	0,392	0,447	0,075	0,068	-0,322	1,000	0,027	0,014	0,019	0,019
pickup_hour_range	0,019	0,018	-0,001	-0,040	-0,001	0,008	0,026	-0,022	-0,001	0,178	0,014	0,025	-0,016	0,004	0,001	0,018	0,027	1,000	0,918	-0,013	-0,013
dropoff_hour_range	0,015	0,017	-0,002	-0,033	-0,001	0,014	0,029	-0,020	-0,001	0,175	0,014	0,022	-0,011	0,005	0,001	0,024	0,014	0,918	1,000	-0,013	-0,013
pickup_date_range	-0,006	0,002	0,001	-0,002	0,013	-0,012	-0,008	0,010	0,001	0,018	-0,006	0,004	0,009	-0,002	0,001	-0,011	0,019	-0,013	-0,013	1,000	1,000
dropoff_date_range	-0,006	0,002	0,001	-0,002	0,013	-0,012	-0,008	0,009	0,001	0,018	-0,006	0,004	0,009	-0,002	0,001	-0,011	0,019	-0,013	-0,013	1,000	0,998

3. ¿Generaron nuevos features?

solo se separaron los tipo dates en fecha y hora por separado para poder discretizarlas

4. ¿Encontraron valores atípicos?¿Cuáles?¿Qué técnicas utilizaron y qué decisiones tomaron?

había muchos nan's,como son menos del 3% en cada tabla, los borre directamente

el datos atípico que encontre es en la columna Passenger_count de las 3 tablas que tenia varios ceros, se reemplazo con la moda

5. ¿Qué columnas tenían datos faltantes?

¿En qué proporción? ¿Qué se hizo con estos registros?

Mes	**Abril**	**Mayo**	**Junio**
Iniciales	3288250	3513649	3307234
Luego del procesamiento	3197560	3411853	3207347
Nulos	90690	101796	99887
% de nulos:	2,758%	2,897%	3,020%

Visualizaciones

Plantear las preguntas de investigación y mostrar los gráficos que permitieron llegar a las respuestas. Seleccionar aquellos que permitan entender cómo se distribuyen los datos, cómo se relacionan la variables etc.

Grafico de violin de trip_distance vs fare_amount

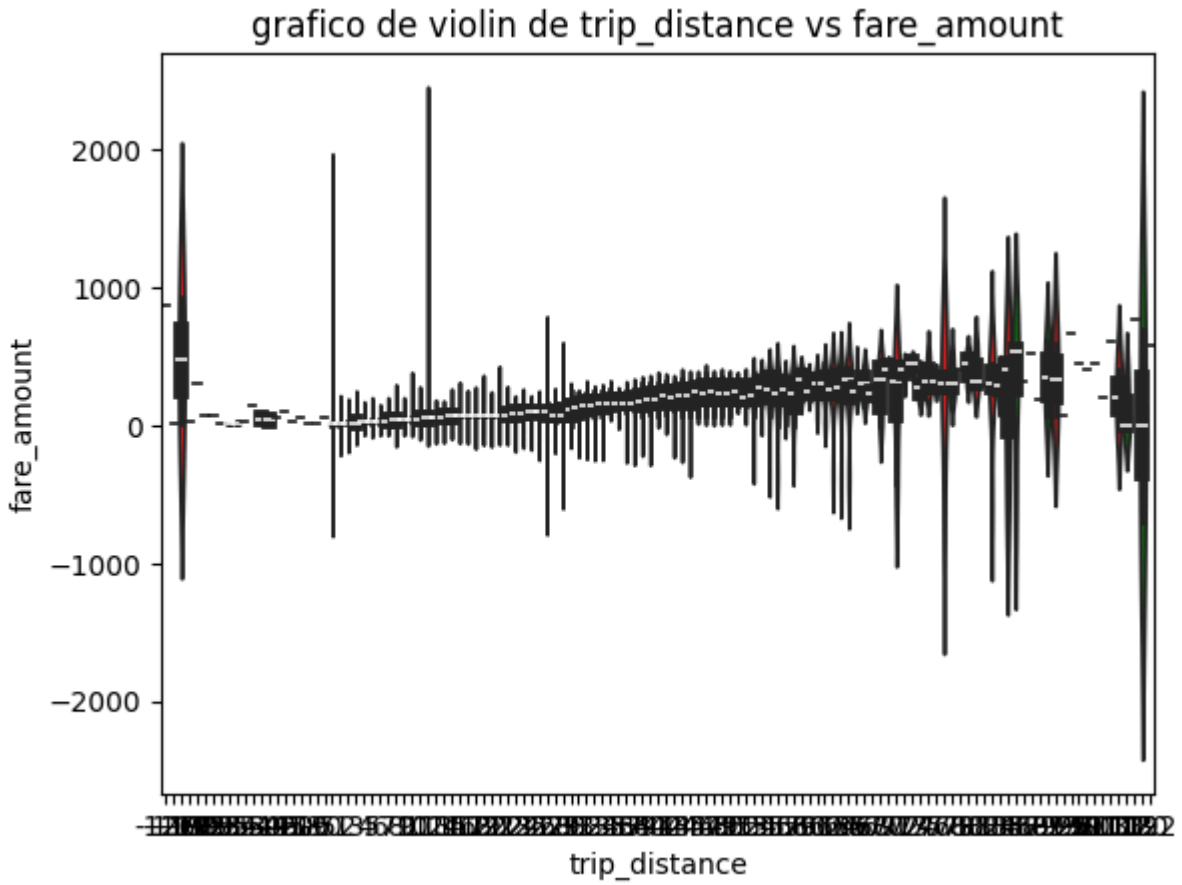


Grafico de violin de trip_distance vs total_amount

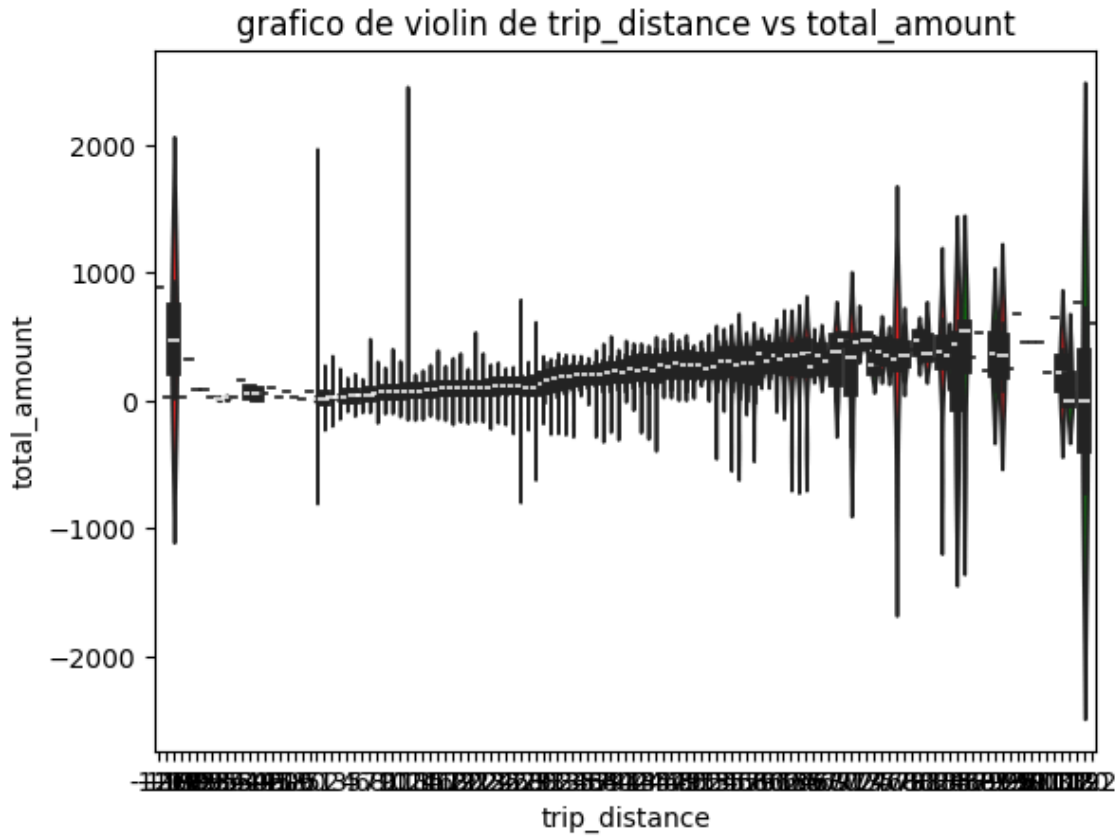
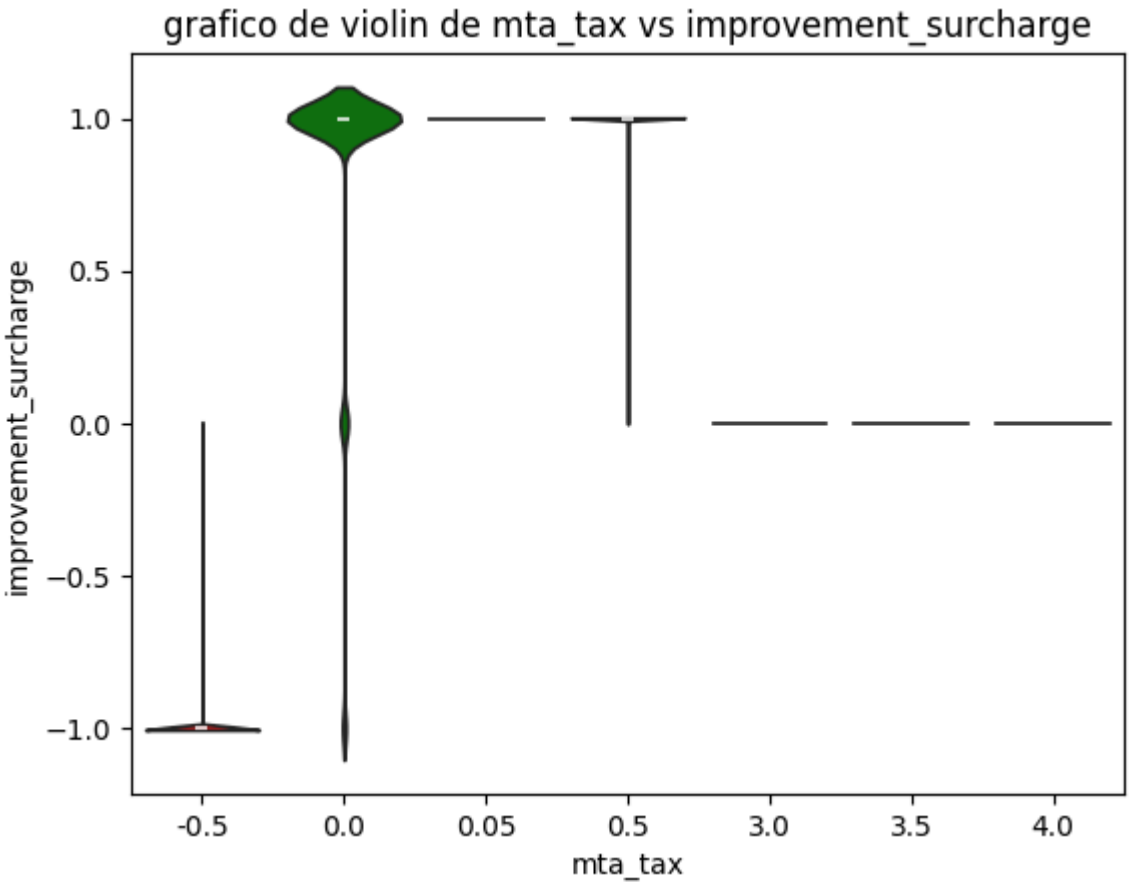


grafico de violin de mta_tax vs improvement_surcharge



Comentar brevemente qué se está visualizando en cada caso y por qué los eligieron.

Ejercicio 2

Descripción del Dataset

El conjunto de datos utilizado en este trabajo proviene de observaciones meteorológicas diarias tomadas durante aproximadamente 10 años en diversas estaciones de Australia. El objetivo es predecir si lloverá al día siguiente (variable RainTomorrow) a partir de datos climáticos del día actual.

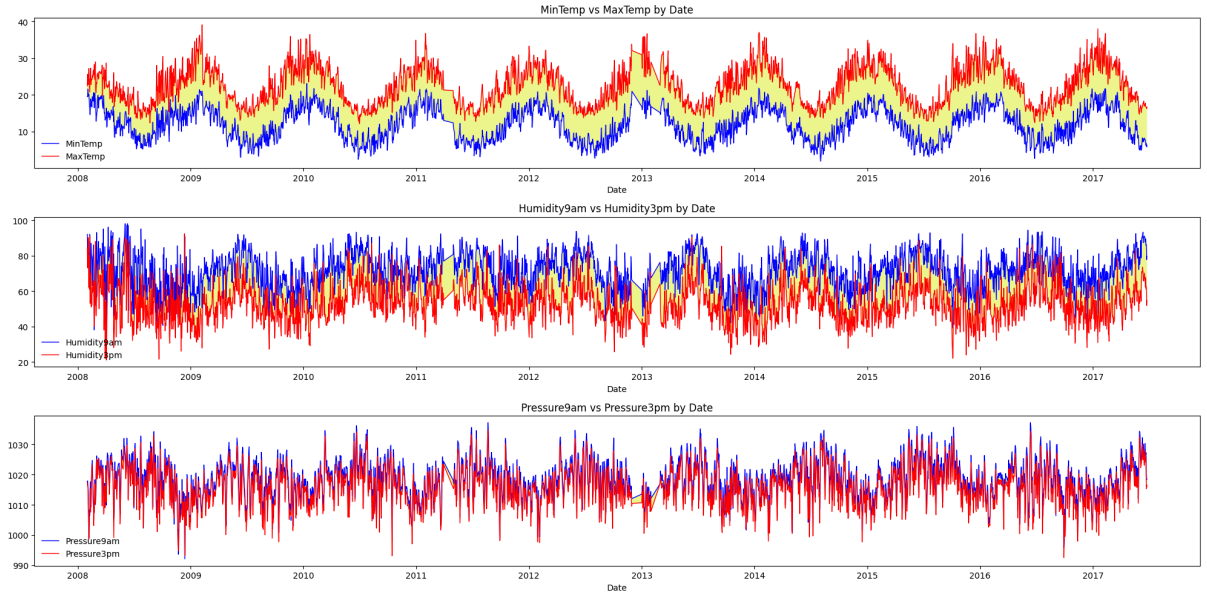
Conjuntos de datos empleados:

- "weatherAUS.csv": Contiene observaciones meteorológicas diarias, como temperatura, viento, presión, y lluvia.
- "aus_coordinates.xlsx": Incluye las regiones de Australia junto con sus coordenadas geográficas.

Los dos conjuntos de datos fueron combinados para incluir únicamente las regiones de Nuevo Gales del Sur y Victoria. El dataset resultante tiene 74,492 observaciones y 25 variables, de las cuales 8 son categóricas y 17 son numéricas

Variables destacables:

- Variables de temperatura: MinTemp, MaxTemp, Temp9am, Temp3pm — Estas variables proporcionan información clave sobre el rango térmico durante el día.
- Lluvia: Rainfall, RainToday, RainTomorrow — Cruciales para el objetivo de la predicción. RainToday y RainTomorrow son variables categóricas que indican si ha llovido o se espera lluvia.
- Viento: WindGustDir, WindGustSpeed, WindDir9am — Información sobre la dirección y velocidad del viento, que puede influir en las precipitaciones.
- Presión y nubes: Pressure9am, Pressure3pm, Cloud9am, Cloud3pm — Indicadores atmosféricos importantes para modelar condiciones meteorológicas.
- Localización: Location, Region, Coordinates — Utilizadas para situar las observaciones geográficamente.



El gráfico anterior muestra los rango en que varían las variables Temp,Humidity, Presure

a) Análisis Exploratorio y preprocesamiento de datos

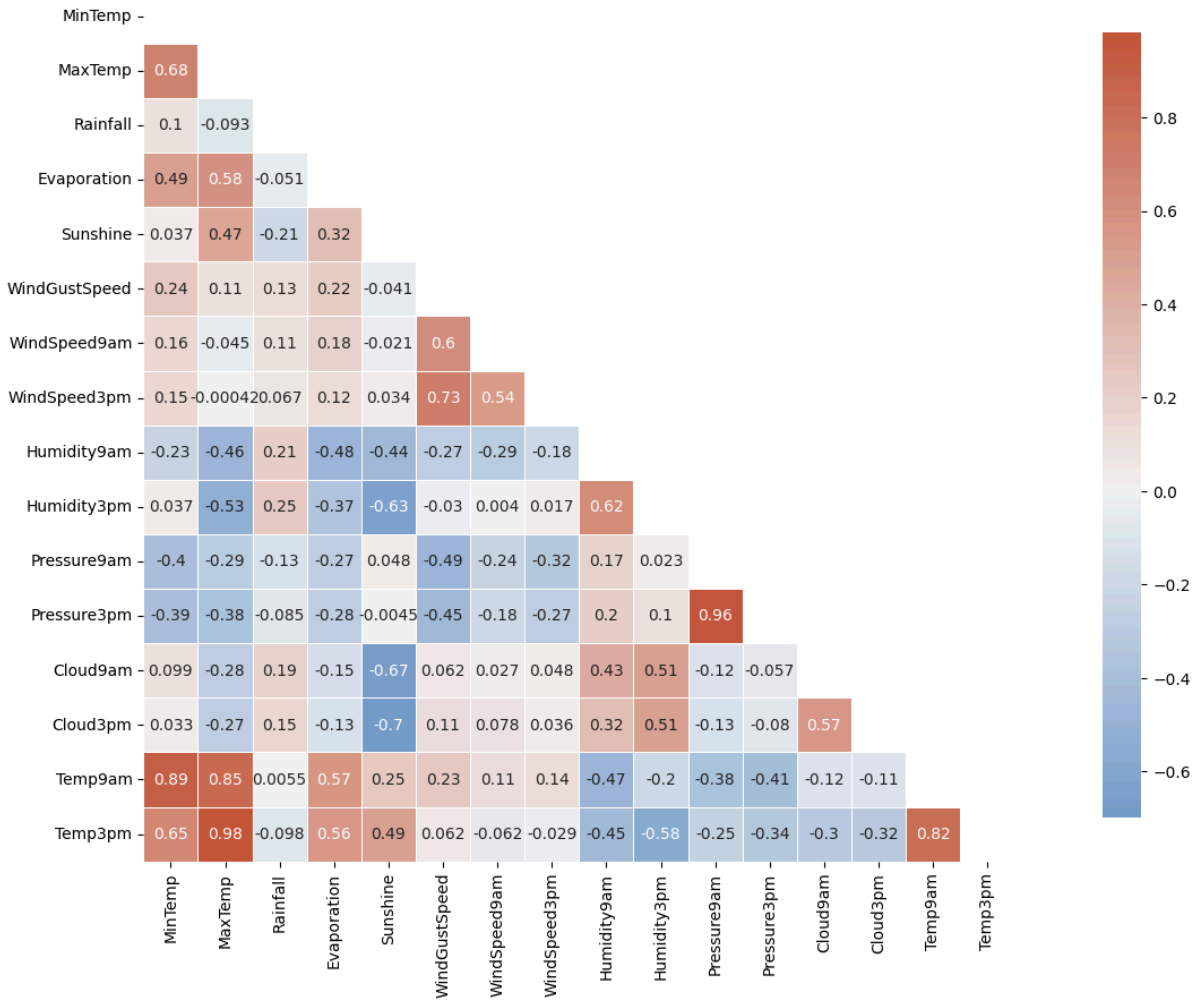
Valores faltantes

Variables Categóricas:

- Se identificaron datos faltantes en las variables de viento (WindGustDir, WindDir9am, WindDir3pm) entre un 3% y 8%. Inicialmente se imputaron agrupando por Region y Location, pero se encontró que usar SimpleImputer con moda mejora las métricas en los modelos.
- Los registros con datos faltantes en la variable objetivo RainTomorrow (alrededor de 2%) fueron eliminados para evitar problemas de sesgo y fuga de datos.
- Se aplicó Binary Encoding a las variables categóricas restantes para reducir la dimensionalidad sin perder información importante.

Variables Numéricas:

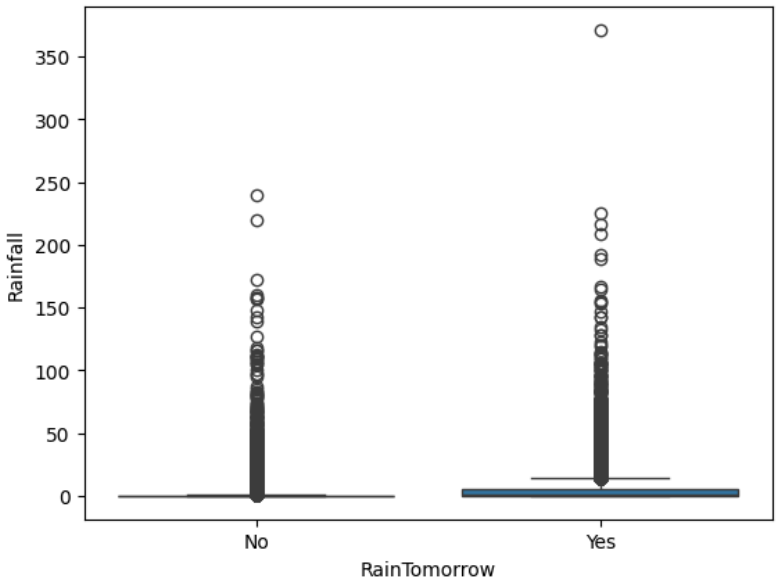
- Varias variables como Evaporation, Sunshine, Pressure9am, Pressure3pm, Cloud9am, y Cloud3pm mostraron datos faltantes significativos. Se utilizaron distintas técnicas de imputación (mean, mode, y KNN) para completar estos valores según el patrón de cada variable.
- Las variables de viento (WindGustSpeed, WindSpeed9am, WindSpeed3pm) se imputaron mediante métodos de regresión para mantener la coherencia entre ellas.
- Finalmente, se aplicó escalado y normalización a las variables numéricas para asegurar que todas las características contribuyan de manera equitativa al modelo StandardScaler.



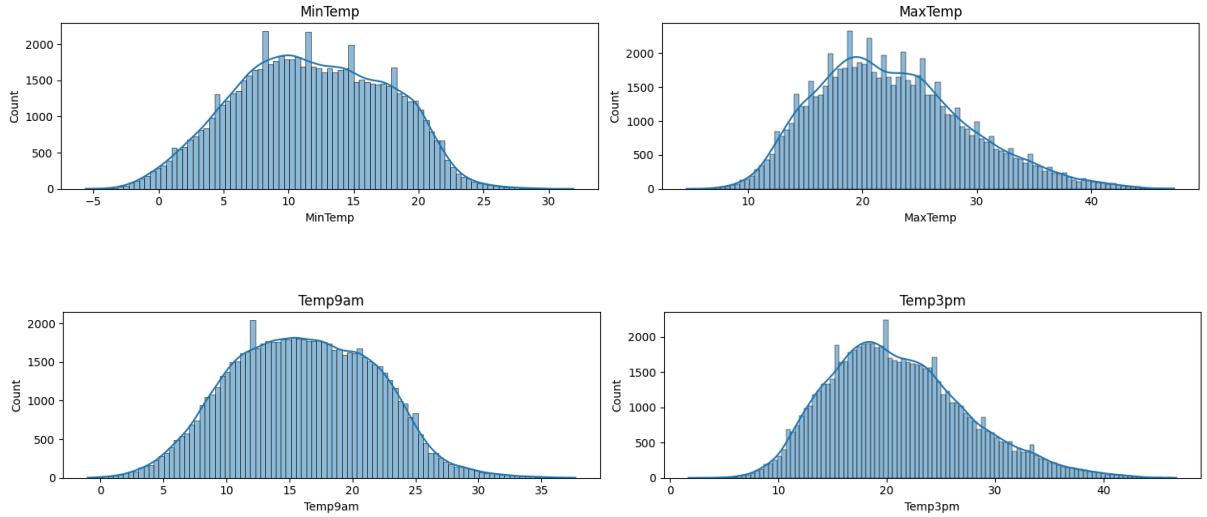
Análisis valores atípicos

- Rainfall mostró valores máximos de hasta 371 mm, mucho mayores al tercer cuartil (0.8 mm), sugiriendo la presencia de valores atípicos extremos.
- Otras variables como WindGustSpeed, WindSpeed9am, WindSpeed3pm y variables de temperatura presentaban outliers menores al 1.5%.

La variable Rainfall es la única con valores faltantes altos 17%, Evaporation y las velocidades del viento están entre 1% a 2%. Se utilizó RobustScaler para reducir el impacto de valores extremos. En primera instancia se eliminó la variable Rainfall, provocando una disminución en las métricas por tal motivo se decidió dejarla.



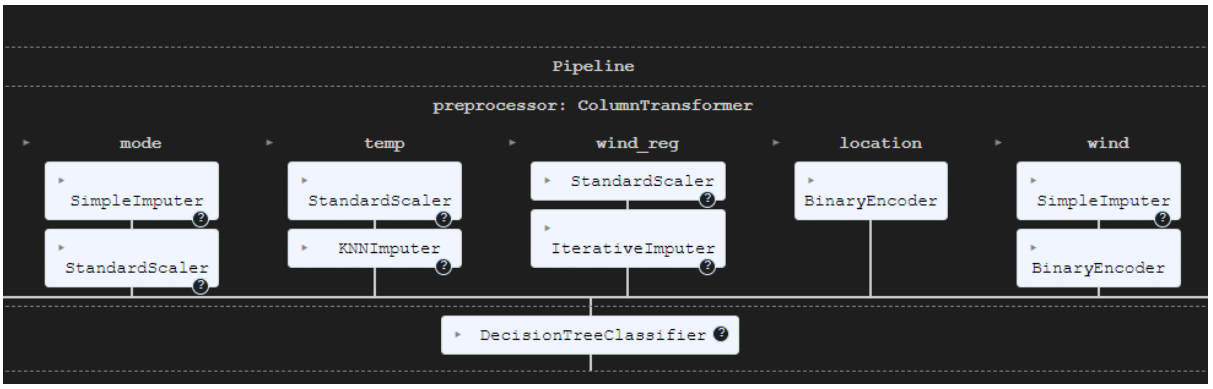
Las demás variables tienen valores faltantes menores al 1%, se aplica Standard Scaler para mantener una distribución normal, como las temperaturas (MinTemp, MaxTemp, Temp9am, Temp3pm).



Transformación de variables

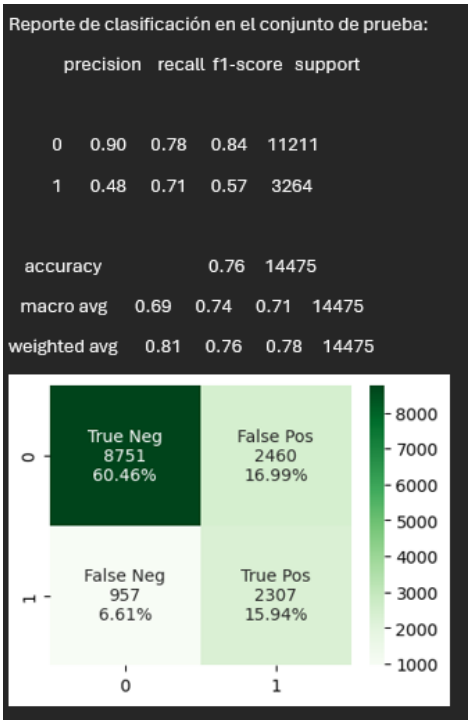
Para manejar todas estas transformaciones de manera eficiente, se implementó un Pipeline de Preprocesamiento, ColumnTransformer que aplica las técnicas apropiadas a cada tipo de variable.

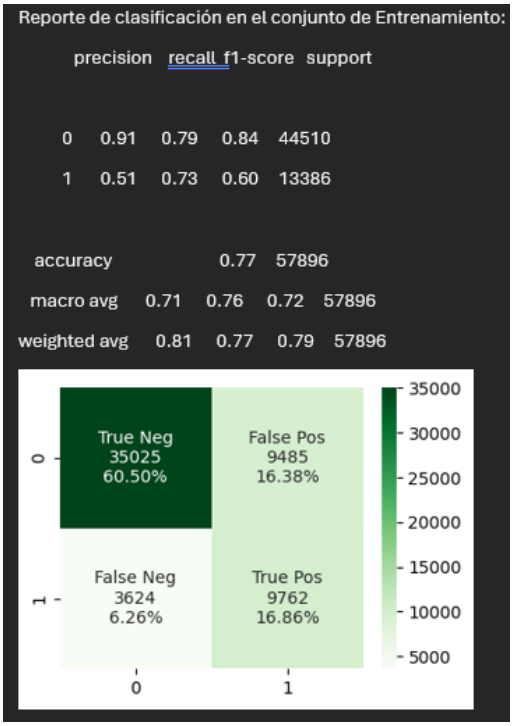
Se transforman las variables Date en año, mes , día y Coordinates en altitud y longitud.



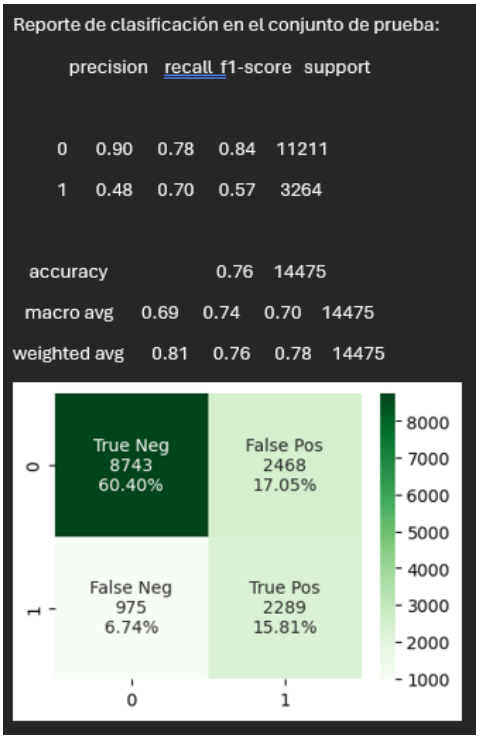
Para mejorar el rendimiento de los modelos, se realizaron diversas pruebas, como la imputación de datos mediante agrupación (clase que diseñe para agrupar por Region - Location- valor), media y moda. Dado que el conjunto de datos estaba desbalanceado (con un 77% de 'No' y un 23% de 'Yes'), se consideró la posibilidad de generar datos sintéticos mediante MICE con una tasa de remuestreo del 30%. Además, se probó utilizar un selector de características (SelectKBest con f_classif, k=15) para seleccionar las más relevantes. Sin embargo, estas estrategias no resultaron efectivas para mejorar el rendimiento de los modelos, por lo que finalmente se decidió no utilizarlas en el pipeline final.

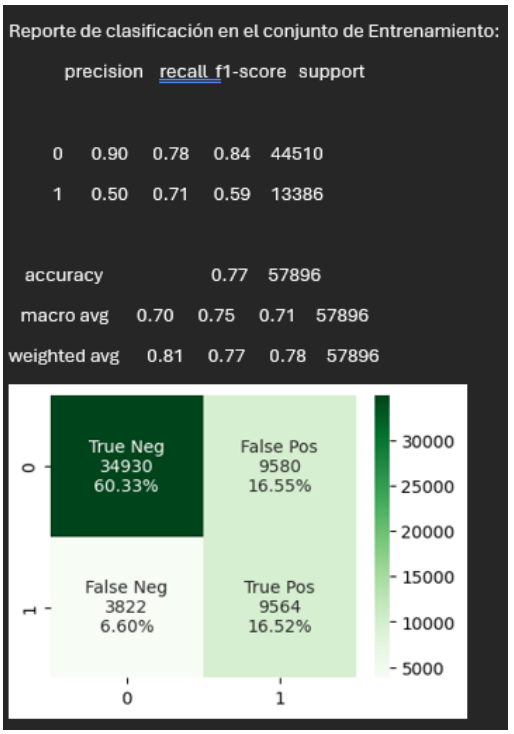
- Con MICE y SelectKBest
  - DecisionTreeClassifier



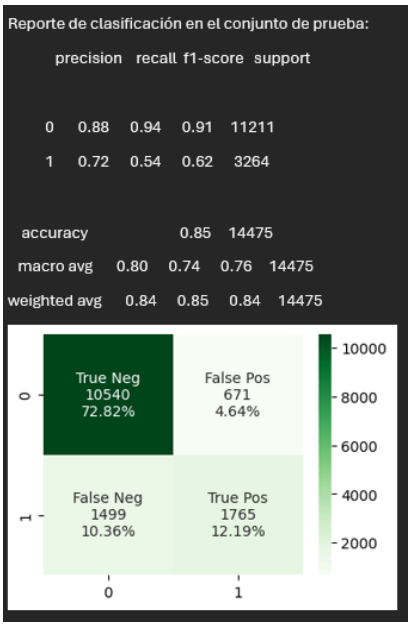


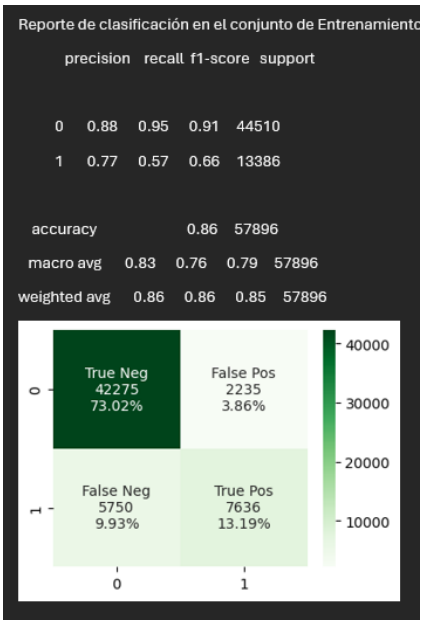
- RandomForestClassifier





- XGBoost





b) Entrenamiento y Predicción

Modelo1: DecisionTreeClassifier

- Parámetros de optimización

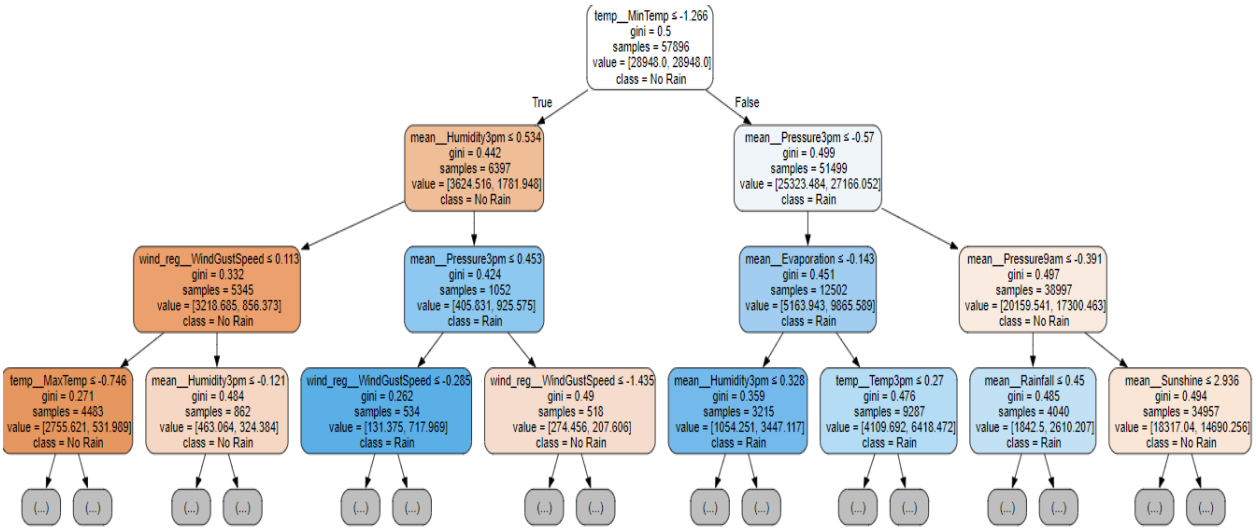
  Se utilizó “StratifiedKFold” con 5 folds porque mantiene la proporción de clases ('No' y 'Yes') en cada división del conjunto de datos.

  La métrica que utilizo es “recall”, dado que el objetivo del Problema es predecir si lloverá (clase "yes"), nos enfocamos en mejorar el recall de la clase "yes" para minimizar los falsos negativos.

```
hiperparams_decision_tree = {

    'classifier__class_weight': ['balanced', None],

    'classifier__criterion': ['gini', 'entropy', 'log_loss'],

    'classifier__max_depth': [None, 1, 3, 5, 8, 10, 20, 30],

    'classifier__max_features': [None, 'sqrt', 'log2'],

    'classifier__min_samples_leaf': [1, 2, 4],

    'classifier__min_samples_split': [2, 5, 10],

}
```

La métrica más importante max_depth, por que si no se coloca un rango apropiado el modelo sobreajusta.

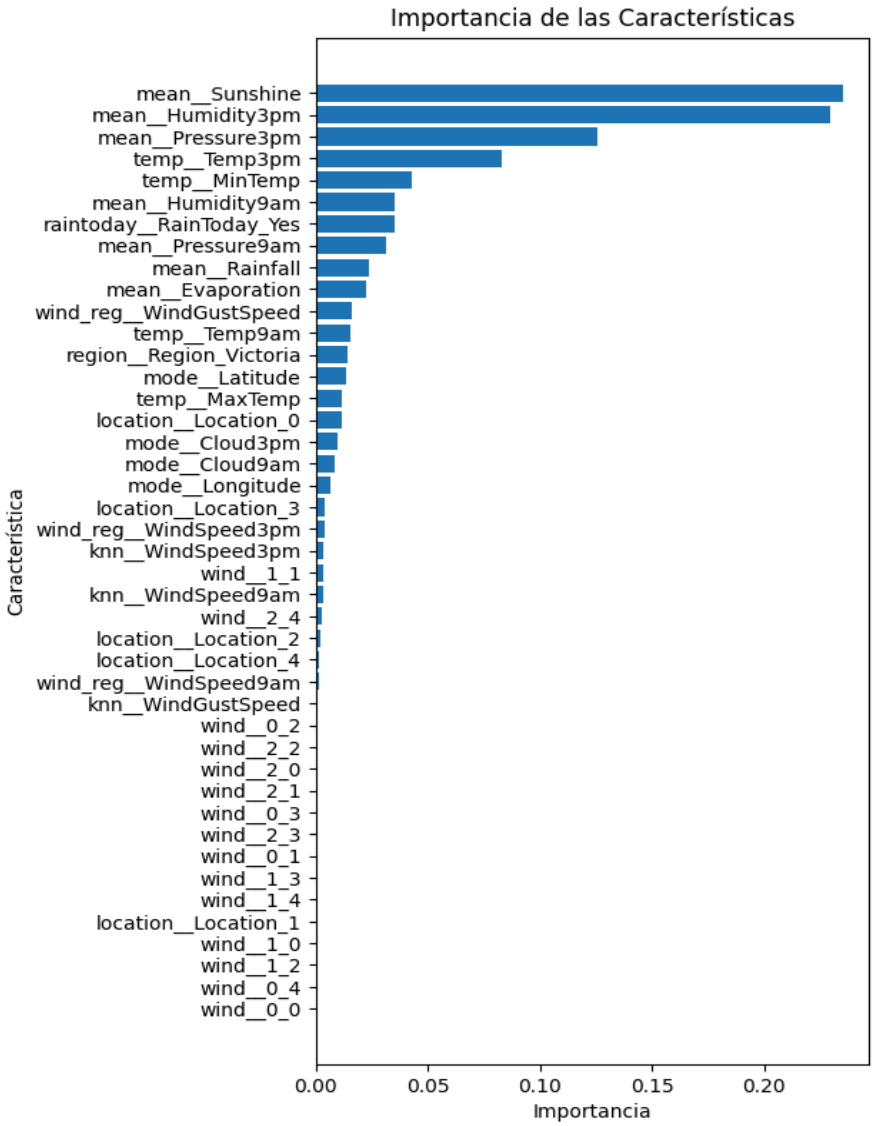
- Árbol de decisión



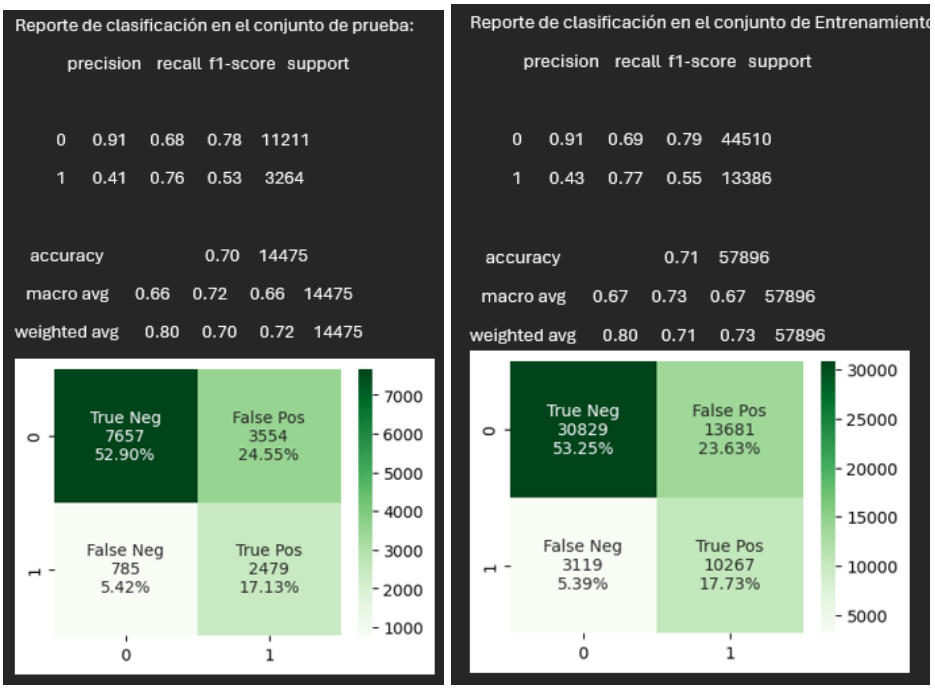
El árbol de decisión comienza evaluando la variable temp__MinTemp, estableciendo un umbral en -1.27. Si la temperatura mínima es menor o igual a este valor, el árbol continúa dividiendo en función de la humedad a las 3 pm (mean__Humidity3pm). Por ejemplo, si la humedad es baja (<= 0.53), se evalúa la velocidad de las ráfagas de viento (wind_reg__WindGustSpeed) para tomar decisiones adicionales.

Por otro lado, si la temperatura mínima es mayor a -1.27, el árbol se enfoca en la presión atmosférica a las 3 pm (mean__Pressure3pm). Un valor bajo de presión sugiere la necesidad de evaluar otras variables como la evaporación promedio (mean__Evaporation) y la temperatura a las 3 pm (temp__Temp3pm).
Estas son las reglas de decisión del modelo.

- Features más importantes



- Performance conjunto de evaluación y conjunto de entrenamiento



  - Accuracy: El modelo obtuvo un 70% de accuracy en el conjunto de prueba. Esto indica que el 70% de las predicciones totales (días lluviosos y no lluviosos) fueron correctas.
  - Precision: La precisión para la clase No RainTomorrow (días sin lluvia) es alta, 0.91, lo que significa que el modelo es eficaz prediciendo correctamente la clase más frecuente. Sin embargo, la precisión para Yes RainTomorrow (días lluviosos) es baja, 0.41, indicando un número considerable de falsos positivos (predicciones incorrectas de lluvia).
  - Recall: El recall para la clase No RainTomorrow es 0.68, lo que indica que el modelo detecta correctamente el 68% de los días sin lluvia. Para Yes RainTomorrow, el recall es más alto, 0.76, lo que significa que el modelo tiene una buena capacidad para identificar días lluviosos, pero podría estar cometiendo errores al confundir días no lluviosos como lluviosos.
  - F1-Score: El F1-score refleja un equilibrio entre precisión y recall. Para No RainTomorrow, es 0.78, y para Yes RainTomorrow, es 0.53, lo que indica un rendimiento moderado en la detección de días lluviosos.

El rendimiento en entrenamiento y prueba es consistente, mostrando patrones similares en la distribución de verdaderos y falsos positivos y negativos. Esto sugiere que el modelo generaliza bien y no está sobreajustado a los datos de entrenamiento. Aunque el modelo es efectivo al predecir días sin lluvia (alta precisión en la clase No RainTomorrow), tiene dificultades con la clase menos frecuente (Yes RainTomorrow), lo que resulta en una baja precisión para identificar días lluviosos. Este comportamiento es esperable debido al desbalance en las clases.

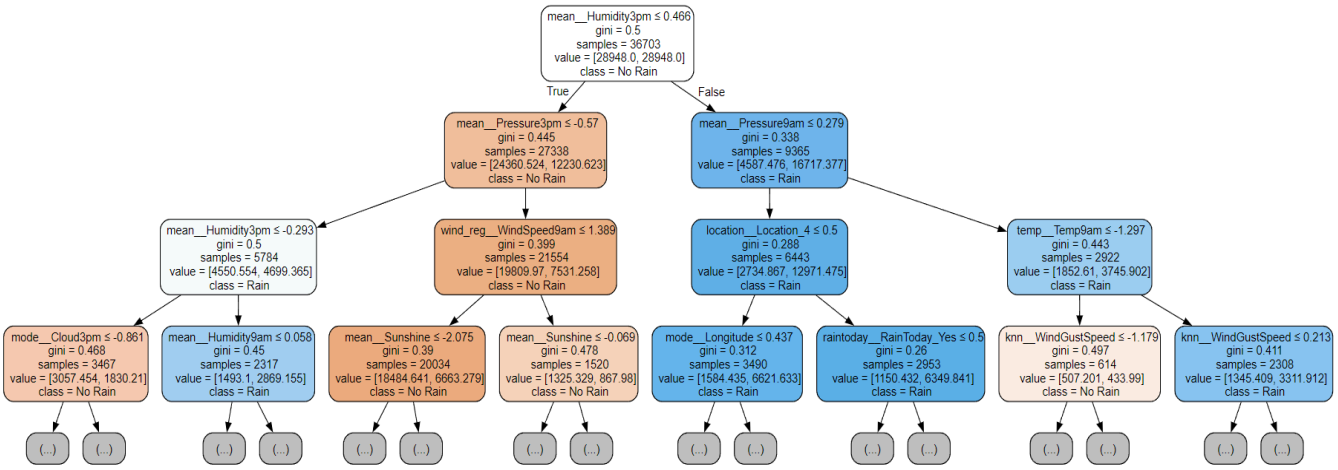
Modelo 2: RandomForestClassifier

- Parametros de optimizacion

```
param_grid_random_fores = {

    'classifier__n_estimators': [100, 200, 300, 500],

    'classifier__criterion': ['gini', 'entropy'],

    'classifier__max_depth': [None, 1, 3, 5, 8, 10, 20, 30, 40, 50],

    'classifier__min_samples_split': [2, 5, 10],

    'classifier__min_samples_leaf': [1, 2, 4],

    'classifier__class_weight': ['balanced', 'balanced_subsample'],

}
```

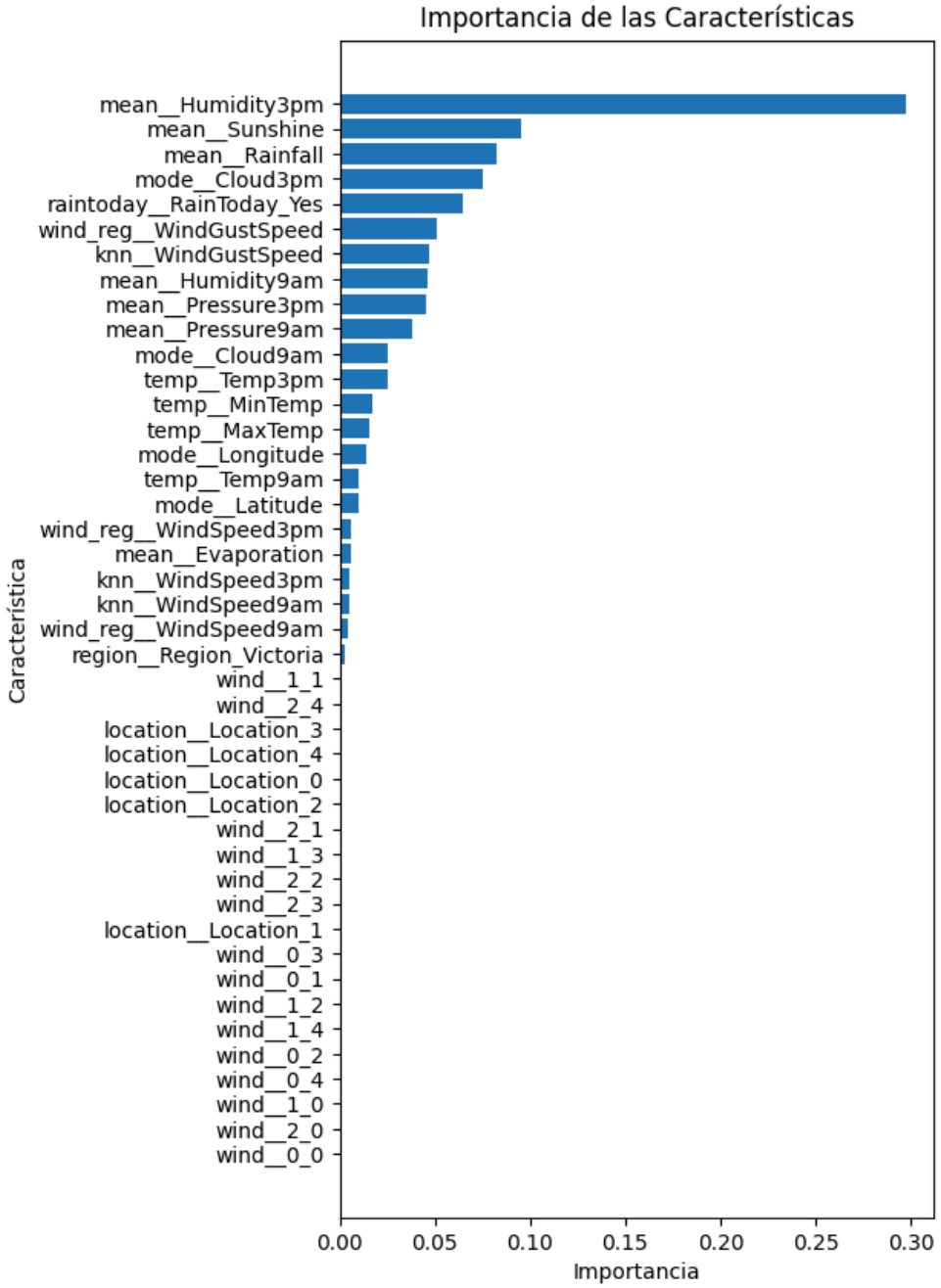
max_depth es la métrica más importante junto con criterion.

- Conformación final de uno de los árboles generados



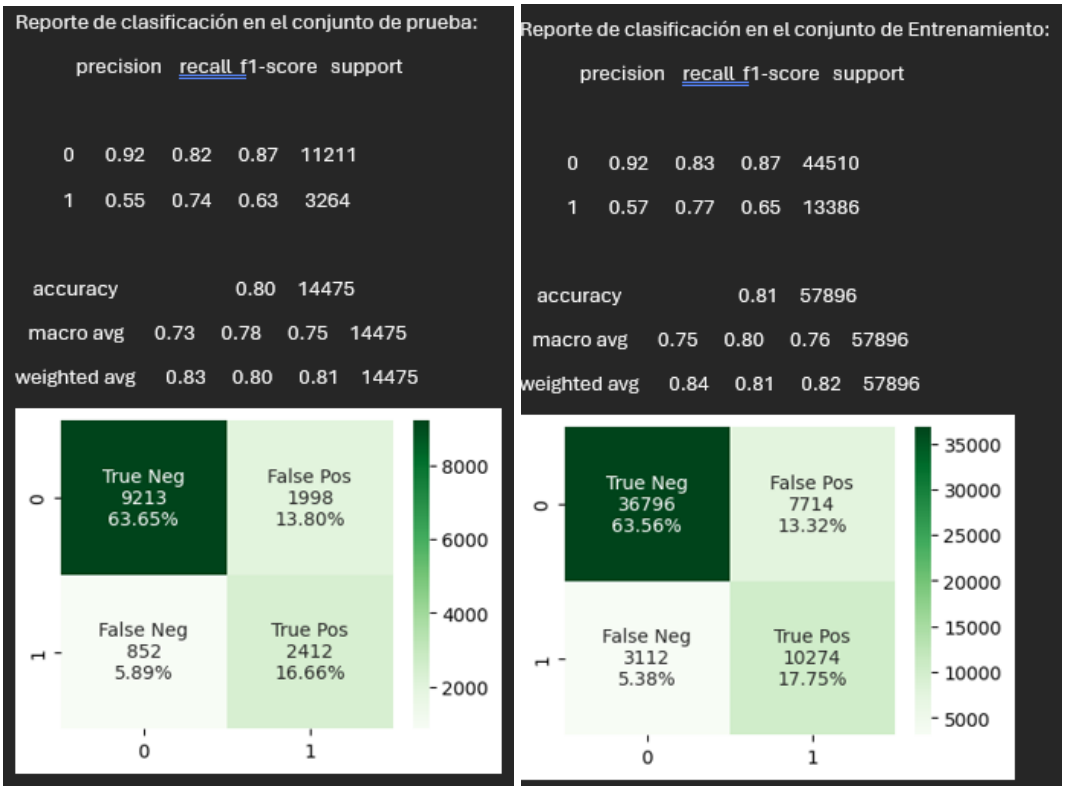
El modelo toma decisiones basadas en condiciones relacionadas con la humedad y la presión atmosférica. La primera bifurcación se basa en el valor de `mean__Humidity3pm`, que divide el conjunto de datos en dos ramas principales: una para valores menores o iguales a 0.47 y otra para valores superiores. En la primera rama, el siguiente criterio es `mean__Pressure3pm`, que se utiliza para dividir aún más, indicando que la presión atmosférica a las 3 p.m. es importante para predecir si lloverá. Otras condiciones incluyen la humedad a las 9 a.m. y el viento, lo que sugiere que la combinación de estas variables es clave para identificar patrones de lluvia al estar primeras en el árbol.

- Features más importantes



Al compararlo con DecisionTreeClassifier en el ranking de los primeros 5, la variable Rainfall (Precipitación en las 24 horas hasta las 9am) entra en escena.

- Performance conjunto de evaluación y conjunto de entrenamiento



  - Accuracy: El modelo obtuvo un 80% de accuracy, mostrando una mejora con respecto al modelo de árbol de decisión (70%).
  - Precision y Recall:
    - Clase No RainTomorrow: Alta precisión (0.92) y recall (0.82), lo que significa que el modelo identifica bien los días sin lluvia, con pocos falsos positivos.
    - Clase Yes RainTomorrow: Precisión moderada (0.55) y recall (0.74), mostrando una mejora en la detección de días lluviosos comparado con el árbol de decisión.
  - F1-Score: Mejora en el equilibrio entre precisión y recall, lo que indica que el modelo Random Forest maneja mejor la detección de días lluviosos que el árbol de decisión.

El rendimiento es consistente entre el entrenamiento y la prueba, con valores similares para las métricas clave. Esto indica que el modelo generaliza bien sin señales de sobreajuste. La mejora en el recall de la clase Yes RainTomorrow indica que el modelo Random Forest es más efectivo para identificar días lluviosos en comparación con el árbol de decisión.

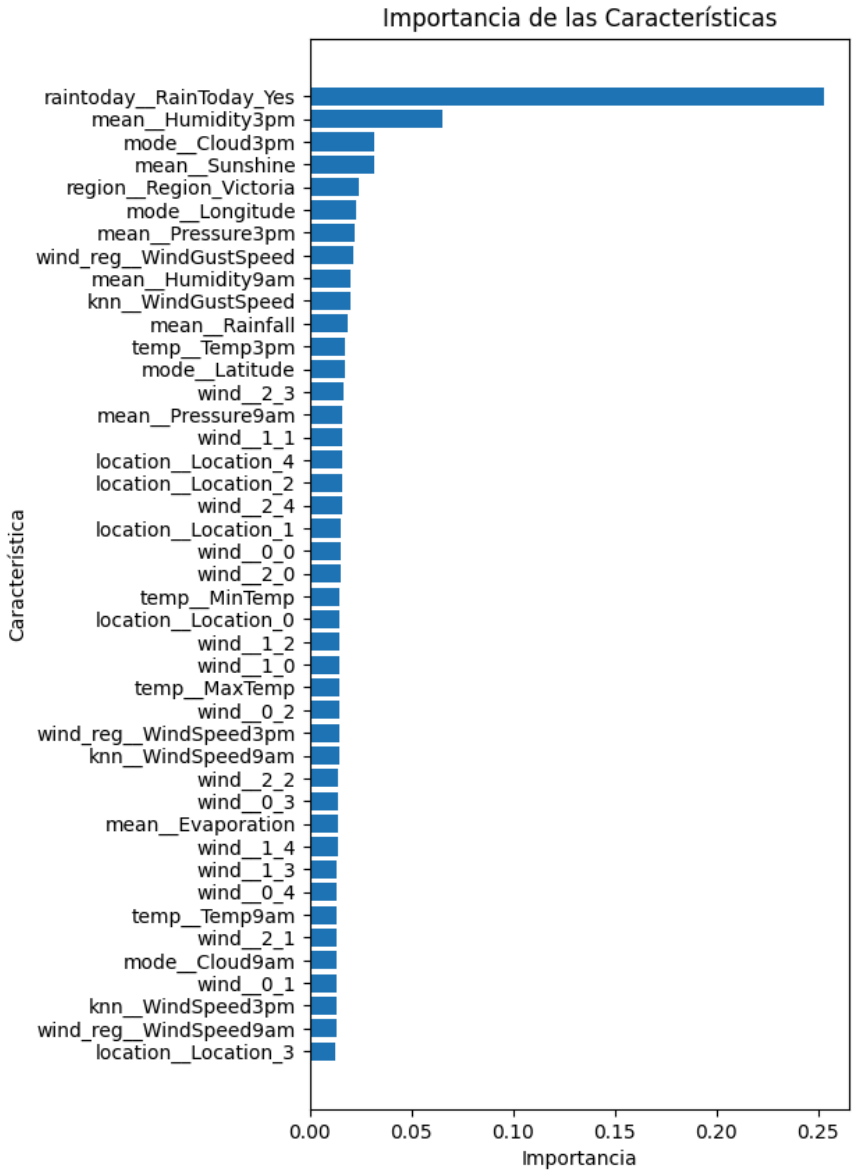
Modelo 3: XGBoost

- Parametros de optimizacion

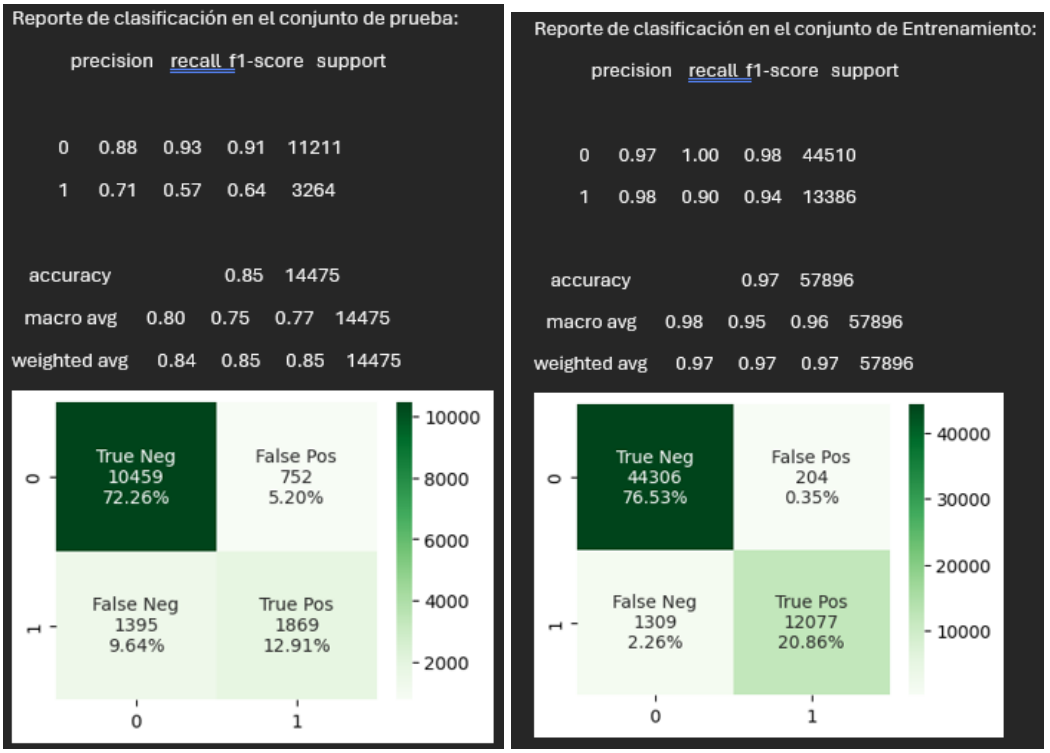
```
hiperparams_xgboost = {

    'classifier__n_estimators': [100, 200, 300, 500],
```

```
    'classifier__max_depth': [3, 5, 7, 9],

    'classifier__learning_rate': [0.01, 0.1, 0.3],

    'classifier__subsample': [0.5, 0.7, 1],

    'classifier__colsample_bytree': [0.5, 0.7, 1],

    'classifier__gamma': [0, 1, 5],

    'classifier__reg_alpha': [0, 0.1, 1],

    'classifier__reg_lambda': [0, 1, 5],

}
```

- Features más importantes



- Performance conjunto de evaluación y conjunto de entrenamiento



- Accuracy:(85%), representa un incremento significativo frente al Random Forest (80%) y el árbol de decisión (70%).
- Precision, Recall y F1-Score:
  - Clase No RainTomorrow:
    - Precisión: 0.88 - Es ligeramente más baja que la precisión del Random Forest, el modelo sigue siendo confiable al identificar días sin lluvia.
    - Recall: 0.93 - Identificó el 93% de los días sin lluvia, el valor más alto de los tres modelos, comete menos falsos positivos.
    - F1-Score: 0.91 - Buen equilibrio entre precisión y recall para esta clase.
  - Clase Yes RainTomorrow:
    - Precisión: 0.71 - Mejor precisión que el árbol de decisión y el Random Forest, lo que significa que tiene menos falsos positivos al predecir días lluviosos.
    - Recall: 0.57 - Detecta el 57% de los días lluviosos, menor en comparación con el Random Forest.
    - F1-Score: 0.64 - Similar al de Random Forest.

El modelo muestra una mejora notable en la precisión general y la reducción de falsos positivos para días lluviosos en comparación con los otros modelos. Aunque el recall para la clase Yes RainTomorrow es más bajo que en el Random Forest pero aun así sigue destacándose por una mayor accuracy global y menor cantidad de errores de clasificación en los días no lluviosos.

c) Cuadrado de resultado

Modelo	Accuracy (Prueba)	Precision (Yes)	Recall (Yes)	F1-Score (Yes)	TN%	FP%	FN%	TP%
Decision Tree	70%	0.41	0.76	0.53	52.9%	24.5%	5.4%	17.1%
Random Forest	80%	0.55	0.74	0.63	63.7%	13.8%	5.9%	16.7%
XGBoost	85%	0.71	0.57	0.64	72.3%	5.2%	9.6%	12.9%

Se elegiría el modelo XGBoost para predecir si lloverá o no al día siguiente porque:

- Tiene la mayor precisión global (85%), lo que indica un mejor rendimiento general en la clasificación.
- La precisión alta para la clase Yes (71%) significa que genera menos falsos positivos y da resultados más confiables para días lluviosos.
- El recall de las clases Yes es menor en comparación con el Random Forest, la reducción de falsos positivos y el equilibrio general de las métricas hacen que XGBoost sea más eficiente.

XGBoost ofrece un buen equilibrio entre precisión y eficiencia, lo que lo hace ideal para el objetivo de predecir si lloverá al día siguiente.

Ejercicio 3

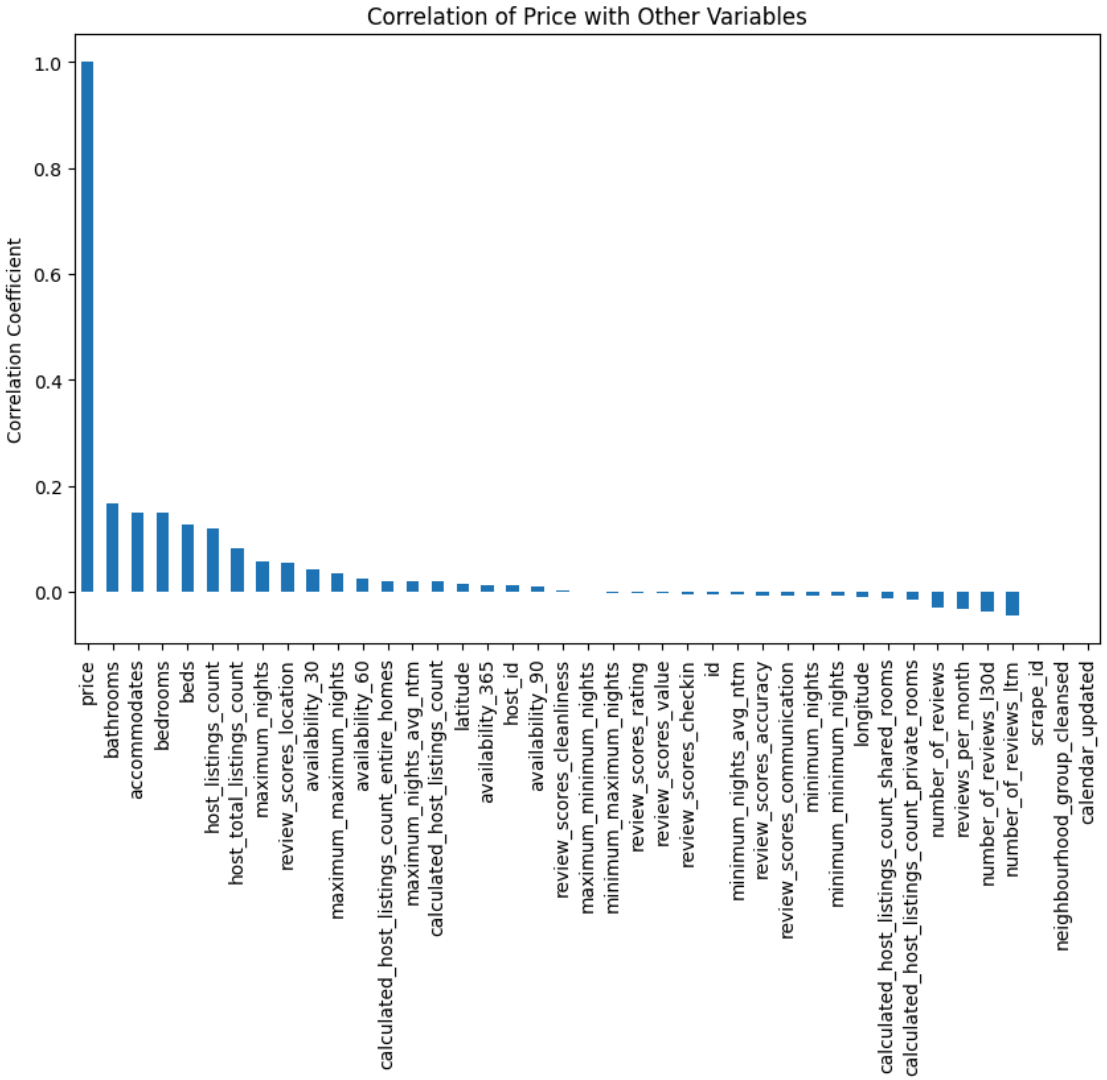
En este ejercicio el objetivo es predecir el precio de alquiler de departamentos. Trabajamos con un dataset de 34.061 departamentos con 75 características.

En primer lugar, procederemos con la limpieza de los datos: valores faltantes, valores atípicos, normalización. Luego, compararemos tres modelos: regresión lineal múltiple, XGBoost y Random Forest, para determinar cuál de ellos sería el más eficiente.

a) Análisis Exploratorio y preprocesamiento de datos

El dataset *"Detailed Listings data"* contiene una gran cantidad de características, por lo que, para simplificar su análisis, comenzaremos eliminando algunas variables que elegimos con los siguientes graficos.

Gráfico que muestre la correlación entre el precio y las demás variables numéricas :



Usamos también gráficos con boxplots con el precio y cada variable separada en categorías (bining). Eso permite observar gráficamente si existe una relación entre la variable y el precio, especialmente en el caso de una relación no lineal o de una relación cuyo coeficiente de correlación parece menor debido a la presencia de valores atípicos.

Esta conversión se realizó para todas las variables numéricas, aquí algunos ejemplos.

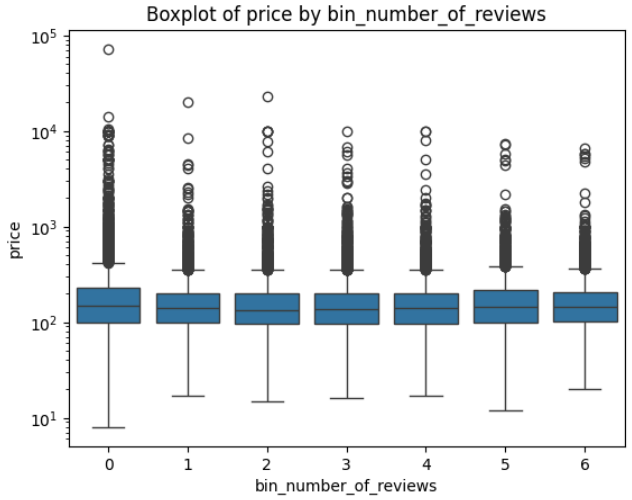


Gráfico en el que no se observa una relación evidente (variable que no va a interesarnos) :

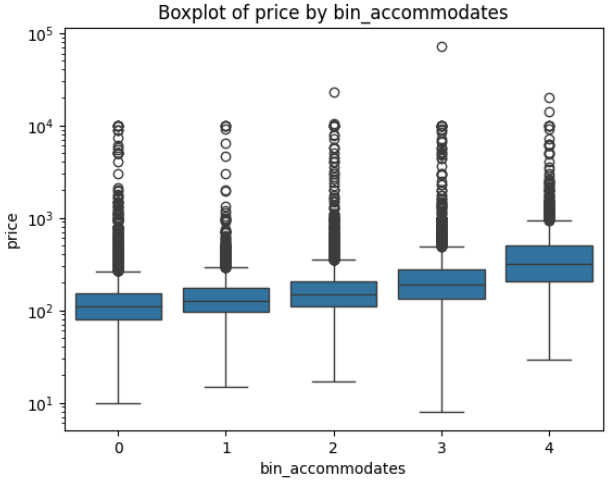


Gráfico en el que se observa una relación que parece lineal (variable que vamos a utilizar directamente :

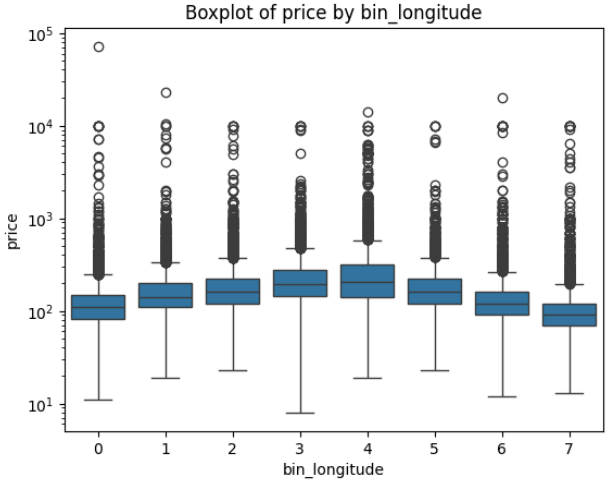


Gráfico en el que se observa una relación no lineal (variable que vamos a utilizar despues de ponerla en categorías) :

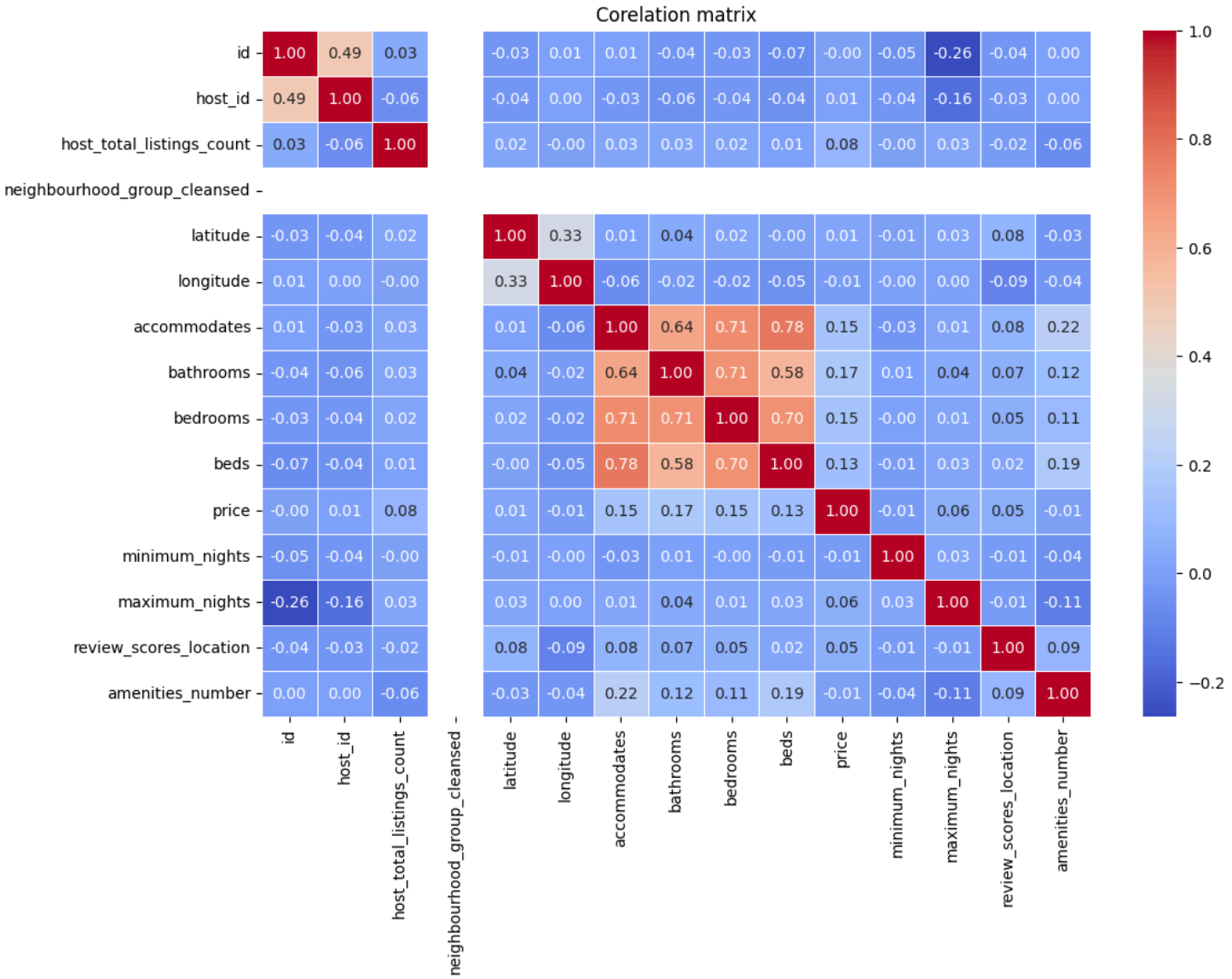
Teniendo en cuenta los dos elementos anteriores, conservamos las siguientes columnas :

- Columns being kept :
  - 'host_id', 'neighbourhood_cleansed', 'latitude', 'longitude', 'property_type', 'room_type', 'accommodates', 'bathrooms', 'bathrooms_text', 'bedrooms', 'beds', 'amenities', 'price', 'minimum_nights', 'maximum_nights', 'host_total_listings_count', 'review_scores_location', 'amenities'

Después de realizar la limpieza anterior, podemos analizar con más detalle la relación entre el precio y nuestras variables.

Datos Numéricos

Matriz de correlación :

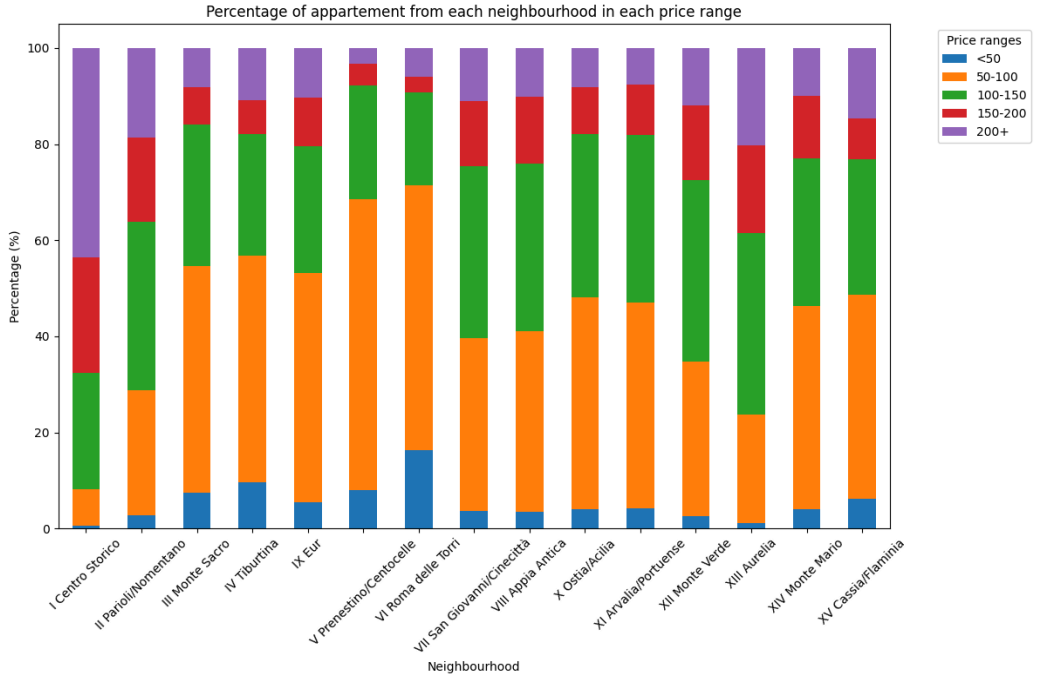


Podemos ver que las variables no están muy correlacionadas con el precio, debe ser mejor después de la limpieza de datos.

Datos no numéricos

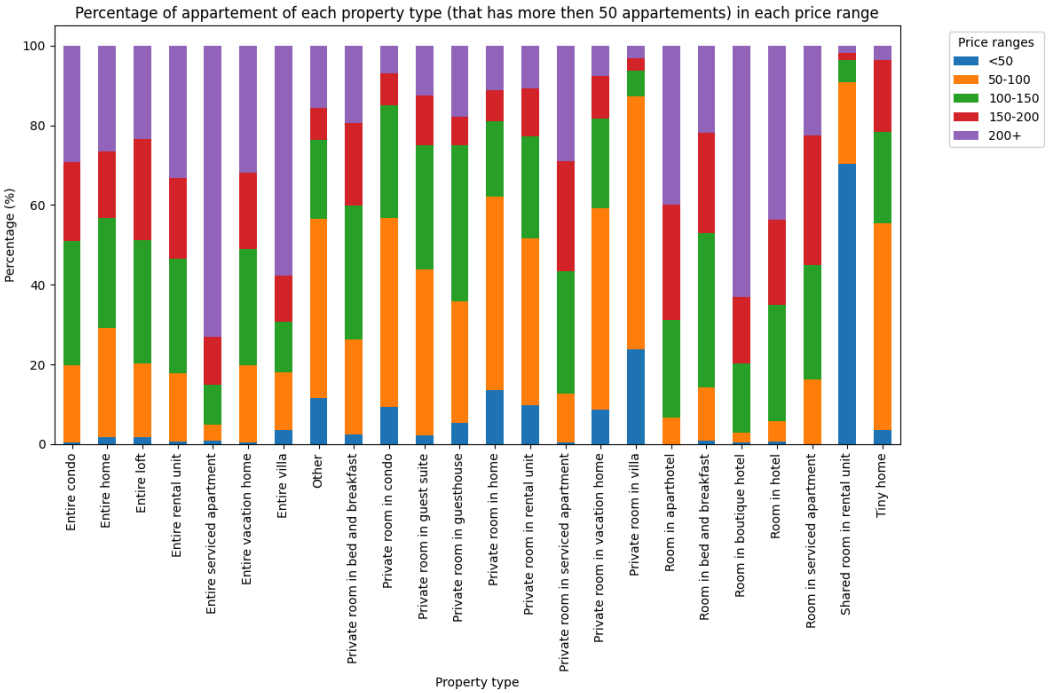
Neighbourhood

El gráfico siguiente representa en cada barrio el porcentaje de apartamentos de ese barrio que se encuentra en cada categoría de precio.



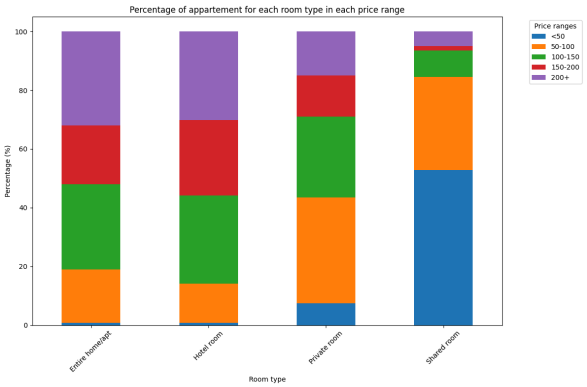
Apartment type

El mismo gráfico para la variable tipo de propiedad da un resultado confuso porque el número de categorías es muy grande. Además, algunas categorías contienen muy pocos elementos. Podemos agrupar los elementos que están en categorías con menos de 50 filas para tener un gráfico más legible.



Room type

El mismo grafico para room_type



Valores faltantes

En este conjunto de datos no hay valores numéricos menores que 0, ni caracteres "-" que indiquen un valor faltante, ni duplicados.

Porcentaje de valores NaN :

host_total_listings_count	0.000000
neighbourhood_cleansed	0.000000
neighbourhood_group_cleansed	100.000000
latitude	0.000000
longitude	0.000000
property_type	0.000000
room_type	0.000000
accommodates	0.000000
bathrooms	11.044890
bathrooms_text	0.135052
bedrooms	2.771498
beds	11.226916
price	11.053698
minimum_nights	0.000000
maximum_nights	0.000000
review_scores_location	14.920290
amenities_number	0.000000

Los NaNs se eliminan de manera diferente según las columnas:

Column being **deleted**

- Neighbourhood_group_cleansed : too much data is missing

Column were NaN are being **replaced by average**

- Review_scores_location

Column were NaN are being **replaced thanks to another column** :

- Bathrooms with Bathrooms_text

Columns were NaNs are replaced with **MICE** (Multivariate Imputation by Chained Equations)

- Accommodates : number of people that can sleep in the apartment
- Bathroom
- Bedroom
- Beds
  -> These columns are being replaced with MICE because they are all linked with one another thus the values we will find are going to be good.

Columns were NaNs won't be replaced

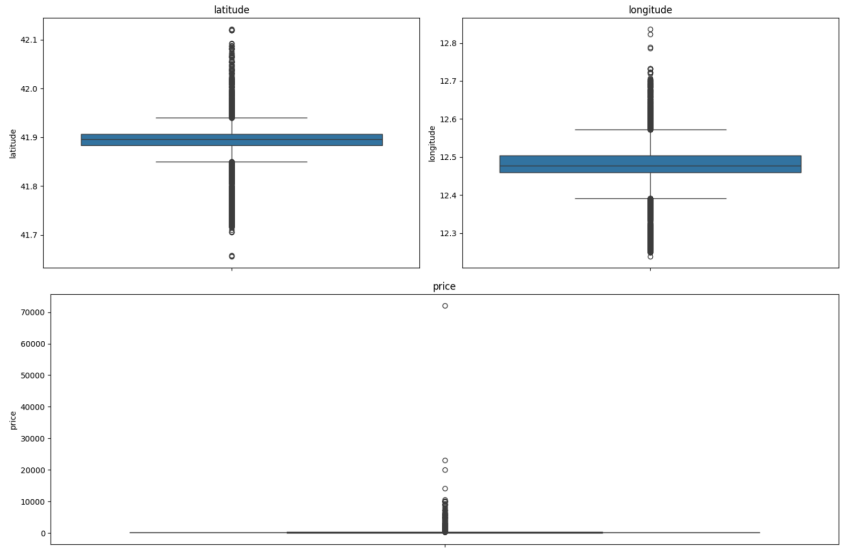
- price

Se elige eliminar los NaNs de la columna precio en lugar de reemplazarlos porque es el target y esto crearía un sesgo.

Análisis valores atípicos

Comenzamos trazando los boxplots de cada variable para tener una idea general de la cantidad de valores atípicos.

Aquí algunos ejemplos:



Porcentaje de valores atípicos

Calculado utilizando IQR :

	variable	cant outliers	porcentaje
9	maximum_nights	6563	21.662926
0	host_total_listings_count	3812	12.582519
1	latitude	2827	9.331265
7	price	2300	7.591761
2	longitude	1792	5.914972
10	review_scores_location	1561	5.152495
4	bathrooms	1519	5.013863
3	accommodates	1446	4.772907
5	bedrooms	782	2.581199
8	minimum_nights	727	2.399657
6	beds	420	1.386322
11	amenities_number	109	0.359783

Calculado utilizando el Z-Score

```
id: 0.00% of outliers
host_id: 0.00% of outliers
host_total_listings_count: 18.46% of outliers
latitude: 7.03% of outliers
longitude: 4.50% of outliers
accommodates: 0.28% of outliers
bathrooms: 25.81% of outliers
bedrooms: 37.99% of outliers
beds: 0.82% of outliers
price: 6.17% of outliers
minimum_nights: 1.64% of outliers
maximum_nights: 0.00% of outliers
review_scores_location: 3.36% of outliers
amenities_number: 0.04% of outliers
```

Con el Z-score, las columnas Bedrooms, Bathrooms y Host Total Listings score tienen muchos valores atípicos. Para estas columnas, los valores atípicos se reemplazan por el valor promedio de la columna, utilizando IQR para detectarlos.

Para las otras columnas, utilizamos el método Z-score.

Si eliminamos todos los valores atípicos excepto bedroom y bathroom, obtenemos el mejor resultado, pero perdemos casi la mitad de los datos.

Si reemplazamos todos los valores atípicos, obtenemos un resultado menos favorable.

Si eliminamos los valores atípicos excepto bathrooms, bedrooms y host_total_listings_count, obtenemos un resultado intermedio y perdemos "solo" unas 5000 filas aproximadamente.

Por lo tanto, optamos por eliminar los valores atípicos excepto bathrooms, bedrooms y host_total_listings_count.

Ejemplo de comparación de los resultados para la regresión lineal múltiple.

Results for linear regression when deleting outliers :

```
Train :
Mean Squared Error: 2563.313475535611
Root Mean Squared Error: 50.62917612933881
R² Score: 0.4889742224797212
Test :
Mean Squared Error: 2476.5187846018425
Root Mean Squared Error: 49.7646338738852
R² Score: 0.5151255346538651
```

Results for linar regression when imputing outliers :

```
Train :
Mean Squared Error: 3442.507580563748
Root Mean Squared Error: 58.67288624708817
R² Score: 0.37968029217204335
Test :
Mean Squared Error: 3538.6486804100705
Root Mean Squared Error: 59.48654201086217
R² Score: 0.3805520190536077
```

Results for linear regression when inputting some of the outliers (bedrooms, bathrooms and host_total_listings_count)

```
Train :
Mean Squared Error: 3098.840295822466
Root Mean Squared Error: 55.66722820315797
R² Score: 0.4861681005653814
Test :
Mean Squared Error: 2909.5969080867776
Root Mean Squared Error: 53.94067952933832
R² Score: 0.478094187036603
```

Transformación de variables

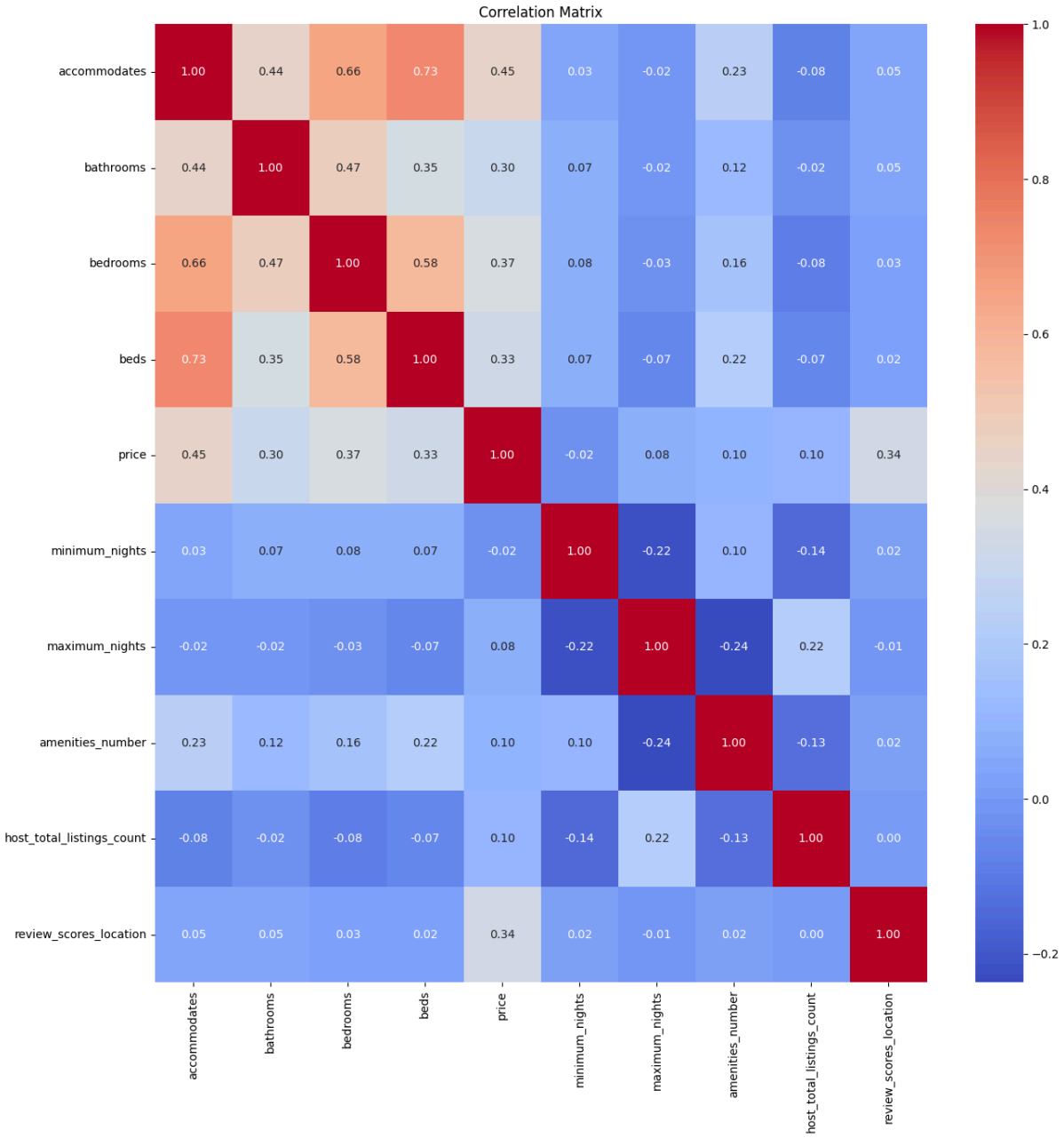
- Para la columna property_type, agrupamos los elementos en categorías más grandes.

- Para las otras columnas que no son numéricas, aplicamos una codificación binaria con OneHotEncoding.

- Para las columnas de tipo numérico cuya relación con el precio no es lineal, hacemos una categorización y luego una codificación binaria.
- Para las otras columnas de tipo numérico, aplicamos una transformación min-max.

b) Entrenamiento y Predicción

Después de la limpieza de datos, obtuvimos la siguiente nueva matriz de correlación :



Podemos ver que las correlaciones entre el precio y las demás variables son más fuertes que anteriormente.

Modelo de regresión lineal

El modelo de regresión lineal múltiple es el más simple.

Nos da el siguiente resultado :

```
Train :
Mean Squared Error: 3098.840295822466
Root Mean Squared Error: 55.66722820315797
R² Score: 0.4861681005653814
Test :
Mean Squared Error: 2909.5969080867776
Root Mean Squared Error: 53.94067952933832
R² Score: 0.478094187036603
```

Significance of different metrics

- The value of rmse is the average distance between the real and the predicted price.
- The value of mse is rmse². The lower rmse and mse we get, the better.
- R^2 is the proportion of the variability of the price that can be explained by the features. The closest it is to 1 the better the model explains the variability.

Result análisis :

- Our value of R^2 is about 0.48, which means our model explains 48% of the variance in the price.
- The rmse is lower in the tests than in the training, which is good because it means our model isn't overfitting.
- The standard deviation for the price is 71 (see result of the code below), thus an RMSE of 54 is okay even though it could probably be improved by optimizing the model more.

Con la validación cruzada obtenemos los siguientes resultados:

```
Cross-validation (training set) :
Mean RMSE (5-folds): 55.74
Standard deviation of RMSE (5-folds): 1.03
```

Elegimos 5 folds ya que el conjunto de datos es grande, por lo que parece suficiente.

Encontramos un RMSE promedio similar al que obtuvimos sin validación cruzada, y una desviación estándar pequeña, lo que indica la estabilidad del modelo.

Modelo XGBoost

Para el modelo XGBoost, probamos diferentes valores para los siguientes parámetros :

- n_estimators : total number of boosting iterations (trees)
- max_depth : maximum depth of each tree
- learning_rate: how much the model adjusts at each step
- subsample : proportion of the training data used for each round

The smallest rmse with a small test-train difference is with parameters :

- n_estimators = 200
- max_depth = 3
- learning_rate = 0.1
- subsample = 0.8

With

```
Cross-validation (training set) for current params:
Mean RMSE (5-folds): 53.32
Standard deviation of RMSE (5-folds): 1.14
Train :
Mean Squared Error: 2670.2222936202866
Root Mean Squared Error: 51.67419369105131
R² Score: 0.5570098874335265
Test :
Mean Squared Error: 2768.8253437109715
Root Mean Squared Error: 52.619628882299914
R² Score: 0.5173964496506809
```

The results are better than with the linear regression model even though a bigger difference could have been expected.

Modelo Random Forest

Para el modelo de Random Forest, con los parámetros básicos, obtenemos un modelo que sobre ajusta los datos de entrenamiento, con un RMSE en el entrenamiento más de 2 veces superior al RMSE de prueba.

```
Cross-validation (training set) :
Mean RMSE (5-folds): 54.81
Standard deviation of RMSE (5-folds): 0.84

Train :
Mean Squared Error (MSE): 418.29
Root Mean Squared Error (RMSE): 20.45
R² Score: 0.93

Test :
Mean Squared Error (MSE): 2948.72
Root Mean Squared Error (RMSE): 54.30
R² Score: 0.49
```

Para solucionar este problema, podemos limitar la profundidad de los árboles del bosque.

Aquí están los resultados con varios valores diferentes de profundidad máxima :

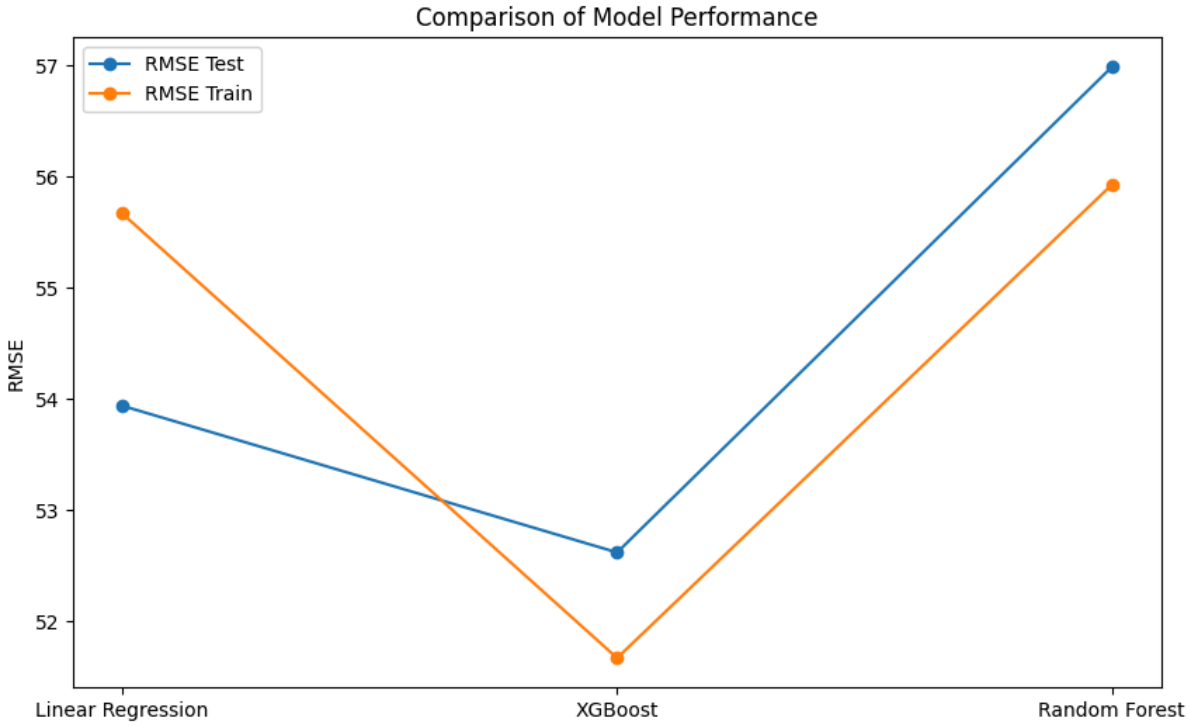
```
Testing for max_depth = 5
Cross-validation (training set) for max_depth=5:
Mean RMSE (5-folds): 60.56
Standard deviation of RMSE (5-folds): 1.19

Testing for max_depth = 7
Cross-validation (training set) for max_depth=7:
Mean RMSE (5-folds): 58.03
Standard deviation of RMSE (5-folds): 1.20

Testing for max_depth = 9
Cross-validation (training set) for max_depth=9:
Mean RMSE (5-folds): 56.44
Standard deviation of RMSE (5-folds): 1.14

Summary of results:
   max_depth    train_mse  train_rmse  train_r2     test_mse  test_rmse  \
0          5  3581.082550   59.842147  0.405898  3549.441763  59.577192
1          7  3128.693553   55.934726  0.480949  3247.887888  56.990244
2          9  2641.207239   51.392677  0.561823  3089.130386  55.579946

    test_r2
0  0.381336
1  0.433896
2  0.461568
```

Una profundidad máxima de 7, por ejemplo, nos permite limitar el sobreajuste sin aumentar demasiado el valor del RMSE.

c) Cuadrado de resultado

Para los mejores parámetros (entre los que hemos probado) de cada modelo, obtenemos los siguientes resultados:

Model	RMSE test	RMSE train	R^2
Model 1 (Linear Regression)	53.94	55.67	0.48
Model 2 (XGBoost)	52.62	51.67	0.52
Model 3 (Random Forest)	56.99	55.93	0.43



El modelo que parece ser el más eficaz en nuestra situación es el modelo XGBoost.

Ejercicio 4

En este ejercicio, trabajamos con un dataset que contiene las características de 750 canciones de Spotify, y el objetivo es clasificarlas utilizando un algoritmo de K-Means para determinar diferentes grupos de música.

Etapa 1: Limpieza del Dataset

- **Análisis de datos faltantes**: En este dataset no se encontraron datos faltantes.
- **Análisis de datos raros**: También se constató que no había datos raros.
- **Eliminación de datos duplicados**: Se eliminaron 14 líneas que eran duplicadas.
- **Análisis de outliers**: A través de un análisis de IQR, se identificaron muchos datos que podrían considerarse outliers. Sin embargo, decidimos mantenerlos, ya que en el contexto de los datos musicales es normal tener valores muy diferentes. Además, eliminar esos outliers significaría perder casi 350 datos, lo que podría afectar el análisis.

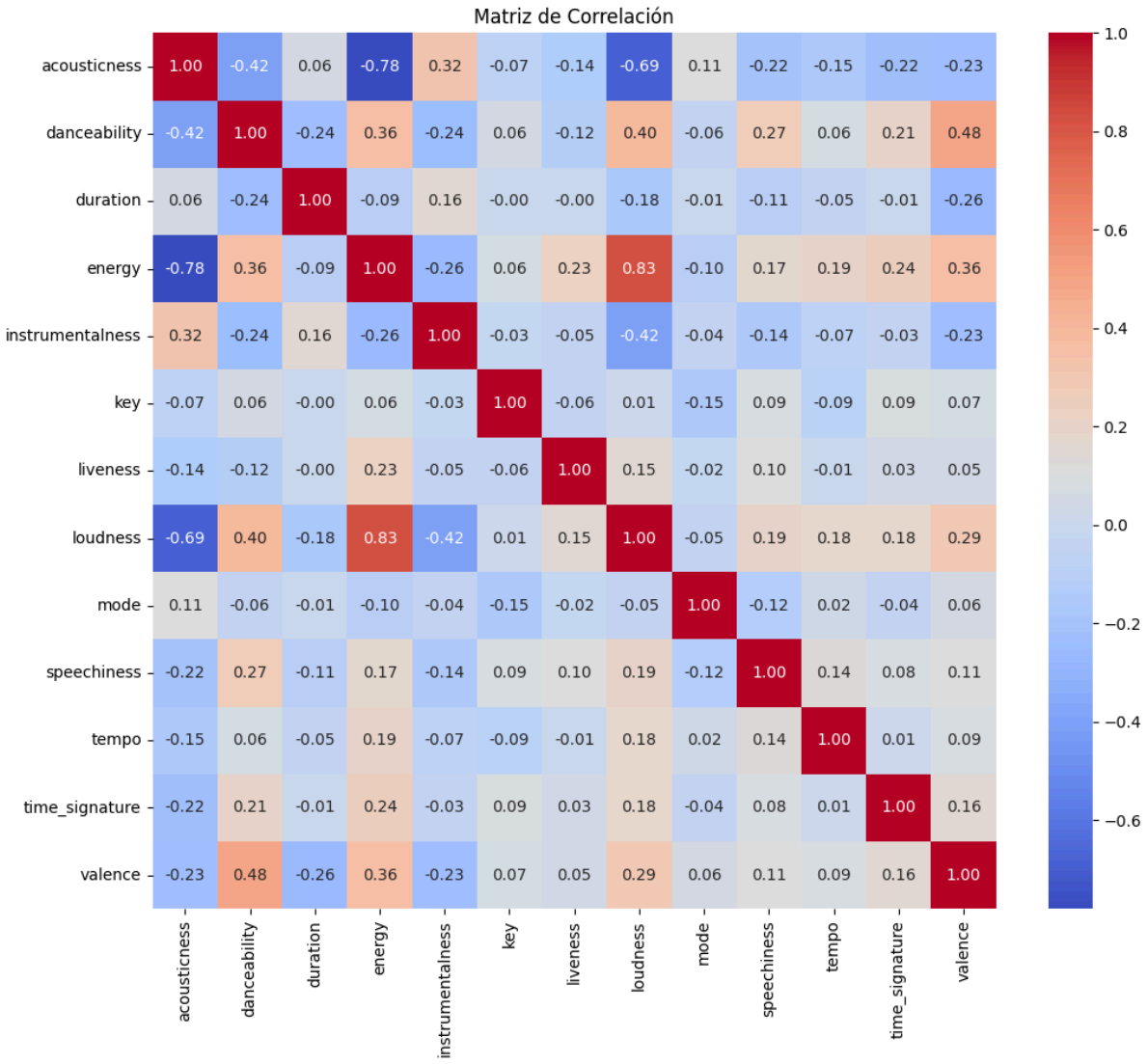
Etapa 2: Normalización de Datos

Para utilizar K-Means de manera efectiva, normalizamos los datos.

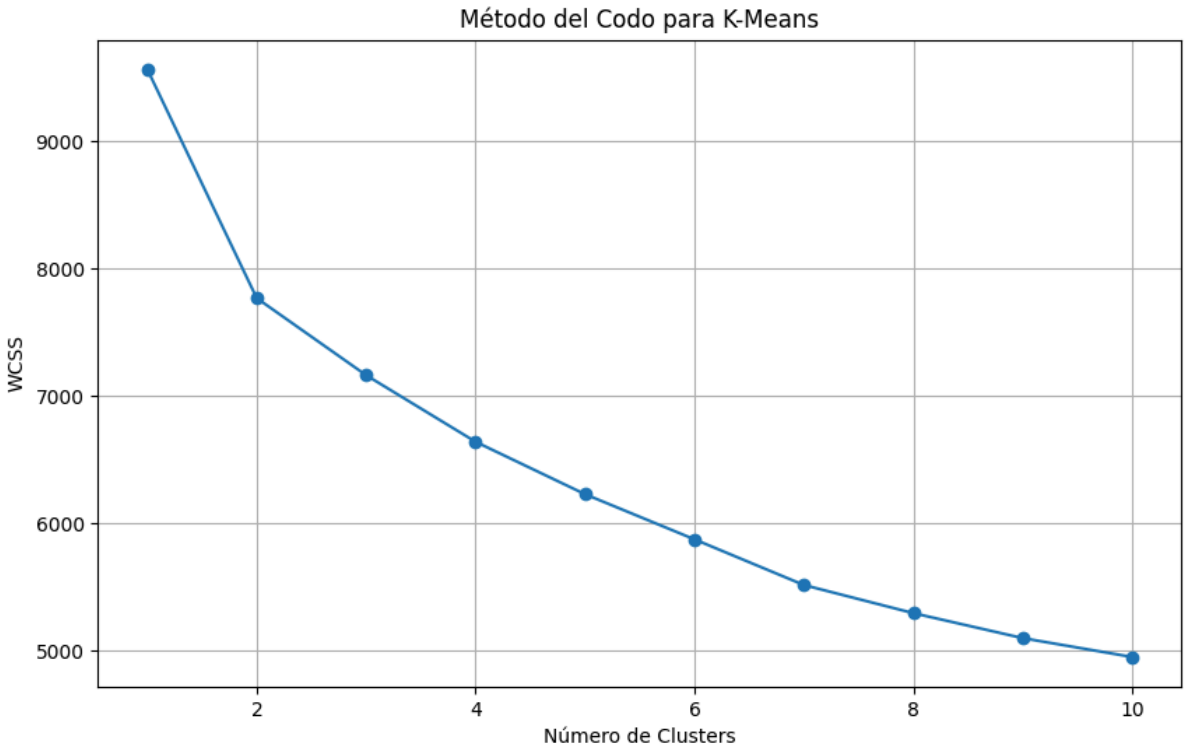
Hemos probado dos maneras de normalizar los datos : MinMaxScaler y StandardScaler.

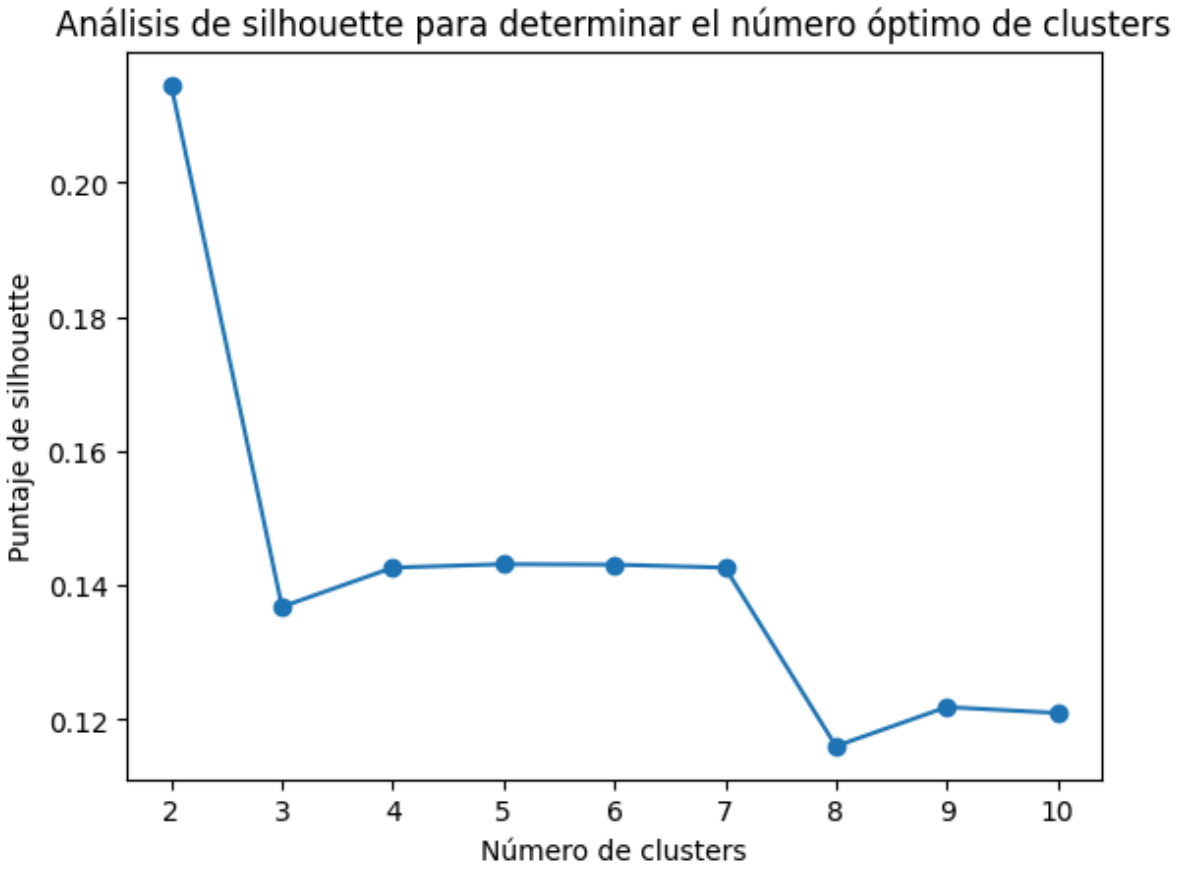
Etapa 3: Reducción de Dimensionalidad

Realizamos un análisis de la matriz de covarianza para visualizar la correlación entre las variables. Las variables más importantes que se identificaron son : **loudness**, **danceability**, **energy**, **valence**, **acousticness** e **instrumentalness**.

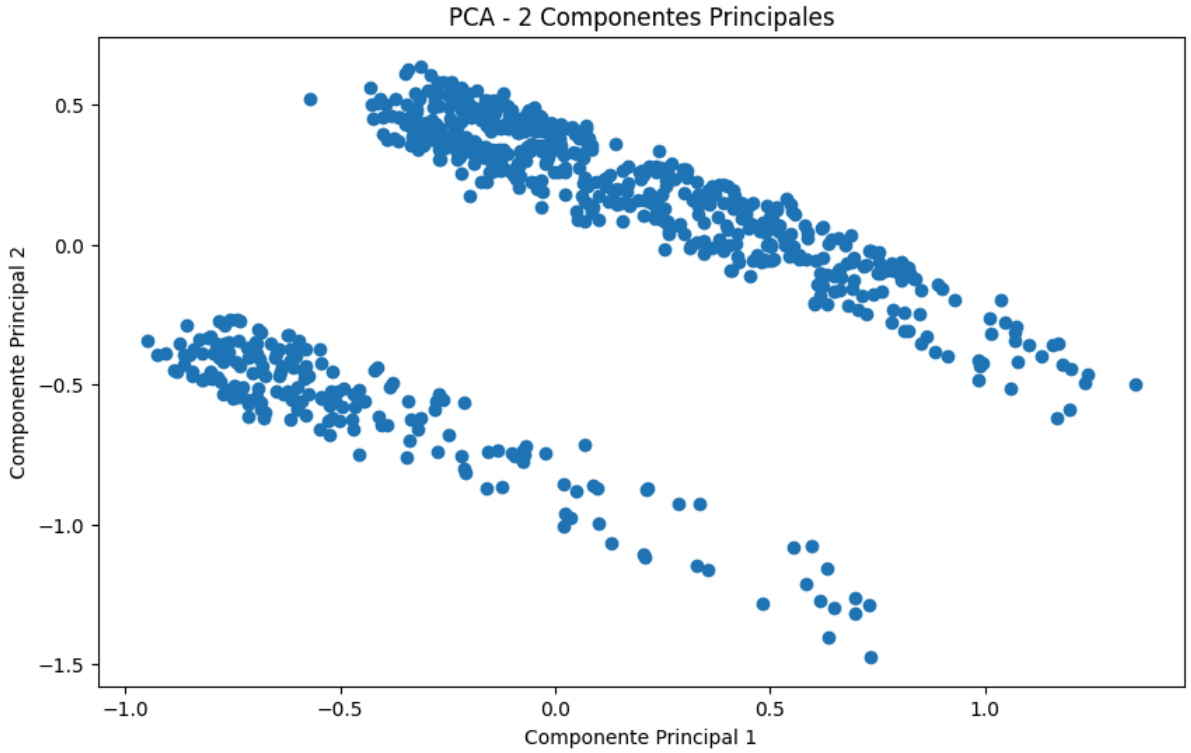


Posteriormente, se llevó a cabo un análisis para determinar el número óptimo de clusters utilizando el método del codo. Se forma un codo a dos clusters, lo que indica que podría ser la cantidad óptima. Adicionalmente, el análisis de Silhouette confirmó que 2 clusters son los más apropiados.


Método del Codo para K-Means
WCSS
9000
8000
7000
6000
5000
2
4
6
8
10
Número de Clusters

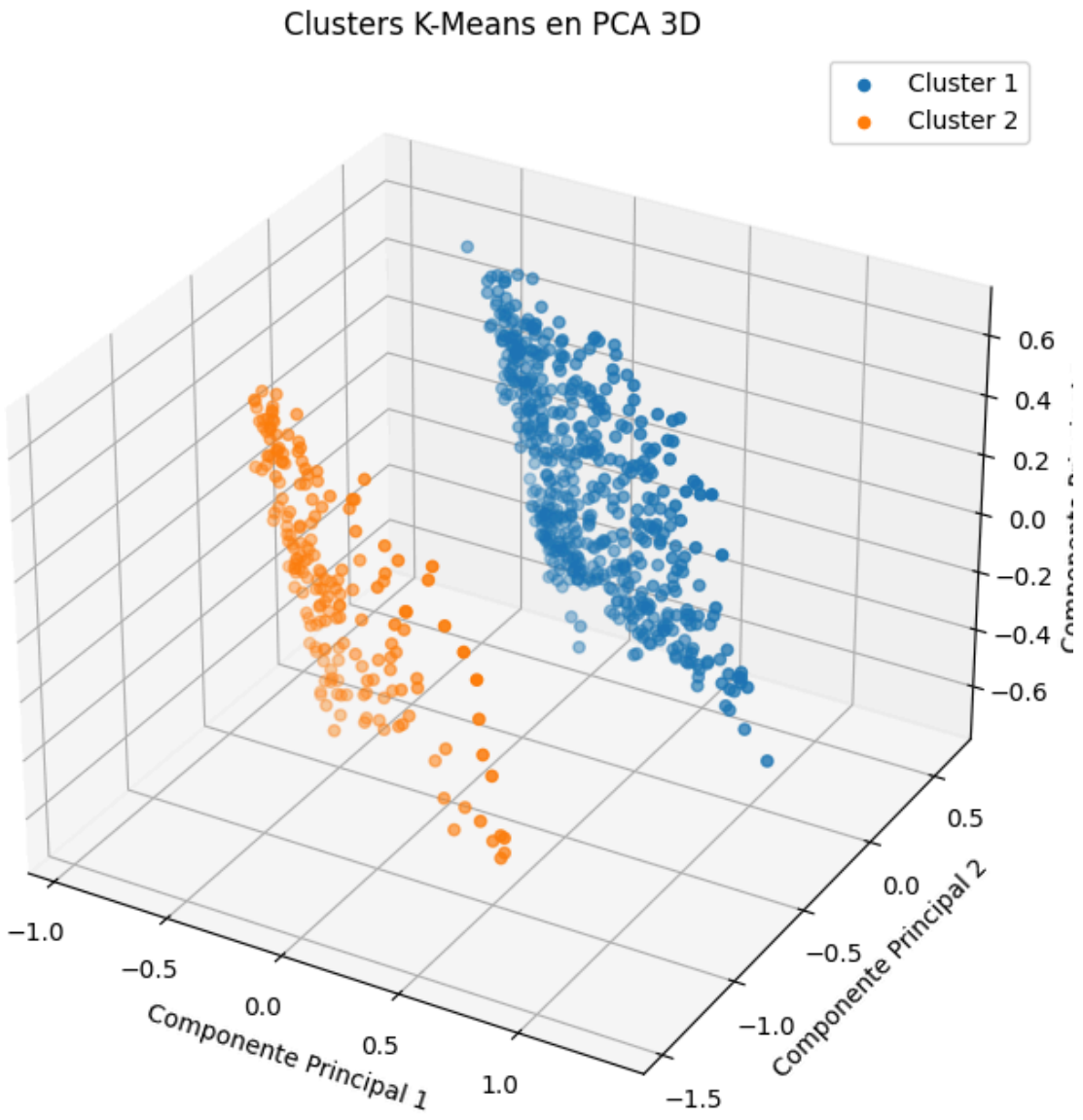

Análisis de silhouette para determinar el número óptimo de clusters
Puntaje de silhouette
0.20
0.18
0.16
0.14
0.12
2
3
4
5
6
7
8
9
10
Número de clusters

Para visualizar mejor los grupos formados, realizamos un PCA (Análisis de Componentes Principales) que permitió reducir las dimensiones del dataset. Esto mostró que se formaron dos grupos, lo que corroboró que 2 clusters son óptimos.



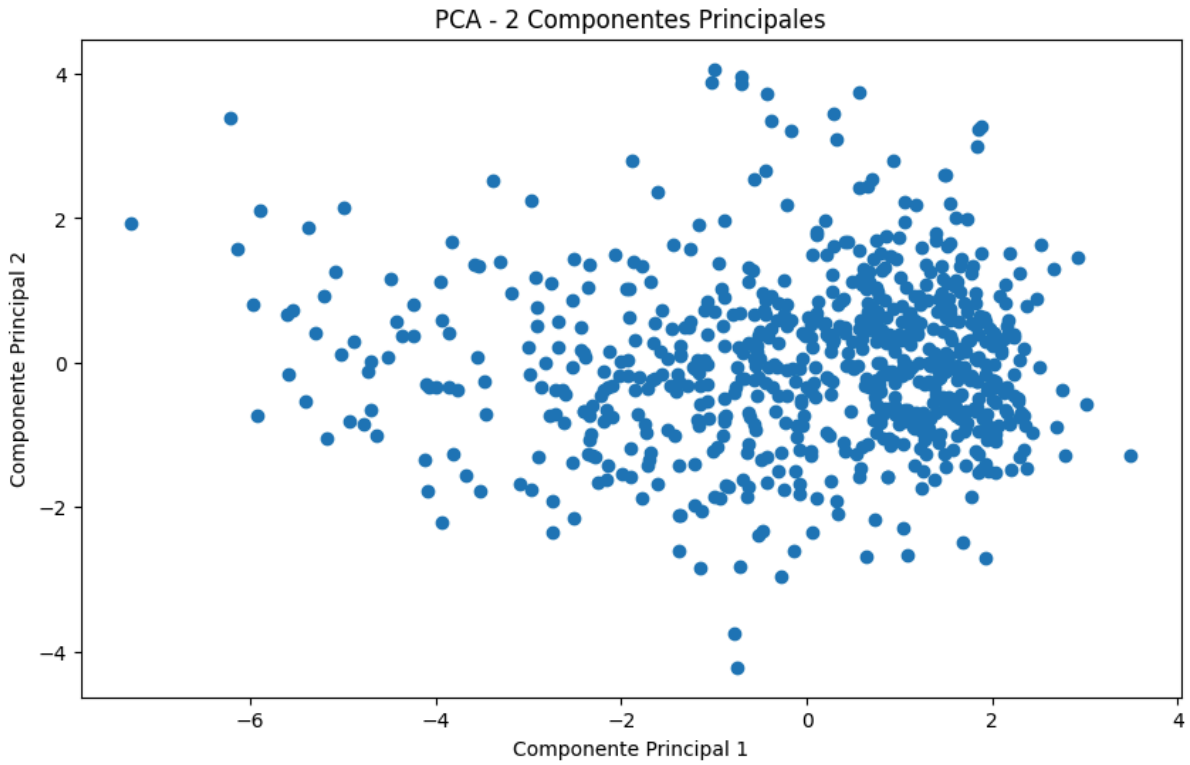
Los resultados fueron esperados, ya que los dos grupos son claramente distintos.

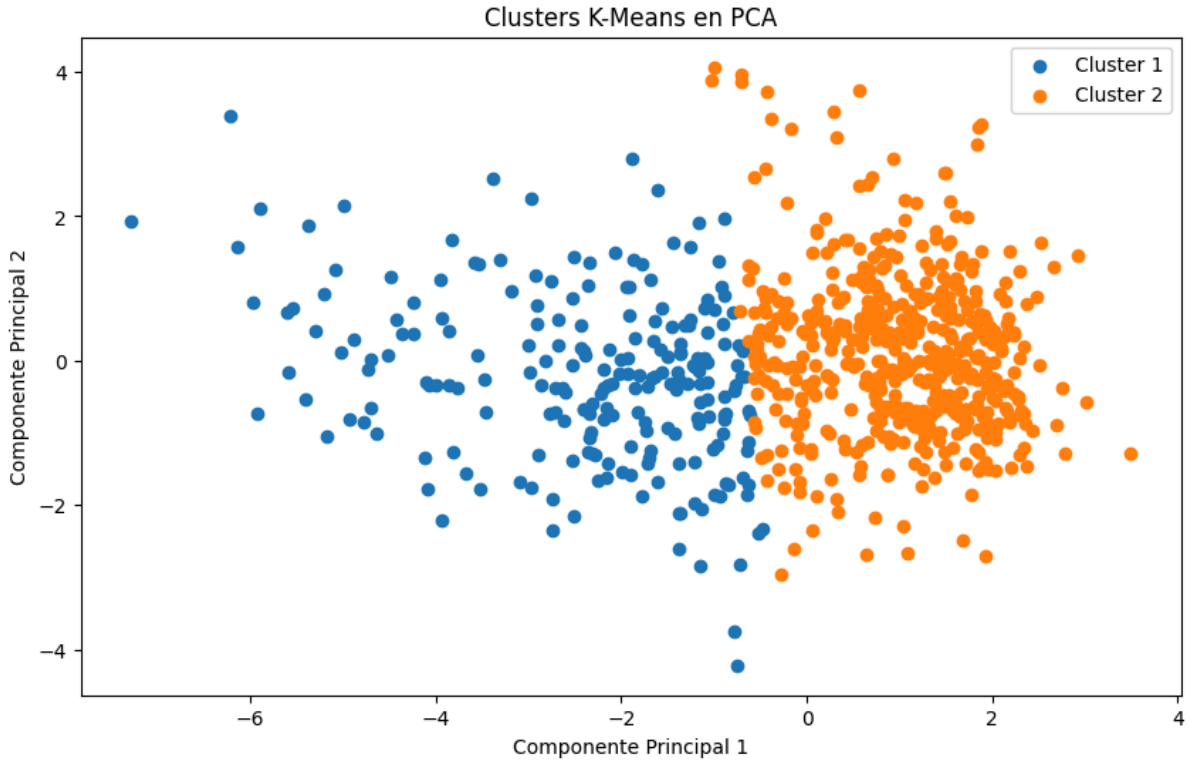
Tenemos Silhouette Score de **0,45** (frente à **0,30** si no se había realizado la reducción de dimensionalidad). Sin embargo, es importante señalar que los dos grupos obtenidos solo se diferenciaban por el valor de la variable **mode**, que es binaria. Esto no resulta particularmente interesante, dado que los grupos formados no utilizan las otras variables.

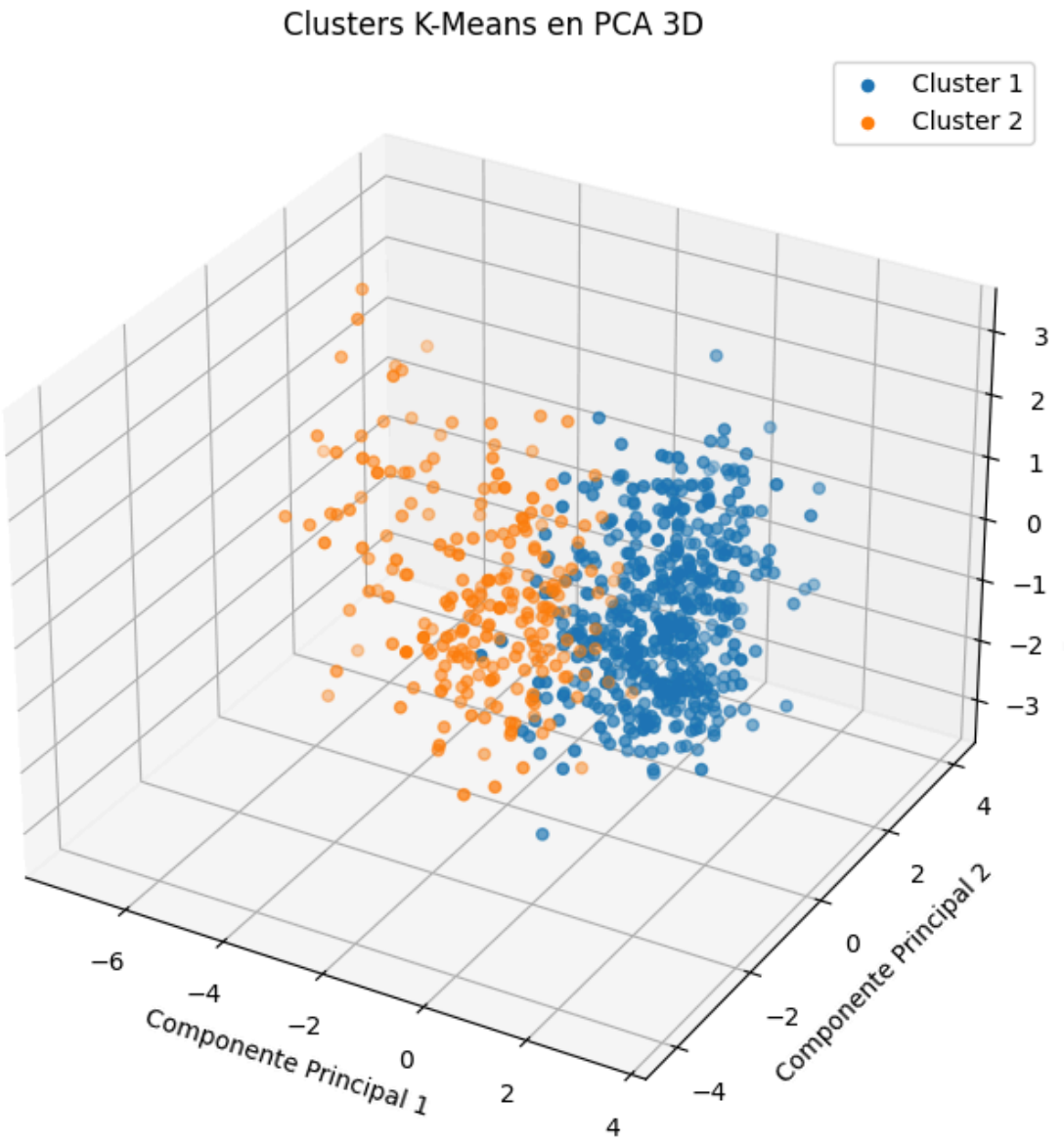


Para mejorar los resultados, realizamos la misma operación utilizando una normalización con **StandardScaler**, lo que dio lugar a resultados diferentes.

Con el método del codo y el análisis de Silhouette, se determinó que el número óptimo de clusters seguía siendo 2. Sin embargo, la visualización de grupos mediante PCA no mostró distinciones claras.


PCA - 2 Componentes Principales
Componente Principal 2
Componente Principal 1
4
2
0
−2
−4
−6
−4
−2
0
2
4


Clusters K-Means en PCA
Cluster 1
Cluster 2
Componente Principal 2
Componente Principal 1
4
2
0
−2
−4
−6
−4
−2
0
2
4



Los grupos formados en este último análisis tuvieron un Silhouette Score más bajo: **0,39** (frente a **0,21** sin la reducción de dimensionalidad). No obstante, los grupos obtenidos resultaron ser más interesantes:

- **Grupo 1**: canciones con energía, danceability y loudness, positivas.
- **Grupo 2**: canciones más tranquilas, acústicas y temas más negativas.

Además, intentamos realizar más agrupaciones. Al considerar 4 grupos, el Silhouette Score fue aún más bajo: **0,27** (frente a **0,14** sin la reducción de dimensionalidad). Los grupos analizados pueden clasificarse de la siguiente manera:

- **Grupos 0 y 2**: menos acústicos, con mayor loudness, energy y danceability. La diferencia entre ambos radica en la **key** y el **mode**, resultando que el Grupo 2 tiene canciones más positivas (mejor **valence**) que el Grupo 0.
- **Grupos 1 y 3**: representan canciones más calmadas y acústicas. Sin embargo, el Grupo 3 es un poco menos acústico y presenta más energía, loudness y danceability.

Se puede establecer una escala que va desde canciones tranquilas y negativas hasta canciones energéticas y positivas: **1 < 3 < 0 < 2**.

Otro método que hemos utilizado fue utilizar un RandomForest para calcular la importancia de cada variable y quedarnos solo con las más relevantes. Luego, aplicamos el mismo método y obtuvimos 4 grupos similares, con un Silhouette Score de **0,37**.

Conclusión del ejercicio

El análisis del dataset de 750 canciones de Spotify permitió identificar patrones musicales mediante K-Means. Aunque el número óptimo de clusters se determinó en 2, estos se diferenciaron principalmente por la variable **mode**, lo que limita la riqueza de la clasificación.

Al aplicar la normalización con **StandardScaler**, se generaron grupos más interesantes, aunque el Silhouette Score disminuyó. Este enfoque resaltó la importancia de otras variables como **energy**, **danceability** y **loudness**.