# MEMORIA TFM
# “FIREMAN INTERVENTIONS”

**BEATRIZ ARÉVALO SÁNCHEZ**

**MASTER DATA SCIENCE**

El primer propósito de este TFM era conocer donde se produciría la siguiente intervención de los bomberos en la ciudad de Nueva York, en este caso ese dato nos lo proporcionaría la columna *FIRE_BOXX*. Tras limpiar el dataset, no fui capaz de hacerlo, con lo cual cambié mi objetivo y decidí que iba a ser conocer el tiempo de duración de una intervención, la columna *TOTAL_INCIDENT_DURATION.*

1. **TFM_primeros_pasos**

En este notebook lo que se realiza es la transformación de alguna columna de los dos dataset para poder unirlos en uno solo. El dataset resultante es el llamado "df_nulls".

En el dataset principal, *"Incidents_Response_to_by_Fire_Companies"*, está compuesto por 2.518.758 filas y 24 columnas.

1. IM_INCIDENT_KEY: Identificador único para cada incidente que sirve como clave primaria
2. FIRE_BOX: Identificador del área de la caja de alarma contra incendios en la que ocurrió el incidente, único por municipio
3. INCIDENT_TYPE_DESC: El código y la descripción del tipo de categoría de incidente
4. INCIDENT_DATE_TIME: La fecha y hora en que se registró el incidente en el sistema de Despacho Asistido por la Computadora
5. ARRIVAL_DATE_TIME: La fecha y hora en que la primera unidad llegó a escena
6. UNITS_ONSCENE: Número total de unidades que llegaron a la escena.
7. LAST_UNIT_CLEARED_DATETIME: La fecha y hora en que se completó el incidente y la última unidad despejó la escena.
8. HIGHEST_LEVEL_DESC: El nivel de alarma más alto que recibió el incidente.
9. TOTAL_INCIDENT_DURATION: El número total de segundos desde que se creó el incidente hasta que se cerró
10. ACTION_TAKEN1_DESC: El código y la descripción de la primera acción tomada.
11. ACTION_TAKEN2_DESC: El código y la descripción de la segunda acción tomada.
12. ACTION_TAKEN3_DESC: El código y la descripción de la tercera acción tomada.
13. PROPERTY_USE_DESC: El código y la descripción del tipo de calle o edificio donde ocurrió el incidente
14. STREET_HIGHWAY: El nombre de la calle donde ocurrió el incidente.
15. ZIP_CODE: El código postal donde ocurrió el incidente.
16. BOROUGH_DESC: El distrito donde ocurrió el incidente.
17. FLOOR: El piso del edificio donde ocurrió el incidente.
18. CO_DETECTOR_PRESENT_DESC: Indicador de cuándo estaba presente un detector de $CO^2$
19. FIRE_ORIGIN_BELOW_GRADE_FLAG: Indicador de cuando el incendio se originó por debajo del grado
20. STORY_FIRE_ORIGIN_COUNT: Historia en la que se originó el fuego.
21. FIRE_SPREAD_DESC: Que lejos se extendió el fuego del objeto de origen

22. DETECTOR_PRESENCE_DESC: Indicador de cuándo estaba presente un detector
23. AES_PRESENCE_DESC: Indicador de cuando un sistema de extinción automática está presente
24. STANDPIPE_SYS_PRESENT_FLAG: Indicador de cuando una tubería vertical estaba presente en el área de origen del incendio

El segundo dataset, *"FIRE_BOX",* tiene 16.286 filas y 4 columnas:

1. Long: Punto longitudinal donde esta ubicada la caja de alarma
2. Lat: Punto latitudinal donde esta ubicada la caja de alarma
3. FIRE_BOX: Identificador del área de la caja de alarma contra incendios en la que ocurrió el incidente, único por municipio
4. Adress: Dirección del Fire Box

El merge de ambos se ha realizado por la columna FIRE_BOX, con el inconveniente de que al no tener en el segundo dataset todas las cajas de alarmas que hay, se ha perdido gran cantidad de los datos.

## 2. TFM_Clean_Nulls

En este paso lo que se ha realizado ha sido la limpieza de los valores NaN, se han eliminado columnas que no se iban a utilizar. Además, muchas de ellas estaban únicamente con valores nulos. En el notebook esta explicado, paso por paso, lo que he ido realizando.

## 3. TFM_ML_Nuevo

En este notebook, con el fin de evitar los problemas de memoria que tuve en la pasada entrega he hecho varias cosas.

Lo primero es modificar dos variables que voy a usar para predecir mis modelos. Estas son, CODE_INCIDENT y CODE_PROPERTY.

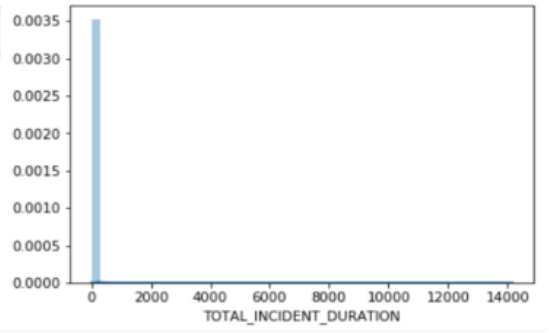
Con ambas he realizado los mismos pasos:

- Obtener los cuantiles, con esto he observado que el 98% (aprox.) de los puntos se concentran en unas pocas variables, 16 en el caso de CODE_PROPERTY y 36 para CODE_INCIDENT.
- Incluir en una variable OTHERS el resto de los datos.
- Para la columna CODE_PROPERTY elimino, además, todos los valores 'UUU' ya que no nos aportan información.

Para la variable objetivo TOTAL_INCIDENT_DURATION al hacer un *describe* podemos ver que la media esta en torno a 24 minutos, pero el valor máximo son 14.164 minutos, con lo cual podemos deducir que hay *outliers.* Al graficarlo se ve más claro.

```
df4['TOTAL_INCIDENT_DURATION'].describe()

count    855101.000000
mean         24.618280
std          63.720123
min           0.000000
25%          13.266667
50%          18.050000
75%          25.433333
max       14164.033333
Name: TOTAL_INCIDENT_DURATION, dtype: float64
```

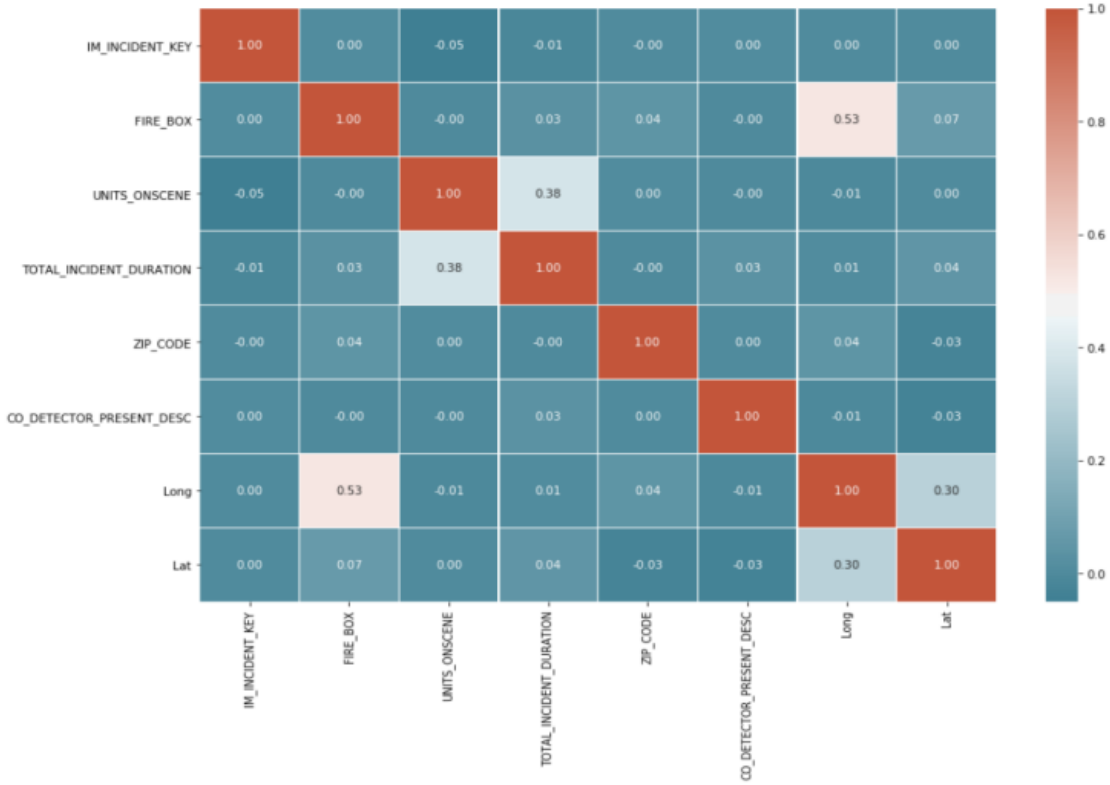



Para tratar estos valores perdidos he utilizado una función que he obtenido del siguiente enlace:

https://towardsdatascience.com/5-ways-to-detect-outliers-that-every-data-scientist-should-know-python-code-70a54335a623

Decido tomar por *outlier* todo valor que este por encima de 200 minutos.

He realizado una matriz de correlación para poder ver si hay relación entre las variables y el resultado es este:




Tres de las columnas que voy a usar en mis modelos son variables categóricas, CODE_PROPERTY, CODE_INCIDENT y BOROUGH_DESC. Aplico sobre ellas *get_dummies* para poder hacer las predicciones.

## METODOLOGÍA

Como modelos para poder llevar a cabo el objetivo de este proyecto he decidido utilizar los siguientes algoritmos de *Machine Learning*

- Regresión Lineal
- K-neighbors
- Random Forest

Las métricas que voy a usar con los tres modelos son MAE, RMSE Y R2.

## RESULTADOS

Tras aplicar los algoritmos comentados en metodología, los resultados obtenidos fueron los siguientes:

### Regresión Lineal

| **Métricas** | |
|---|---|
| MAE | 9.774716 |
| RMSE | 16.10889 |
| R2 | 0.285279 |

### KNN

| **Métricas** | |
|---|---|
| MAE | 10.225962 |
| RMSE | 16.477331 |
| R2 | 0.253257 |

### Random Forest

| **Métricas** | |
|---|---|
| MAE | 9.808306 |
| RMSE | 16.119014 |
| R2 | 0.285381 |

En los tres modelos podemos observar que el $R^2$ es muy bajo, lo cual significa que las variables que estamos utilizando no explican la duración de los incidentes.

El modelo que mejores resultados ofrece es la regresión lineal, aunque seguramente si se hubieran aplicado los hiperparámetros óptimos para KNN y Random Forest los resultados de ambos mejorarían.

## CONCLUSIONES

Los conocimientos con los que contaba cuando comencé el máster, pese a haber hecho el curso anterior de Python y Estadística aplicada, eran bastante pobres. He encontrado muchas dificultades para poder realizar el TFM y ello me ha llevado a perder mucho tiempo “googleando” buscando la información pertinente.

Considero que la diferencia entre el trabajo que presente en diciembre y este es notable, he conseguido entrenar los modelos y obtener unos resultados.