

# Proyecto final

Iván Arriola, Federico Miquelerena, Damián Rovetta

12-07-2023

## Introducción

Esto es un análisis descriptivo de los datos del tráfico de Montevideo, Uruguay. Hemos tomado los registros desde enero de 2021 hasta mayo de 2023 y nuestro interés es saber el comportamiento de la velocidad y el volumen de tráfico (variables explicativas) dependiendo de varias variables que iremos desarrollando a lo largo de la investigación.

## Datos

### Descripción general de los datos

Todos los datos fueron sacados de Catalogo de Datos Abiertos de **gub.uy**. En particular, los datos elegidos son los siguientes:

- Conteo vehicular en las principales avenidas de Montevideo
- Velocidad promedio vehicular en las principales avenidas de Montevideo
- Ubicación de sensores de medición de conteo vehículos

Los tres dataset son mantenidos por la Intendencia de Montevideo.

### Descripción de variables

Originalmente los datos vienen presentados de la siguiente forma:

#### Conteo vehicular en las principales avenidas de Montevideo

- **cod\_detector**: Numérico - ID de la cámara que monitorea un carril específico para detectar vehículos.
- **id\_carril**: Numérico - Número del carril monitoreado (1, 2, 3, ...).
- **fecha**: Fecha, AAAA-MM-DD - Día en que se realizó la medición.
- **hora**: hh:mm:ss - Hora en que se realizó la medición.
- **dsc\_avenida**: Texto - Nombre de la avenida donde se mide el tráfico.
- **dsc\_int\_anterior**: Texto - Nombre de la vía desde donde vienen los vehículos.
- **dsc\_int\_siguiente**: Texto - Nombre de la vía hacia donde se dirigen los vehículos.
- **latitud**: Float - Latitud del lugar de medición.
- **longitud**: Float - Longitud del lugar de medición.
- **volumen**: Numérico - Cantidad de vehículos detectados en el carril en los últimos 5 minutos.
- **volumen\_hora**: Numérico - Cantidad de vehículos detectados en el carril en la última hora.

## **Velocidad promedio vehicular en las principales avenidas de Montevideo**

- **cod\_detector:** Numérico - ID de la cámara que monitorea un carril específico para detectar vehículos.
- **id\_carril:** Numérico - Número del carril monitoreado (1, 2, 3, ...).
- **fecha:** AAAA-MM-DD - Día en que se realizó la medición.
- **hora:** hh:mm:ss - Hora en que se realizó la medición.
- **dsc\_avenida:** Texto - Nombre de la avenida donde se mide el tráfico.
- **dsc\_int\_anterior:** Texto - Nombre de la vía desde donde vienen los vehículos.
- **dsc\_int\_siguiente:** Texto - Nombre de la vía hacia donde se dirigen los vehículos.
- **latitud:** Float - Latitud del lugar de medición.
- **longitud:** Float - Longitud del lugar de medición.
- **velocidad\_promedio:** Numérico - Promedio de las velocidades de los vehículos que circularon por el carril durante los últimos 5 minutos.

## **Ubicación de sensores de medición de conteo vehículos**

- **dsc\_avenida:** Texto - Nombre de la avenida donde se encuentra el sensor o cámara y donde se mide el tránsito.
- **dsc\_int\_anterior:** Texto - Nombre de la vía que forma el cruce desde donde vienen los vehículos.
- **dsc\_int\_siguiente:** Texto - Nombre de la vía que forma el cruce donde está el sensor. En general, el sensor se encuentra un poco antes de esta vía. El sentido de circulación será desde el cruce con **dsc\_int\_anterior** hacia el cruce con **dsc\_int\_siguiente**.
- **latitud:** Float - Coordenada que indica la latitud de la ubicación del sensor.
- **longitud:** Float - Coordenada que indica la longitud de la ubicación del sensor.

Sobre estos datos en particular, son *100 sensores* que se van cambiando de ubicación mes a mes.

## **Base de datos**

Debido a que los datos están estrechamente relacionados y a su vez son sumamente masivos, hemos decidido utilizar una base de datos quedando de la siguiente manera.

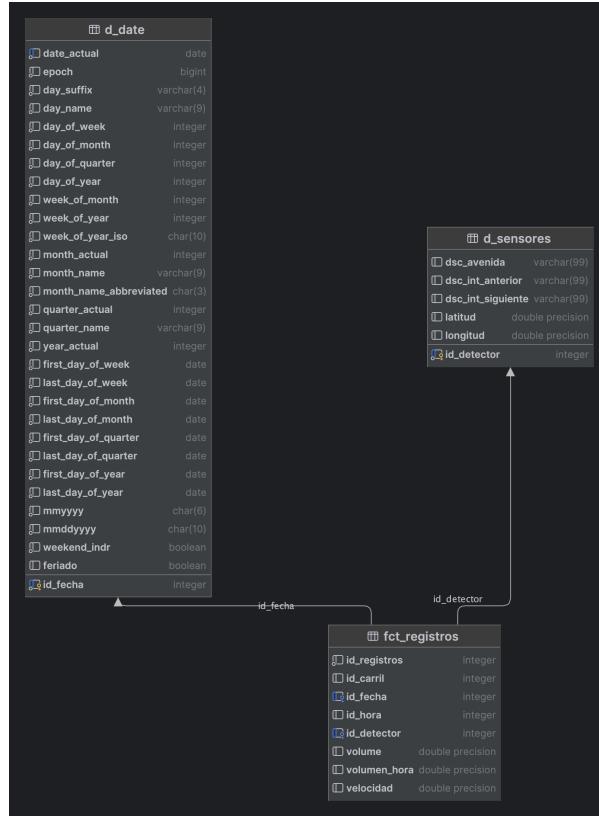


Figure 1: Diagrama de la base de datos

Nuestra tabla principal será `fct_registros`.

#### Tabla: fct\_registros

- Cantidad de datos: 85386695.
- Variables de la tabla:
  - `id_registros`: Numérico (*Primary Key*).
  - `id_carril`: Numérico.
  - `id_fecha`: Numérico (*Foreign Key*, vinculado con `d_sensores`). La fecha de la que fue tomada el registro, tiene el formato *YYYY-MM-DD*
  - `id_hora`: Numérico. Hora en la que fue tomado el registro con formato *HHMM*.
  - `id_detector`: Numérico (*Foreign Key*, vinculado con `d_date`).
  - `volume`: Numérico. Cantidad de vehículos que pasaron en los últimos 5 minutos.
  - `volumen_hora`: Numérico. Cantidad de vehículos que pasaron en la última hora.
  - `velocidad`: Numérico. Velocidad promedio de los vehículos registrados en los últimos 5 minutos. Unidad en km/h

#### Tabla: d\_sensores

- Cantidad de datos: 273
- Variables de la tabla:
  - `id_detector`: Numérico (*Primary Key*).
  - `dsc_avenida`: Texto. Calle donde se encuentra el sensor.

- **dsc\_int\_anterior**: *Texto*. Cruce previo de la calle en **dsc\_avenida**.
- **dsc\_int\_siguiente**: *Texto*. Cruce posterior de la calle en **dsc\_avenida**. Estas dos juntas nos dirá que cada sensor se encuentra en *Avenida* entre *Anterior* y *Siguiente*.
- **latitud**: *Numérico continuo*.
- **longitud**: *Numérico continuo*. Junto a **latitud** nos indica las coordenadas geográficas del sensor.
- **barrio**: *Texto*. Esta variable fue creada a partir del paquete **geouy**

**Tabla: d\_date**

- Cantidad de datos: 3652
- Variables de la tabla:
  - **id\_fecha**: *Numérico (Primary Key)*
  - **date\_actual**: *Fecha*. Secuencia de fechas desde el 01-01-2021 con formato *YYYY-MM-DD*
  - **epoch**
  - **day\_suffix**: *Texto*. Fecha del día abreviado.
  - **day\_name**: *Texto*. Nombre del día
  - **day\_of\_week**: *Numérico*. Día de la semana que indica 1 como lunes, 2 como martes, etc.
  - **day\_of\_month**: *Numérico*. Fecha del mes, va desde 1 hasta 31.
  - **day\_of\_quarter**: *Numérico*. Día del cuatrimestre.
  - **day\_of\_year**: *Numérico*. Día del año, del 1 al 366.
  - **week\_of\_month**: *Numérico*. Semana de cada mes, valores del 1 al 5.
  - **week\_of\_year**: *Numérico*. Semana del año, valores del 1 al 53.
  - **week\_of\_year\_iso**: *Texto*. Variable que combina el año, la semana del año y el día de la semana.
  - **month\_actual**: *Numérico*. Mes del año tomado como número, enero como 1, febrero como 2 y así sucesivamente.
  - **month\_name**: *Texto*. Mes del año traducido en texto, de enero a diciembre
  - **month\_name\_abbreviated**: *Texto*. Mes del año en formato abreviado.
  - **quarter\_actual**: *Numérico*. Indica el cuatrimestre correspondiente con números del 1 al 4.
  - **quarter\_name**: *Texto*. Indica el cuatrimestre en formato de texto, primero, segundo, tercero y cuarto.
  - **year\_actual**: *Numérico*. Indica el año.
  - **first\_day\_of\_week**: *Fecha*. Indica el primer día de la semana que corresponde tal fecha.
  - **last\_day\_of\_week**: *Fecha*. Indica el último día del rango de la semana correspondiente.
  - **first\_day\_of\_month**: *Fecha*. Límite inferior que indica a qué mes corresponde cada fecha.
  - **last\_day\_of\_month**: *Fecha*. Límite superior que indica a qué mes corresponde cada fecha.
  - **first\_day\_of\_quarter**: *Fecha*. Límite inferior que indica a qué cuatrimestre corresponde cada fecha.
  - **last\_day\_of\_quarter**: *Fecha*. Límite superior que indica a qué cuatrimestre corresponde cada fecha.
  - **first\_day\_of\_year**: *Fecha*. Límite inferior que indica a qué año corresponde cada fecha.
  - **last\_day\_of\_year**: *Fecha*. Límite superior que indica a qué año corresponde cada fecha.
  - **mmyyyy**: *Numérico*. Secuencia de caracteres que indica el mes y el año en formato MMYYYY
  - **mmddyyyy**: *Numérico*. Secuencia de caracteres que indica el mes, la fecha y el año en formato MMDDYYYY.
  - **weekend\_indr**: *Lógico*. TRUE si la fecha tiene como día de la semana sábado o domingo, FALSE en caso contrario.
  - **feriado**: *Lógico*. TRUE si la fecha correspondiente coincide con días feriados en Uruguay, FALSE en caso contrario.

## Análisis exploratorio

En nuestro proyecto tenemos datos que tienen una dimension geo-espacial, por lo que es importante tener en cuenta que la informacion que tenemos no es homogenea. Tambien es importante tener en cuenta que la informacion que tenemos es de un periodo de tiempo acotado.

Dicho esto, para empezar, me parecio adecuado comprobar la integridad de los datos, es decir, ver si tenemos datos faltantes o datos que no tienen sentido.

### Datos faltantes y nulos

```
##      atributo cant_total cant_null cant_0 porc_null porc_0
## 1    velocidad 85386695          0 8873659 0.000000 10.39232
## 2 volumen_hora 85386695          0 8873659 0.000000 10.39232
## 3      volume 85386695          0 8873659 0.000000 10.39232
## 4     id_fecha 85386695          0          0 0.000000 0.00000
## 5   id_detector 85386695        2274          0 0.002663 0.00000
```

En 2274 datos se perdio la informacion de la ubicacion del sensor, por lo que no sabemos de que calle se trata. En el 10.39232% de los datos se detecto velocidad 0 y en el mismo porcentaje se detecto volumen 0, se tiene que averiguar si son en los mismos registros o no.

```
##      cant_total cant_0 porc_0
## 1    85386695 8873659 10.39232
```

Las cantidades coinciden, por lo que se puede asumir que son registros donde el sensor no detecto ningun vehiculo. Indicando que el 10.39232% de los datos son registros donde no se detecto ningun vehiculo.

Sobre los datos faltantes de la ubicacion del sensor, son datos que no se pueden recuperar, por lo que se tendran que descartar.

Se quiso averiguar en que fecha se perdio la informacion de la ubicacion del sensor, para ver si se podia recuperar la informacion de otra forma, pero no se pudo.

```
##      id_fecha cant_total cant_null porc_null
## 1 20210724      131083      259 0.197585
## 2 20210725      131607      287 0.218074
## 3 20210726      132574      288 0.217237
## 4 20210727      132635      288 0.217137
## 5 20210728      131988      288 0.218202
## 6 20210729      130631      288 0.220468
## 7 20210730      130224      288 0.221157
## 8 20210731      129410      288 0.222548

##      id_registros      id_carril      id_fecha      id_hora
## Min.   :91804143   Min.   :2   Min.   :20210724   Min.   : 0
## 1st Qu.:91954682   1st Qu.:2   1st Qu.:20210726   1st Qu.: 610
## Median :921116584   Median :2   Median :20210728   Median :1205
## Mean   :92111331   Mean   :2   Mean   :20210728   Mean   :1192
## 3rd Qu.:92265955   3rd Qu.:2   3rd Qu.:20210730   3rd Qu.:1800
## Max.   :92403719   Max.   :2   Max.   :20210731   Max.   :2355
##      id_detector      volume      volumen_hora      velocidad
## Mode:logical      Min.   : 0.000   Min.   : 0.00   Min.   : 0.00
```

```

##  NA's:2274      1st Qu.: 2.000    1st Qu.: 24.00    1st Qu.:37.00
##              Median : 4.000      Median : 48.00    Median :43.00
##              Mean   : 4.978      Mean   : 59.74    Mean   :38.39
##              3rd Qu.: 7.000      3rd Qu.: 84.00    3rd Qu.:47.00
##              Max.   :37.000      Max.   :444.00    Max.   :90.00
##      fecha          hora
##  Min.   :2021-07-24  Min.   : 0
##  1st Qu.:2021-07-26  1st Qu.: 610
##  Median :2021-07-28  Median :1205
##  Mean   :2021-07-27  Mean   :1192
##  3rd Qu.:2021-07-30  3rd Qu.:1800
##  Max.   :2021-07-31  Max.   :2355

```

Se puede ver que los datos faltantes de la ubicacion del sensor van desde el 24/07/2021 hasta el 31/07/2021. Tambien se puede ver que todos los datos faltantes son del carril 2. Quizá se podrían recuperar los datos de la ubicacion del sensor revisando los datos originales pero es irrelevante ya que son pocos datos y no afectan al análisis.

## Ubicacion de los sensores

Los datos que tenemos son de 100 sensores ubicados todos en Montevideo y estos van cambiando de ubicación cada mes. Por lo que el primer paso es ver cuantas ubicaciones distintas tenemos y cuantos sensores hay en cada ubicación.

Para mostrarlo, hemos decidido utilizar un mapa de Montevideo con los barrios y mostramos la cantidad de sensores ubicados en el. En el mapa se puede ver que los sensores están ubicados en 42 de los 62 barrios de Montevideo. Los barrios que tienen sensores son 42 sobre 62 siendo los barrios de Buceo, Centro, Pocitos y Unión con más de 20 sensores.

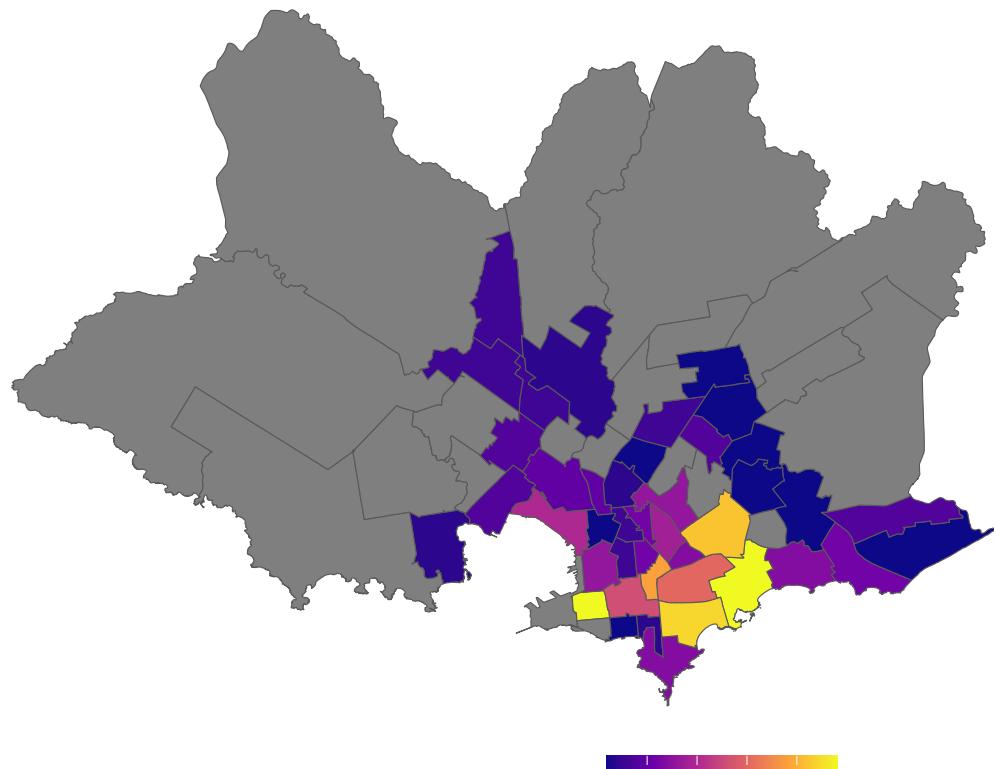


Figure 2: Mapa de Montevideo con cantidad de sensores por barrio.

Ahora quiero mostrar la cantidad de datos que tenemos por ubicacion, para ver si hay alguna ubicacion en particular que tenga mas o menos datos que las demás.

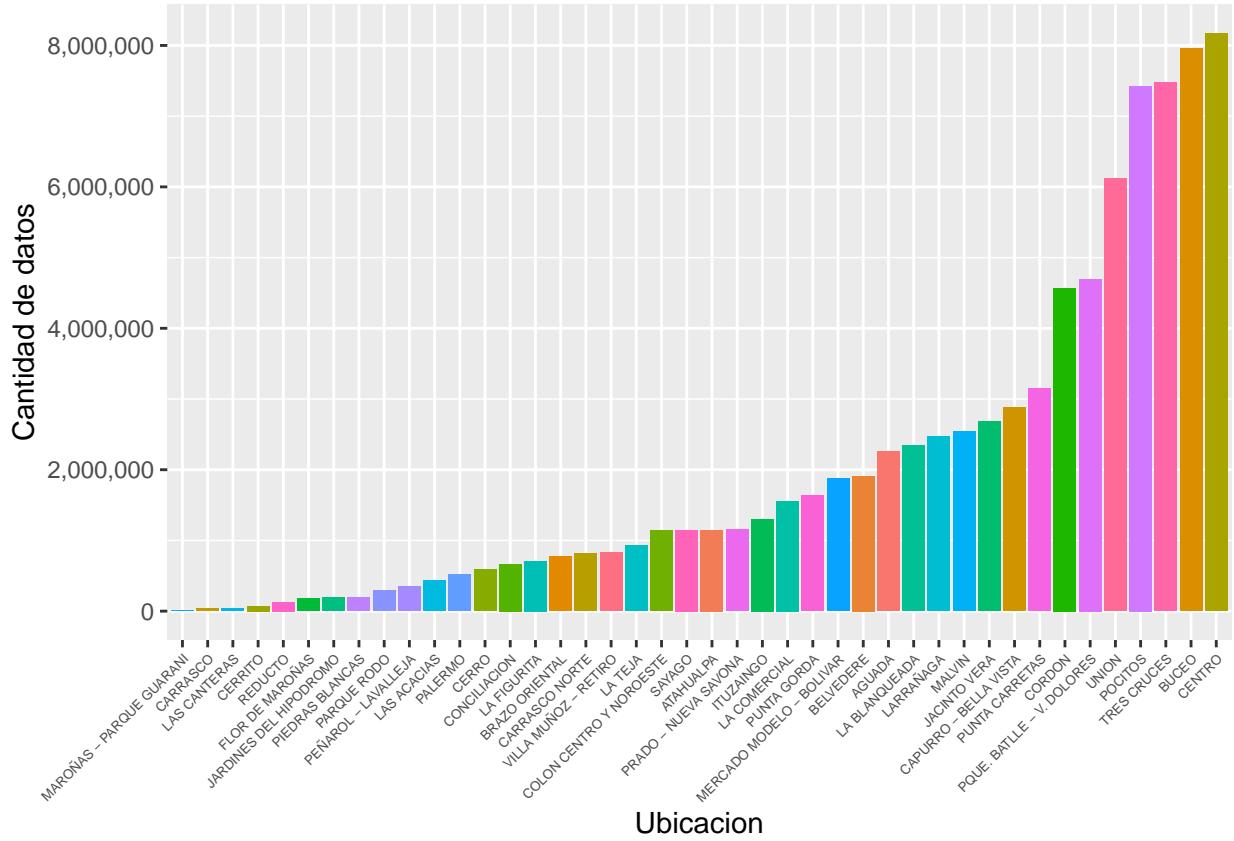


Figure 3: Cantidad de datos por ubicacion

Se puede observar que la cantidad de datos por ubicacion no es para nada homogenea. Los barrios con mayor cantidad de datos aportados al dataset son Union, Pocitos, Tres Cruces, Buceo y Centro. Por otro lado Maroñas, Carrasco, Las Canteras y Cerrito son los que menos datos aportan.

Ahora me interesaria saber cuales son los barrios mejores representados en el dataset, es decir, cuales son los barrios que tienen mas datos por metro cuadrado.

primero calculo el area de cada barrio y luego calculo la cantidad de datos por metro cuadrado.

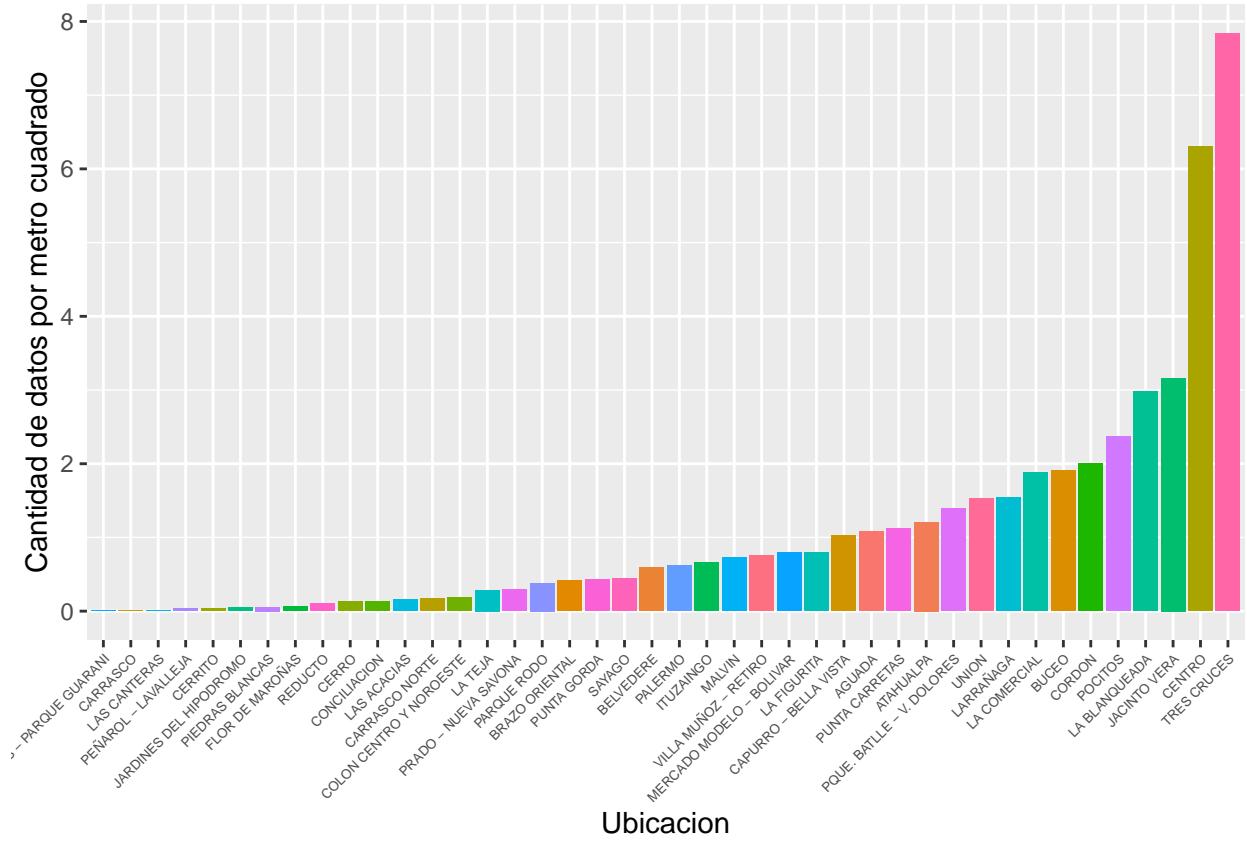


Figure 4: Cantidad de datos por ubicacion ponderado por area

Se puede observar que los barrios con mayor cantidad de datos por metro cuadrado son Pocitos, La Blanqueada, Jancito Vera, Centro y Tres Cruces. Por otro lado Maroñas, Carrasco, Las Canteras, Peñarol y Cerrito son los que menos datos aportan por metro cuadrado.

## Principales variables

Las variables que se van a analizar son las siguientes: - **volume**: Numérico. Cantidad de vehiculos que pasaron en los últimos 5 minutos. - **volumen hora**: Numérico. Cantidad de vehiculos que pasaron en la ultima hora. - **velocidad**: Numérico. Velocidad promedio de los vehiculos registrados en los ultimos 5 minutos. Unidad en km/h

## Velocidad

Resumen de la variable velocidad

```
##      cant_0 minimo primer_cuartil mediana tercer_cuartil maximo promedio    desvio
## 1 8873659      0           22        32          41       144 31.31286 17.07865
```

Ahora veamos la distribucion de la velocidad registrada Para mejor visualizacion, voy a dejar de lado los datos donde la velocidad es 0

```

## Warning: Using 'size' aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use 'linewidth' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.

```

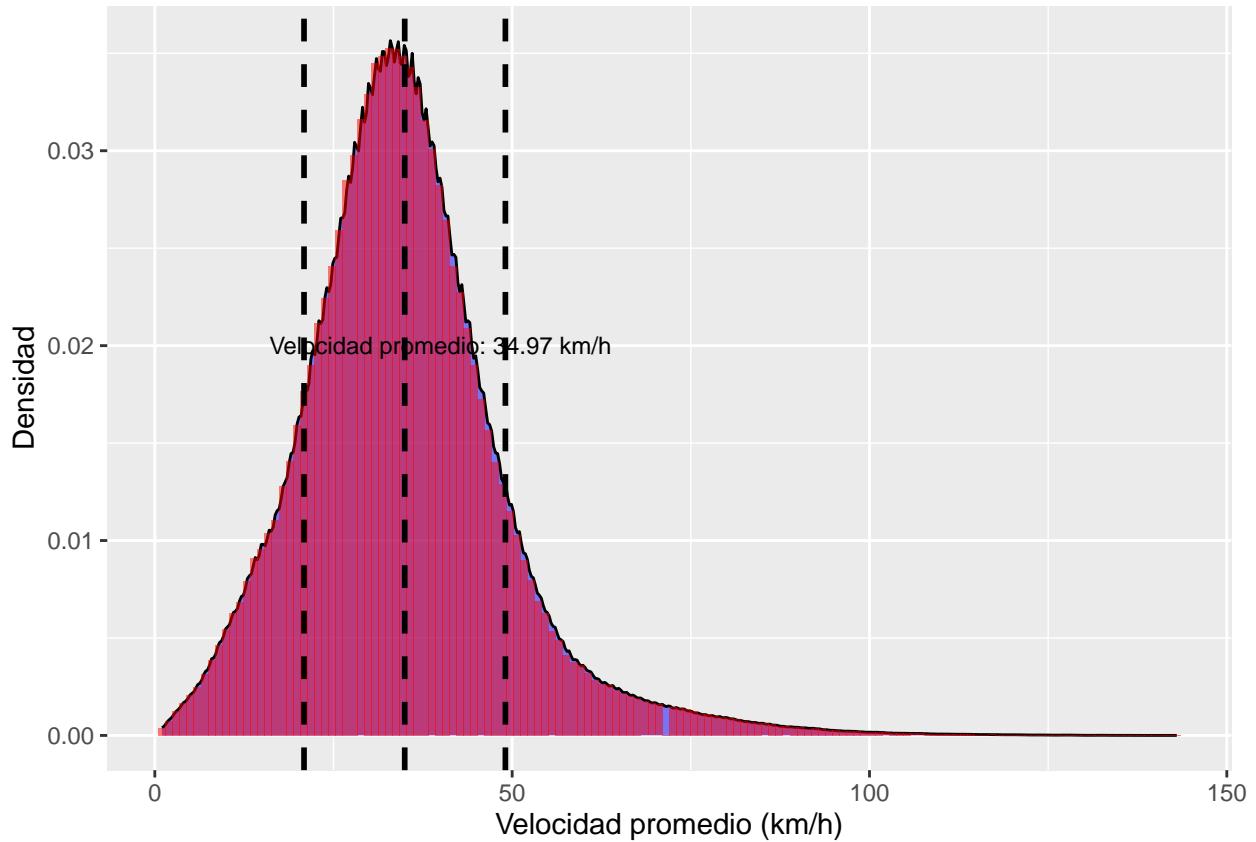


Figure 5: Distribucion de velocidad promedio

Se puede observar que la distribucion de la velocidad es normal, con una media de 31.31 km/h y un desvio de 17.08 km/h. El 68% de los datos se encuentran entre 14.23 km/h y 48.39 km/h.

## Volumen

Resumen de la variable volumen

minimo, maximo, promedio, desvio, cuartiles

```

##     cant_0 minimo cuartil_1 mediana cuartil_3 maximo promedio    desvio
## 1 8873659      0        3       11       26      659 17.27588 19.18341

```

Se ve que la cantidad minima de vehiculos registrados en 5 minutos es 0 y la maxima es 659. El promedio de volumen es 17.28 vehiculos y el desvio estandar es 19.18 vehiculos.

Ahora veamos la distribucion del volumen registrado Para mejor visualizacion, voy a dejar de lado los datos donde el volumen es 0

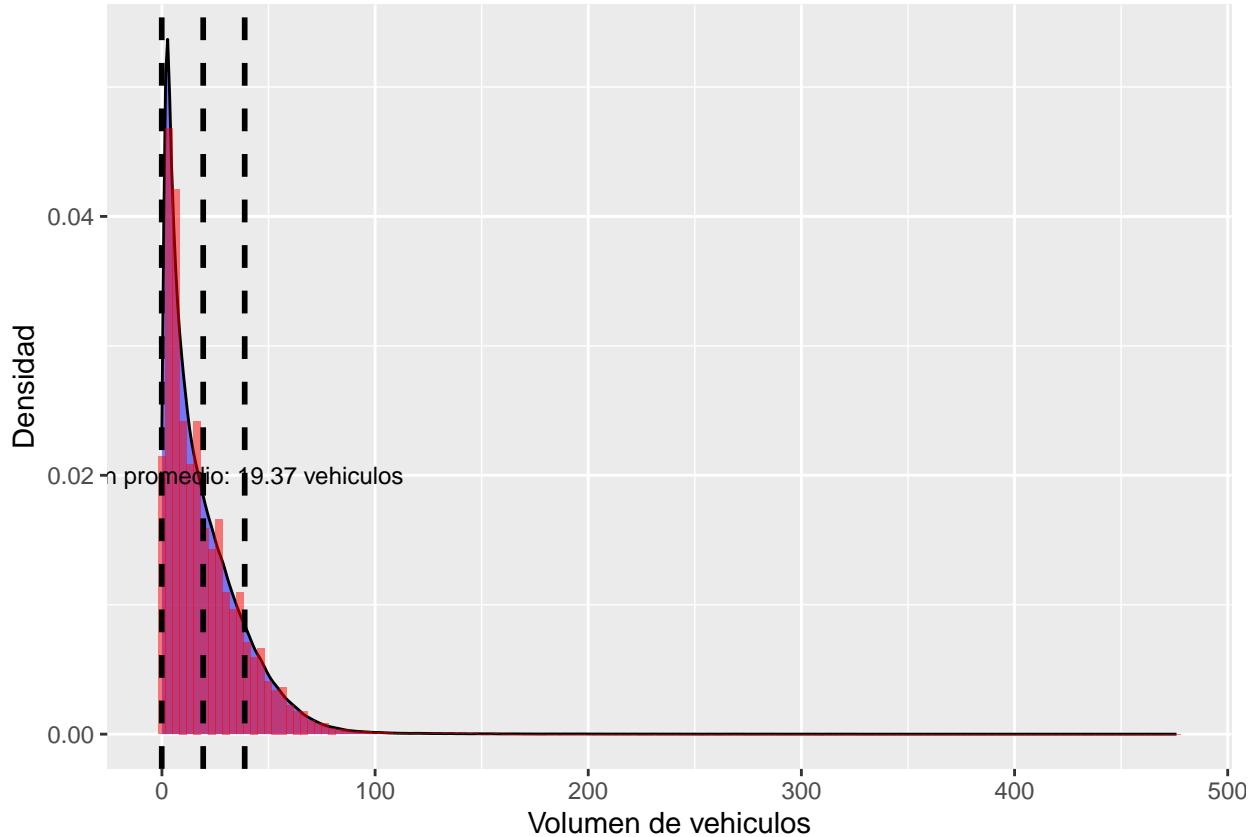


Figure 6: Distribucion de volumen

Se puede observar que la mayoría de los valores son menores a 100, en particular el 75% de los datos son menores a 26 vehículos.

## Preguntas de Investigacion

Las preguntas que dirigen este análisis son las siguientes: 1. ¿Existe alguna correlación entre el volumen y la velocidad? 2. ¿Cuáles son las calles con los mayores promedios de velocidad en Montevideo? ¿Con qué frecuencia se cometan excesos de velocidad? 3. ¿Cómo va variando el volumen y velocidad medidos a través del TIEMPO?

### **¿Existe alguna correlación entre el volumen y la velocidad?**

Haremos un gráfico de puntos para visualizarlo. Para los datos usaremos una muestra aleatoria de toda la base de datos

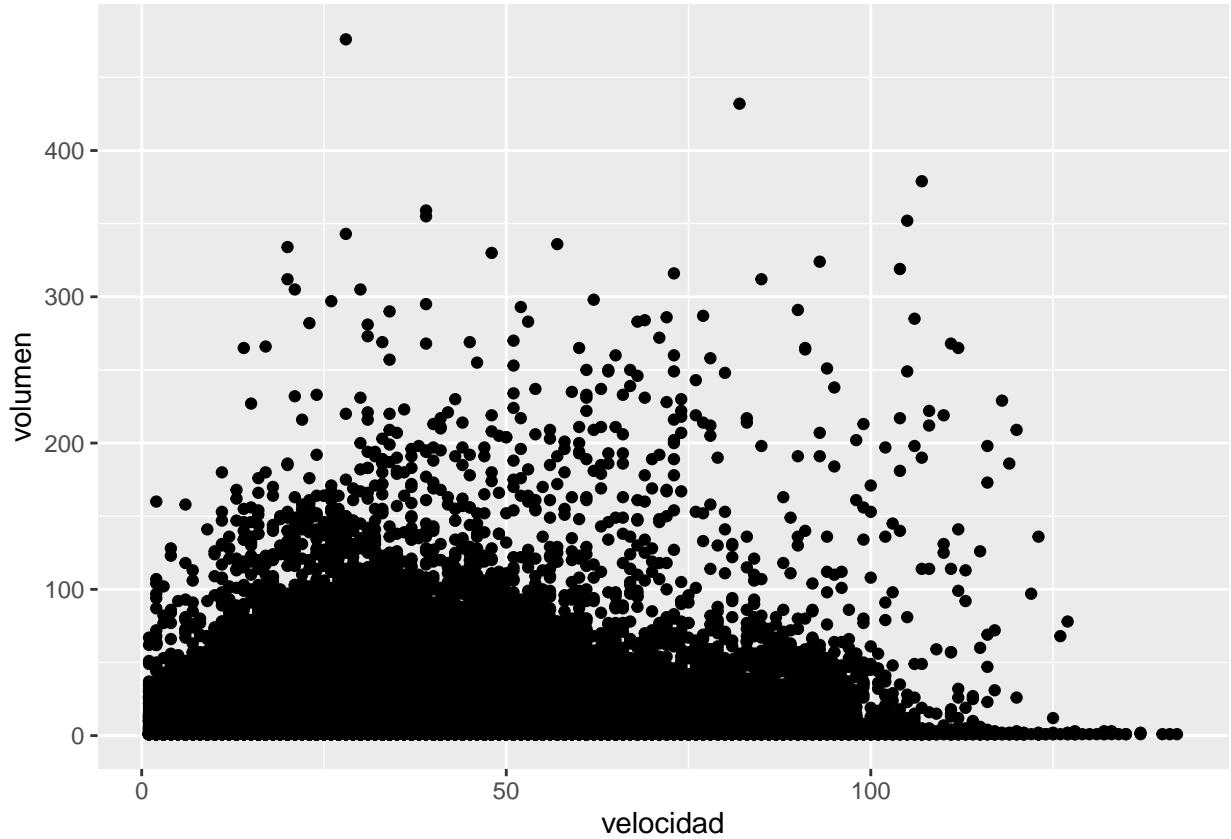


Figure 7: Grafico de puntos de velocidad y volumen

Definitivamente **no hay una relacion lineal** entre velocidad y volumen

```
##           velocidad      volumen
## velocidad  1.00000000 -0.03175882
## volumen    -0.03175882  1.00000000
```

La correlacion dio -0.03, lo que indica que no hay una correlacion lineal entre las variables.

WICKHAM, HADLEY. 2023. *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data*.  
O'REILLY MEDIA.