# 75.06 Organización de Datos

1C - 2019

Trabajo Práctico 1

## Análisis Exploratorio de Datos

GRUPO 2

Rodríguez, Agustín

Scetta, María de los Ángeles

GitHub: https://github.com/nonsignificantp/organizacion_datos

# Introducción

Durante el primer cuatrimestre de 2019 realizamos un análisis exploratorio sobre datos provenientes de Jampp, una compañia dedicada al *retargeting* y *engagement* de apps móviles. Jampp nos proporcionó 4 datasets diferentes, cada uno de ellos representa una etapa en el modelo de negocios de *demand-side platforms*. Las etapas que se describen son:

1. **Auctions**: Registro de todas las *real-time biddings* que ocurrieron en el *ad-exchange* durante el periodo de 1 semana comprendida entre 2019-03-05 00:00:00 y 2019-03-13 23:59:59 UTC.
2. **Events**: Registro de todas las interacciones que el usuario realiza dentro de una aplicación móvil cliente de Jampp.
3. **Clicks**: Registro e información relacionada de todos los clicks que ocurrieron sobre un ad impreso de Jampp.
4. **Installs**: Eventos explicitos o implicitos de instalaciones de apps promocionadas por Jampp.

Acompañando a los datasets, Jampp incorporo metadatos descriptivos de cada una de las columnas presentes en ellos. Es importante remarcar que, para preservar tanto la privacidad de los usuarios así como el bien activo que la data representa para Jampp, la mayor parte de las columnas se encuentran ofuscadas mediante algoritmos de hashings, transformaciones lineales u otro método que permita preservar la anonimidad de la información. Una descripción general de los datasets resumiendo atributos importantes puede verse en la tabla I (ver **tabla I**).

### Entorno de trabajo

El análisis exploratorio de datos se realizó enteramente con el uso de python 3.7.3 a través del IDE *jupyter notebooks*, más específicamente *jupyter lab*. Los paquetes más utilizados durante el trabajo fueron pandas, numpy, matplotlib, seaborn entre otros. Una copia exacta del entorno de trabajo para utilizar con *conda* puede encontrarse en el archivo *requirements.txt* dentro del github que acompaña a este trabajo.

---

| Nombre | Formato | Compresión | Tamaño | Registros | Columnas | NaN |
|---|---|---|---|---|---|---|
| auctions | csv | gzip | 1.4G | 19571319 | 7 | 14% |
| events | csv | gzip | 715M | 2494423 | 22 | 28% |
| clicks | csv | gzip | 1.6M | 26351 | 20 | 15% |
| installs | csv | gzip | 770K | 3412 | 18 | 28% |

**Tabla I:** Atributos generales de los cuatro datasets. El tamaño para cada archivo fue calculado en base al archivo csv comprimido con el comando de linux **ls -lh**. La columna NaN representa el porcentaje de celdas con *missing values* en cada dataset.

# Auctions

El dataset *auctions.csv* colecciona todas las instancias de subastas o *bidding* ocurridas en el *ad-exchange* entre el 2019-03-05 00:00:00 y 2019-03-13 23:59:59 en horario UTC. Está comprendido por 19.571.319 registros y 7 columnas, con un total de 136.999.233 celdas. Los valores nulos se concentran en tan solo 1 columna y conforman el 14% del total de las celdas. Descartamos la columna *auction_type_id* debido a que contiene valores nulos en el total de su extensión. La figura 1 resume el estado de completitud del dataset (ver **figura 1**). Considerando que la variable *country* del dataset posee un único valor en su extensión, y habiendo sido informados que los dataset provienen del país Uruguay es entonces que decidimos cambiar el huso horario de los datos de UTC a GMT-3.

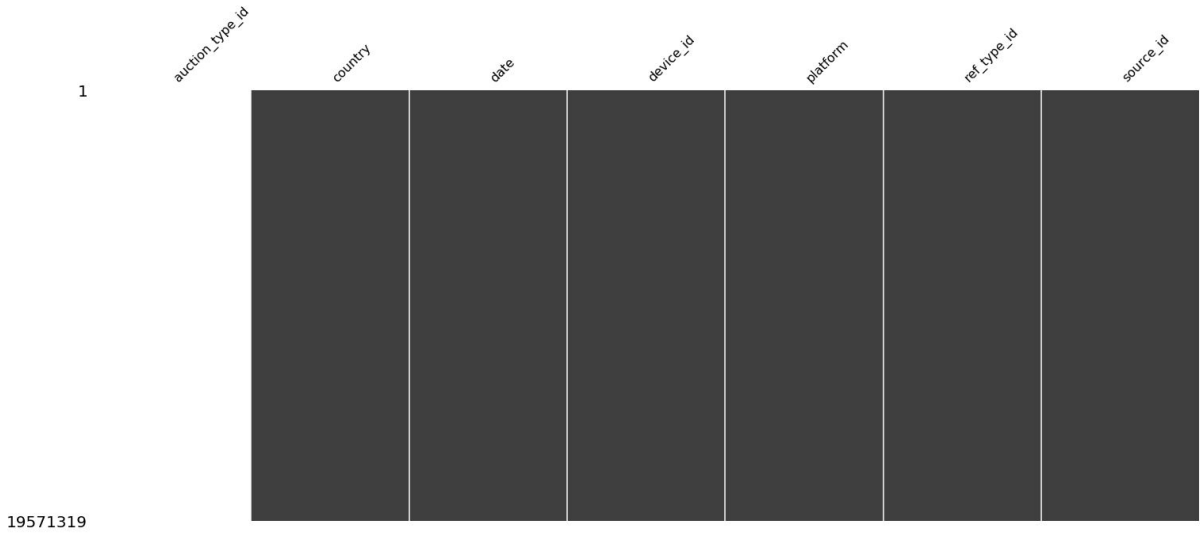


**figura 1**: Matriz mostrando el estado de completitud del dataset generado utilizando el paquete missingno de python. En el extremo izquierdo se muestra el primer (1) y el último índice (19.571.319) de las filas. Cada columna de la imagen representa una columna del dataset. El color negro representa celdas que contienen datos diferentes de nulos, el color blanco representa datos nulos. En el dataset auctions todos los datos nulos se concentran bajo la columna *auctions_type_id*.

---

La columna *date* recoge los timestamp de cada una de las subastas. Abarca un total de 9 días, con un pico de frecuencia máximo observado durante el dia 2019-03-10 con 2.950.749 subastas realizadas. El dia con menor subastas realizadas fue el 2019-03-04. No se descarta la presencia de un subregistro por ser el primer dia que aparece en el dataset.

Realizamos una visualización de la serie de tiempo tomando dos tipos de agregación temporal, diaria y horaria (ver **figura 2**). En una segunda instancia identificamos los días que abarca el dataset que fueron fines de semana y los días que fueron días de semana. Una vez identificados, separamos la serie de tiempo según dia-de-semana y fin-de-semana y observamos las diferencias en cuanto a frecuencia que se producen durante un momento de la semana y el otro (ver **figura 2**). Se debe considerar que los días de semana están menos representados que los días de semana debido a la cantidad de los mismos en el dataset.

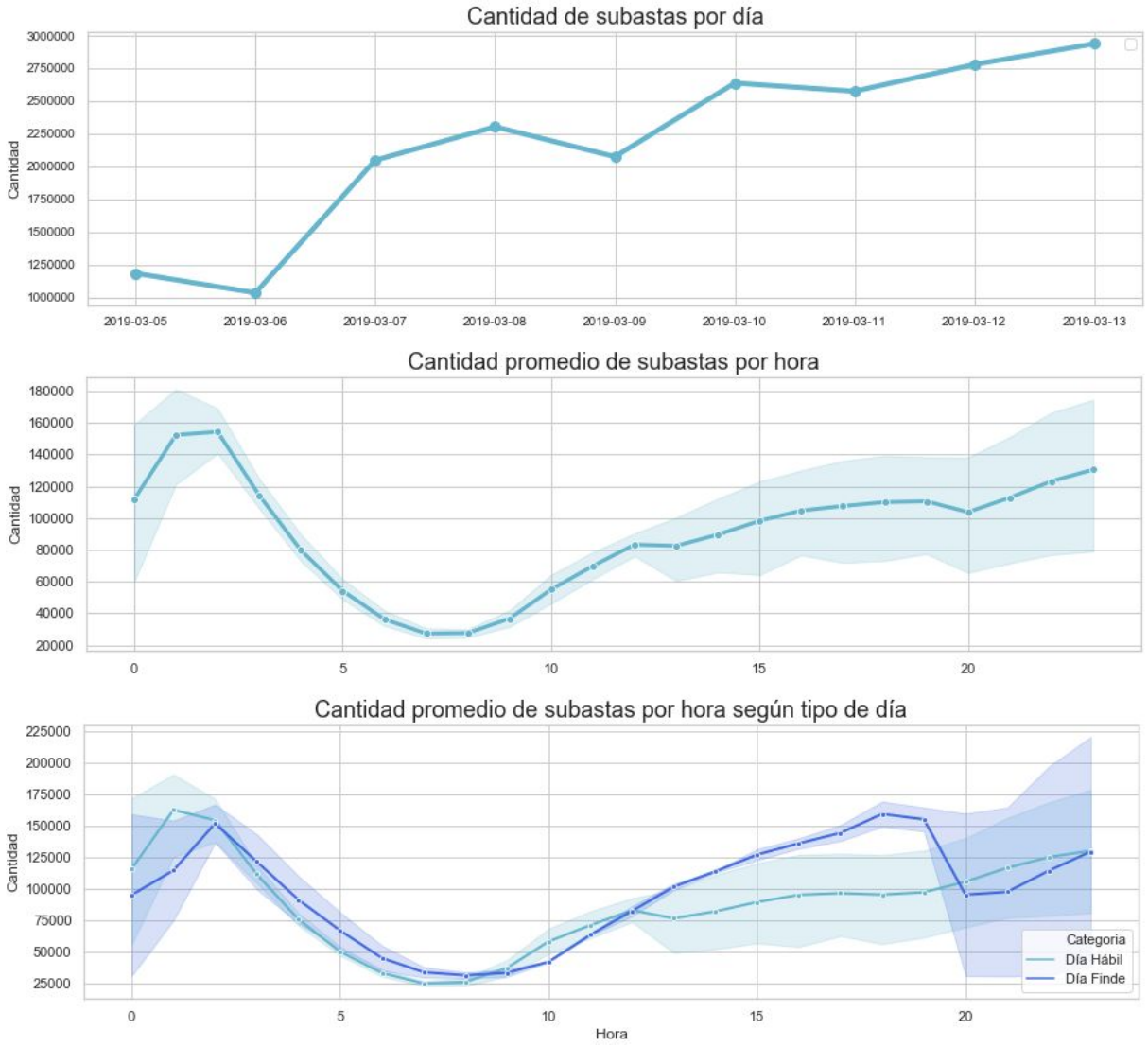


**Figura 2**: Serie de tiempo de la cantidad de subastas con diferentes agregaciones horaria. El horario en la imagen se encuentra en UTC. **Arriba**: Agregación diaria de la cantidad de subastas realizadas. **Medio**: Cantidad de subastas por hora y su error estándar. **Abajo**: agregación horaria según días hábiles y fines de semanas.

---

Las columnas *platform* y *ref_type_id* representan variables del market-store y del ad-exchange. Cada una posee dos valores únicos que se relacionan íntimamente entre ellos. La figura 3 recoge la proporción observada para cada uno de los valores, dado que ambas columnas son iguales, la proporción de los mismos se mantiene igual (ver **figura 3**).

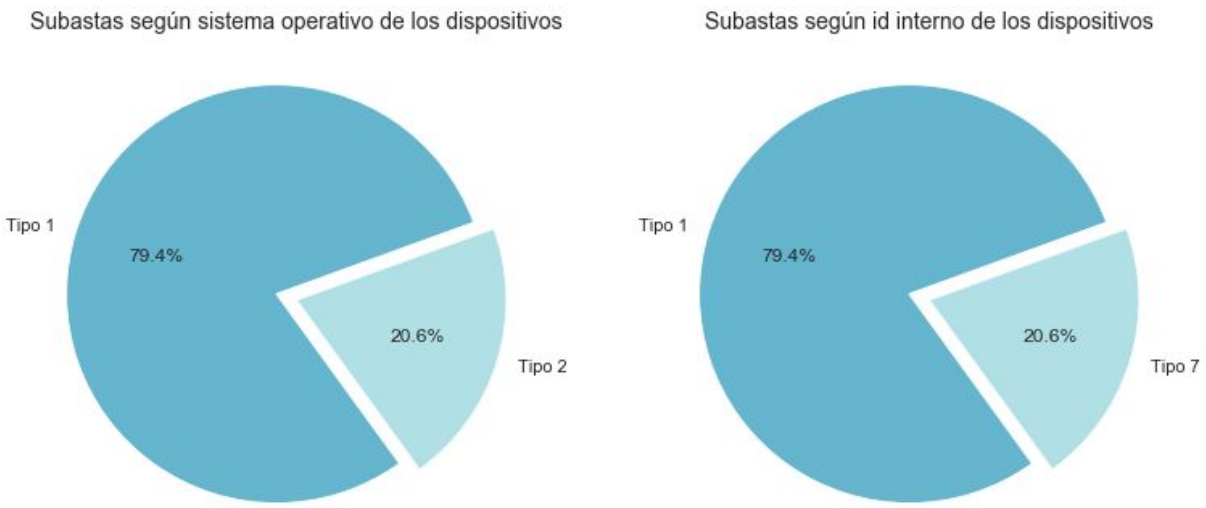


**Figura 3**: Gráficos de torta mostrando proporciones de las columnas platform (**izquierda**) y ref_type_id (**derecha**).

---

Finalmente, la columna source_id presenta 5 valore únicos que hacen referencia a las 5 fuentes de subastas. El source_id 0 es el valor más frecuente con 13.354.597 de subastas emitidas por el mismo. La figura 4 sintetiza la cantidad de subastas por dia emitidas por cada *source_id* (ver **figura 4**).

---

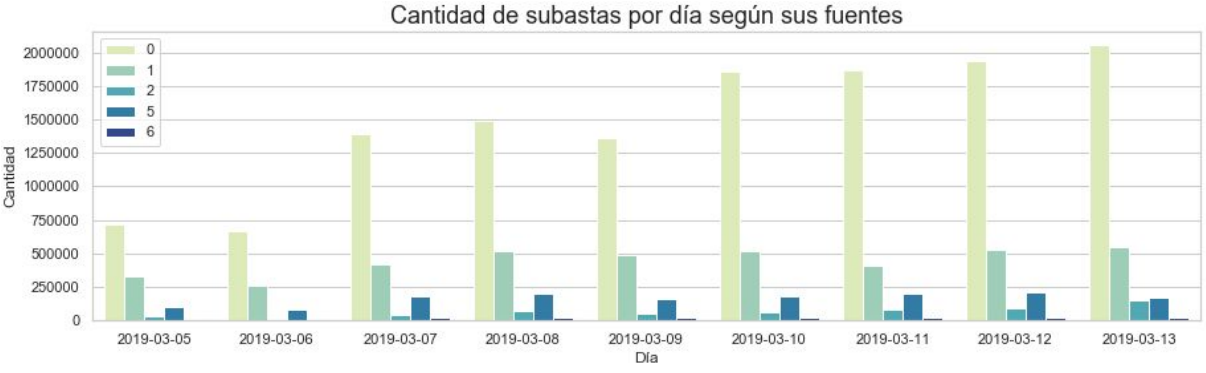


**Figura 4**: Gráfico de barras mostrando cantidad de subastas por dia agrupada por la fuente que generó la subasta que aparece en la columna *source_id*. El source_id 0 muestra consistentemente ser el más generador de subastas en relación al resto.

# Events

El dataset *events.csv* colecciona todas las interacciones que el usuario realiza dentro de una aplicación móvil cliente de Jampp. Está comprendido por 2.494.423 registros y 22 columnas, con un total de 54.877.306 celdas. Los valores nulos se encuentran repartidos entre 14 columnas y conforman el 28% total de las celdas. Descartamos las columnas *device_os_version*, *device_city*, *trans_id*, *user_agent*, *device_brand*, *carrier*, *device_os* y *connection_type* debido a la gran presencia de valores nulos al azar que presentan en su extensión. La figura 5 muestra el estado de completitud del dataset (ver **figura 5**).

La columna *date* representa el timestamp en UTC de cada uno de los eventos registrados. Puesto que la columna *countrycode* presenta el mismo valor en toda su extensión y sabiendo que este valor representa a Uruguay, transformamos el timestamp de UTC a GMT-3. El dataset abarca la ventana temporal desde 2019-03-04 hasta 2019-03-13, 10 dias en total. El dia 2019-03-12 representa el dia con mas eventos registrados, con un total de 299.502, mientras que 2019-03-04 registra la menor cantidad de eventos con un total de 46.898 debido al subregistro del total de las horas en ese día (ver **figura 6**).

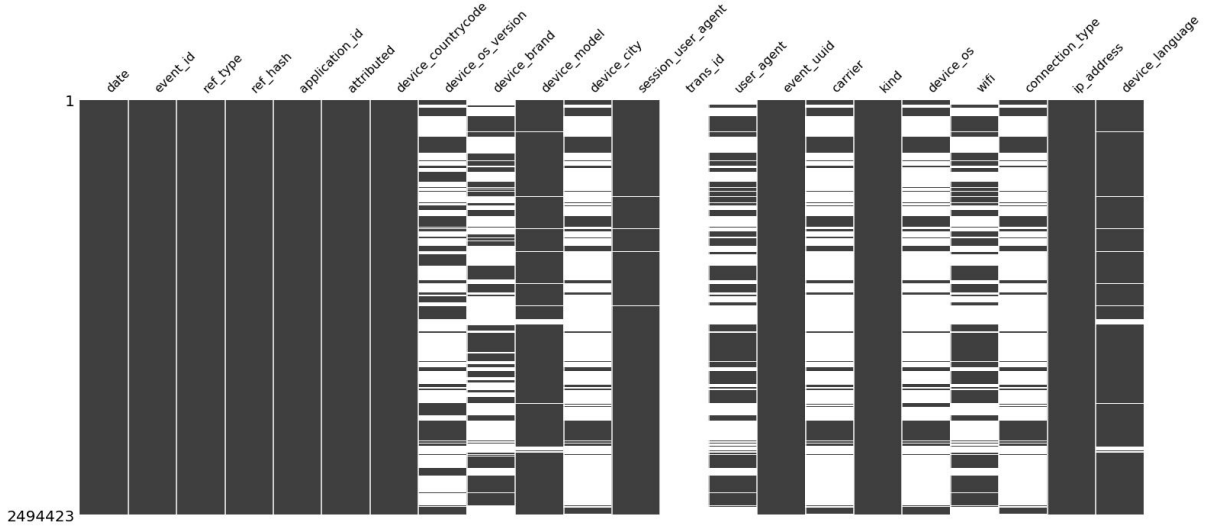


**figura 5**: Matriz mostrando el estado de completitud del dataset generado utilizando el paquete missingno de python. En negro se muestran las celdas completas, los espacios muestran los valores nulos.

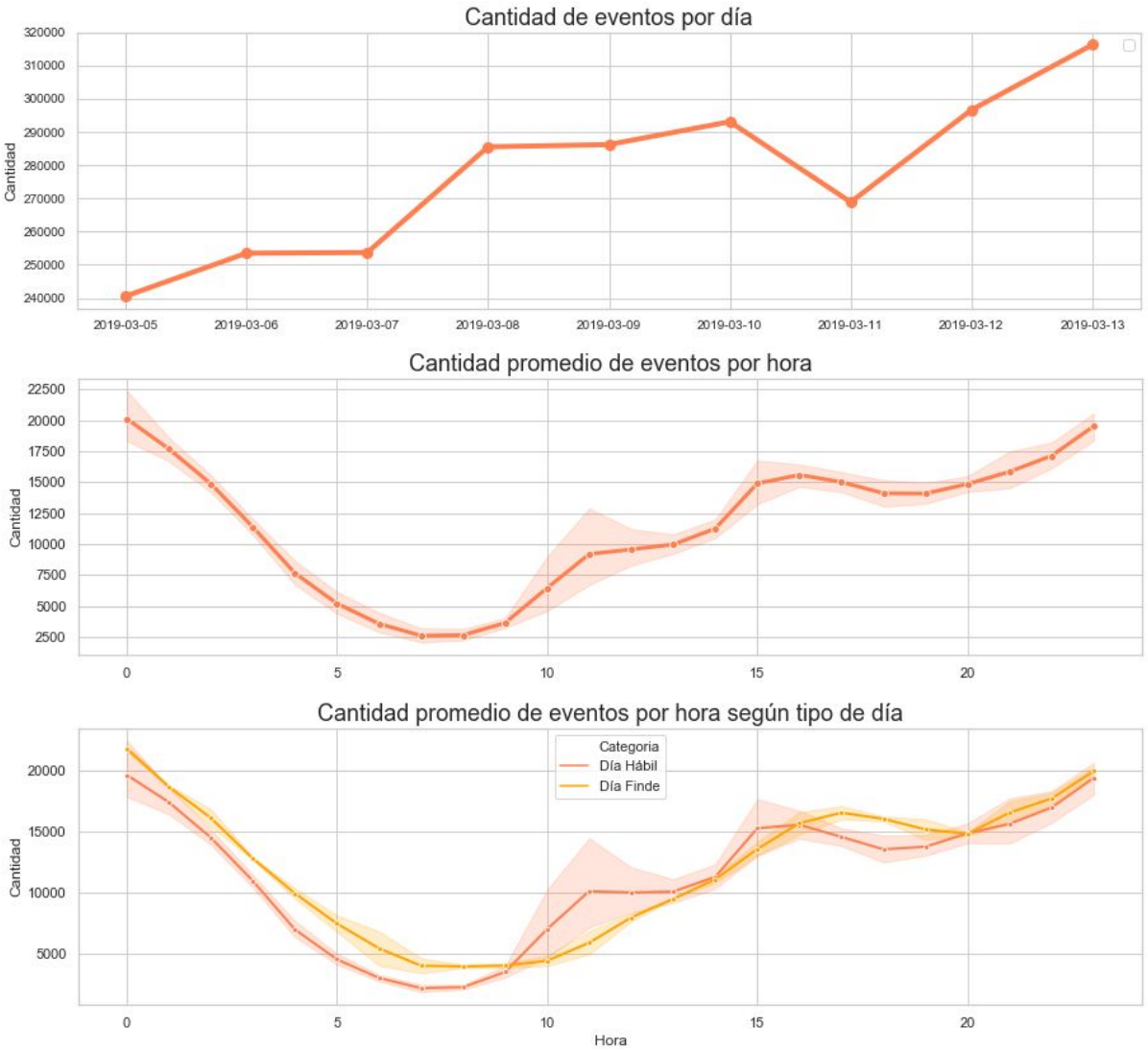


**Figura 6**: Serie de tiempo de la cantidad de eventos con diferentes agregaciones horaria. El horario en la imagen se encuentra en UTC. **Arriba**: Agregación diaria de la cantidad de eventos registrados.. **Medio**: Cantidad de eventos por hora y su error estándar. **Abajo**: agregación según días hábiles y fines de semanas.

---

En la figura 6 examinamos la frecuencia de subastas horarias y agrupamos las subastas en dia-de-semana y fin-de-semana para visualizar cambios en el volumen de subastas (ver **figura 6**). Del gráfico se desprende que no existen diferencias evidentes entre el nivel de subasta de días de semana vs. fines de semana, y que los valles y picos se ubican a la madrugada y la medianoche respectivamente.

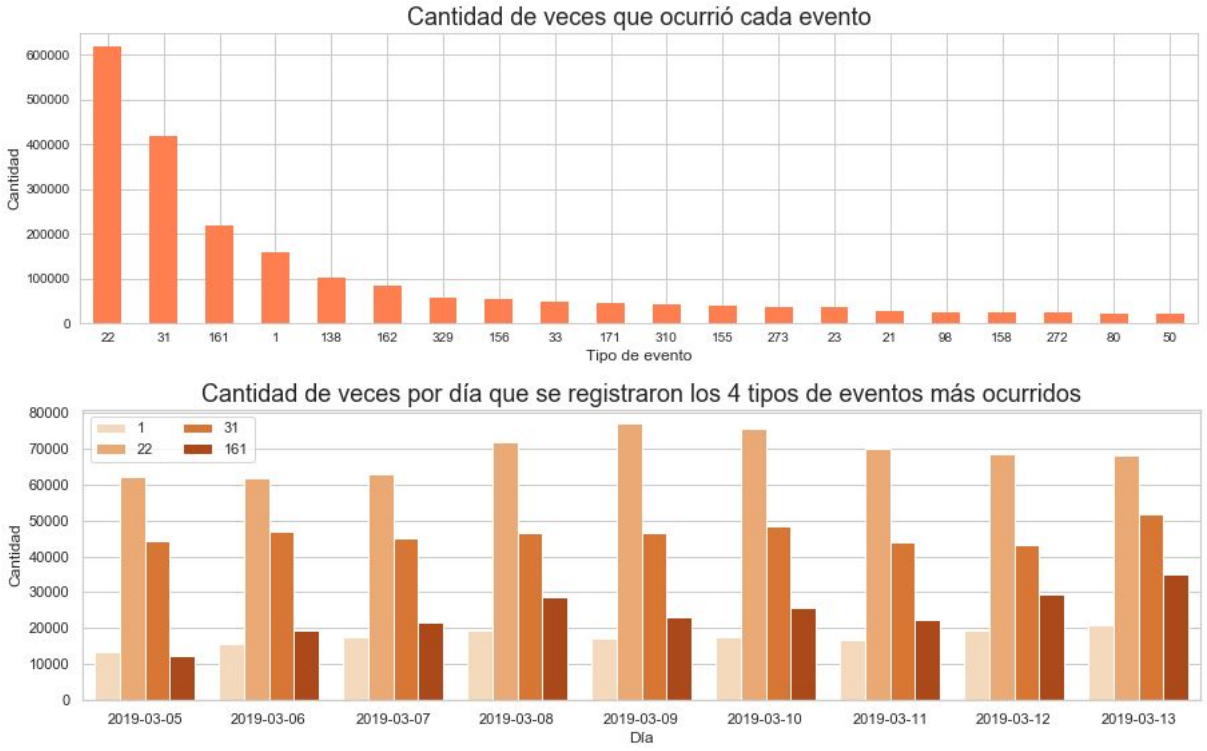


**Figura 7**: Visualización de la frecuencia de tipos de eventos según identificador y según dia. **Arriba**: Cantidad de eventos según tipo de evento. **Abajo**: Cantidad de los 4 eventos mas frecuentes segun dia.

---

La columna *events_id* recoge el identificador de diversos eventos que los usuarios pueden realizar dentro de una aplicación. El evento de tipo 22 fue el más frecuentemente observado, con 618.228 registros. La figura 7 visualiza el frecuencia de los diferentes identificadores de eventos (ver **figura 7**). Posteriormente realizamos un subset de los eventos con más de mil registros y visualizamos el volumen de cada agrupado por día (ver **figura 7**).

Dependiendo de la *app-store* de procedencia de la aplicación móvil, la columna ref_type puede tomar dos tipos de valores, una para la app-store de google y el otro para la de apple. La figura 8 visualiza la proporción de cada uno de los valores dentro de *ref_type* (ver **figura 8**). Por último, visualizamos la proporción de eventos que registran haberse realizado con la red inalámbrica wifi activada (ver **figura 8**).

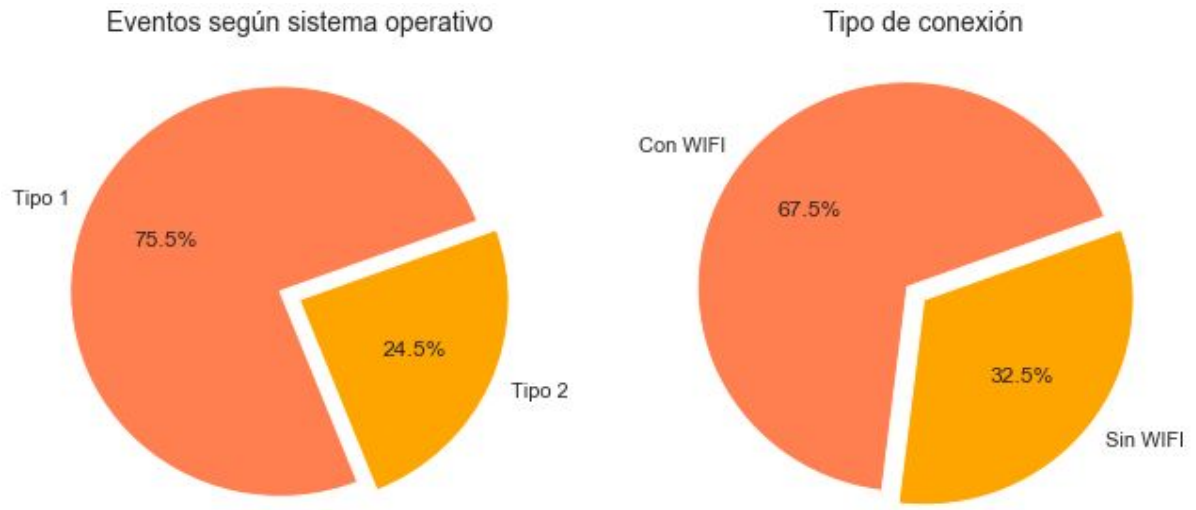


**Figura 8**: Visualización de la proporción de eventos según *ref_type* (izquierda) y según wifi (derecha).

---

# Clicks

El dataset *clicks.csv* representa instancias de click registradas sobre ads pertenecientes a 7 clientes de Jampp. Está comprendido por 26.351 registros y 20 columnas, con un total de 527.020 celdas. Los valores nulos se encuentran repartidos entre 9 columnas y conforman el 15% total de las celdas. Destacamos que la columna *action_id* es la única en su tipo ya que presenta valores nulos a lo largo de toda su extensión, es por esto que no será tenida en cuenta para el análisis posterior. De la misma manera, también se retira la columna *wifi_connection* por no dar información en su contenido. La figura 9 resume el estado de completitud del dataset (ver **figura 9**).

La columna *created* recoge los timestamps en UTC del momento en que cada click es registrado. Dado que la columna *country_code* posee un único valor en su extensión, y habiendo sido informados por personal de Jampp que la data proviene de uruguay, decidimos pasar el huso horario del timestamp de UTC a GMT-3.

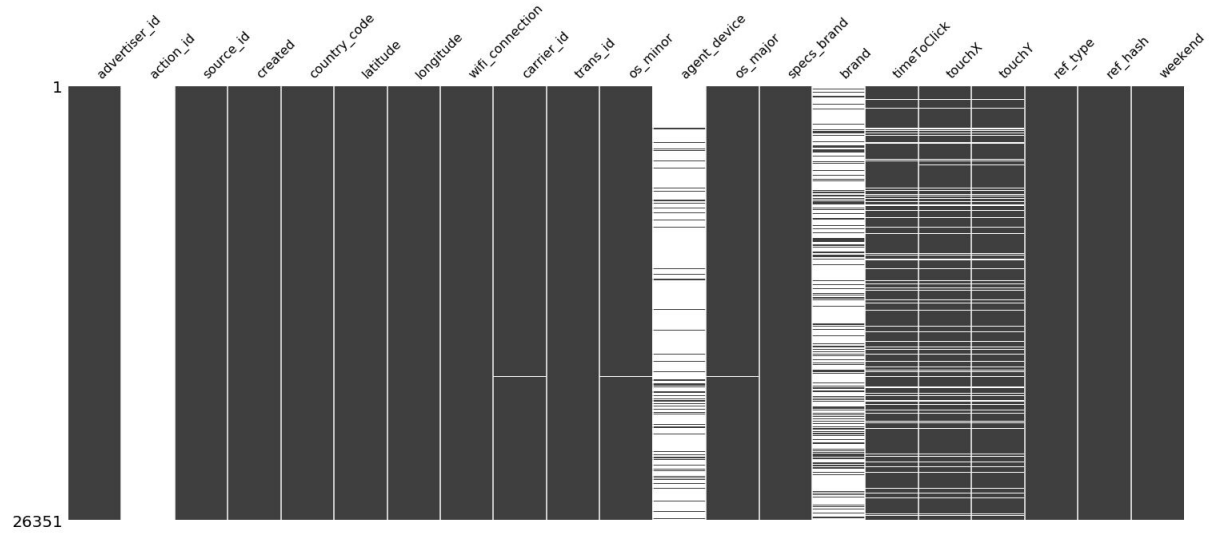


**figura 5**: Matriz mostrando el estado de completitud del dataset generado utilizando el paquete missingno de python. En negro se muestran las celdas completas, los espacios muestran los valores nulos.

---

El dataset se extiende en una ventana temporal que abarca 9 días, comenzando 2019-03-04 22:00:00 y se extiende hasta 2019-03-13 20:00:00. El pico de frecuencia se observó durante el día 2019-03-12 con un total de 4856 clicks realizado, mientras que la menor frecuencia se observó durante el día 2019-03-04 con solamente 9 clicks realizados. La figura 10 muestra la cantidad de clicks con agregación diaria y horaria durante los 9 días (ver **figura 10**). No descartamos errores del sampleo como explicación para el subregistro de datos presentes en los primeros dos días de la serie temporal.

La agregación horaria de la serie temporal reveló picos de frecuencia ubicados alrededor de las 8 de la mañana y de las 10 de la noche, con valles ubicados a las 4 de la madrugada y las 5 de la tarde. Agrupando los días según si son días de semana o fines de semana se aprecia una diferencia en la frecuencia de clicks por hora (ver **figura 11**), con un coeficiente de determinación de Pearson de -0.58. Debido a que el subregistro previamente mencionado afecta principalmente a los fines de semana, se debe tener recaudo al interpretar las diferencias.

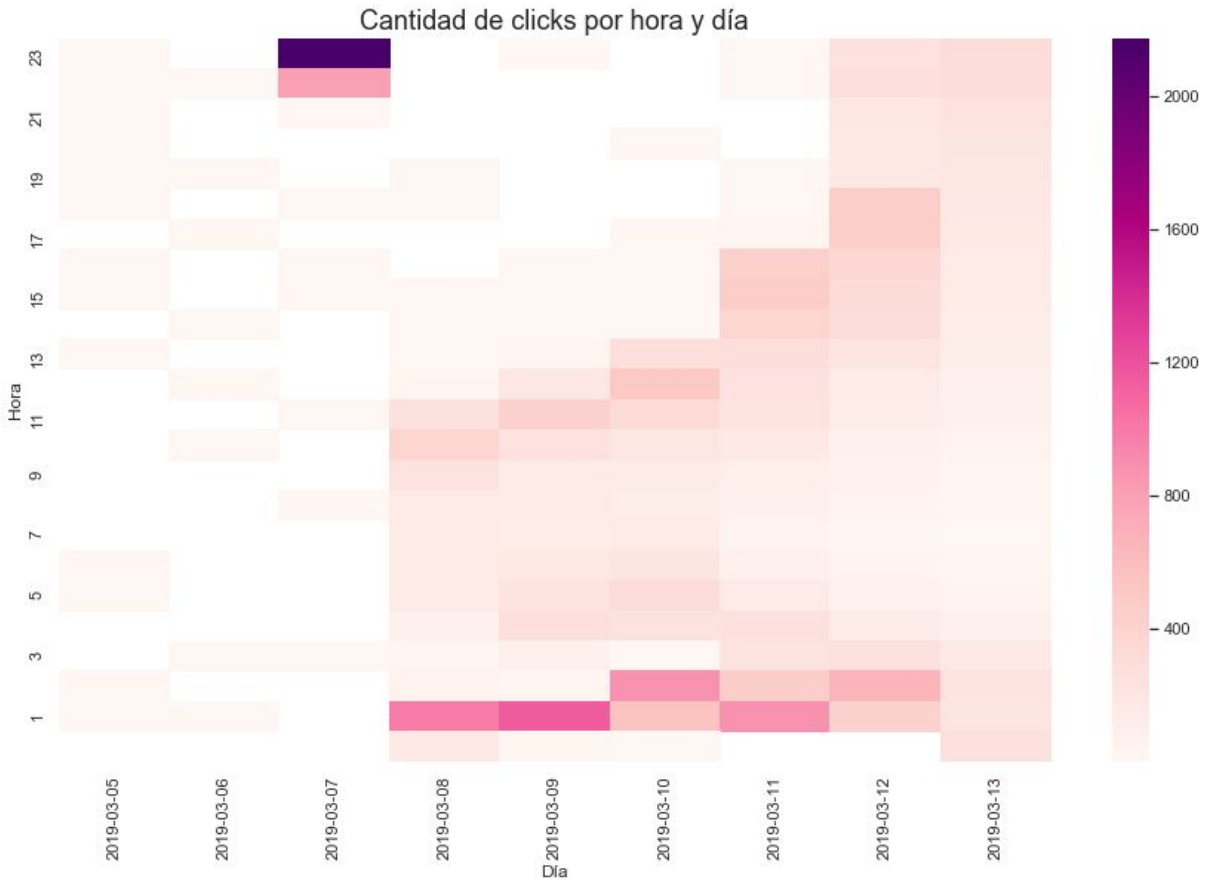


**figura 10**: Heatmap representando la cantidad de clicks realizados por hora y por día durante los 9 días de ventana temporal del dataset clicks.

---

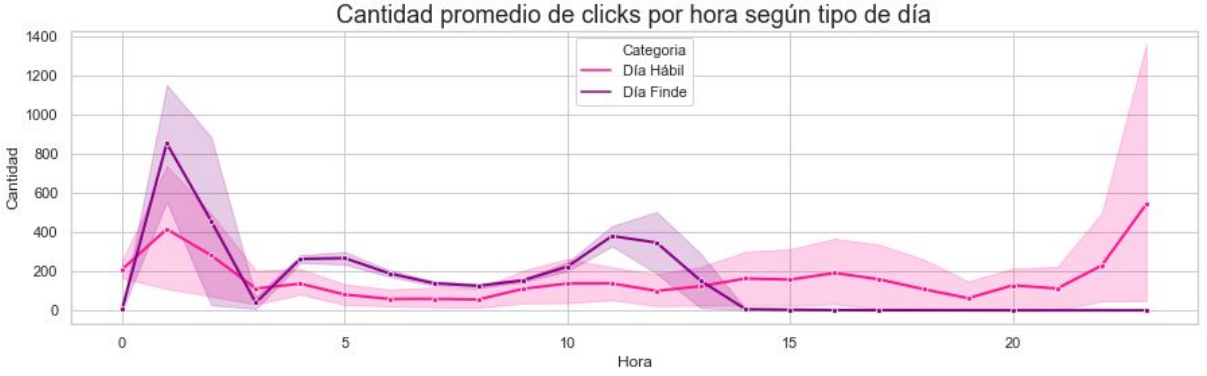


**figura 11**: Serie temporal mostrando la cantidad de clicks por hora agrupadas en días de semana (días hábiles) y fines de semana (dias finde).

Cada evento de click registrado en el dataset se acompaña del par *touchX* y *touchY* representando la posición en la que el click fue realizado en la pantalla del dispositivo. Mediante la inspección visual de la distribución de los puntos a lo largo del eje Y podemos definir 3 zonas características según la densidad de clicks. La figura 12 muestra la posición de los clicks en el espacio durante los primeros 30 segundos, entre 30 y 60 segundos y luego de los 60 segundos según la columna *timeToClick* (ver **figura 12**). La tabla II muestra la proporción de clicks que cae en cada una de las regiones entre las líneas rojas del gráfico (ver **tabla II**).

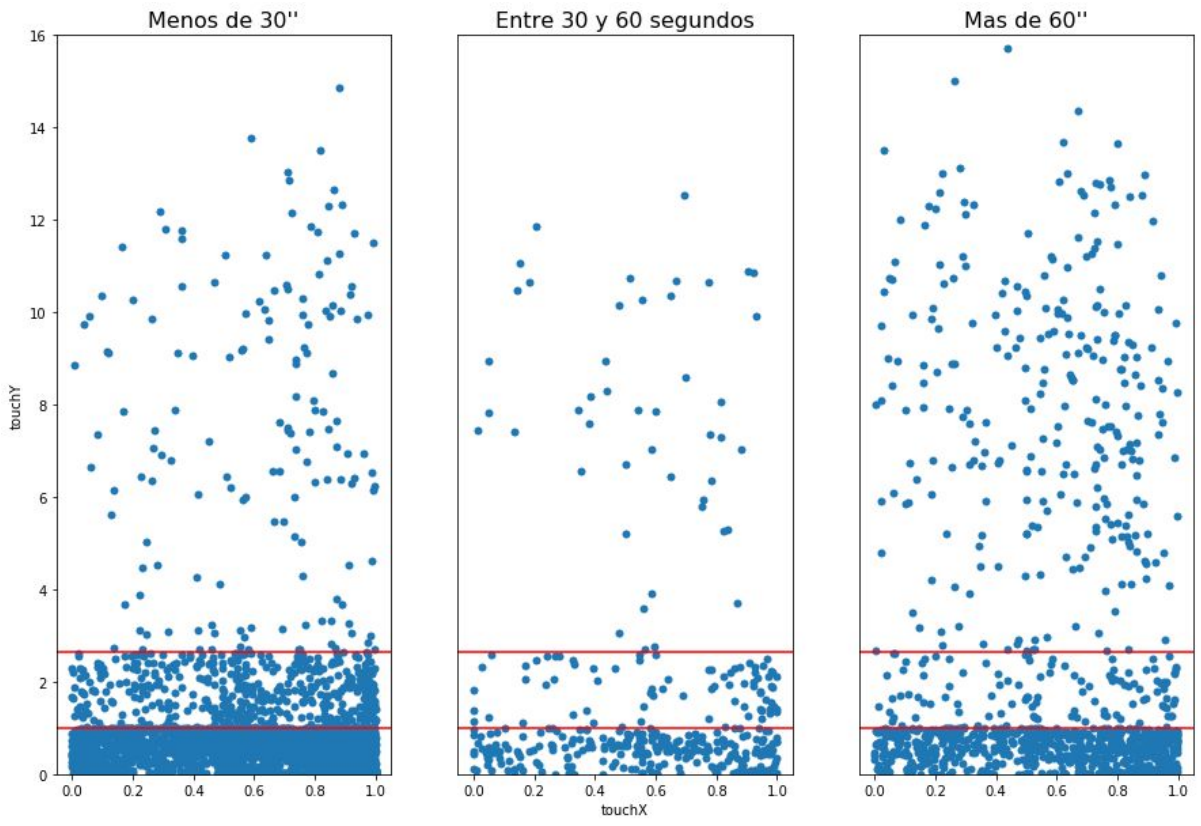


**figura 12**: Posición XY de cada click proyectado sobre la pantalla táctil del dispositivo (puntos azules). Recreamos cada punto en el espacio XY y agrupamos en tres momentos temporales. De izquierda a derecha: los clicks que ocurrieron en los primeros 30", entre los 30" y 60" y los clicks luego de los 60". Tres áreas definidas por el eje Y quedan delimitadas por líneas rojas.

| Cuadrante | Menos de 30” | Entre 30” y 60” | Más de 30” |
|---|---|---|---|
| Superior | 83% | 12% | 5% |
| Media | 68% | 20% | 12% |
| Inferior | 65% | 10% | 25% |

**Tabla II:** Proporción de clicks que cayeron en cada régions por unidad de tiempo agrupado. La suma marginal de cada una de las final es 100%.

---

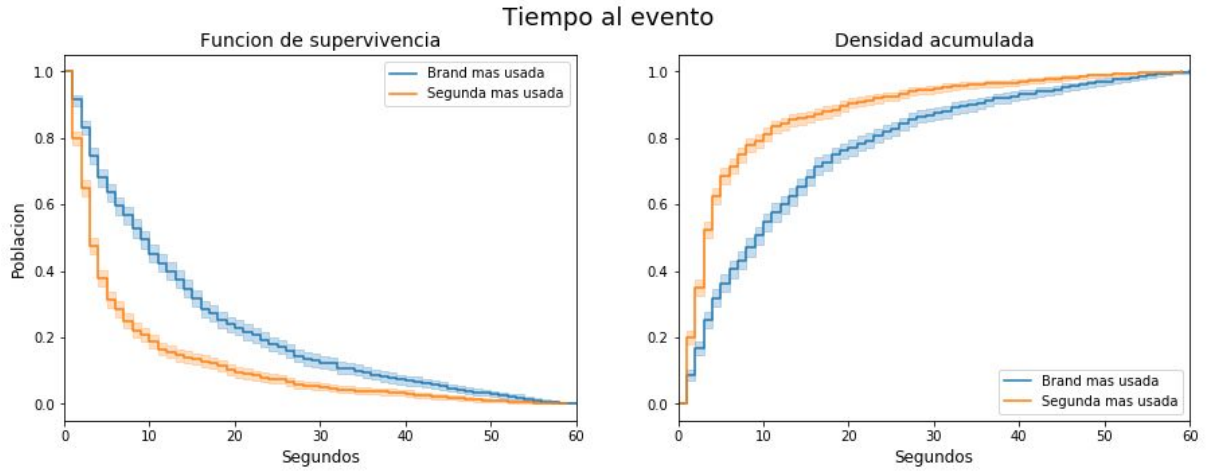


**figura 13:** Análisis de supervivencia del tiempo en segundos que le lleva al usuario realizar el click agrupado por las dos marcas más utilizadas de celular. **Izquierda**: Función de supervivencia de porcentaje de usuarios que realizaron clicks por unidad de tiempo. **Derecha**: Densidad acumulada.

---

La columna *timeToClick* informa acerca del tiempo en segundos que transcurrió desde la impresión del ad hasta la realización del click sobre la pantalla del dispositivo. Realizamos un análisis de supervivencia para observar las diferencias en *timeToClick* tomando como variable explicativa las primeras dos marcas más comunes de celulares especificadas por la columna *brand*. La figura 13 muestra la función de supervivencia (izquierda) y la densidad acumulada para el *riesgo* de realizar un click entre los grupos de la primera y segunda marca (ver **figura 13**). Utilizando un modelo Cox de riesgo proporcional encontramos que el HR de la primera marca para hacer click es de 0.56 (IC 95% 0.51 - 0.62). Concluimos que la

marca de celulares más frecuente según la columna *brand* tiene una frecuencia de clicks de aproximadamente la mitad de la frecuencia de la segunda marca. La columna *source_id* hace referencia a la identificación interna de la aplicación donde se genero el click, *carrier_id* indica el identificador del operador móvil del dispositivo donde se genero el click y *ref_hash* es un indicador único de usuarios. En la figura 14 resumimos los valores dentro de cada columna y presentamos los 10 usuarios que más clicks realizaron (ver).

---

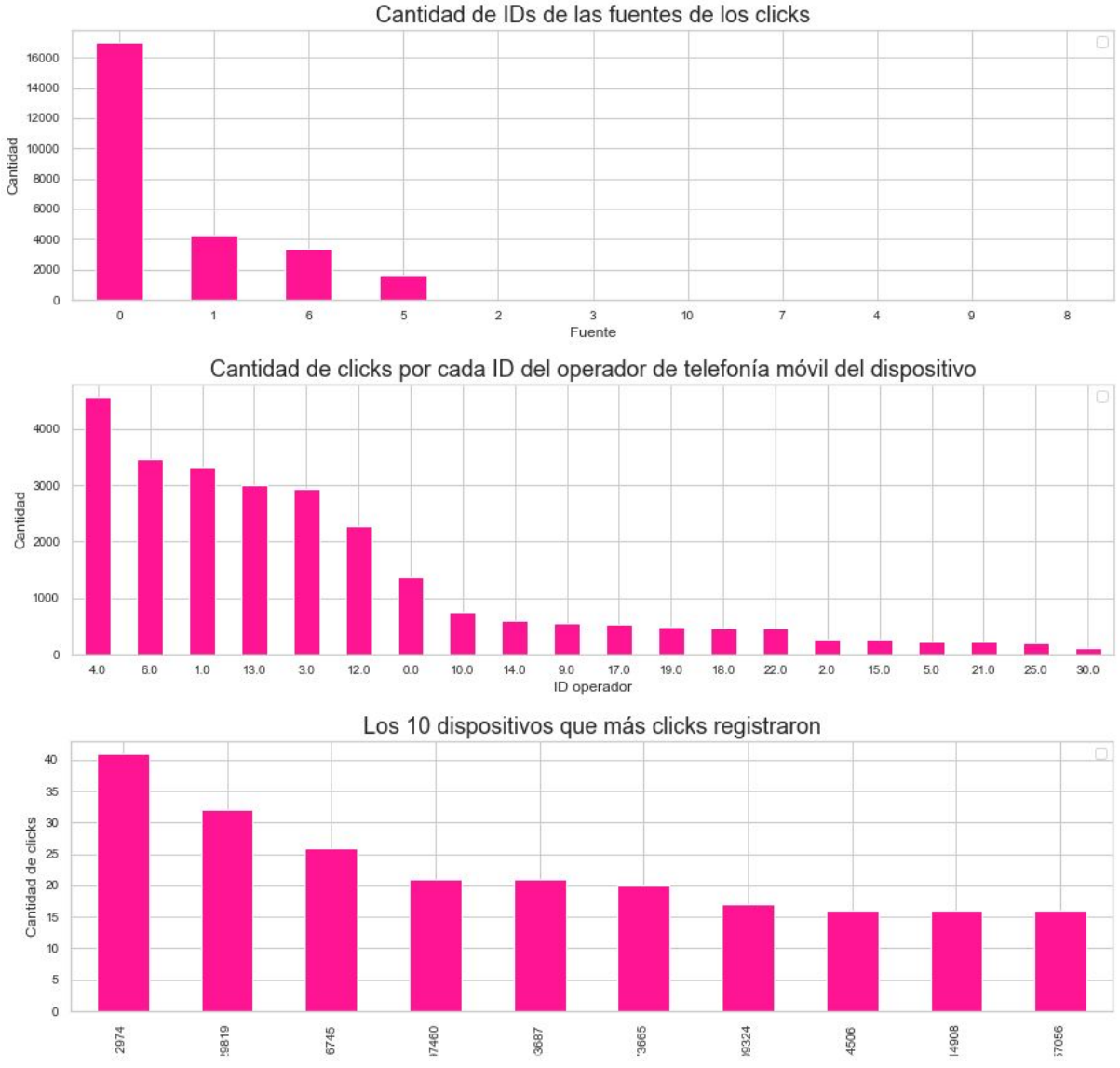


**figura 13:** Gráficos de barras representando la cantidad absoluta de cada una de las variables en las columnas *source_id*, *carrier_id* y *ref_hash* respectivamente.

# Installs

El dataset *installs.csv* representa instancias atribuidas implícita o explícitamente en las que un usuario instaló una de las 31 aplicaciones móviles clientes de Jampp especificadas en la columna *application_id*. El dataset está comprendido por 3.412 registros y 18 columnas, con un total de 61.416 columnas. Los valore nulos se encuentran repartidos en 9 columnas y conforman el 29% de las celdas. Las columnas *click_hash* y *trans_id* presentan valore nulos a lo largo de toda su extensión, es por esto que no serán tenidas en cuenta en el posterior análisis. Tampoco serán tenidas en cuenta las columnas kind y event_uuid debido a presentar casi 75% de valores nulos y no aportar información. La columna *attributed* presenta el valor *falso* en toda su extensión, por lo que también se decidió removerla. La figura 14 resume el estado de completitud del dataset (ver **figura 14**).

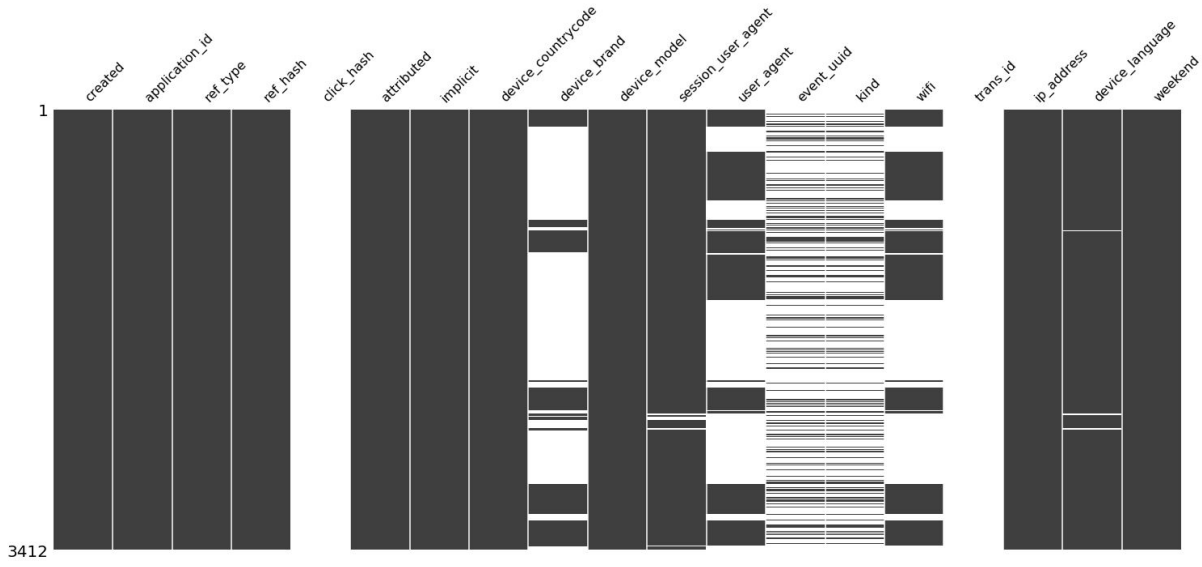


**figura 14:** Matriz mostrando el estado de completitud del dataset generado utilizando el paquete missingno de python. En negro se muestran las celdas completas, los espacios muestran los valores nulos.

La columna *created* recoge los timestamps en UTC del momento en que cada instalación ocurre. El timezone de cada timestamp fue transformado a GMT-3 al igual que los otros datasets para reflejar la hora en Uruguay, país de donde provienen los datos.

El dataset se extiende durante 10 días, desde 2019-03-04 21:00:00 hasta 2019-03-13 20:00:00. El pico máximo de frecuencia durante la ventana temporal se registró el 2019-03-12 10:00:00 con 35 instalaciones en una hora. La figura 15 muestra la actividad horaria durante las 24 horas de los 10 días (ver **figura 15**). La agregación temporal horaria de la serie de tiempo revela que el valla de actividad se encuentra alrededor de las 4 de la madrugada, en cambio, la actividad es máxima luego de las 12 del mediodía y se extiende hasta las 12 de la noche.

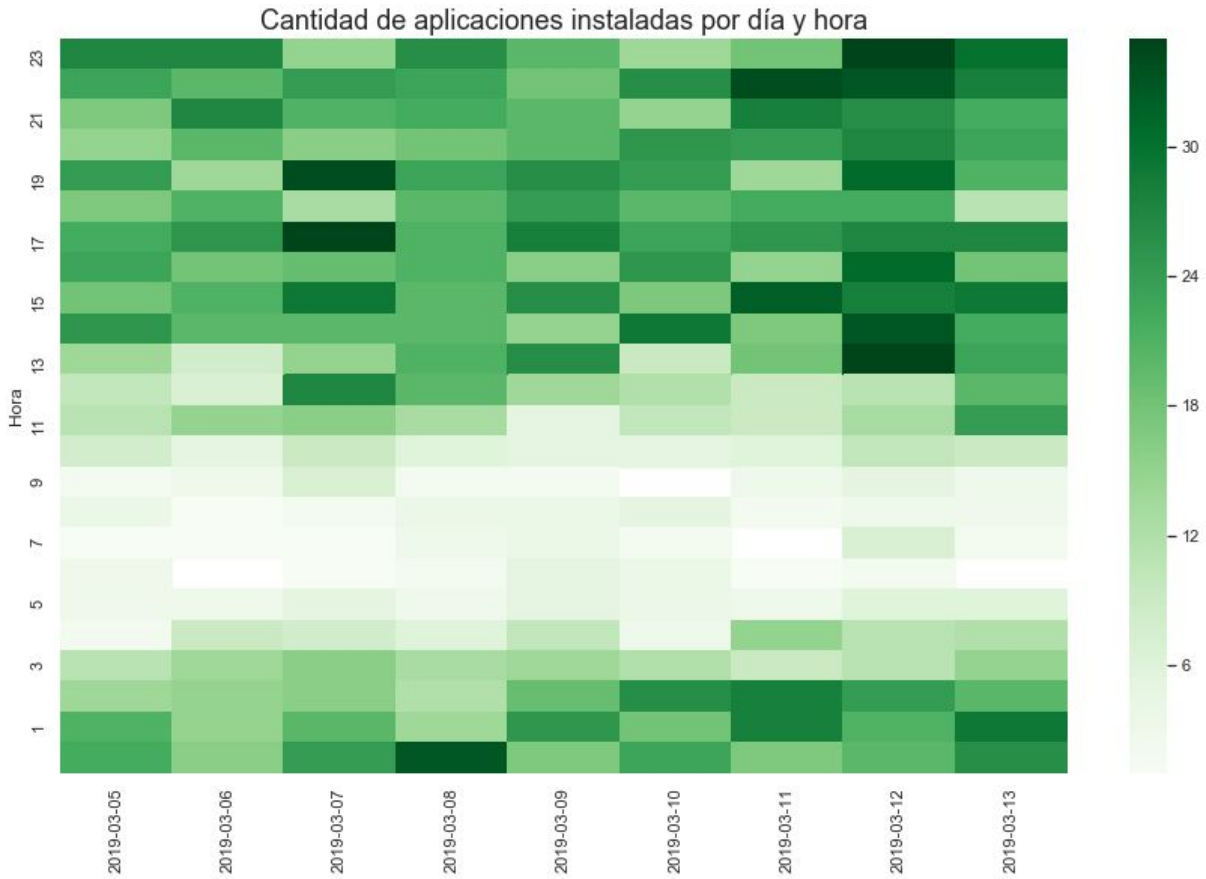


**figura 15**: Heatmap representando la cantidad de clicks realizados por hora y por día durante los 9 días de ventana temporal del dataset clicks. El huso horario se encuentra en UTC.

La columna *ref_hash* especifica un identificador único para cada usuario que realizó una instalación. De los 3.412 registros presentes en el dataset, identificamos 3.008 usuarios únicos. Del total, 2.629 usuarios (87%) realizaron solo una instalación, mientras que 379 (17%) usuarios registran haber instalado más de una aplicación. Las columnas *ref_type*, *device_model*, *session_user_agen* y *device_language* incorporan información de atributos como el market-store (apple o google), del modelo de dispositivo, características propias del dispositivo y del lenguaje respectivamente, la tabla II muestra un resumen general de las mismas (ver **Tabla II**). Con estas variables identificamos a cada uno de los usuarios, sus atributos y asignamos un *outcome* binario representado si habia instalado más de una (1) o solo una (0) aplicacion. Convertimos cada una de las variables categorías previamente mencionadas en dummies creando un dataset de dimensiones 2.975 x 457. Redujimos la dimensionalidad del mismo utilizando PCA a 30 componentes y posteriormente corrimos un TSNE con 2 componentes. La figura 16 muestra el resultado del proceso (ver **Figura 16**) y los clusters de usuarios con mas de una o una instalación acercados por sus similitudes y separados por sus diferencias.

---

| **Nombre** | **Cantidad** | **Missing** | **Únicos** | **Frecuente** |
|---|---|---|---|---|
| ref_type | 3.412 | 0 | 2 | 189... |
| device_model | 3.411 | 1 | 415 | 233... |
| session_user_agent | 3.364 | 48 | 12 | http-kit/2.0 |
| device_language | 3.378 | 34 | 30 | 330... |

**tabla III**: Valores descriptivos de las columnas ref_type, device_movel, session_user_agent y device_language. La columna frecuente reúne el valor más frecuente dentro de la variable, si muy largo se trunco.

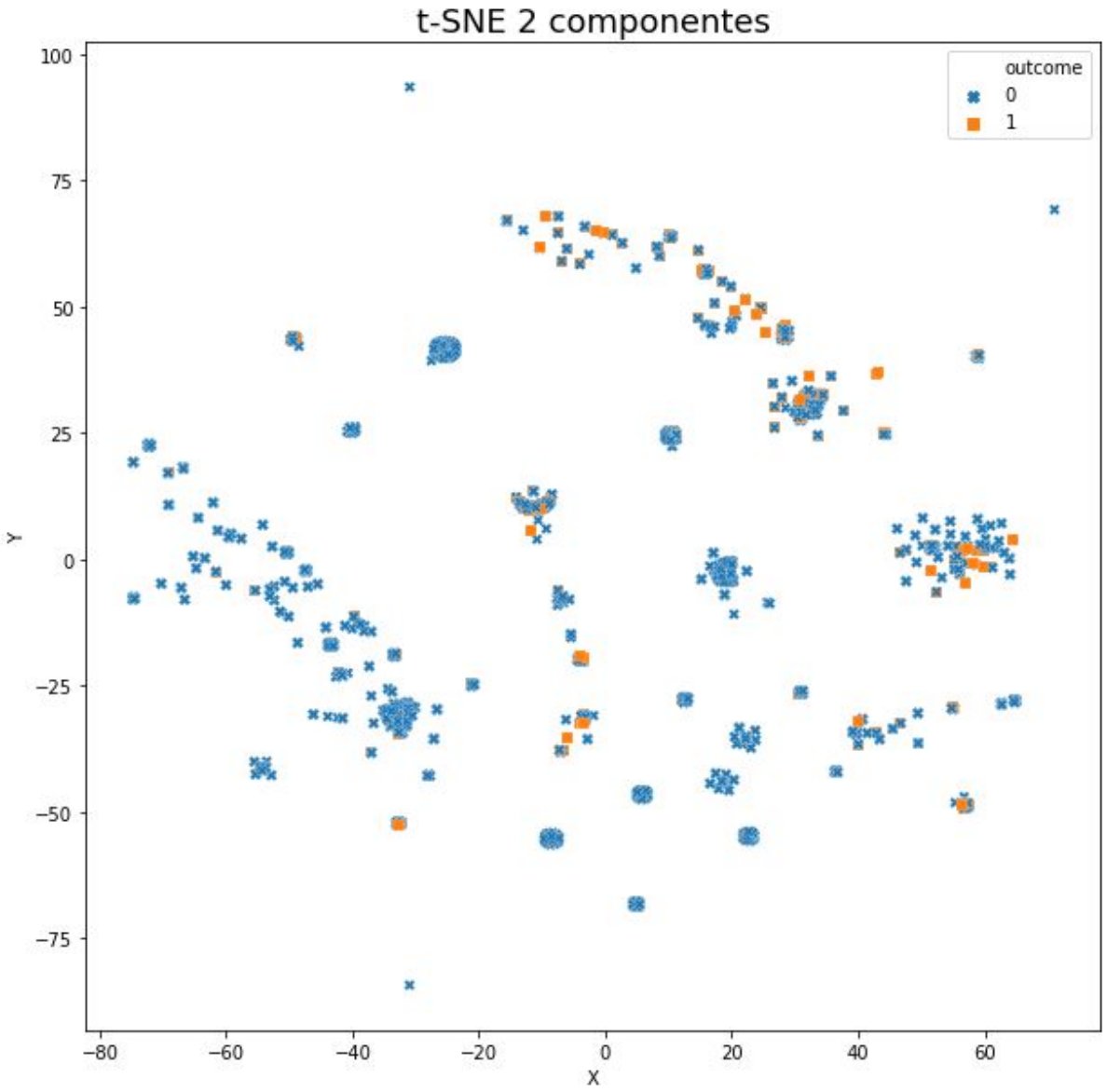


**figura 16:** Resultado del t-SNE de dos componentes mostrando clusters según atributos de los usuarios y con marcadores mostrando el outcome asociado a cada usuario. El outcome describe 1 si instalo más de una aplicación o 0 si solo instalo una aplicación.

---

La figura 17 muestra la proporción de valores dentro de las columnas *country* y *wifi* (ver **figura 17**). Por último, la figura 18 muestra la cantidad de veces que cada aplicacion se instalo según su application_id, a su vez se muestran las dos más instaladas agregadas por dia (ver **figura 18**).

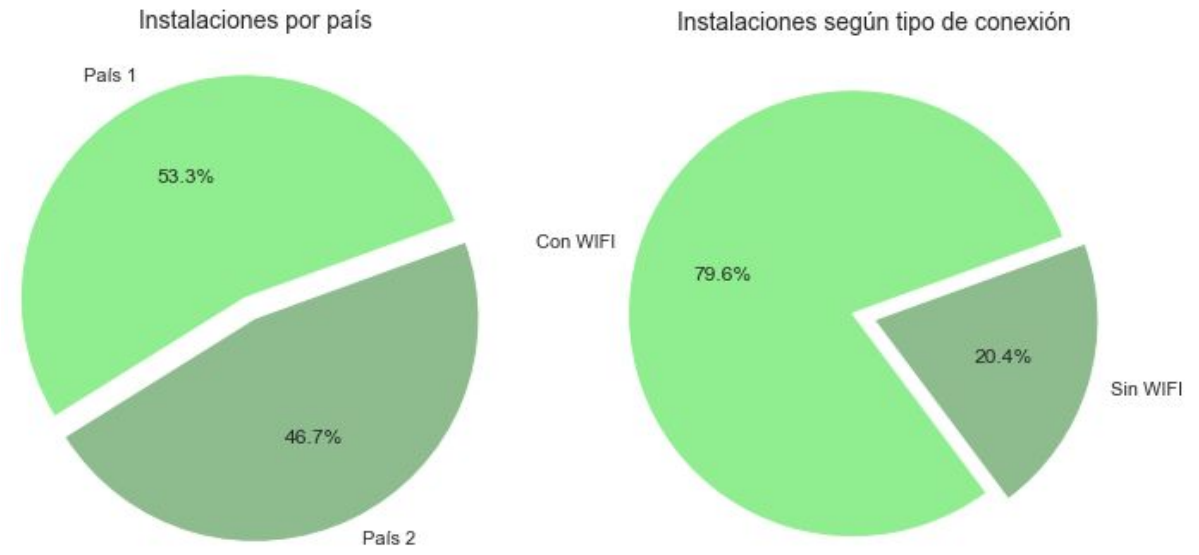


**figura 17:** Gráficos de torta para las columnas representando la proporción de valores para las columnas *country* y *wifi*.

---

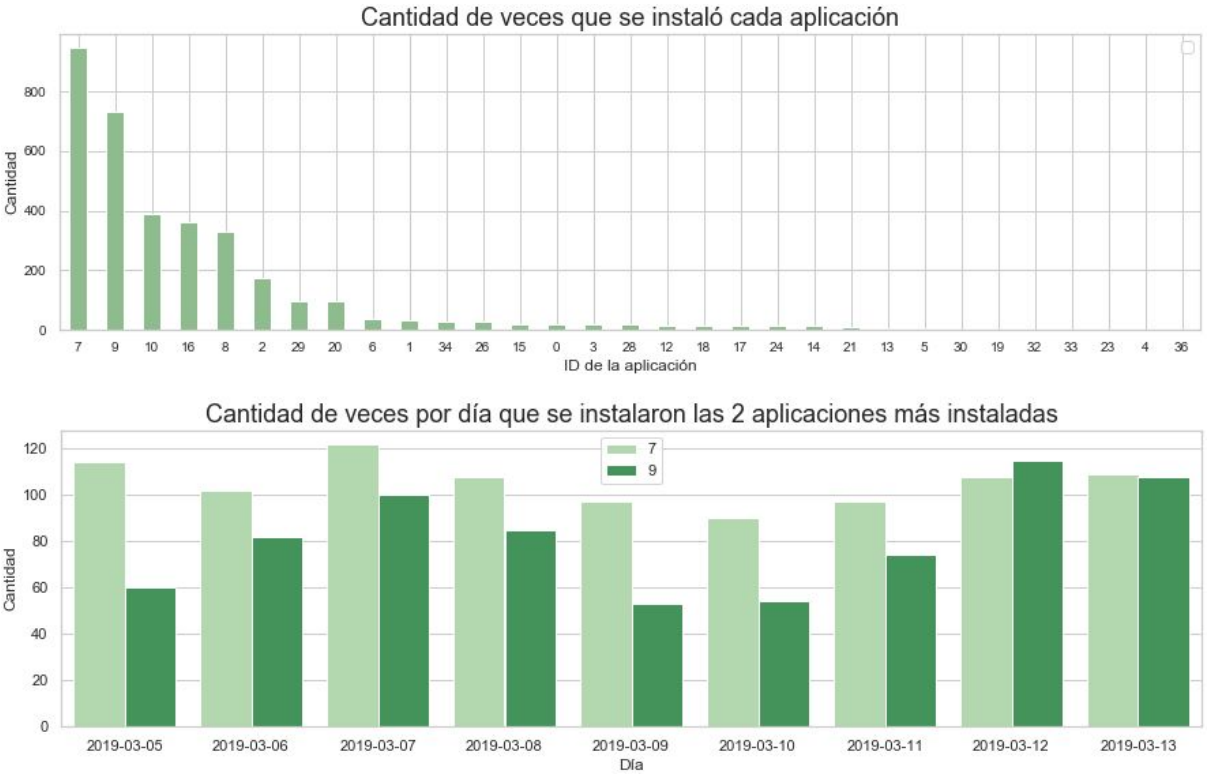


**figura 18:** Gráficos de barras mostrando la cantidad de installs para cada id de aplicación (**arriba**) y la cantidad de installs por dia para las dos aplicaciones más instaladas (**abajo**).