



Inteligencia artificial avanzada para la ciencia de datos

*Reto*

***Etapas 3. Preparación de los datos***

**Diego Arturo Padilla Domínguez - A01552594**

**Keyuan Zhao - A01366831**

**Carolina Herrera Martínez - A01411547**

**Cutberto Arizabalo Nava - A01411431**

**Jose Pablo Cobos Austria - A01274631**

Campus Querétaro

14 de octubre de 2022

# Etapa 3: “Data preparation”

## Selección de datos iniciales

Debido a que en el dataset todas las columnas son importantes para el objetivo, no descartamos ningún dato de momento para nuestros siguientes procesos. Para ello, dividimos las columnas con el rango de 1 al 3 según la importancia para realizar nuestras modelaciones:

Nombre	Importancia
PHONE_ID	3
timestamp	3
bts_id	1
lat	3
lon	3

## Formateo de datos (I)

Antes de iniciar con la construcción de datos, es necesario aumentar la calidad de estos para poder agilizar la velocidad al momento de crear atributos derivados. Para esto, se realiza un primer formateo:

- **Ordenamiento de los datos**

Lo primero que se realizó fue ordenar los datos, a base de las columnas ‘PHONE\_ID’ y ‘timestamp’, esto con el fin de tener una mejor visualización de los datos, además de permitirnos realizar ciertas operaciones de una manera más sencilla.

- **Conversión de ‘timestamp’**

El formato de ‘timestamp’ es de tipo objeto, donde en este formato no nos es de utilidad, por ello lo reformateamos a tipo date, con ello podemos obtener los datos que necesitamos de la fecha.

## Construcción de datos (I)

La primera construcción consistió en la creación de nuevos datos que nos ayudarán a observar los comportamientos de los viajes, para ello se crearon las columnas:

- **Distancia**

Para saber si un dispositivo se está moviendo o no, necesitamos saber si hubo un cambio de ubicación y de que distancia fue este, para ello se ocuparon los campos de 'lat' y 'lon'; con la ayuda de una función matemática pudimos calcular la distancia entre dos puntos.

- **Tiempo (hrs)**

Necesitamos saber cuánto tiempo estuvo parado un dispositivo en un mismo sitio con el fin de saber si su viaje terminó o no. Debido a que los registros que tenemos son de un solo día, se optó por reducir solamente a hora del día.

- **Velocidad**

Para determinar la velocidad en la que se trasladó un dispositivo de un punto a otro se utilizó la fórmula de la velocidad con los datos de tiempo y distancia calculados previamente.

## **Limpieza de datos (II)**

- **Remoción de dispositivos inmóviles**

Al ver el Dataframe resultante se pudo observar que había ciertos dispositivos que jamás se movieron de lugar durante todos los registros que tenemos disponibles, por ello decidimos eliminar todos aquellos registros que correspondieran a esos dispositivos que nunca se movieron.

- **Remoción de viajes imposibles**

Como sabemos, uno de nuestros supuestos es que nuestro viaje no puede ir a más de 150 km/h, por ello se removieron todos aquellos registros en donde hubiese una velocidad mayor a dicha velocidad. Esto se realizó de manera iterativa junto con el cálculo de distancia, tiempo y velocidad, debido a que al actualizar los registros los cálculos también deben de actualizarse.

## **Construcción de datos (II)**

La primera detección de datos faltantes es la ubicación de cada una de las antenas en sus respectivas comunas, para ello se llevaron a cabo distintas acciones:

- **Creación de nuevo archivo de antenas**

Se creó un nuevo archivo en el que se contienen cada una de las antenas con sus respectivas coordenadas geográficas.

- **Obtención de ubicaciones**

Con la ayuda de la librería de Geopy se obtuvieron las respectivas comunas de cada una de las antenas.

## **Integración de datos (I)**

- **Combinación de ambos conjuntos de datos**

Ya teniendo ambos dataframes se unieron basándose en el 'bts\_id' esto con el fin de tener un dataframe unificado en donde pudiéramos ver en qué comuna estuvo cada dispositivo en cada una de sus respectivas conexiones.

## **Construcción de datos (III)**

- **Creación de nuevo ID de antenas.**

Se pudo observar que hay antenas con una misma ubicación, pero con un ID distinto, por ello se optó por concatenar 'lat' y 'lon', para que con ello tuviéramos un ID único por cada ubicación.

## **Limpieza de datos (III)**

Ya teniendo todos los datos en un solo archivo, haber calculado y obtenido todos los datos necesarios, hay columnas que ya no son necesarias, debido a que ya cumplieron su propósito, por ello se optó por eliminarlas:

- Velocidad
- bts\_id
- lat
- lon

## **Construcción de datos (IV)**

Con los datos limpios se creó una matriz en donde se encuentran cada uno de los viajes realizados.

Cada uno de los viajes tenía que cumplir las siguientes condiciones (al no cumplir con alguna de las condiciones el viaje se corta e inicia uno de inmediato):

Viajes no mayores a 2 horas

Viajes no mayores a 25 km

## **Limpieza de datos (IV)**

Había registros en donde de uno a otro se rompían las condiciones, así que pasaban estos registros como viaje directamente a la matriz, por ello se procedieron a borrar dichos registros.

## **Construcción de datos (V)**

Los datos actuales de origen y destino están dados en coordenadas, pero esto no nos es útil, por ello se transformaron estos datos en los nombres de cada comuna con su respectiva coordenada.

## **Limpieza de datos (V)**

Dentro de los datos existen algunas repeticiones de comunas con diferentes artículos, por ejemplo: la barrenechea y lo barrenechea. A pesar de que son de la misma comuna, pero con distinto artículo, se remplazaron estos datos y estandarizarlos en una sola comuna.

## **Disociación por modelos**

A partir de este punto la preparación de datos varía dependiendo de cada modelo.

## **Modelo 0 (Heatmap viajes origen-destino)**

### **Construcción de datos (I)**

Se creó una nueva matriz con base en la matriz general de viajes, en donde se agruparon los viajes por origen y destino, con ello se obtuvo la cantidad de viajes que hubo entre cada una de las comunas.

## **Modelo 1 (Regresión lineal)**

### **Construcción de datos (I)**

Se generó una matriz que contiene la cantidad de viajes entre cada una de las comunas en cada hora del día con base en la matriz general de viajes.

Se obtuvieron los datos de cada comuna de la cantidad de iglesias, escuelas, hospitales y lugares de interés.

Se unieron ambos conjuntos de datos.