Tipología y Ciclo de Vida de los Datos: PRA2 - Limpieza y validación de los datos

Autores: Joel Bustos - Iván Ruiz

Junio 2020

Table of Contents

[Introducción 2](#_Toc40947866)

[Presentación. 2](#_Toc40947867)

[1. Descripción del dataset. ¿Por qué es importante y qué pregunta/problema pretende resolver? 3](#_Toc40947868)

[2. Integración y selección de los datos de interés a analizar. 5](#_Toc40947869)

[2.1 Resumen de tratamientos previos 9](#_Toc40947870)

[2.2 Carga del nuevo archivo tras el procesado de datos. 17](#_Toc40947871)

[3. Limpieza de los datos. 19](#_Toc40947872)

[3.1 ¿Los datos contienen ceros o elementos vacíos? ¿Cómo gestionarías cada uno de estos casos? 19](#_Toc40947873)

[3.2 Identificación y tratamiento de valores extremos 21](#_Toc40947874)

[4. Limpieza de los datos. 23](#_Toc40947875)

[4.1 Selección de los grupos de datos que se quieren analizar/comparar (planificación de los análisis a aplicar) 23](#_Toc40947876)

[4.2 Comprobación de la normalidad y homogeneidad de la varianza 23](#_Toc40947877)

[4.3 Aplicación de pruebas estadísticas para comparar los grupos de datos 23](#_Toc40947878)

[5. Representación de los resultados a partir de tablas y gráficas 24](#_Toc40947879)

[6. Resolución del problema. A partir de los resultados obtenidos, ¿cuáles son las conclusiones? ¿Los resultados permiten responder al problema? 25](#_Toc40947880)

[Contribuciones 26](#_Toc40947881)

[Bibliografía 26](#_Toc40947882)

**¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.¡Error! Referencia de hipervínculo no válida.**

# Introducción

## Presentación.

A lo largo de esta segunda práctica, vamos a tratar de profundizar en el análisis planteado durante la primera parte de la asignatura. Así, hacemos referencia tanto al repositorio que creamos, como a la documentación generada a través del siguiente enlace: <https://github.com/iruiper/Cyberattacks-History>

A pesar de las dificultades que plantea el set de datos sobre el que trabajaremos, nos gustaría tratar de cerrar las inquietudes y motivaciones que nos llevaron, en primera instancia, a trabajar sobre la problemática de los ataques de ciberseguridad. En este sentido, nos gustaría iniciar esta segunda exposición rescatando la motivación planteada en el proyecto de obtención de datos:

*“Los equipos de seguridad han necesitado incorporar, cada vez más, perfiles técnicos en el área de la ciberseguridad. Estos equipos técnicos, normalmente con un conocimiento muy específico, en ocasiones no disponen de demasiadas herramientas que les permitan ser proactivos y anticiparse a las nuevas tendencias y técnicas de ciberataque. De esta forma, acaban adaptando un comportamiento reactivo, realizando tareas de mantenimiento y de respuesta ante incidentes.”*

*“Nos planteábamos, como contexto para la presente práctica, recopilar datos históricos de ciberataques con el objetivo de crear un modelo predictivo que sirviese de soporte al equipo de seguridad de una empresa. Idealmente, estudiando lo que está ocurriendo en relación a delitos cibernéticos, los equipos internos de las distintas entidades, podrían tratar de prepararse mejor contra aquellos riesgos a los que los modelos estadísticos les pudieran sugerir que se encuentran más expuestos.”*

Bajo esta situación, vamos a tratar de plantear un problema concreto que podría analizarse utilizando los datos del sitio web <https://www.hackmageddon.com/>.

Cabe remarcar que para abordar este segundo proyecto, será necesario aplicar tareas de procesado de datos, ya que la calidad de la información extraída durante la primera práctica, no es la adecuada. De esta forma, se pretenderán resolver problemas de calidad, tales como la falta de integración, la existencia de datos incompletos e inconsistentes, o incluso la carencia de variables relevantes para el análisis.

A pesar de que la propuesta de práctica se centra en tareas de limpieza y acondicionado mediante R, también haremos uso del lenguaje de programación Python, con el objetivo de ampliar las herramientas destinadas a mejorar la calidad de datos, y así, potenciar su posterior análisis.

# 1. Descripción del dataset. ¿Por qué es importante y qué pregunta/problema pretende resolver?

Tal y como se expone en la introducción, disponemos de un set de datos con información de ataques cibernéticos producidos desde el año 2017 hasta la actualidad.

Con esta información, cabe preguntarse hasta qué punto podemos anticiparnos a los problemas que sobrevendrán a una entidad, o cómo estos datos pueden ayudar en asuntos financieros y presupuestarios, para responder a preguntas tales como:

* ¿Hasta qué punto mi compañía se encuentra más expuesta a ciberataques, en función del sector en el que se encuentra?
* ¿Existe algún periodo del año en que debería ampliar el presupuesto y los medios necesarios, al estar más expuesto a recibir ciberataques?
* ¿A qué tipología de ciberataque mi compañía está más expuesta? ¿Debería por ello, analizar la documentación cualitativa, así como los detalles técnicos referentes a este tipo de ataque?
* ¿Hasta qué punto el encontrarme en un mundo globalizado, estoy expuesto a recibir ataques internacionales?
* ¿Existen atacantes bien conocidos, con pautas concretas, a los que me encuentre particularmente expuesto?

Todas las preguntas anteriores, podrían resolverse a partir del análisis de la información contenida en el set de datos obtenido durante la primera práctica, ya que:

* Disponemos de datos sobre las entidades o sectores afectados por los ciberataques que aparecen documentados en el dataset.
* Disponemos de la fecha en la que se han producido los distintos ataques.
* Sabemos el alcance del impacto de los ataques, dado que existe información sobre si el ataque afecto a un país en concreto, o a varios de ellos.
* Tenemos información acerca de la autoría de los ataques.
* Conocemos la tipología de ataque de los distintos incidentes reportados.

En concreto, para que los resultados y los contrastes que llevemos a cabo sean lo más concretos y realistas posibles, vamos a centrarnos en el caso de que **formemos parte del equipo de seguridad de un organismo público**, por lo que nuestros análisis cuantitativos y cualitativos, tratarán de poner en relieve las diferencias entre nuestro sector y los demás. Esta distinción, también puede tener como derivada interesante, averiguar el nivel de gasto e inversión acometido por entidades de otros sectores en materia de ciberseguridad, y a partir de nuestra evaluación del riesgo específico, estudiar si puede ser necesaria la aplicación de nuevas partidas presupuestarias para la defensa contra estas amenazas.

Como hemos visto en los materiales didácticos de Subirats, Pérez y Calvo [1], existen muchos retos diferentes a la hora de integrar y asegurar la calidad de los datos, con el objetivo de dar respuesta a las inquietudes de cualquier analítica de datos.

Adicionalmente, según se desprende de dicho material, así como se muestra en los ejemplos basados en sets de datos estructurados, gran parte de los problemas de calidad se originan en variables cuantitativas. Estas, posteriormente, serán objeto de análisis con el fin de obtener algún tipo de conocimiento.

En nuestro caso, la principal dificultad presente en el set de datos seleccionado, es precisamente la escasez de variables numéricas. En consecuencia, gran parte del reto al que nos enfrentamos, y gran parte de los problemas de procesado de datos que vamos a desarrollar a lo largo de la práctica, irán dirigidos a “crear” datos numéricos, así como estadísticos que permitan resolver las cuestiones planteadas.

# 2. Integración y selección de los datos de interés a analizar.

El primer reto que deberemos resolver en esta etapa de procesado de datos, consistirá en integrar el conjunto de datos recopilados durante la primera práctica. En concreto, es de vital importancia destacar la presencia de heterogeneidad entre las distintas fuentes de información ya que, aunque todas procedan del mismo sitio web, presentan estructuras de datos distintas.

A modo de recordatorio, a continuación, se exponen brevemente los distintos conjuntos de datos recopilados del sitio web <https://www.hackmageddon.com/>.

* **Timeline:** Contiene ciberataques producidos desde el 2019 hasta la actualidad. Esta base de datos, está formada por informes quincenales, que presentan una estructura de datos específica.
* **Master Table 2018:** Ciberataques producidos durante el año 2018 contenidos en una única tabla.
* **Master Table 2017:** Ciberataques producidos durante el año 2017. Al igual que en el caso anterior, se encuentran agregados en forma tabular, pero con una estructura de campos distinta.

En consecuencia, el primer problema con el que nos encontramos, es la existencia de varios archivos csv con una estructura de campos específica y con información de distintos periodos temporales.

El primer paso pues, consistirá en unir todos los archivos en bruto, con el objetivo de crear un único conjunto de datos, que servirá para identificar las tareas de procesado necesarias para cada campo, así como la tipología de datos presente en estos.

Cabe remarcar que este conjunto de datos, no pretende ser el archivo definitivo con la información procesada, a partir del cual, se realizarán tareas de minería de datos, sino que es un primer borrador, sobre el que se analizarán las distintas estrategias de procesado de datos que se deberían de aplicar, para obtener el conocimiento necesario, que permita resolver los planteamientos propuestos.

# Almacenamos el set de datos bruto en el frame "attacks\_Raw" para un análisis preliminar de algunos de los campos que utilizaremos  
attacks\_Raw <- read.csv2(file='DatosAtaques\_2017\_2020\_RAW.csv',stringsAsFactors = TRUE)  
  
# Mostramos la estructura del archivo recién cargado  
str(attacks\_Raw)

## 'data.frame': 4468 obs. of 11 variables:  
## $ ID : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ Date : Factor w/ 1054 levels "01/01/2017","01/01/2018",..: 1 1 1 1 71 104 104 104 104 138 ...  
## $ Author : Factor w/ 564 levels ""," LulzSecITA",..: 17 7 166 7 38 7 289 7 7 186 ...  
## $ Target : Factor w/ 3518 levels ""," City of Del Rio",..: 2547 2954 1070 2743 1385 1417 1231 969 2080 2528 ...  
## $ Description : Factor w/ 4453 levels ""," Malaysia's Computer Emergency Response Team (MyCERT) reveal the details of a campaign carried out by APT40, ta"| \_\_truncated\_\_,..: 3591 3443 1072 4046 92 1412 1522 1040 3895 1885 ...  
## $ Attack : Factor w/ 327 levels "\"view as\" vulnerability",..: 3 280 220 177 91 177 97 177 9 263 ...  
## $ Target.Class: Factor w/ 25 levels "C Manufacturing",..: 13 2 13 16 13 13 7 16 15 23 ...  
## $ Attack.Class: Factor w/ 13 levels " CC",">1","CC",..: 4 6 3 3 10 4 3 3 3 4 ...  
## $ Country : Factor w/ 158 levels "",">1","AE","AF",..: 47 129 139 139 47 59 16 139 139 2 ...  
## $ Link : Factor w/ 1144 levels "","http://abcnews.go.com/Politics/fbi-probing-attempted-hack-trump-organization-officials/story?id=47652150",..: 268 174 68 714 1042 44 1032 724 815 69 ...  
## $ Tags : Factor w/ 927 levels "","#OpIsrael, #OpUSA, Anonymous",..: 352 831 213 754 381 556 456 261 582 245 ...

A partir de esta carga inicial, observamos que existen campos con mucha información cualitativa, que resultarán irrelevantes para resolver los problemas planteados. En consecuencia, una primera estrategia consistirá en realizar tareas de **reducción de dimensionalidad**, descartando aquellos atributos que no vamos a necesitar:

* **ID**: Identificador único dentro de cada informe.
* **Target**: Nombre de la entidad atacada.
* **Description**: Explicación detallada de cada incidente.
* **Attack**: De manera análoga a **Description**, es un campo con información detallada sobre de ataque realizado.
* **Link**: Enlace la URL a la noticia del incidente reportado.
* **Tags**: Contiene hashtags o etiquetas con palabras clave contenidas en la descripción del ciberataque. Únicamente presente en *Master Table 2017.*

Por otra parte, el campo fecha (**Date**) tiene un nivel de granularidad excesivo, ya que resultará muy difícil encontrar varios ataques producidos en un mismo día. Por ello, el nivel de granularidad que definiremos será el número de ataques reportados en un mes concreto. Para realizar este cambio de granularidadagregar la información contenida en un mismo mes, creando así, dos variables que contendrán los campos *Año – mes.*

En relación al tipo de ataque, *Attack**Class*, al ser una variable categórica, se analizarán los valores contenidos en ella.

# Analizaremos la calidad de los datos de la variable Tipo de Ataque  
  
tipoAtaque <- attacks\_Raw$Attack.Class  
table(tipoAtaque)

## tipoAtaque  
## CC >1 CC CE   
## 1 1 1469 232   
## CW CW? Cyber Crime Cyber Espionage   
## 50 1 1094 172   
## Cyber Warfare H Hacktivism N/A   
## 33 58 38 5   
## Not Found   
## 1314

Sobre la tabla anterior, observamos que:

* Será necesario homogenizar valores. Existen distintas representaciones/etiquetas para un mismo valor, como, por ejemplo: “*Cyber Crime*” - “*CC*” o “*Cyber Warfare*” - “*CW*”.
* Se deberán de tratar los valores ausentes representados a través de las etiquetas ***“****N/A”* y *“Not Found”*.
* Existen etiquetas que contienen el carácter *“?”.*

Para el tipo de entidad, *Target Class*, se realizará el mismo análisis que en *Attack Class***.** De esta forma, el rango de valores contenidos en esta variable es:

# Analizaremos la calidad de los datos de la variable Entidad Atacada  
  
tipoEntidad <- attacks\_Raw$Target.Class  
table(tipoEntidad)

## tipoEntidad  
## C Manufacturing   
## 82   
## D Electricity gas steam and air conditioning supply   
## 46   
## E Water supply, sewerage waste management, and remediation activities   
## 7   
## G Wholesale and retail trade   
## 86   
## H Transportation and storage   
## 38   
## I Accommodation and food service activities   
## 71   
## J Information and communication   
## 201   
## K Financial and insurance activities   
## 193   
## L Real estate activities   
## 4   
## M Professional scientific and technical activities   
## 68   
## N Administrative and support service activities   
## 33   
## Not Found   
## 1314   
## O Public administration and defence, compulsory social security   
## 263   
## O Public administration, defence, compulsory social security   
## 167   
## P Education   
## 201   
## Q Human health and social work activities   
## 281   
## R Arts entertainment and recreation   
## 119   
## S Other service activities   
## 57   
## U Activities of extraterritorial organizations and bodies   
## 25   
## V Fintech   
## 88   
## X Individual   
## 672   
## Y Multiple Industries   
## 352   
## Y Multiple targets   
## 78   
## Y Multiple Targets   
## 9   
## Z Unknown   
## 13

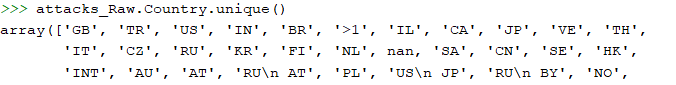
Sobre la tabla anterior, observamos que:

* Cada valor está formado por un Identificador único (Primera letra), seguido de una breve descripción. En este caso, utilizaremos el identificador único para encontrar inconsistencias en los datos y realizar tareas de homogenización.
* Existen
* valores perdidos, representados mediante las etiquetas “*Not Found*” y “*Z-Unknown”*.

En relación al autor, campo *Author*, dado el alto número de posibles valores (564 niveles), discretizaremos esta variable con el objetivo de distinguir, si los ataques son producidos por delincuentes u organizaciones bien identificadas, o por autores anónimos o desconocidos. De esta forma, se creará una clase con los valores ‘*Desconocido’* o ‘*Conocido’* que podría ayudarnos a identificar si estamos más expuestos a ataques de redes conocidas, y por lo tanto, dedicar recursos a analizar sus sistemas y tipos de ataques; o si necesitamos mecanismos de defensa mucho más heterogéneos debido al alto número de atacantes anónimos.

Por último, en relación al país afectado, *Country*, observamos que el número de casos documentados es muy alto (158 niveles). De la misma forma que con el campo *Author***,** se discretizarán los valores de estas variables en función del *Continente*en el que se encuentre cada país.

Por otra parte, existen tipologías de ataques de rango internacional, cuya etiqueta es ‘*>1’*. Del mismo modo, el set de datos contiene observaciones con distintos países separados por saltos de línea. A estos casos, también se les asignará el valor ‘*International’.*



Finalmente, este campo también contiene la presencia de valores perdidosque deberán ser tratados.

## 2.1 Resumen de tratamientos previos

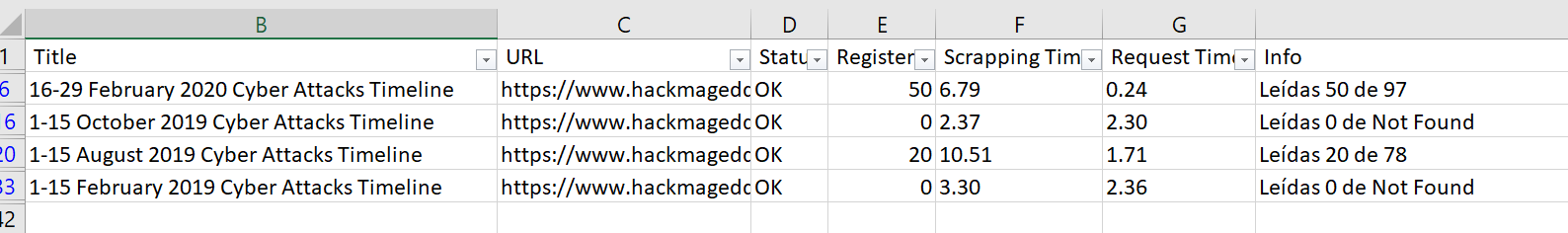
Como indicábamos en la sección introductoria, en el proyecto de limpieza de datos que hemos planteado, puede que uno de los principales retos con nuestro set de datos, sea precisamente todo lo relativo a tareas de **integración, selección, reducción y conversión de datos**, porque partimos de una situación en la que ni siquiera tenemos una información totalmente tabulada y con las variables cuantitativas que necesitaríamos para el estudio.

Adicionalmente, en los procesos de obtención de datos, es posible que se produzcan errores por distintos motivos, o que incluso en el propio informe de la web no se haya registrado el dato. Sea por el motivo que sea, hay que tomar una decisión sobre qué hacer con dichos valores, como se nos indica en [1].

Una de las posibilidades pasaría por completar los datos manualmente, como indica Osborne [3], o corregir los procesos de obtención de datos, contemplando nuevas particularidades en la fase de scraping.

Durante la primera práctica, se implantó un proceso de control de calidad, en el cual se generaba un archivo *csv* con información referente a cómo había ido el proceso de *scraping*. De esta forma, a través de este archivo, se podrían detectar errores durante el proceso de extracción de datos, y relanzar dichos procesos a partir de funcionalidades concretas del scraper, con el objetivo de completar el set de datos.

A continuación, se muestra una captura del archivo *scraping 2020-05-17 08.46.29.csv*, contenido en la carpeta *logging* del repositorio de Github. En esta imagen, se resaltan procesos en los cuales, la extracción de datos no ha sido completa o ha fallado.



Con el objetivo pues, de obtener toda la información de las fuentes de origen, se ha relanzado el *scraper* para las casuísticas concretas reflejadas en el archivo de *logging*, creando un nuevo fichero que será integrado con el resto de archivos *csv*. Esta nueva fuente de información, *error-files.csv,* está disponible en la carpeta *data/00\_raw* del repositorio de Github.

Antes de realizar la integración de las fuentes de información, será necesario aplicar un conjunto de controles, que asegurarán la calidad de datos. Estos controles, consistirán en filtrar los posibles errores contenidos en la fecha de reporte, eliminando aquellos registros, que no pertenezcan al periodo temporal contenido en cada archivo. Adicionalmente, en esta fase, se seleccionarán y adecuarán los nombres de los campos presentes en todas las estructuras de datos.

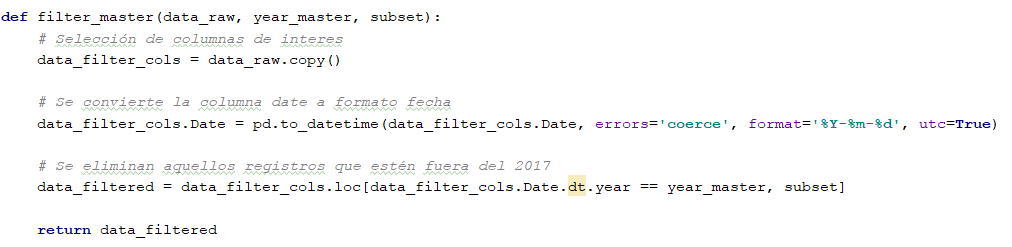


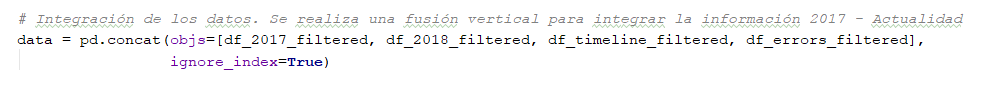
Ilustración 1. Código de limpieza de las Master Tables

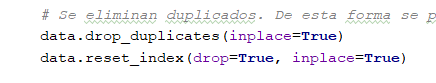
En el caso del Timeline, se añadirá un nivel más de control, eliminando aquellos registros cuya fecha de ataque sea posterior a la fecha de Reporte.



Ilustración 2. Código de limpieza del Timeline.

Una vez filtrados los archivos originales, y seleccionado el conjunto de datos comunes en ellos, el siguiente paso consistirá en integrarlos verticalmente, con el objetivo de crear una única estructura de datos, sobre la cual aplicar las de tareas de limpieza restantes.



El siguiente paso, consistirá en garantizar la unicidad de las observaciones, eliminado duplicidades de datos, generadas al reportar el mismo ataque en distintos reportes.

A continuación, el siguiente paso consistirá en procesar el campo *Attack* Class, con el objetivo de solucionar las casuísticas detectadas durante el análisis previo, realizado en al inicio de esta sección. Para ello, se homogenizarán los datos a través de la siguiente tabla.

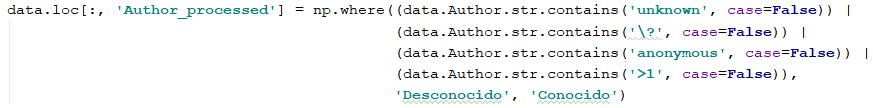
|  |  |
| --- | --- |
| CE | Cyber Espionage |
| CW | Cyber Warfare |
| CC | Cyber Crime |
| H | Hacktivism |
| >1 | Multiple |
| UK | Unknown |

Posteriormente, en el apartado 3.1, se explicará como se han tratado los valores perdidos, categorizados a través de la clase ‘Unknown’.

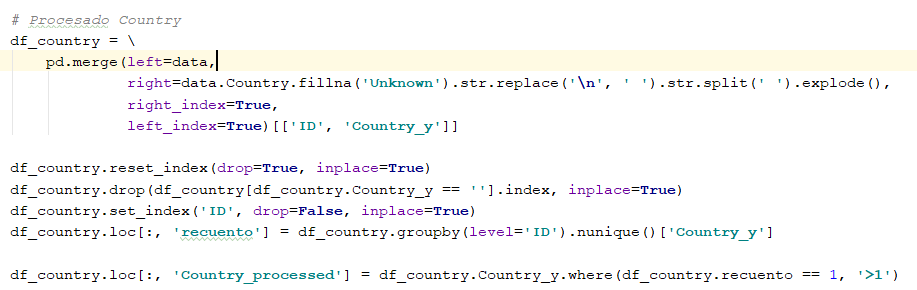
Adicionalmente, a partir de este campo, crearemos una nueva variable dicotómica ProblemasQC, que informará de aquellas observaciones que han sufrido algún tipo de procesado especial, que pueda afectar a los resultados obtenidos en análisis posteriores.

Una vez realizadas las tareas de homogenización del tipo de ataque, el siguiente paso consistirá en procesar el campo Author. En este caso, se realizará una discretización en autores conocidos o desconocidos.

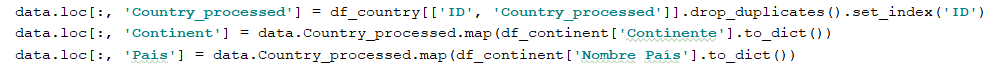
A través del análisis del rango de posibles valores que toma esta variable, para determinar si un autor es desconocido, deberá contener el carácter ‘?’ o tener las etiquetas: ‘unknown’, ‘anonymous’, ‘>1’.



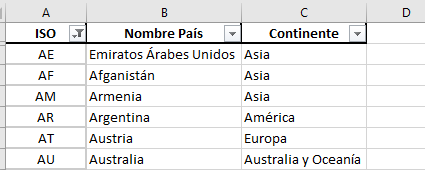
El siguiente paso, consistirá en procesar la información presente en el campo *Country.* Para ello, primero se asignará la etiqueta ‘*>1’* a aquellos valores que presenten diversos países separados por saltos de línea.



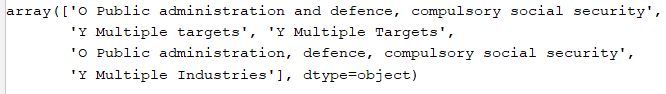
Una vez codificado este valor, se realizará una discretización asignando el continente al cual pertenece cada país, a través del código ISO, contenido en el campo *Country*.



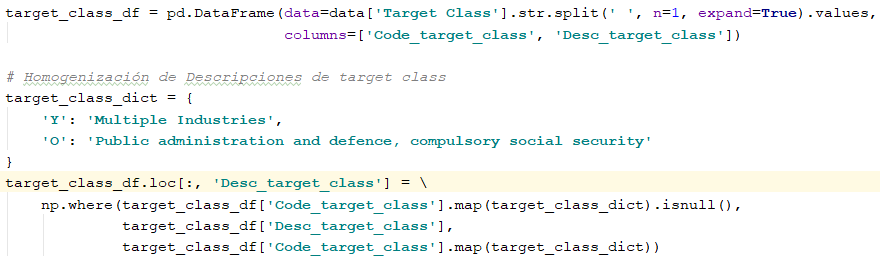
Para realizar esta discretización, se hará uso del documento externo *continente \_country.xlsx,* contenido en la carpeta *data/99\_aditional* del repositorio de Github.

Este *Excel* está compuesto por los siguientes campos:

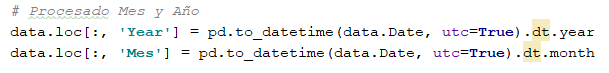
El siguiente campo a procesar será *Target.* En este caso, tal como hemos comentado al inicio de esta sección, nos encontramos ante un problema de inconsistencia de datos.



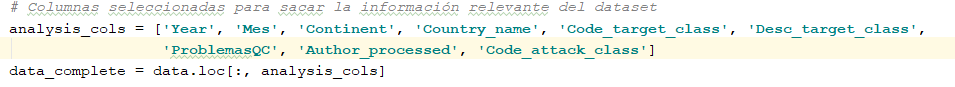
En este caso, existen dos descripciones distintas para las tipologías ‘O’ e ‘Y’. De esta forma, se deberá de realizar una homogenización de los datos.



Finalmente, a partir de la variable *Date,* crearemos dos nuevas variables *Año* y *mes,* que servirán para aumentar la granularidad de los datos y tener observaciones más significativas.



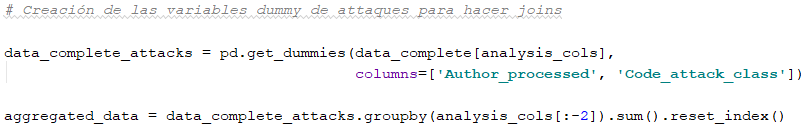
Una vez realizadas todas las tareas de limpieza, homogenización y conversión de datos, se seleccionarán aquellos campos de interés para nuestros análisis.



Con la información seleccionada, el siguiente paso, consistirá en crear un conjunto de variables numéricas que recojan el número de ataques por tipología, en función del periodo de tiempo en el que se originan; el lugar dónde se producen; y el objetivo que afectan, es decir, el tipo de entidad atacada.

De esta forma, se creará una estructura del tipo *crosstab* [2], que contendrá la distribución de los ataques según su tipología. Esta información, nos será de utilidad a la hora de tratar de analizar si hay algún tipo de ataque que afecte en mayor medida a nuestra entidad. Además, podremos estudiar el tipo de relación existente entre distintas tipologías de ataques, en cuestión de tendencias, correlaciones, etc.

Adicionalmente, también será de interés, conocer el número de atacantes conocidos o desconocidos, con el objetivo de poder realizar análisis de tendencias, y averiguar a qué tipología de atacante estamos más expuestos. Por este motivo, se añadirá esta variable cuantitativa, a las ya mencionadas anteriormente.



Como último paso previo a la realización de tareas de análisis, volcaremos la información transformada en el fichero ***EstadisticasAtaques2017\_2020\_Input.csv****,* contenido en la carpeta *Data/01\_Clean* del repositorio de Github.

Tal y como comentamos previamente, se complementa parte del código R que incluye el presente informe, con el lenguaje de programación *Python* a través del código *data\_processing.py*, con el cual, se resuelven todos los problemas que hemos expuesto y analizado anteriormente.

A modo de resumen, a continuación, se expondrán las variables finales que componen el fichero *EstadisticasAtaques2017\_2020\_Input.csv.*

* **Year:** Año en el que se producen los ataques recogidos para cada observación.
* **Mes**. Mes en el que se producen los ataques recogidos para cada observación.
* **Code\_target\_class**. Código del tipo de entidad afectada por el número de ataques recogidos para cada observación. Este campo contiene las letras iniciales de la variable original *Target Class*.
* **Desc\_target\_class**. Contiene el descriptivo del tipo de entidad, afectada por el número de ataques recogidos para cada observación.
* **Country\_name**: nombre del país afectado por el ataque. Este, ha sido obtenido a partir del código ISO de la variable *Country.*
* **Continent:** Continente al que pertenece el país afectado por el ataque.
* **ProblemasQC:** Identificador que señala si la observación concreta ha tenido problemas de calidad identificados en la etapa de generación de la información. Nos indica que los valores cuantificados pueden ser imprecisos, estando clasificados en categorías genéricas, o en entidades no identificadas.
* **Author\_processed\_conocido**: Número de ataques con autor conocido e identificable que se han producido para ese tipo de entidad, en el país, mes y año correspondiente. Variable creada a partir de la variable original *Author*.
* **Author\_processed\_conocido**. Número de ataques con autor desconocido e identificable que se han producido para ese tipo de entidad, en el país, mes y año correspondiente. Variable creada a partir de la variable original *Author*.
* **Code\_attack\_class\_CC**: Número de ataques del tipo “Cyber Crime” que se han producido para ese tipo de entidad, en el país, mes y año correspondiente. Variable creada a partir del recuento de casos, a través de la variable original *Attack* *Class*.
* **Code\_attack\_class\_CE**: Número de ataques del tipo “Cyber Espionage” que se han producido para ese tipo de entidad, en el país, mes y año correspondiente. Variable creada a partir del recuento de casos, a través de la variable original *Attack* *Class.*
* **Code\_attack\_class\_CW**: Número de ataques del tipo “Cyber Warfare” que se han producido para ese tipo de entidad, en el país, mes y año correspondiente. Variable creada a partir del recuento de casos, a través de la variable original *Attack* *Class.*
* **Code\_attack\_class\_H**: Número de ataques del tipo “Hacktivism” que se han producido para ese tipo de entidad, en el país, mes y año correspondiente. Variable creada a partir del recuento de casos, a través de la variable original *Attack* *Class*.
* **Code\_attack\_class\_UK**: Número de ataques de tipo *‘Unknown’* que se han producido para ese tipo de entidad, en el país, mes y año correspondiente. Variable creada a partir del recuento de casos, a través de la variable original *Attack* *Class*.
* **Code\_attack\_class\_>1**. Número de ataques con múltiples tipologías, codificadas a través de valor “>1”, y que se han producido para ese tipo de entidad, en el país, mes y año correspondiente. Variable creada a partir de conteo de casos, a través de la variable original *Attack* *Class*
* **NumeroAtaques:** Número de ataques total producido para un tipo de entidad, país, mes y año concreto. Variable creada a partir del conteo de casos de la variable original *Attack Class.*

## 2.2 Carga del nuevo archivo tras el procesado de datos.

A continuación, cargaremos el nuevo fichero de *input*, y explicaremos brevemente cómo vamos a avanzar en las siguientes secciones con dichos datos.

# Almacenamos el nuevo set de datos en el frame "attacks\_Input" para una validación adicional, y para explicar algunos de los tratamientos que llevaremos a cabo a lo largo del análisis  
attacks\_Input <- read.csv2(file='processed\_data.csv',stringsAsFactors = TRUE)  
attacks\_Input$Year <- as.factor(attacks\_Input$Year)  
attacks\_Input$Mes <- as.factor(attacks\_Input$Mes)  
  
# Creamos una nueva variable con el número total de ataques por observación  
attacks\_Input$NumeroAtaques <- attacks\_Input$Code\_attack\_class\_.1+attacks\_Input$Code\_attack\_class\_CC+attacks\_Input$Code\_attack\_class\_CE+attacks\_Input$Code\_attack\_class\_CW+attacks\_Input$Code\_attack\_class\_H+attacks\_Input$Code\_attack\_class\_UK  
  
# Mostramos la estructura del archivo recién cargado  
str(attacks\_Input)

## 'data.frame': 1748 obs. of 16 variables:  
## $ Year : Factor w/ 4 levels "2017","2018",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Mes : Factor w/ 12 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Continent : Factor w/ 7 levels "África","América",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ Country\_name : Factor w/ 112 levels "Afganistán","Alemania",..: 15 15 18 33 33 33 33 33 33 33 ...  
## $ Code\_target\_class : Factor w/ 21 levels "C","D","E","G",..: 7 21 13 1 6 7 10 11 12 13 ...  
## $ Desc\_target\_class : Factor w/ 21 levels "Accommodation and food service activities",..: 11 19 5 12 1 11 15 3 16 5 ...  
## $ ProblemasQC : logi FALSE FALSE FALSE FALSE FALSE FALSE ...  
## $ Author\_processed\_Conocido : int 1 0 0 0 0 1 0 0 1 0 ...  
## $ Author\_processed\_Desconocido: int 0 1 1 3 3 5 1 1 4 10 ...  
## $ Code\_attack\_class\_.1 : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ Code\_attack\_class\_CC : int 1 1 1 3 3 6 1 1 5 10 ...  
## $ Code\_attack\_class\_CE : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ Code\_attack\_class\_CW : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ Code\_attack\_class\_H : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ Code\_attack\_class\_UK : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ NumeroAtaques : int 1 1 1 3 3 6 1 1 5 10 ...

Podemos hacer una comprobación sobre una de las variables más relevantes para nuestros análisis numéricos: el número de ataques totales que estamos analizando, segmentándolos en las distintas tipologías que nos indica el atributo original *Attack* *Class*.

# Análisis de integridad en la carga: comprobación de la completitud del dataset y la consistencia entre variables numéricas.  
  
colSums(attacks\_Input[7:15])

## ProblemasQC Author\_processed\_Conocido   
## 228 785   
## Author\_processed\_Desconocido Code\_attack\_class\_.1   
## 3899 2   
## Code\_attack\_class\_CC Code\_attack\_class\_CE   
## 2786 468   
## Code\_attack\_class\_CW Code\_attack\_class\_H   
## 104 1206   
## Code\_attack\_class\_UK   
## 118

cat("Suma del número de ataques por categoría: ",sum(colSums(attacks\_Input[8:15])[3:8]),"\n")

## Suma del número de ataques por categoría: 4684

cat("Suma del número de ataques de acuerdo a si se conoce el atacante: ",sum(colSums(attacks\_Input[8:15])[1:2]),"\n")

## Suma del número de ataques de acuerdo a si se conoce el atacante: 4684

cat("Valor acumulado en la variable NumeroAtaques: ",sum(attacks\_Input$NumeroAtaques))

## Valor acumulado en la variable NumeroAtaques: 4684

Con lo anterior, ya podemos considerar que tenemos un primer set de datos, con información consistente, exacta, única y válida; garantizando de esta forma, un nivel de calidad óptimo, que nos permitirá realizar tareas de análisis.

Aun así, cabe remarcar que quedarán pendientes tareas de procesado de datos, como la normalización/estandarización de variables; el análisis de valores extremos, o la verificación de suposiciones de normalidad y homocedasticidad con el objetivo de poder aplicar distintas pruebas estadísticas.

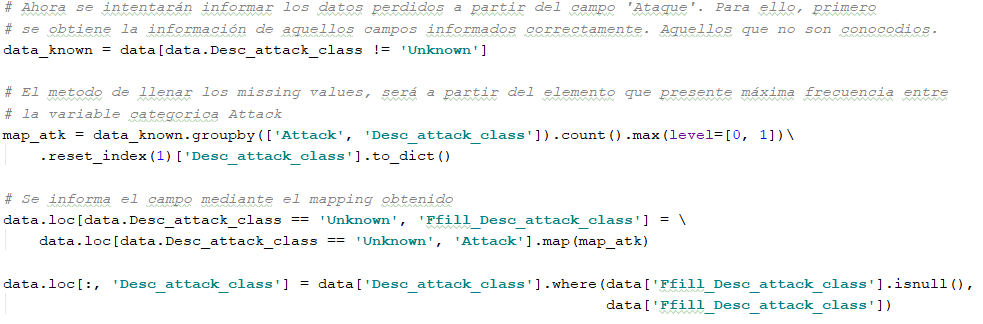
# 3. Limpieza de los datos.

## 3.1 ¿Los datos contienen ceros o elementos vacíos? ¿Cómo gestionarías cada uno de estos casos?

En el dataset utilizado para la realización de esta práctica, nos hemos encontrado ante un problema de pérdida total de la información*.* En este caso, a través del fichero de *logging*,se han detectado un conjunto de registros que, por errores en la etapa de extracción de datos, no se han leído adecuadamente, reduciendo de esta forma el número total de observaciones posibles. En este caso, para solucionar esta casuística, se ha relanzando el proceso de *scraping.*

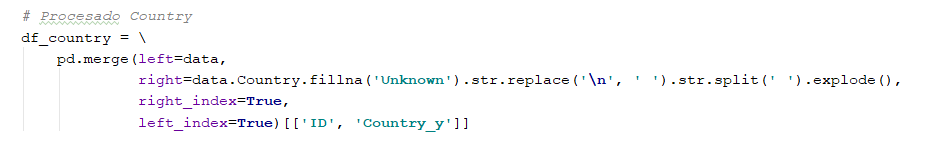
Por otra parte, a lo largo de la etapa de acondicionamiento de datos, se han detectado observaciones que contienen valores perdidos, generando así, un problema de pérdida parcial de la información. En este sentido, para solventar dicha casuística, se han empleado metodologías distintas, en función de la variable afectada.

La primera variable que contiene *missing Values,* es *Attack Class.* Para este campo, se han considerado como valores perdidos, las etiquetas que contenían el carácter ‘*?’* o los valores ‘*unknown’, ‘not found’* y *‘n/a’.* En este caso, debemos diferenciar dos metodologías distintas de tratamiento de valores perdidos.

La primera, intentará solventar los valores representados por la etiqueta *‘Not found’.* Esta, hace referencia a la falta de información producida por un error durante la etapa de obtención de datos. Ante tal situación, una de las opciones propuestas por *Osborne [3]¸* sería la de recoger los datos de forma manual, siempre que suponga una inversión de tiempo aceptable. En concreto, existen un total de 1314 casos *Not Found,* por lo que esta opción ha sido descartada. La alternativa, será realizar la imputación de estos valores perdidos, a través de una medida de tendencia central. De esta forma, se calculará cual es la clase de ataque más representativa, es decir, aquella con una frecuencia de aparición mayor, en función de la variable *Attack.*

La segunda metodología empleada, será la de asignar una constante, tanto a aquellos valores perdidos representados por las etiquetas ‘*unknown’, ‘?’* y *‘n/a’*, como a los valores *‘not found’,* que no se han imputado correctamente.

La siguiente variable que contiene registros vacíos es *Country*. En este caso, se les asignará la constante ‘*desconocido’,* y en función del análisis que realicemos, descartaremos o no la información referente a esta variable*.* De esta forma, podremos aprovechar el resto de campos informados, para realizar análisis que no requieran información del país. Esta técnica de análisis mencionada, es conocida como *pairwise [4].*

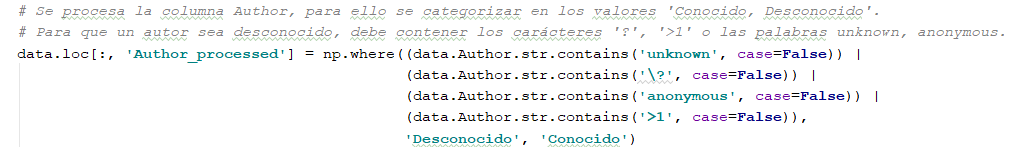


Otra variable en la que encontramos valores perdidos es *target\_class.* En este caso, los valores perdidos se representan a través de las etiquetas *‘Not Found’* y *‘Z Unknown’.* Del mismo modo que con el atributo *Attack Class,* nos encontramos ante una pérdida parcial de la información producida por errores en la etapa de extracción de los datos. En concreto, la volumetría de valores perdidos para este campo (‘*Not Found’*) es de 1382 registros, por lo que se ha decidido asignarles la clase *‘Z Unknown’,* y realizar técnicas de *pairwise* en las taras de análisis y extracción de conocimiento. Adicionalmente, cabe destacar que se ha descartado la posibilidad de eliminar todas observaciones que contienen valores perdidos, debido la gran cantidad de información que se perdería.



Finalmente, la última variable que contiene valores perdidos es *Author.* Tal como se expone en la introducción, uno de los objetivos será analizar si estamos estadísticamente más expuestos, a ataques realizados por autores conocidos o desconocidos. La finalidad, será saber si debemos centrar nuestros recursos en la creación sistemas de defensa versátiles, o focalizados en técnicas de ataque características de cada autor.

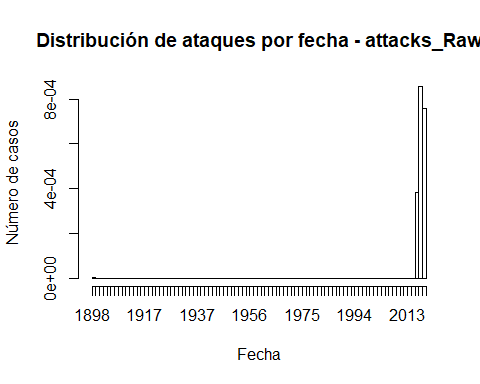
En este sentido, para el campo *Autor,* asignaremos a todos los valores perdidos la variable *‘Desconocido’.* En concreto, se consideran las etiquetas ‘*unknown’, ‘anonymous’, ‘>1’,* y aquellas clases que contienen el símbolo ‘*?’,* como valores perdidos.



## 3.2 Identificación y tratamiento de valores extremos

Otro problema adicional de la fecha es que podría haberse registrado mal en origen. Podemos hacer un par de breves análisis sobre este campo, que nos confirmarán la necesidad de analizar y limpiar dicho campo:

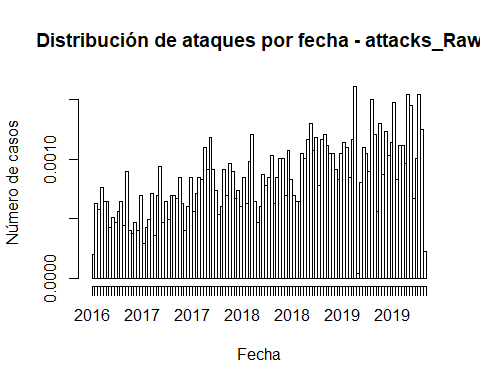
# Extracción y conversión del campo que registra la fecha del incidente  
fechas <- as.Date(attacks\_Raw$Date, format="%d/%m/%Y")  
  
# Análisis gráfico de la distribución de incidentes por fecha  
hist(fechas, breaks=100, main="Distribución de ataques por fecha - attacks\_Raw", xlab = "Fecha", ylab="Número de casos")



# Se observa que necesariamente hay valores erróneos (el típico registro de valor nulo y/o conversión a 01/01/1900). Veámoslo en modo tabla.  
table(format(fechas,"%Y"))

##   
## 1900 2017 2018 2019 2020   
## 4 951 1338 1693 482

# Analicemos de nuevo la distribución de casos por fecha si eliminamos los casos de 1900  
hist(fechas[format(fechas,"%Y")!="1900"], breaks=100, main="Distribución de ataques por fecha - attacks\_Raw", xlab = "Fecha", ylab="Número de casos")



# 4. Limpieza de los datos.

## 4.1 Selección de los grupos de datos que se quieren analizar/comparar (planificación de los análisis a aplicar)

## 4.2 Comprobación de la normalidad y homogeneidad de la varianza

## 4.3 Aplicación de pruebas estadísticas para comparar los grupos de datos

# 5. Representación de los resultados a partir de tablas y gráficas

# 6. Resolución del problema. A partir de los resultados obtenidos, ¿cuáles son las conclusiones? ¿Los resultados permiten responder al problema?

# Contribuciones

|  |  |
| --- | --- |
| Investigación Previa Redacción de las respuestas Desarrollo código | JBP – IRP JBP – IRP JBP – IRP |

# Bibliografía

[1] Subirats, Laia - Pérez, Diego O. - Calvo, Mireia (2019). “Introducción a la limpieza y análisis de los datos”, Universidad Oberta de Catalunya

[2] Bock, Tim (2019). “What is a Crosstab”, Display R Blog. [en línea] [Última consulta: 15/May/2020] <https://www.displayr.com/what-is-a-crosstab/>

[3] Osborne, Jason W. (2013). “Best practices in data cleaning: A complete guide to everything you need to do before and after collecting your data.” Thousand Oaks, CA. Sage Publications.

[4]