.UBAfiuba
FACULTAD DE INGENIERÍA

TA047 - Ciencia de Datos Dr. Ing. Rodríguez - 2°C 2024

Trabajo Práctico 1 - Grupo 05

EJ1: Análisis Exploratorio

El objetivo del ejercicio fue realizar el análisis exploratorio de los datos, aplicar técnicas de exploración y de preprocesamiento (de valores nulos y outliers) para poder responder algunas preguntas planteadas sobre dichos datos, utilizando 3 dataset (que unificamos en 1 para el análisis) de viajes de taxi amarillos en EEUU del año 2023 en los meses de Enero, Febrero y Marzo.

El dataset (unificado) está conformado por 21 columnas (features) con 9.384.487 de registros.
Incluye campos que registran las fechas y horas de salida y llegada, las distancias de los viajes, las tarifas detalladas, los tipos de tarifas, los tipos de pago y los recuentos de pasajeros informados por el conductor. Los datos utilizados en los conjuntos de datos adjuntos fueron recopilados y proporcionados a la Comisión de Taxis y Limusinas de la Ciudad de Nueva York (TLC) por proveedores de tecnología autorizados en virtud de los Programas de Mejora de Pasajeros de Taxis y Limusinas (TPEP/LPEP).

Se analizaron los features de variables cualitativas y cuantitativas.
Para las variables cualitativas las que destacaron fueron las fechas de los viajes (la de salida y llegada coinciden en el mismo día), el tipo de pago y VendorID (Un código que indica el proveedor de TPEP que proporcionó el registro).
Para las variables cuantitativas las que destacaron fueron cantidad de pasajeros informados por el conductor, distancia recorrida por cada viaje y el importe total.

Analizando la correlación entre las variables previo al tratamiento de datos, lo cuál fue muy poca , las variables que tuvieron más correlación positiva fuerte fueron entre la VendorID con cantidad de pasajeros (0,1) y con el día de la semana (0,051, que se obtuvo de la fecha del viaje). Una correlación negativa fuerte entre importe total con tipo de pago (-0,077) y con día de la semana (-0,012) .
La distancia recorrida apenas tenía un poco de correlación con el importe total del viaje (0,018).

Luego del tratamiento de datos y volviendo a analizar la correlación entre las variables, las que tuvieron correlación positiva fuerte fueron los features distancia recorrida con importe total (0,92), duración del viaje en minutos con importe total (0,88), y con distancia recorrida (0,83). Respecto de la anterior, la correlación entre VendorID con cantidad de pasajeros se redujo a 0,07. Y entre cantidad de pasajeros y día de la semana se mantuvo con 0.05.
En cuanto a la correlación negativa fuerte los features que tuvieron mayor coeficiente fueron entre importe total con tipo de pago (-0,12 la cual aumentó respecto de la anterior) y con día de la semana (-0,013 casi lo mismo) .

Preprocesamiento de Datos

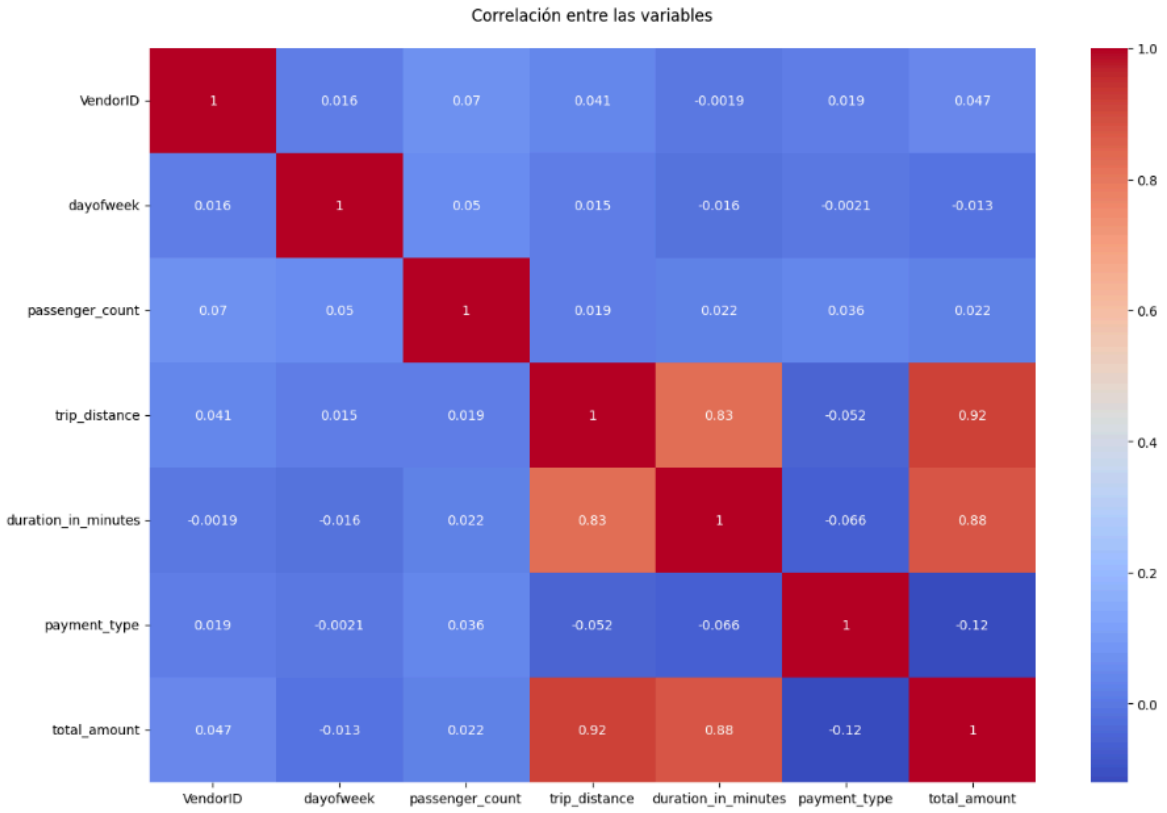
1. ¿Se eliminaron columnas (Nombre de la columna y motivo de eliminación?

   Las columnas que se eliminaron fueron los features airport_fee y Airport_fee (Tarifa por recogida únicamente en los aeropuertos LaGuardia y John F. Kennedy) ya que tenían el 70% y 30% aprox. respectivamente de datos faltantes, y era un porcentaje muy significativo.

2. ¿Detectaron correlaciones interesantes (entre qué variables y qué coeficiente)?

   Como ya mencionamos, después del tratamiento de datos, los que tuvieron correlación positiva fuerte fueron los features distancia recorrida con importe total (0,92), duración del viaje en minutos con importe total (0,88), y con distancia recorrida (0,83).

   En cuanto a la correlación negativa fuerte los features que tuvieron mayor coeficiente fueron entre importe total con tipo de pago (-0,12) y con día de la semana (-0,013) .



3. <u>¿Generaron nuevos features?</u>

   Generamos 7 features nuevas (6 cualitativas y 1 cuantitativa):

   Nombre del día de la semana (**weekday**), identificador del día de la semana (**weekofday**) y nombre del día del mes (**month**). Estos 3 features se generaron a partir de la fecha de llegada del viaje.

   El nombre del tipo de pago (**payment_type_name**), que se utilizó la documentación para reemplazar los IDs de payment_type.

   La descripción del VendorID (**vendor_description**), también obtenida de la documentación.

   La descripción del código de la tarifa (**ratecode_description**), utilizando el feature RatecodeID (código de tarifa final vigente al final del viaje) con su documentación.

   Duración del viaje desde la hora de salida y llegada en minutos (**duration_in_minutes**), utilizando dichas fechas del viaje.

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

Las variables de tipo monto tenían todas outliers. Arreglamos montos negativos que no tienen sentido en conceptos de impuesto e importe del viaje. Consideramos los negativos como mal cargados y los convertimos en positivos. En cuanto al importe total había casos donde el monto era 0 y claramente era un error de carga, ya que representaba un viaje gratis, así que tomamos la decisión de eliminar los registros ya que sólo representaban el 0.018% de los datos.

Una vez corregidos los datos los outliers ya eran por valores extremos y para mantener los registros (aunque alterando su distribución) aplicamos el método de recorte (capping).

Para el feature de cantidad de pasajeros vimos que había más de 6 pasajeros por viaje, y dedujimos que era un caso extremo que se podía dar en pocos casos donde el taxi era un tipo de vehículo de camioneta y podían entrar más de 6 pasajeros. Eran sólo 55 registros así que tomamos la decisión de aplicar el método de eliminación (trimming), ya que no influye demasiado en el análisis.

Para el feature de distancia recorrida vimos que habían viajes muy largos, más de 100000 km donde en EEUU la distancia más larga entre ciudades es de aprox 5000 km. Por lo tanto, los viajes que superan dicha distancia las consideramos mal cargadas. Las cuales eran el 0.0023% del dataset y no influye demasiado, así que las eliminamos. Luego vimos casos extremos que superan los 100 km, por lo que aplicamos el método de recorte para no perderlos.

Por último, para el feature que creamos para la duración de los viajes, nos encontramos que estaban quedando algunas en negativo, así que buscamos los casos mal cargados donde la fecha de llegada era menor o igual a la de salida. Encontramos 3630 registros y al ser pocos, fueron eliminados. En cuanto a los casos extremos, encontramos varios casos que superan los 1500 min, lo que representa aprox. 1 día de viaje sin descanso como máximo, a los cuales les aplicamos el método de recorte (aunque igualmente eran pocos).

5. ¿Qué columnas tenían datos faltantes?
   ¿En qué proporción? ¿Qué se hizo con estos registros?

Los primeros que encontramos y eliminamos fueron los ya mencionados **impuestos del aeropuerto**.

Luego vimos que la **cantidad de pasajeros** habían casos con 0 (156806 registros), el cual tenía sentido un viaje sin pasajeros sólo en los casos donde el viaje haya sido ANULADO (payment_type = 5). Si no se daba esa condición lo consideramos como un valor mal ingresado. Analizando dicho casos, ninguno se

encontraba anulado, así que tomamos la decisión de rellenarlo con el valor promedio, que era 1 pasajero por viaje. No perdimos los datos aunque si modificamos la distribución de los datos, pero manteniendo el promedio.

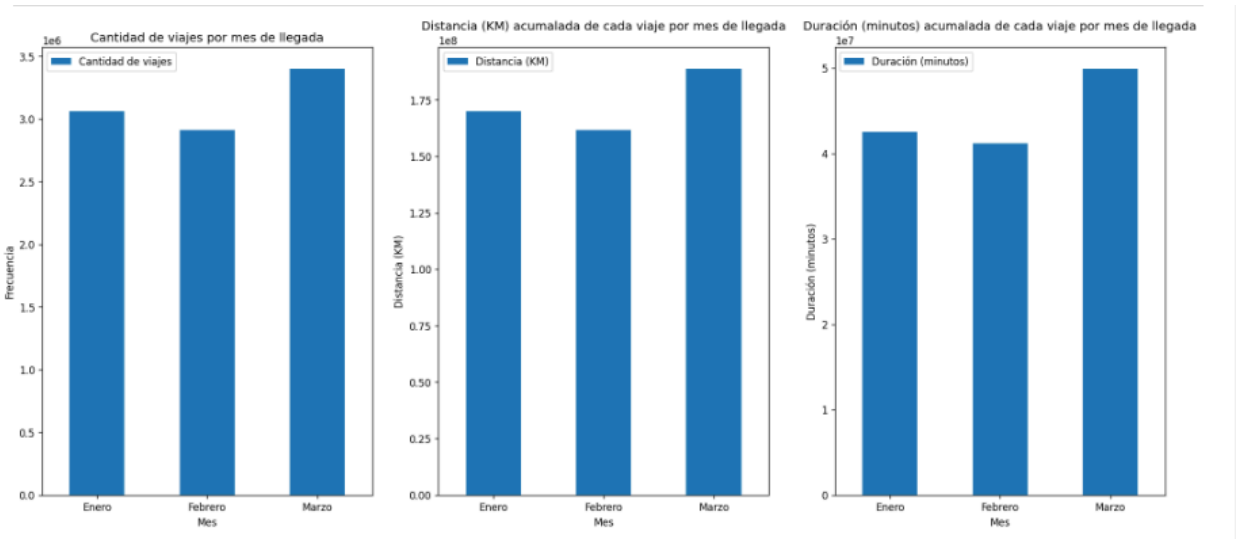
Para el feature **RatecodeID** tenía el 2,5% de los datos faltantes y que también rellenamos con el promedio, tarifa estándar. Además, encontramos un valor mal ingresado (que no se encontraba en la documentación), el ID 99 y viendo que el último ID de la lista el 6, tenía solamente 16 registros, interpretamos que era un valor mal cargado y que correspondía al ID 6 (por ser el último en la lista y además por ser números “parecidos”).

Lo mismo para el feature **store_and_fwd_flag** ("almacenar y reenviar": indica si el registro del viaje se mantuvo en la memoria del vehículo antes de enviarlo al proveedor, porque el vehículo no tenía una conexión con el servidor) tenía el 2,5% de datos faltantes que rellenamos con el valor promedio (“N”), ya que consideramos que no afecta demasiado al análisis.

En el caso del feature **VendorID** no tenía datos faltantes pero si tenía un valor mal cargado, el 6. Ya que según la documentación de los datos los valores posibles son el 1 (Creative Mobile Technologies, LLC.) y el 2 (VeriFone Inc.). Así que lo reemplazamos por el promedio que es el 2.

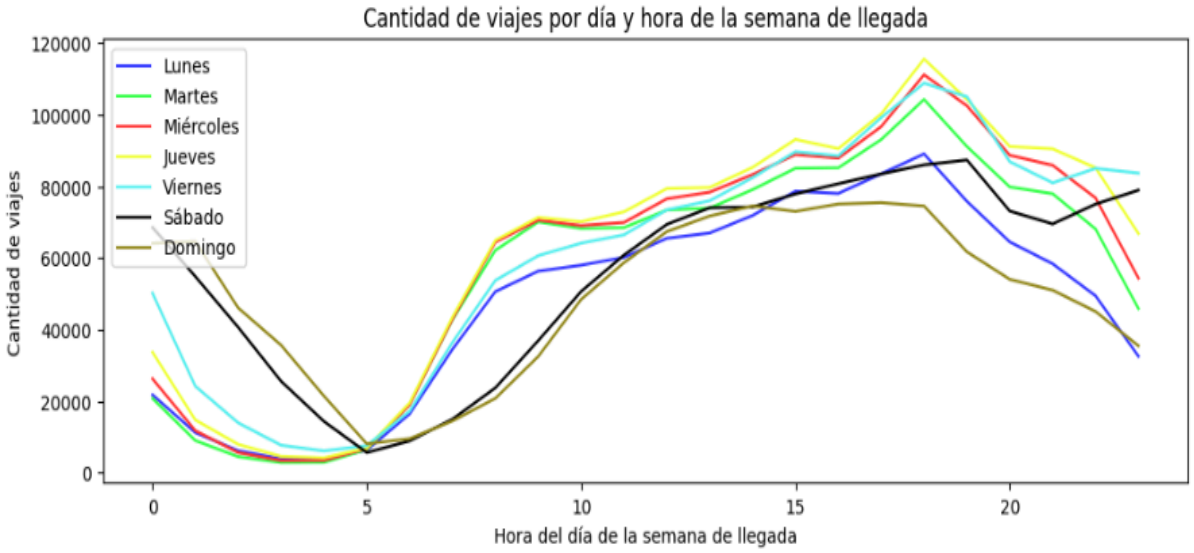
<u>Visualizaciones</u>

● ¿Cantidad de viajes por distancia y duración, respecto al mes de llegada?



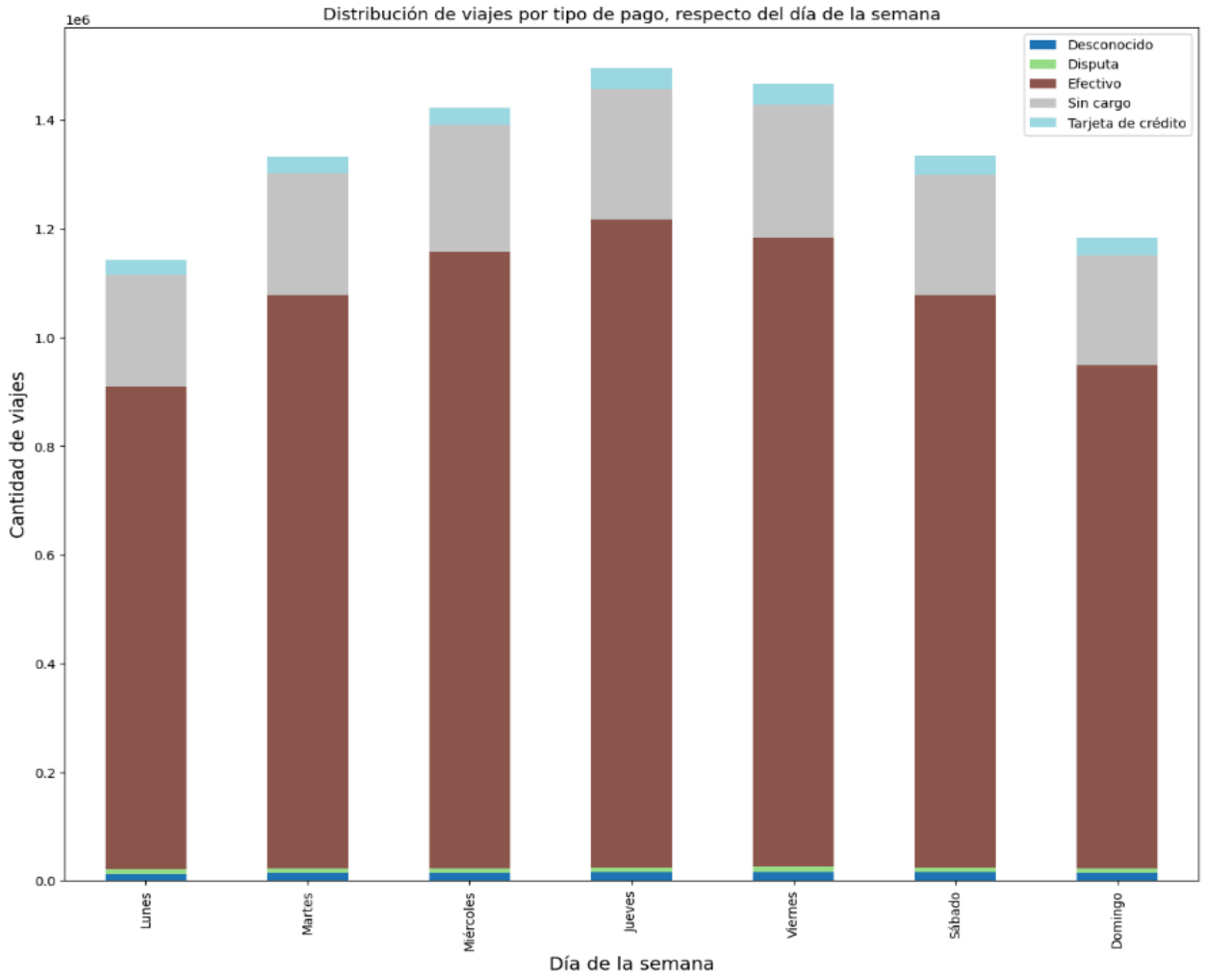
Se puede apreciar que el mes con la mayor cantidad de viajes y con más distancia y duración acumulada es Marzo. Con una cantidad de 3,500,000 viajes acumuló una distancia de 175,000,000 KM y una duración de 50,000,000 minutos (como habíamos visto, un promedio de 14 minutos por viaje).
Luego lo sigue Enero y por último Febrero con la menor cantidad y acumulación.

● ¿Cantidad de viajes por hora y día de la semana de llegada?



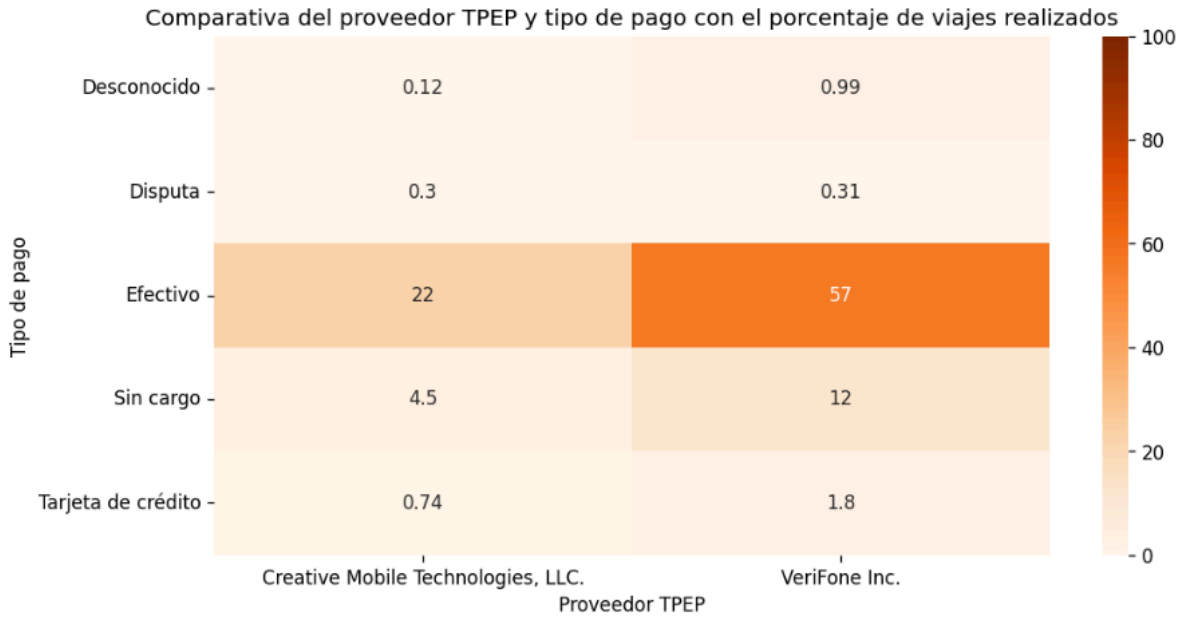
Se puede apreciar que para todos los días, la hora donde hay más viajes es entre las 18 y 19 hs, y las 5hs hay la menor cantidad de viajes. Respecto a los días,Jueves es el que mayor pico tiene, seguido de Miércoles y Viernes, respectivamente. A la madrugada, entre las 00 y 5hs, los fines de semanas son los que tienen mayor cantidad de viajes.

● ¿Cómo es la distribución de los viajes por tipo de pago, respecto del día de la semana de llegada?



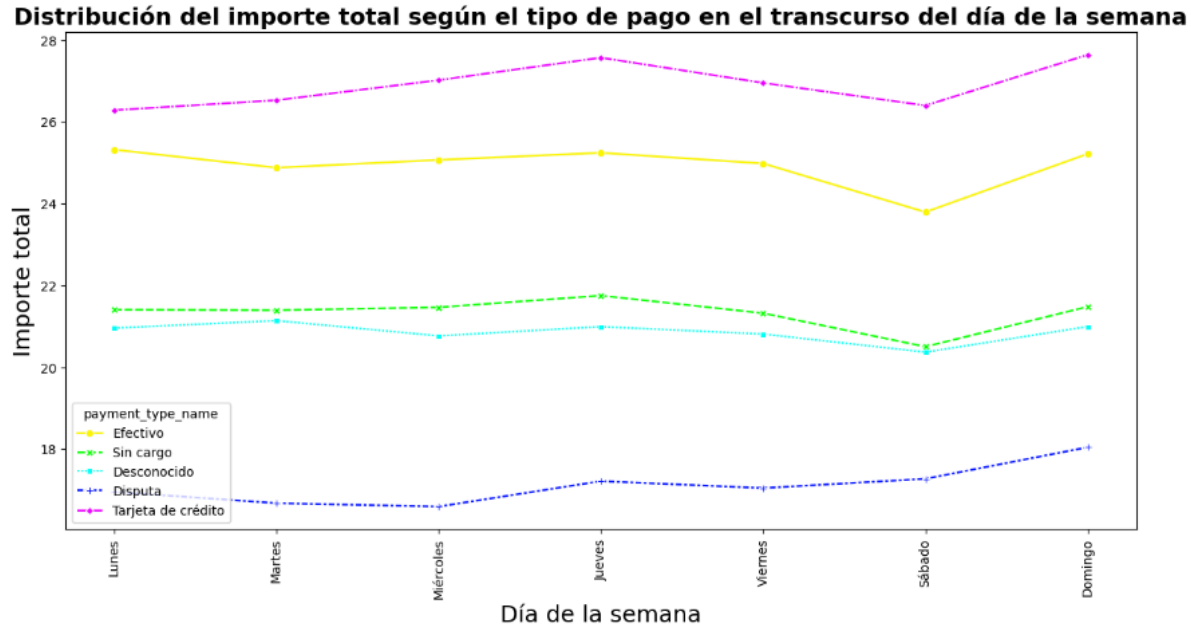
Se puede apreciar que el tipo de pago "Efectivo" es el más utilizado en los 7 días de la semana. Como ya vimos, el Jueves es el que tiene mayor cantidad de viajes, igualmente la distribución de los tipos de pagos son muy similares entre los días.

● ¿Cómo son los viajes por el proveedor de TPEP que proporcionó el registro y tipo de pago?



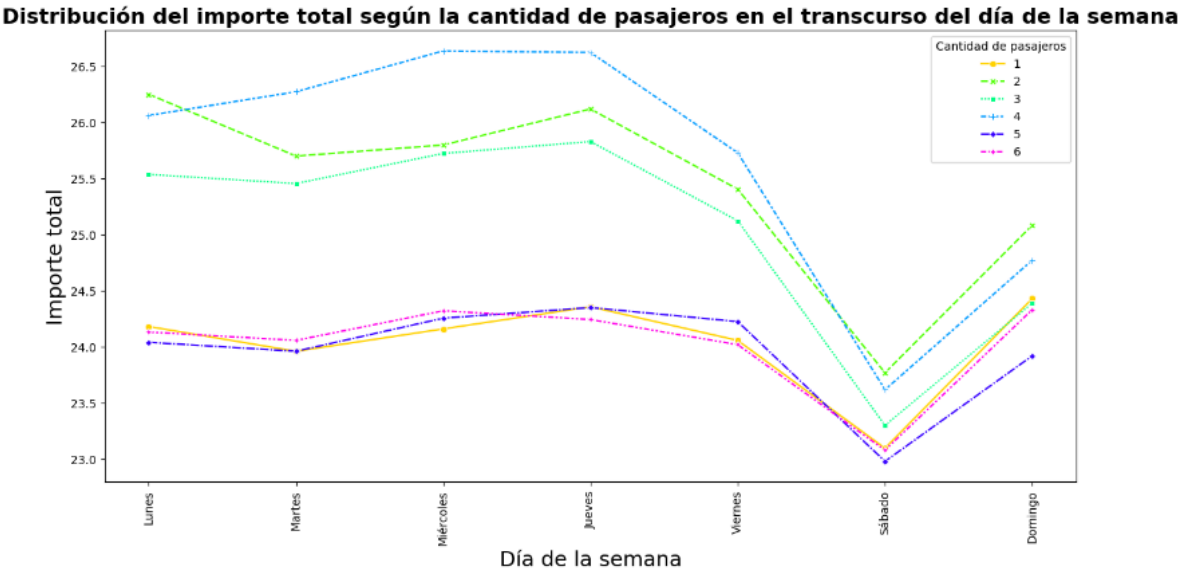
Se puede apreciar, como veníamos viendo que “Efectivo” es el tipo de pago más utilizado para pagar los viajes con el 79%, 57% fue a través del proveedor TPEP VeriFone Inc.

● ¿Cuál es el flujo del importe total según el tipo de pago en el transcurso del día de la semana?



Se puede apreciar que los días donde hay más viajes (Jueves) el importe total con "Efectivo" es un poco más elevado. Se mantiene parejo, excepto los Sábados que son más baratos.
Respecto a los pagos con "Tarjeta de crédito" son un poco más caros, seguramente por el impuesto agregado.

● ¿Cuál es el flujo del importe total según la cantidad de pasajeros en el transcurso del día de la semana?



Se puede apreciar que los días donde hay más viajes (Jueves) el importe total es más elevado con 4 pasajeros por viaje. Se mantiene parejo, excepto los Sábados que son más baratos.
Respecto a la cantidad de pasajeros, los importes más altos son cuando hay entre 4,2 y 3 pasajeros por viaje.

Conclusión:
Podemos concluir que Marzo es un mes clave en términos de volumen de viajes y duración, mientras que el día jueves se destaca como el día con mayor demanda, desde la cantidad de viajes hasta el flujo de pagos y pasajeros. La mayor parte de los viajes se paga en efectivo. También se observa que las horas pico de viajes son entre las 18 y 19 hs, lo que sugiere que se deben considerar estrategias de optimización y planificación para esos períodos críticos.

EJ2: Clasificación

Realizar una breve descripción del dataset: cantidad de registros y columnas, etc. Comentar los features más destacables. Mencionar en cada caso si realizaron transformaciones sobre los datos (encoding, normalización, etc)

Dada un conjunto de datos provenientes de estaciones meteorológicas de Australia, mediante sus datos meteorológicos del día actual, nuestro objetivo es predecir si lloverá o no al día siguiente. Esto se realizará mediante la creación de modelos de clasificación.El conjunto de datos está comprendido por observaciones diarias del clima durante 10 años de distintas localidades de Australia. La variable a predecir es: "RainTomorrow". Es una variable de Yes o No. El criterio de tomar como positivo, o sea Yes, es que en ese día llegó a llover 1mm o más.

En nuestro caso, las Ubicaciones ("Locations") a estudiar son:

- Queensland
- Victoria
- Australia Meridional
- Australia Occidental

El conjunto original del dataset esta compuesto por 23 variables.

Las variables del dataset son:

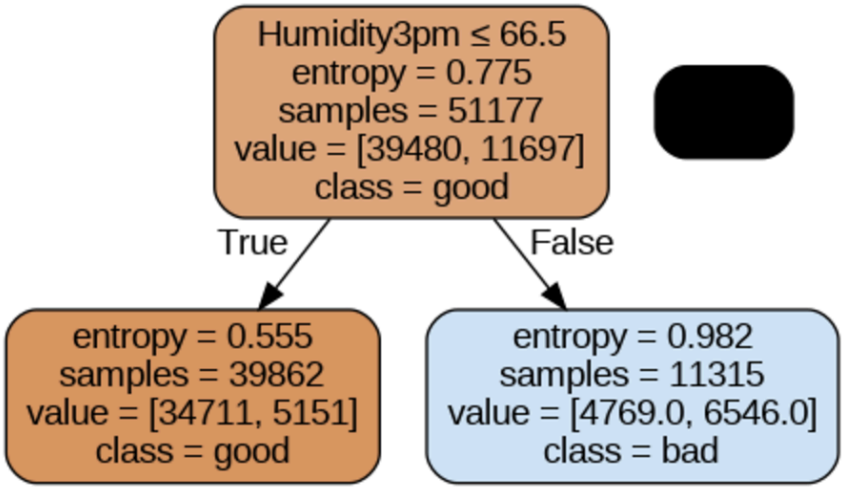
- **Date**: Fecha de la observación de los datos meteorológicos.
- **Location**: El nombre común de la ubicación de la estación meteorológica.
- **MinTemp**: La temperatura mínima en grados centígrados.
- **MaxTemp**: Máxima temperatura en grados centígrados.
- **Rainfall**: La cantidad de lluvia registrada durante el día en mm.
- **Evaporation**: La denominada evaporación en tanque de clase A (mm) en las 24 horas hasta las 9 a. m.
- **Sunshine**: El número de horas de sol brillante en el día.
- **WindGustDir**: La dirección de la ráfaga de viento más fuerte en las 24 horas hasta la medianoche.
- **WindGustSpeed**: La velocidad (km/h) de la ráfaga de viento más fuerte en las 24 horas hasta la medianoche.
- **WindDir9am**: Dirección del viento a las 9 am.
- **WindDir3pm**: Dirección del viento a las 3 pm.
- **WindSpeed9am**: Velocidad del viento (km/h) promediada durante 10 minutos antes de las 9 a. m.
- **WindSpeed3pm**: Velocidad del viento (km/h) promediada durante 10 minutos antes de las 3 p. m.
- **Humidity9am**: Humedad (porcentaje) a las 9 am
- **Humidity3pm**: Humedad (porcentaje) a las 3 pm
- **Pressure9am**: La presión atmosférica (hpa) se redujo al nivel medio del mar a las 9 am
- **Pressure3pm**: La presión atmosférica (hpa) se redujo al nivel medio del mar a las 3 pm
- **Cloud9am**: Fracción del cielo oscurecida por nubes a las 9:00 h. Se mide en "oktas", que son una unidad de octavos. Registra cuántos octavos del cielo están oscurecidos por nubes. Un valor de 0 indica un cielo completamente despejado, mientras que un valor de 8 indica que está completamente nublado.
- **Cloud3pm**: Fracción del cielo oscurecida por nubes a las 15:00 h. Se mide en "oktas", que son una unidad de octavos. Registra cuántos octavos del cielo están oscurecidos por nubes. Un valor de 0 indica un cielo completamente despejado, mientras que un valor de 8 indica que está completamente nublado.
- **Temp9am**: Temperatura (grados C) a las 9 am

- **Temp3pm**: Temperatura (grados C) a las 3 pm
- **RainToday**: Booleano: 1 si la precipitación (mm) en las 24 horas hasta las 9 a. m. supera 1 mm, de lo contrario 0
- **RainTomorrow**: Cantidad de lluvia del día siguiente en mm. Se utiliza para crear la variable de respuesta **RainTomorrow**. Una especie de medida del "riesgo".

Haciendo un análisis de correlación, de cantidad de nulos, se eliminaron las variables de alto contenido de datos nulos. Para las variables que tenían poca cantidad de valores nulos se procedió a eliminar dichos registros. También se eliminaron variables que estaban muy correlacionadas, por lo que no aportarían información extra, este es caso de las variables de temperaturas. Luego se transformó Date por estaciones del año. El conjunto de dataset limpio y filtrado quedó de 17 variables. Luego se procedió a hacer One Hot Encoding para las variables categóricas.
Si bien se intentó imputar valores de forma univariada como multivariada, esto no dieron los resultados esperados, quedamos con las estrategias usadas.

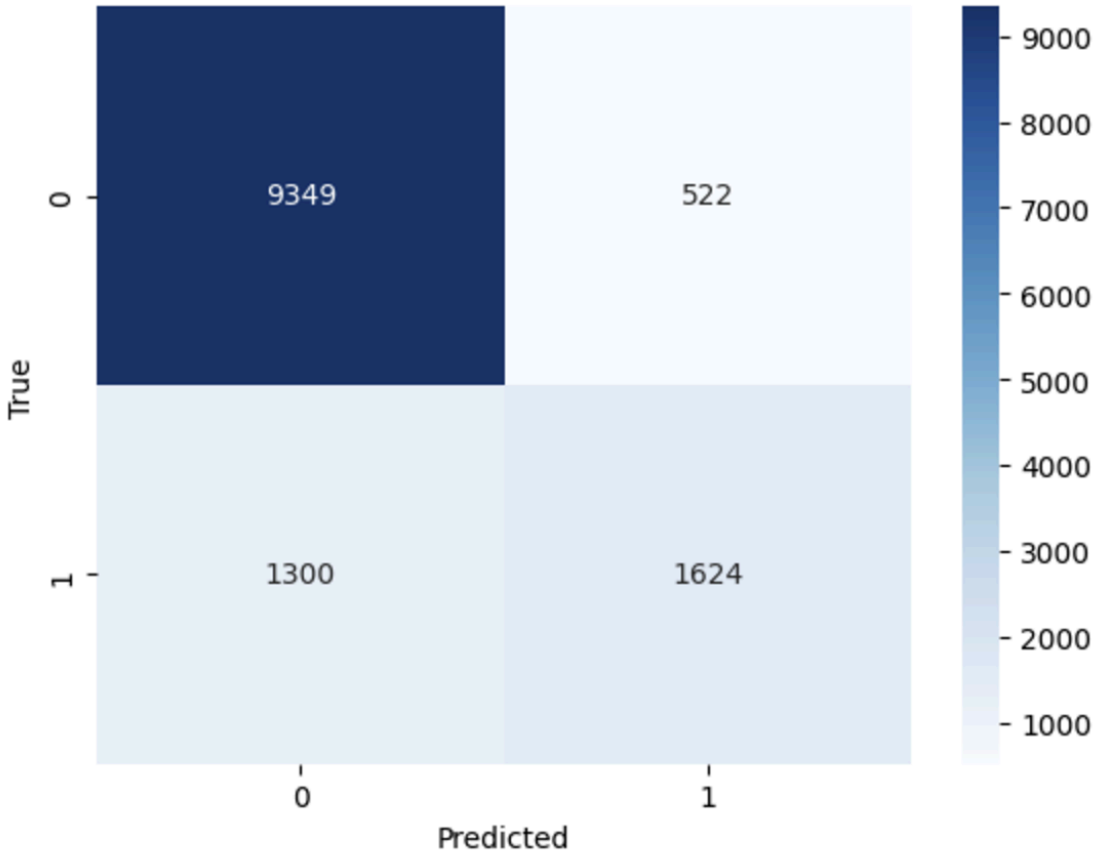
Modelos

1. Arbol de Decision
    - Los hiperparametros a optimizar fueron: criterion, ccp_alpha y max_depth. No se pudieron optimizar más por errores de Timeout.
    - Se utilizó K-fold Cross Validation, de 5 folds.
    - La metrica para optimizar fue f1-score.
    - Mejores hiperparametros encontrados: {'max_depth': 4, 'criterion': 'entropy', 'ccp_alpha': 0.033}
    - Las variables más representativas a la hora de clasificar en el árbol fueron: Humidity3pm, WindGustSpeed y Pressure3pm.



2. Random Forest

- Por GridSearch se optimizaron los hiperparametros: criterion, max_features, min_samples_leaf, min_samples_split, n_estimators.
- Los mejores hiperparametros hallados fueron:
    i. 'criterion': 'gini',
    ii. 'max_features': 'sqrt',
    iii. 'min_samples_leaf': 1,
    iv. 'min_samples_split': 10,
    v. 'n_estimators': 20
- Se utilizó K-fold Cross Validation con 5 folds.
- La métrica para buscar los hiperparametros fue accuracy, f1 y roc_auc.
- Las métricas obtenidas fueron:
    i. Accuracy: 0.858
    ii. Recall: 0.555
    iii. f1 score: 0.641
    iv. Precision: 0.766
- Matriz de confusión



3. XGBoost
    - Optimizamos los hiperparametros: learning_rate, max_depth, min_child_weight, subsample, colsample_bytree y n_estimators.
    - Se utilizó K-fold Cross Validation con 5 folds.

- La métrica que utilizamos para buscar los hiperparametros fue roc_auc.

Mejores modelos

- Árbol de decisión: {'max_depth': 5, 'criterion': 'entropy', 'ccp_alpha': 0.033}
- Random Forest: {'criterion': 'gini', 'max_features': 'sqrt', 'min_samples_leaf': 1, 'min_samples_split': 10, 'n_estimators': 20}
- XGBoost: {'criterion': 'gini', 'max_features': 'sqrt', 'min_samples_leaf': 1, 'min_samples_split': 4, 'n_estimators': 20}

Cuadro de Resultados

Modelo	F1-Test	Precision Test	Recall Test	Accuracy Test
Arbol de Decision	0.57	0.57	0.57	0.80
Random Forest	0.64	0.77	0.56	0.86
XGBoost	0.67	0.77	0.60	0.87

Elección del modelo

Elegimos los modelos de ensamble XGBoost por tener mejores métricas a nivel general.

EJ3: Regresión: AirbnbNY

a) Análisis exploratorio y pre-procesamiento

Datos faltantes

Es un dataset sobre airbnb de New York, que presenta una tabla de (37541,18), en el cual la columna que nos interesa predecir es la de ***price***, nos encontramos que price presenta muchos valores nulos que representa ***39,29%*** y como es la columna a predecir decidimos descartar dichos datos nulos, ya que imputar alteraría nuestras predicciones.

Por otro lado también vemos que la columna **license** presenta más del 85% de datos nulos, por lo que hemos decidido no trabajar con dicha columna.

Por último las columnas **last_reviews** y **reviews_per_month** presenta ambas 30.70% (11540 registros) de datos nulos, por lo que decidimos llenarlo con el valor de moda

Datos outliers

Datos outliers univariados: usamos el método IQR para las columnas price, ya que había pocos datos que tenían un precio exageradamente alto

Por otro lado las columnas **minimum nights** decidimos quedarnos con aquellos donde pedian 365 noches minimo.

Relación de price con los nulos de last_review y reviews per month

Para ver la relación de estas 3 columnas, primero trabajamos con aquellos datos no nulos de last review y reviews per month. A través de 3 gráficos:

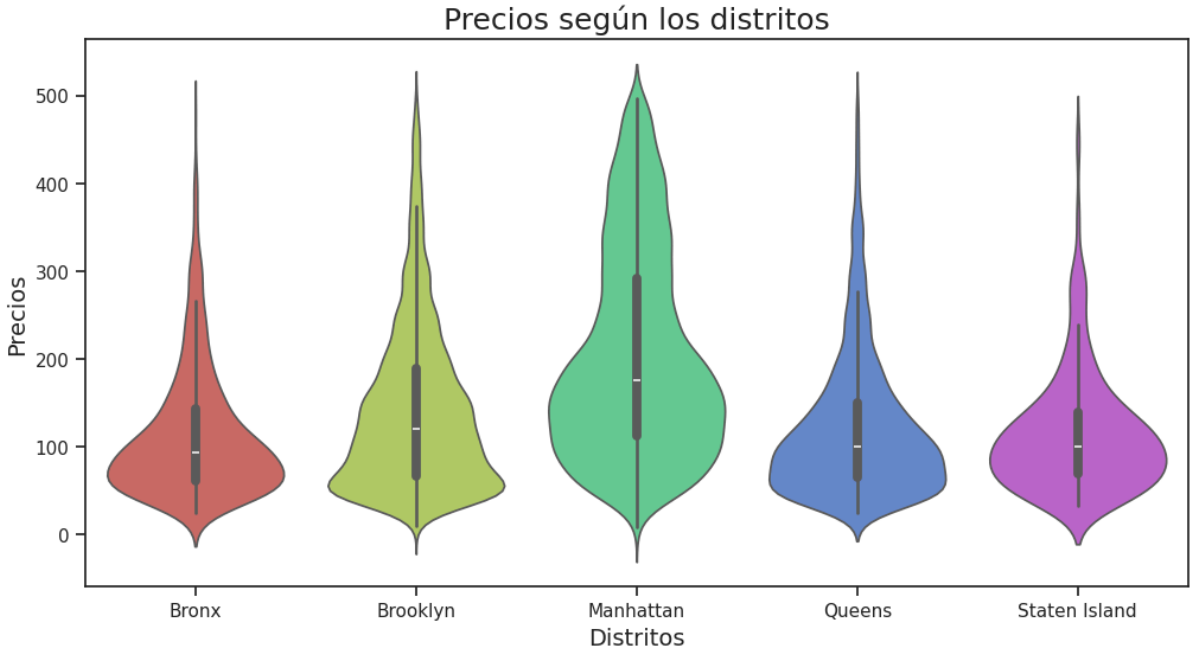
- **1- ScartterPlot :** hemos creado el feature days since last review y asi poder ver la distribución de las reviews en todo el tiempo en relación al precio y hemos visto que la gran concentración y variedad de tipo de viviendas se mantiene dentro de los 2000 días desde su última reseña
- **2- Heatmap:** price con el resto de las variables numéricas no presentan una fuerte relación

- **3- LinePlot:** analizamos cómo cómo evolucionó el precio en el transcurso de los años de reseña para las distintas viviendas, en este caso creamos el feature trimestre usando la columna last review y vimos que las reviews antes del 12 del 2018 no hay registro de las viviendas del tipo Hotel room. A su vez en USA la etapa de pandemia se considera de 01/2020 al 09/2022, por lo que se considera años atípicos no solo en precio sino también en demanda y esto puede traer malas predicciones en nuestro caso.
  Entonces por visualizaciones 1, 2 y 3, hemos decidido trabajar con los datos que tienen registros post-pandemia y aún así nos queda luego imputar 5840 datos nulos en las columnas last_reviews y reviews per month. Para esta imputación vamos a considerar grupos similares como el precio, tipo de vivienda y zona, considerando la moda de la fecha para esos grupos similares y si no existe una moda consideramos tomarlo como la última fecha, ya que son usuarios que aún no han realizado ninguna reseña

<u>Relación de price con las variables categóricas</u>

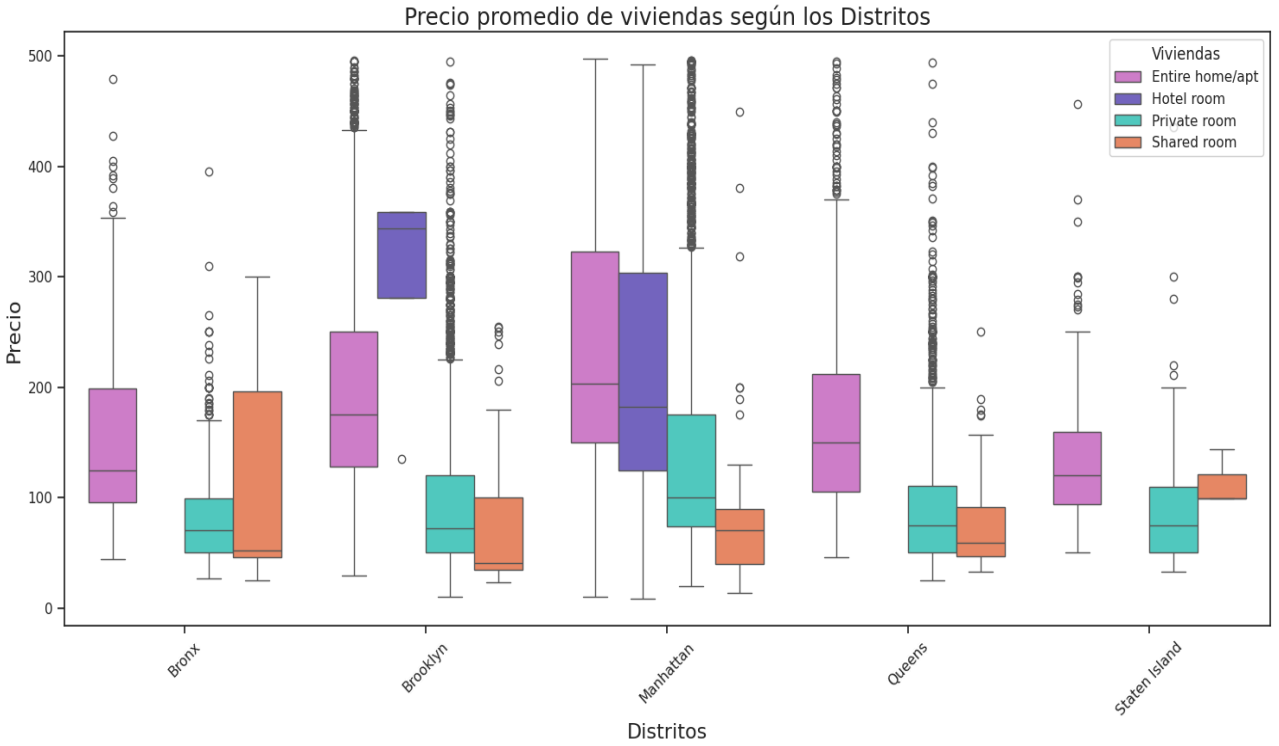
-
- **Variable neighbourhood_group**: para esto decidimos ver los sgte:

  ¿Cúal es el precio promedio del alquiler de en los distintos distritos?
  Manhattan es el distrito más caro con un precio promedio de de 201.73, seguido por Brooklyn con un precio promedio de 142.33 y el más económico es Bronx con un promedio de 110.19



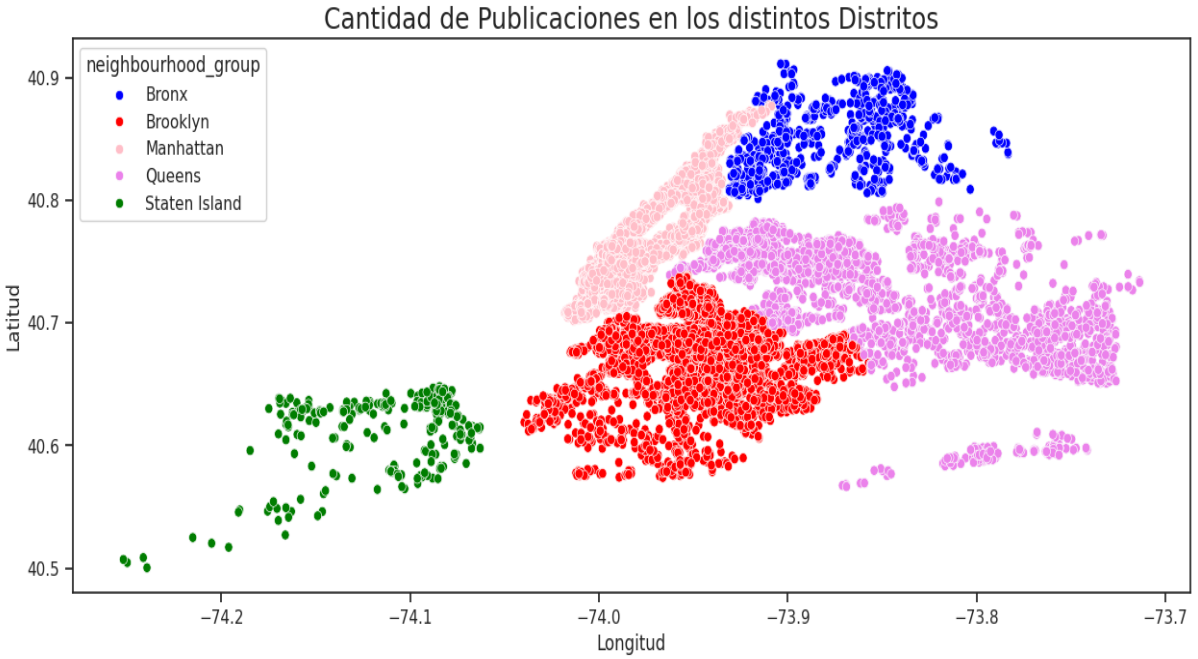
¿Cúal es el precio promedio de alquiler para las distintas viviendas en cada distrito?

1. En Bronx, Queen, Staten Island no hay ofertas de Hoteles.
2. Manhattan tiene el alquiler mas alto respecto a la vivienda del tipo Entire home con un valor promedio de 234.64 y private room con un precio promedio de 142.41. Por otro lado, Staten Island tiene el precio promedio mas baja respecto a la vivienda Entire home con un valor de 138.84
3. El Bronx tiene el precio promedio mas bajo respecto a la vivienda del tipo private room con un valor de 83.40



¿Qué distritos tienen más publicaciones?

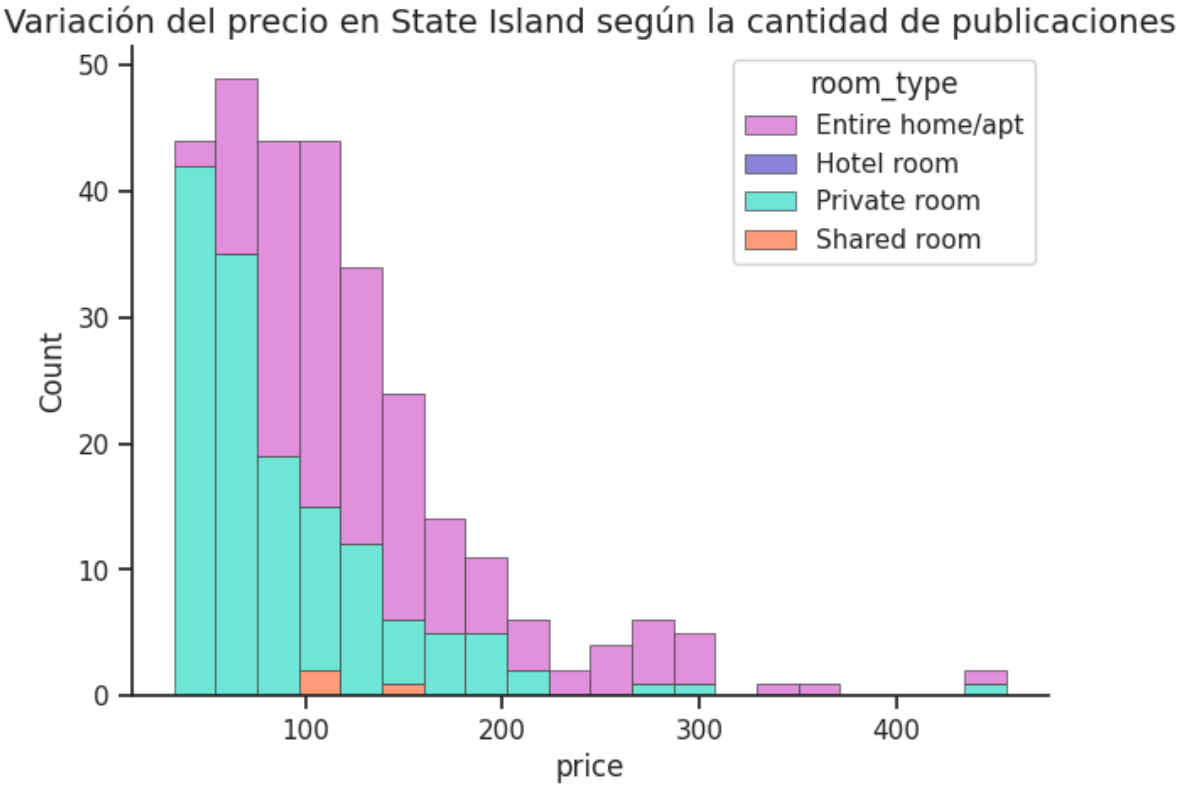
Se observa que Manhattan tiene más publicaciones con un registro de 7997, seguido por Brooklyn con 7132 y el que menos tiene es Staten Island



- **Variables room_type y neighbourhood:**

¿A mayor cantidad de publicaciones, baja los precios? (Ley de oferta y demanda)

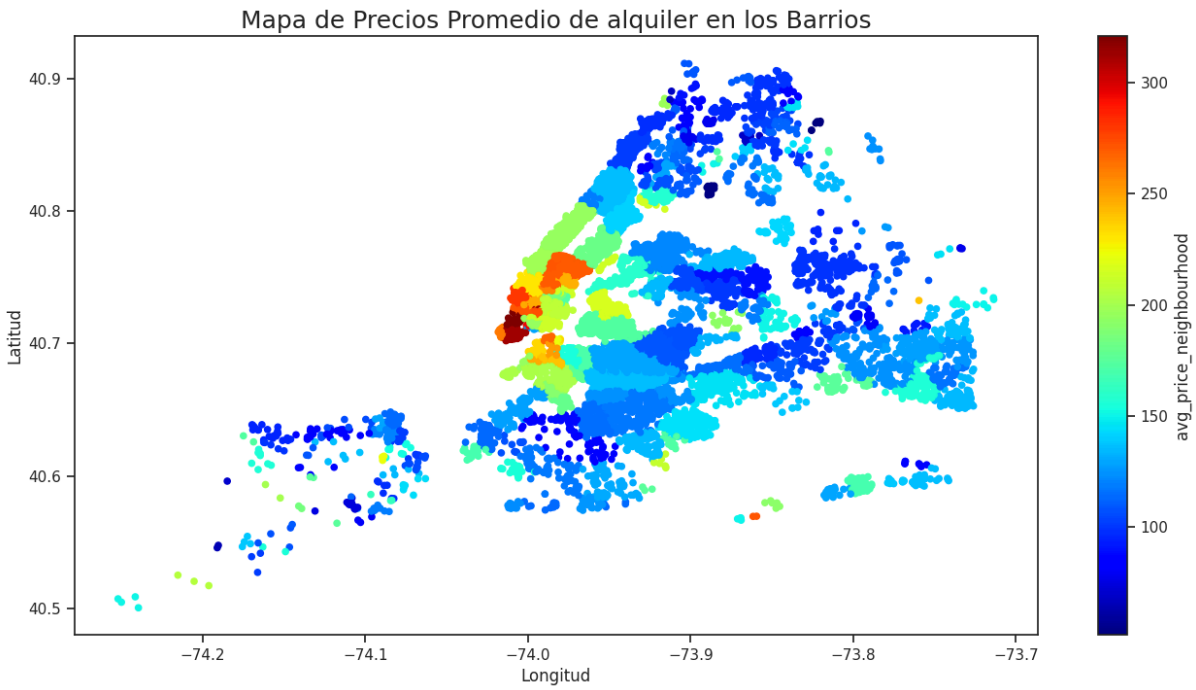
En gral si se cumple, abunda más las publicaciones de menor precio y pocas de mayor precio



¿Qué distritos tienen los barrios más caros y más baratos?

Se puede observar que en Manhattan,Brooklyn y Queens se encuentra los barrios donde es más alto el alquiler

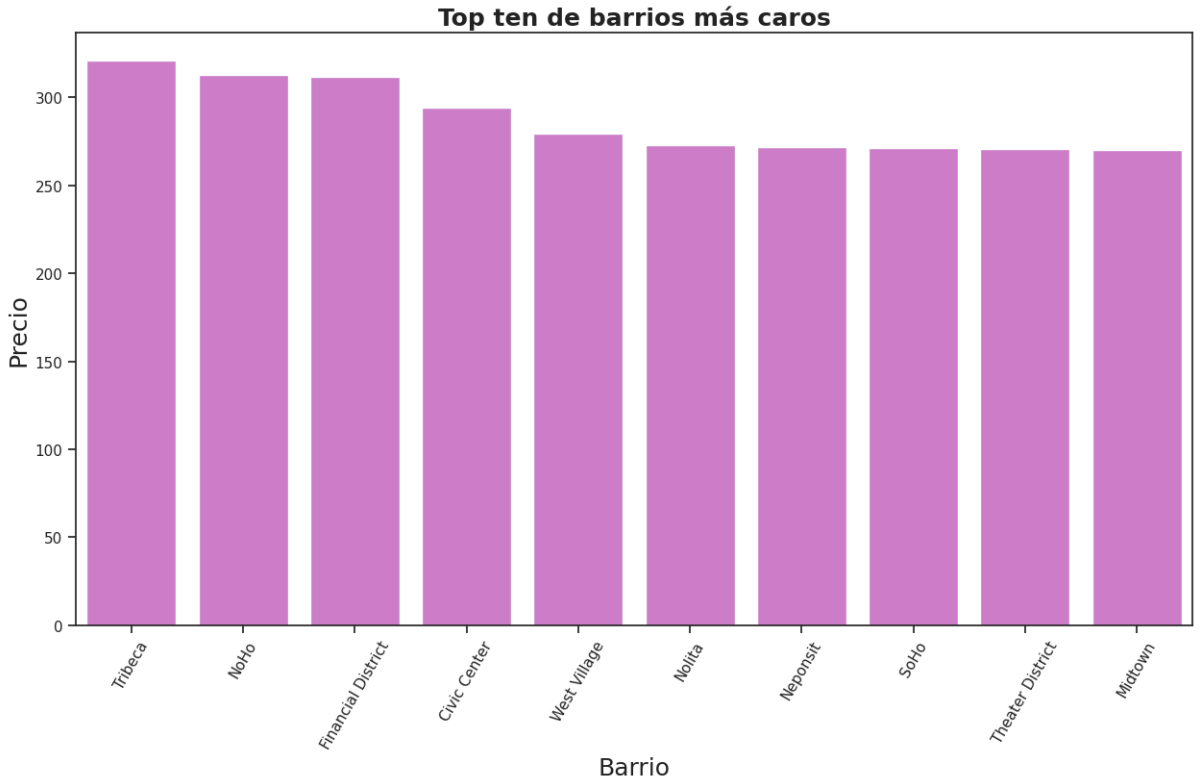
Lo más economicos esta en el distrito de Staten Island



Top ten de los barrios más caros y más ecocómicos

Podemos observar que el barrio más caro es Tribeca y se encuentra dentro de Manhattan con un promedio de $320.91

Dentro de los barrios más economicos se encuentra Co-op City que se encuentra dentro de Bronx y Staten Island con un precio pomedio de $51.50



Features nuevos

Se crea estos features con el fín de tener un un modelo predictivo más robusto y eficiente.

- **tipo_anfitrion:** identifica si es un propietario o un negocio inmobliario
- **total_public_neighbourhood_typeroom:** total de publicaciones por barrio
- **trimestre:** agrupa los registros de forma trimestral

b) Entrenamientos

Encodeo: Hemos usado 2 tipos de encodeo para las variables categóricas

1. OneHotEncoder: usaremos para los features que tienen menos de 10 categorias, se usará OneHotEncode
2. MeanEnconding: usaremos para los features con más de 10 categorias se usará MeanEncoding

Features seleccionados: room_type, neighbourhood_group, neighbourhood, number_of_reviews, number_of_reviews_ltm, total_public_neighbourhood_typerrom, trimestre, tipo_anfitrion

Variables de entrenamiento y testeo: cuando separamos las variables en test y train, previamente usamos una transformación logarítmica a nuestras variables y_test e y_train, para no tener una distribución sesgada o no normal, en el cual para el modelo va ser difícil ajustarse

Evaluación de modelo: para todos los modelos se va evaluar los sgtes errores:

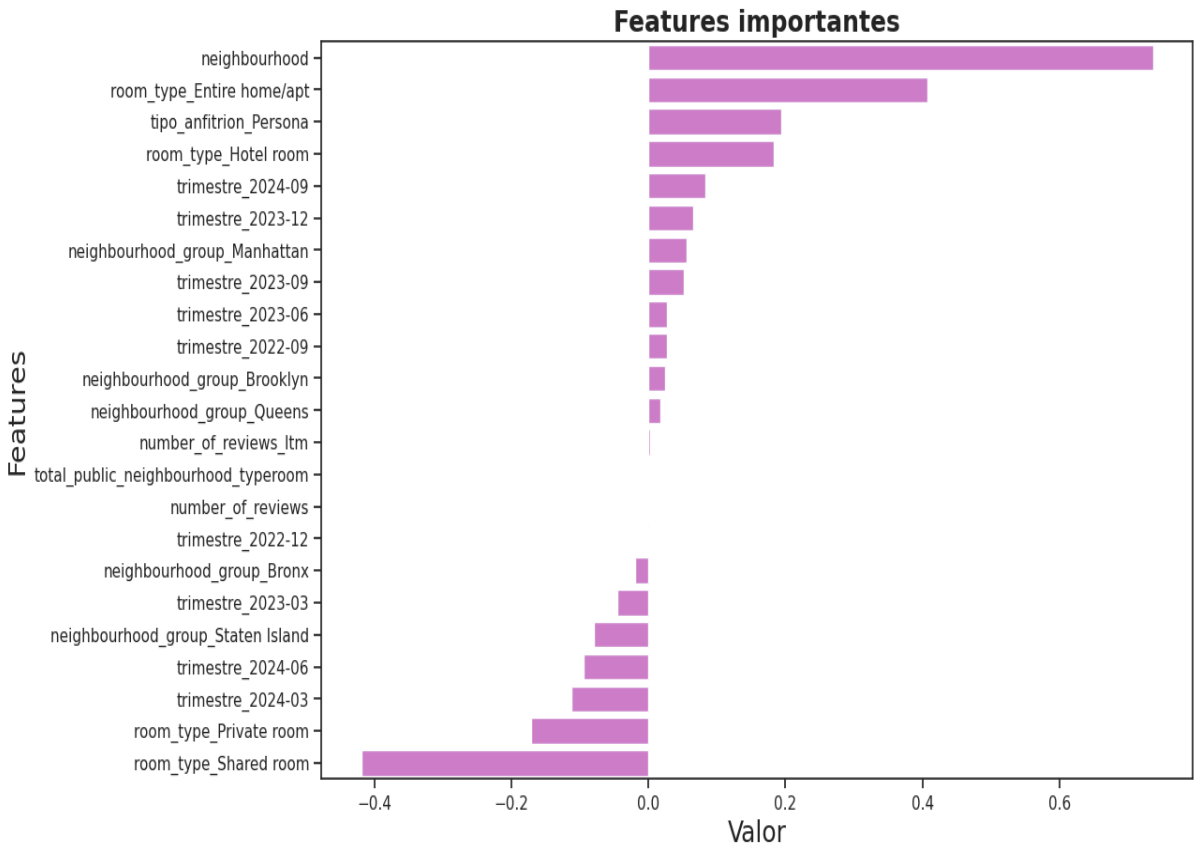
1. Error Cuadrático Medio (MSE)
2. Error Absoluto Medio (MAE)
3. Error Raiz Cuadratica Medio (RMSE)
4. Error coeficiente de determinación (R^2)

Modelos

1. Regresión Lineal
    - Construir un modelo Regresión Lineal Múltiple, identificando los features más importantes para predecir el precio del alquiler.

    En este caso vamos a probar con 2 tipos de funciones, una lineal y otra de transformación polinómica para ver la performance y flexibilidad que tiene la polinómica respecto a la lineal

    Se puede ver que la feature mas importante influye el barrio **(neighbourhood)**, seguido por el tipo de vivienda **Entire home/apt** y de dueño directo, esto es posible, ya que una inmobiliaria también cobra una comisión. El feature nuevo que era cantidad de publicaciones por barrio, se ve que no es de suma importancia y también notamos que las personas buscan más por la zona de **Manhattan**



- Evaluar la performance del modelo en el conjunto de evaluación, explicar todas las métricas. Comparar con la performance de entrenamiento.

La regresión lineal polinómica supera a la regresión lineal simple en todas las métricas de evaluación (MSE, RMSE, MAE, y R^2). El modelo polinómico logra una mejor precisión de predicción y explicación de la variabilidad de los datos en comparación con la regresión lineal simple. Aunque la diferencia no es enorme, el modelo polinómico es más adecuado para predecir el precio del alquiler de un Airbnb en función de las métricas obtenidas. La transformación polinómica permite al modelo capturar mejor las relaciones no lineales entre las características y el precio, lo que mejora su desempeño.

2. XGBoost
   - ¿Utilizaron K-fold Cross Validation?¿Cuántos folds utilizaron? ¿Qué métrica utilizaron para buscar los hiperparámetros?
   - Evaluar la performance del modelo en el conjunto de evaluación, explicar todas las métricas. Comparar con la performance de entrenamiento.

RTA: Hemos probado el modelo con diferentes hiperparametros y optimizado con Cross Validation. La opción 1 nos dió mejor performance. En el cual usamos 5 folds, con los sgtes hiperparametros: learning_rate: 0.05, n_estimators: 200, subsample:0.8,max_depth: 6

y la métrica que consideramos adecuada para buscar hiperparametros es **RMSE** ya que es una métrica muy confiable para ver el error promedio en las predicciones.

Por otro lado R^2 nos dá una buena indicación de cuán bien está nuestro modelo

Como los errores de predicción son relativamente bajos y R^2 tiene casi el 60%, nos da indicio que el modelo tiene un desempeño bueno.

3. Modelo a elección:
    - En este punto se debe entrenar (mediante cross-validation) un modelo elegido por el grupo. Se debe evaluar su performance en entrenamiento y sobre el conjunto de evaluación explicando todas las métricas.

      Para este pto hemos decidido usar Árbol de decisión y Random Forest, ya que queriamos comparar la performance de ambos modelos, considerando los mimos hiperparametros y cross-validation

RTA: Para el modelo de Arbol de decisión nos dió que tiene un rendimiento excelente en el conjunto de entrenamiento, con métricas bajas de error (MSE, RMSE, MAE) y un alto R^2. Esto nos indica que esta sobreajustados, ya que el modelo puede aprender los detalles específicos de los datos de entrenamiento. Sin embargo, cuando se evalúa en el conjunto de prueba, el rendimiento del modelo disminuye significativamente. Esto sugiere que el modelo no es capaz de generalizar bien y que ha sobreajustado (overfitting) los datos de entrenamiento.

Para mejorar el modelo, tenemos que ir a un modelo más complejo como Random Forest o XGBoost, o realizar ajustes en los hiperparámetros al árbol de decisión.

Cuando modelamos con Random Forest muestra un rendimiento superior en el conjunto de prueba en comparación con el Árbol de Decisión. Es más preciso, con un menor error (MSE, RMSE, MAE) y un mejor R^2. En base a estas métricas, Random Forest sería el mejor modelo a diferencia de Arbol de decisión y lo elegimos para modelar ya que su predicción en el precio de alquiler de un Airbnb en esta ciudad tiene mejor rendimiento general y es más capaz de generalizar a nuevos datos.

C) Cuadro de Resultados

Realizar un cuadro de resultados comparando los modelos que entrenaron (entre ellos debe figurar cuál es el que seleccionaron como mejor predictor).

RTA: en función a las métricas, eligiriamos XGBoost ya que:

- MSE (Error Cuadrático Medio): XGBoost tiene el MSE más bajo (0.183378), lo que indica que en promedio sus predicciones son las más precisas.
- RMSE (Raíz del Error Cuadrático Medio): XGBoost también tiene el RMSE más bajo (0.428226), lo que muestra que, en promedio, las predicciones están más cerca de los valores reales.
- MAE (Error Absoluto Medio): XGBoost presenta el MAE más bajo (0.330425), lo que significa que tiene menos errores absolutos en las predicciones, lo que mejora la precisión global.
- R^2 Score (Coeficiente de Determinación): XGBoost tiene el R^2 más alto (0.589600), lo que indica que el modelo es capaz de explicar un mayor porcentaje de la variabilidad de los precios de alquiler.

Por otro lado vemos que Random Forest, tiene un buen desempeño, con un R^2 de 0.557927 pero aún así XGBoost lo supera en todas las métricas. Esto hace que XGBoost sea una opción más fuerte en este caso. Árbol de Decisión: Tiene un MSE y un R^2 considerablemente peores que XGBoost, por lo que no sería una opción preferible. Regresión Lineal Polinómica: Aunque la regresión polinómica mejora la

regresión lineal simple, su desempeño no es tan bueno como el de XGBoost, especialmente en el MSE y el R^2.

Medidas de rendimiento en el conjunto de TEST:

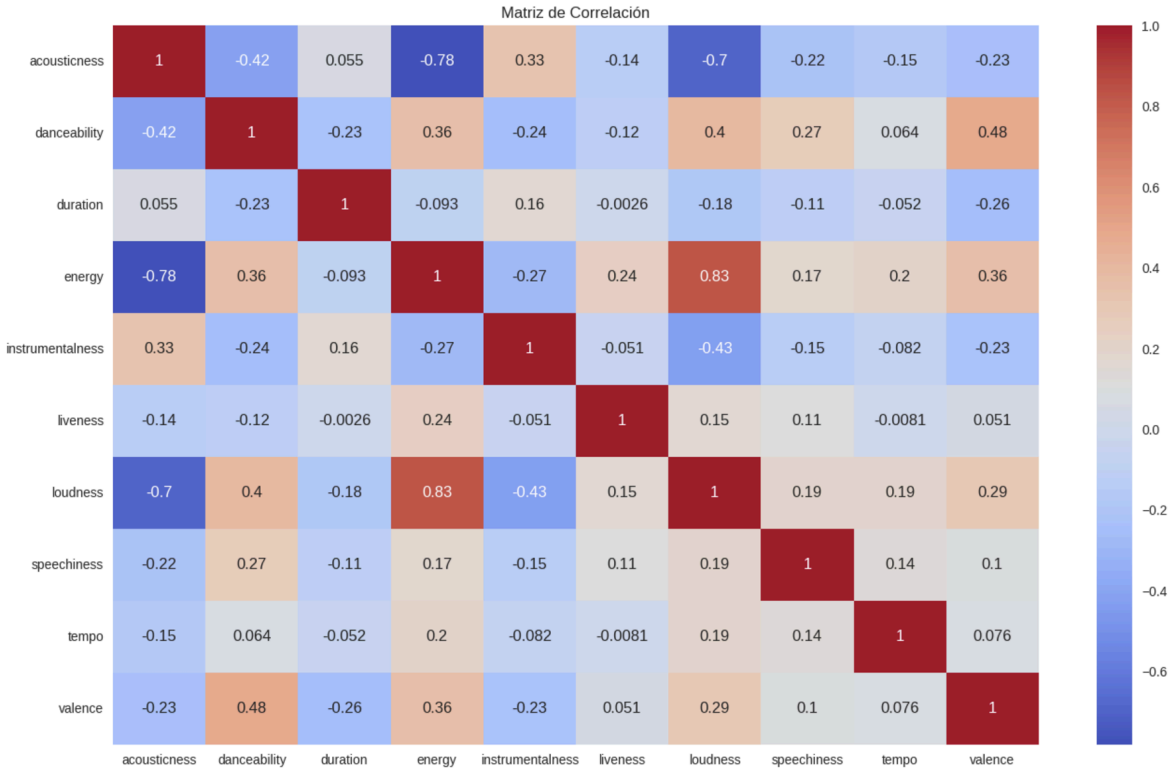
Cuadro comparativo de los modelos

Modelos	MSE	RMSE	MAE	R^2 Score
Regresión Lineal	0.239249	0.554729	0.386307	0.464561
R.L c/transformación polinomica	0.215402	0.554729	0.360873	0.517929
XGBoost	0.183378	0.428226	0.330425	0.589600
Arbol de decisión	0.302073	0.549612	0.404433	0.323959
Random Forest	0.197530	0.444444	0.336104	0.557927

EJ4: Clustering

El objetivo de este ejercicio es encontrar posibles agrupaciones de datos. Para ello primero realizamos un análisis mediante la estadística de Hopkins. El valor hallado dio alrededor de 0.92 , indicando así una fuerte tendencia a clusters de datos.

Tambien se realizo un analisis de correlacion entre los features

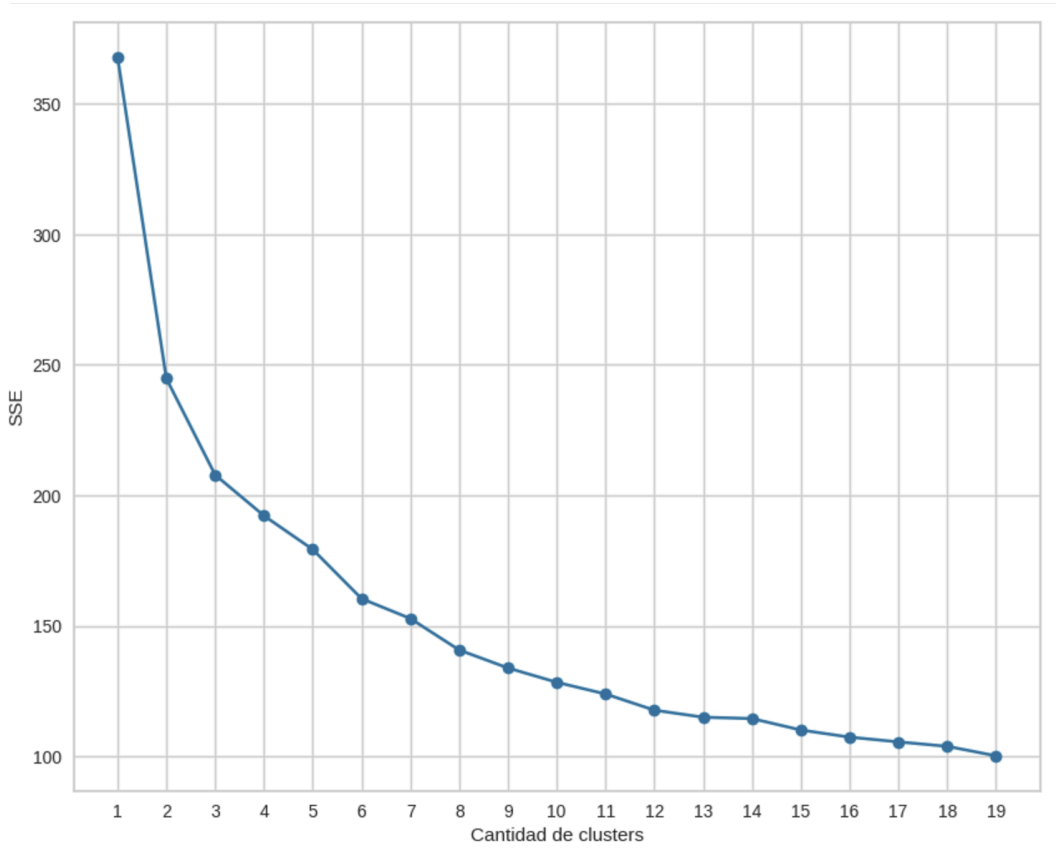


Se observaron fuertes correlaciones con los features:

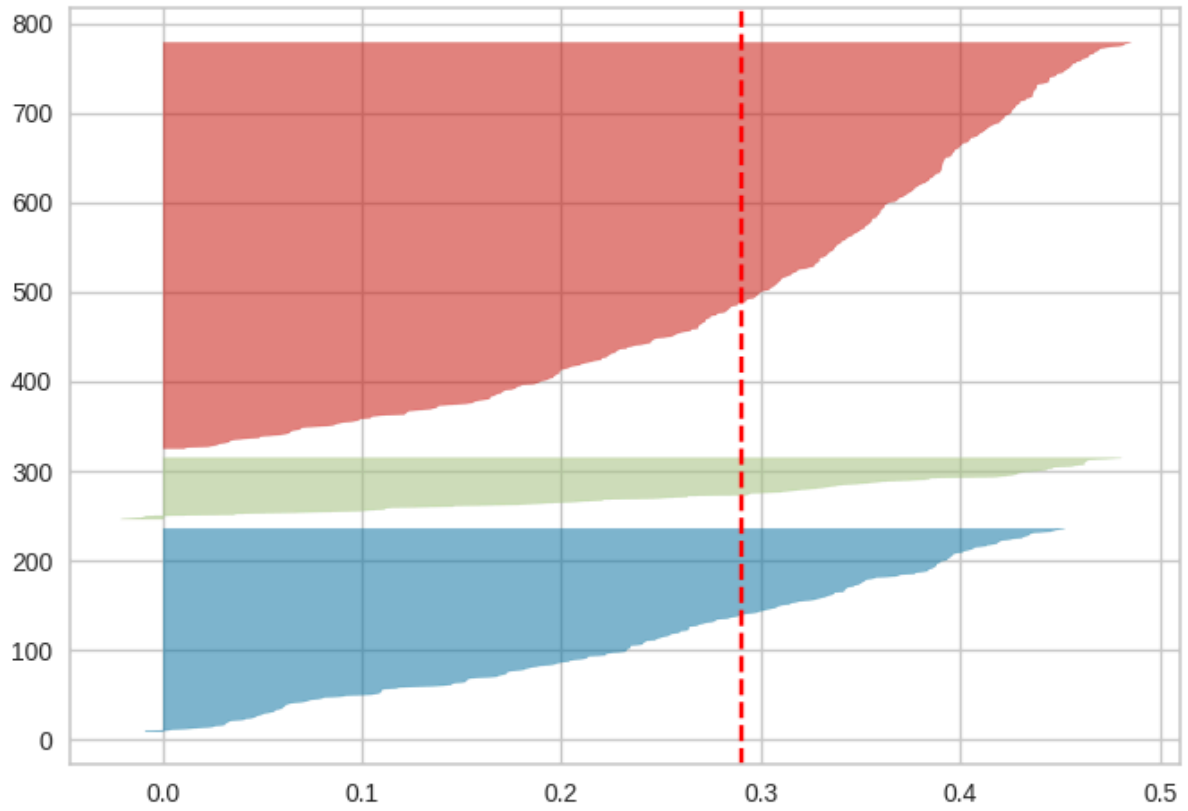
- energy vs acousticness (inversa)
- loudness vs acousticness (inversa)
- loudness vs energy (directa)
- danceability vs valence (directa)

Estas observaciones nos dan idea de como podrian estar agrupados en clusters nuestro dataset. O sea tener presentas las posibles fuerzas de armado de clusters.

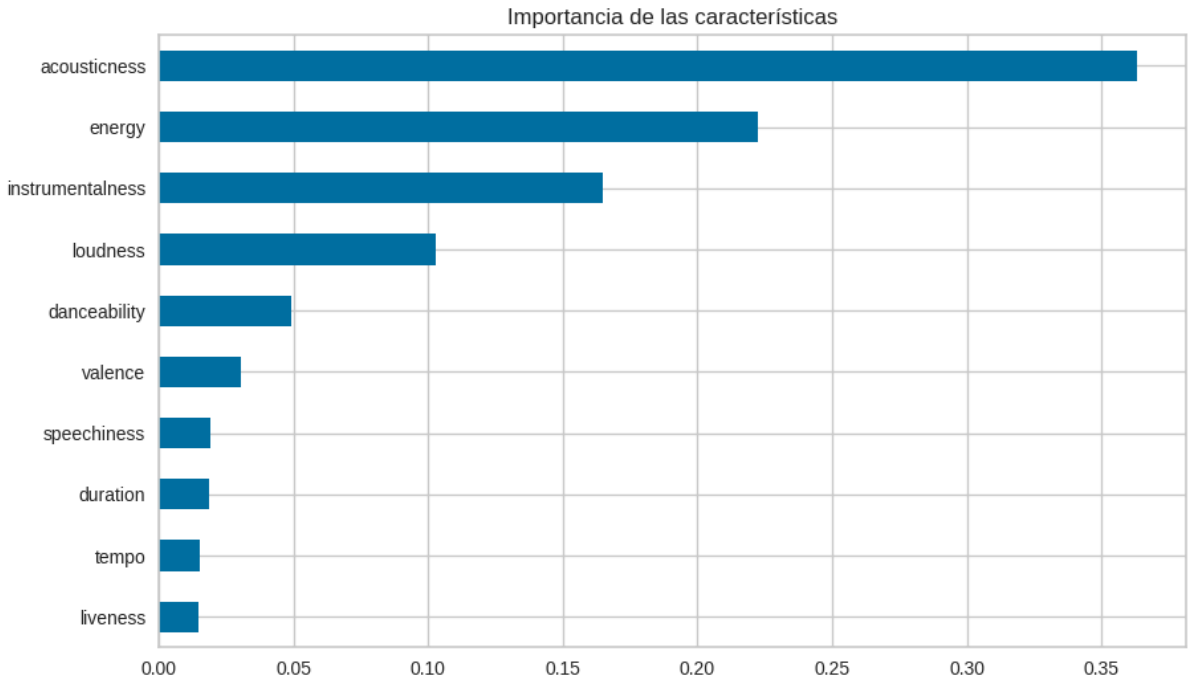
Para hallar la cantidad de grupo de datos óptima, realizamos el análisis del método de Elbow.



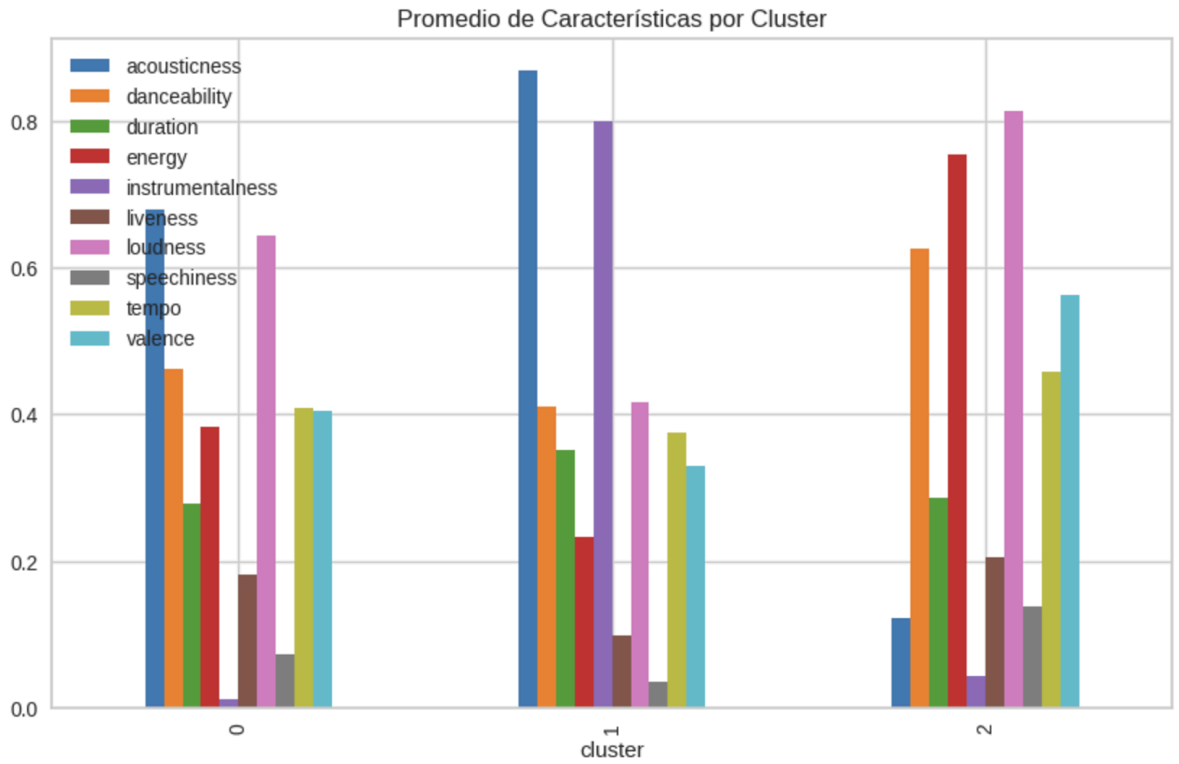
Para seguir investigando la cantidad de grupos de datos, usamos el análisis de mediante el índice de Silhouette para cada cantidad. Los de mayor valor dieron para clusters de cantidad n=2 y n=3. Para ambos dieron valores muy similares. Para definir la cantidad realizamos un gráfico de Silhouette y nos quedamos con aquel que tenga mayor cantidad de valores positivos de silueta, en este caso para n=3.



Se analizaron los centroides de dichos clusters y también a través de un análisis de random forest, se busco la importancia de las variables como muestra en el siguiente gráfico:



Tambien se realizo un grafico de promedio de caracteristicas que tenia cada grupo. Estos valores son los normalizados.



Las fuerzas obtenidas por el análisis de correlación fueron las que terminaron definiendo los clusters.

Se observaron 3 clusters con las siguientes caracteristicas:

- k1: cluster marcado por loudness y acousticness. Suponemos que son personas que les gusta escuchar música de forma intensa y con pasión.
- k2: cluster marcado por instrumentalness y acousticness. Suponemos que son personas que buscan música para relajarse o quizás enfocarse mientras hacen alguna tarea.
- k3: cluster marcado por loudess, energy y danceability. Suponemos que esun grupo de personas que buscan música para bailar.

Tiempo dedicado

Integrante	Tarea	Prom. Hs Semana
Mejia Alan	Análisis de datos, procesamiento de datos, entrenamiento de modelos, realización de gráficos y análisis de métricas.	8 hs
Prieto Pablo Alejandro	Análisis de datos, procesamiento de datos,	8 hs

	realización de visualizaciones y análisis de los resultados.	
Sosa Zoraida Flores	Análisis de datos, procesamiento de datos, realización de visualizaciones	8 hs