Universitat Oberta de Catalunya

Máster en Ciencia de datos

Tipología y ciclo de vida de los datos

PRÁCTICA 1: Web Scraping

Viviendas de lujo en venta de E & V en Barcelona: predicción de precios

Alumnos: Manuel Ernesto Martínez Martín

Vanessa Moreno González

Aula 1

1. Contexto

El proyecto tiene como objetivo recopilar datos sobre el sector inmobiliario de lujo en la provincia de Barcelona, reuniendo información de las características de los inmuebles en venta, como pueden ser la cantidad de habitaciones, los metros cuadrados, la antigüedad o el estado de la propiedad entre otros, que pueden influir en su precio de venta. Estos datos, se emplearán para crear modelos predictivos que permitan estimar el precio de las viviendas en función de sus características. Esta herramienta predictiva, puede servir de ayuda a todos los actores involucrados en el sector inmobiliario. Entre otras cosas, para:

1. Ayudar en la toma de decisiones, ya que los compradores/inversores pueden usar esta herramienta de predicción de precios para identificar viviendas que suponen una buena oportunidad de inversión. Por otro lado, los propietarios de inmobiliarias y los vendedores pueden usarla para decidir el precio de venta óptimo de un inmueble según sus características.
2. Reducir la cantidad de tiempo necesario para determinar el precio adecuado de un inmueble.
3. Los clientes pueden determinar también el precio óptimo de una vivienda, y esto puede servir para negociar el precio de aquellas viviendas que se venden muy por encima del valor predicho.

Se ha seleccionado la web de Engel & Völkers por la gran cantidad de información detallada que ofrece sobre los inmuebles que tiene a la venta. La recopilación de todos estos datos permitirá la predicción de precios de viviendas en venta, mediante técnicas de aprendizaje supervisado, como la regresión lineal múltiple, árboles de regresión o random forest regression. Además, Engel & Völkers está especializada en el sector inmobiliario de lujo, que es el nicho de mercado que queremos predecir.

Para evaluar el rendimiento del algoritmo, al disponer de un conjunto de datos pequeño y para aprovechar todos los datos disponibles para el entrenamiento, se empleará la técnica de evaluación cross validation.

La página web de Engel & Völker es: https://www.engelvoelkers.com/es/

Y la página web en concreto donde realizamos web scraping es:

https://www.engelvoelkers.com/es/search/?q=&startIndex=0&businessArea=residential&sortOrder=DESC&sortField=sortPrice&pageSize=18&facets=bsnssr%3Aresidential%3Bcntry%3Aspain%3Brgn%3Abarcelona%3Btyp%3Abuy%3B

2. Título

El título seleccionado para el dataset es:

Viviendas de lujo en venta de E & V en Barcelona.

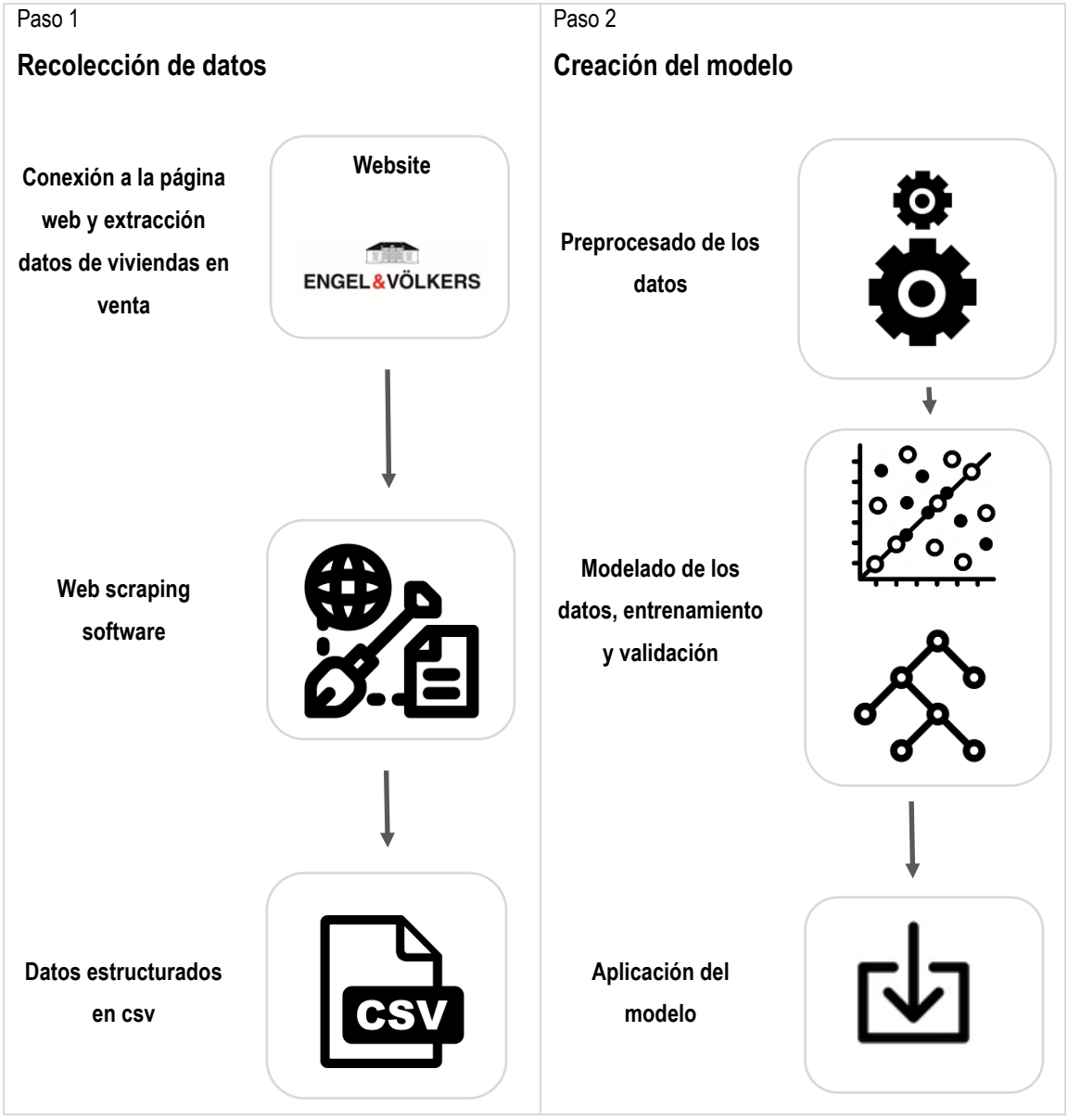
3. Descripción del dataset

El dataset contiene información de 2111 viviendas de lujo, en venta, disponibles en la provincia de Barcelona. La información incluida en este dataset es una recopilación de 25 características de las propiedades, como son el número de habitaciones, el tipo de inmueble, el número de baños, el barrio donde ubica, el año de construcción, el estado del inmueble y la descripción, entre otras, así como el precio de venta.

4. Representación gráfica

Este esquema representa el proyecto elegido. Para representarlo se ha dividido el proyecto en dos pasos:

1. La fase de extracción de datos.
2. La fase de modelización.



5. Contenido

Los campos que encontramos en el dataset "**Viviendas de lujo en venta de E & V en Barcelona"** son los siguientes:

- **id:** campo numérico secuencial, que sirve como identificador único de cada registro del dataset.
- **eav_id:** campo identificador de Engels & Völkers que identifica la propiedad en venta. Cadena alfanumérica compuesta de "W-" seguida de 6 caracteres alfanuméricos.
- **title:** campo tipo texto que incluye una pequeña descripción de la propiedad.
- **subtitle:** campo tipo texto que incluye información que hace referencia a diferentes características de la propiedad como es el tipo de propiedad, (*Casa, Apartamento, Terreno u Obra nueva),* la modalidad en la que se encuentra (*comprar o alquilar)*, el país del inmueble, con el valor único de *España,* la provincia, que en este caso es únicamente *Barcelona* por haber extraído los datos con esta condición, la comarca o el barrio de Barcelona ciudad y el municipio o barrio de Barcelona. En el preprocesado de los datos, este campo se dividirá en cada una de las características del inmueble, creando los campos:
    - **tipe:** campo tipo texto categórico que indica el tipo de propiedad: *Casa, Apartamento, Terreno u Obra nueva.*
    - **modality:** campo tipo texto categórico que indica la modalidad en la que se encuentra el inmueble. Esta puede ser: *comprar o alquilar.*
    - **región:** campo tipo texto que indica la comarca de la provincia de Barcelona o el barrio de Barcelona donde se encuentra el inmueble
    - **municipality:** campo tipo texto que indica el municipio de la provincia de Barcelona o el barrio de Barcelona donde se encuentra el inmueble

  No se creará una variable para los campos país y provincia por ser siempre España y Barcelona respectivamente.

- **n_rooms:** variable cuantitativa discreta que indica el número de estancias, incluyendo salón, cocina… de la propiedad. Incluye valores nulos y valores con el texto “desde” seguido de un valor de estancias.
- **n_bedrooms:** variable cuantitativa discreta que indica el número de habitaciones de la propiedad. Incluye valores nulos y valores con el texto “desde” seguido de un valor de habitaciones.
- **n_bathrooms:** variable cuantitativa discreta que indica el número de baños de la propiedad. Incluye valores nulos y valores con el texto “desde” seguido de un valor de estancias.
- **useful_area:** campo tipo alfanumérico que indica los metros cuadrados de superficie habitable. Esta compuesto por el valor de la superficie seguido del texto m^2. Incluye valores nulos y valores con el texto “desde” seguido de la superficie. En el preprocesado de los datos, este campo se transformará en una variable numérica.
- **built_area:** campo tipo alfanumérico que indica los metros cuadrados de superficie construida. Esta compuesto por el valor de la superficie seguido del texto m^2. Incluye valores nulos. En el preprocesado de los datos, este campo se transformará en una variable numérica.
- **land_area:** campo tipo alfanumérico que indica los metros cuadrados de terreno. Esta compuesto por el valor de la superficie seguido del texto m^2 o ha. Incluye valores nulos. En el preprocesado de los datos, este campo se transformará en una variable numérica.
- **price:** campo tipo alfanumérico que indica el valor de venta del inmueble. Esta compuesto por el valor de venta seguido del texto EUR. Incluye valores nulos, celdas con el texto “a consultar” y celdas con el texto desde seguidas del precio mínimo del inmueble. En el preprocesado de los datos, este campo se transformará en una variable numérica.
- **built_year:** variable cuantitativa discreta que indica que año de construcción del inmueble. Incluye valores nulos.
- **energy_class:** variable tipo texto de naturaleza categórica que indica la eficiencia energética del inmueble. Esta variable toma las siguientes categorías

en orden de más eficiencia a menos: *APLUSPLUS, A+, A, , B, C, D, E, F y G.* Incluye valores nulos.

- **energy_consumption:** variable tipo alfanumérica que indica el consumo energético del inmueble. Esta compuesta por el valor del consumo energético seguido del texto "kWh/m^2*a" que indica las unidades a las que hace referencia el valor. Incluye valores nulos. En el preprocesado de los datos, este campo se transformará en una variable numérica.
- **co2_emissions:** variable tipo alfanumérica que indica las emisiones de CO2 del inmueble. Esta compuesta por el valor de las emisiones seguido del texto "kg/m^2" que indica las unidades a las que hace referencia el valor. Incluye valores nulos. En el preprocesado de los datos, este campo se transformará en una variable numérica.
- **co2_emissions_scale:** variable categórica tipo texto que indica la escala en la que se sitúan las emisiones de CO2 del inmueble. Los valores que toma esta variable, ordenada de menos emisiones a más son los siguientes: *A, B, C, D, E, F y G.* Incluye valores nulos.
- **protected:** variable categórica tipo texto binaria que indica si la propiedad dispone de restricciones en cuanto a su acceso y uso. Toma los valores de VERDADERO y FALSO.
- **status:** variable categórica tipo texto que indica en qué estado de conservación se encuentra la propiedad. Esta toma los valores de*: Bien, Excelente, Muy bien, Necesita renovaciones, Necesita restauración, Otros, Parcialmente renovado, Regular y Renovado*. Incluye valores nulos.
- **parking:** variable numérica que indica el número de plazas de aparcamiento disponibles por los inmuebles.
- **garaje:** variable numérica que indica el número de plazas de aparcamiento disponibles por los inmuebles que disponen de garaje propio.
- **floor_cover:** variable categórica tipo texto que indica el material de revestimiento de los suelos. Incluye valores nulos.

- **property_subclass:** variable categórica tipo texto que indica, dentro del tipo de propiedad, la subclase de esta. Por ejemplo, encontramos las categorías: *Villa, Casa unifamiliar, Apartamento, Loft/Ático…* Incluye valores nulos.
- **terrace_area:** campo tipo alfanumérico que indica los metros cuadrados de superficie construida. Esta compuesto por el valor de la superficie seguido del texto m^2. Incluye valores nulos. En el preprocesado de los datos, este campo se transformará en una variable numérica.
- **heating_type:** variable categórica tipo texto que indica el sistema de calefacción del inmueble. Por ejemplo, encontramos las categorías: *Gas, Corriente Eléctrica, Suelo radiante, Gas Solar…* Incluye valores nulos.
- **location_status:** variable categórica tipo texto que indica el estatus de la localización del inmueble. Los valores que toma son: *Excelente, Muy bien, Bien, Otros y Regular.* Incluye valores nulos.

El período de tiempo al que pertenecen los datos será desde la fecha de publicación de la vivienda en el portal, hasta la fecha de extracción de los datos. (aproximadamente Abril).

6. Propietario

Los datos de la página web https://www.engelvoelkers.com/es-es/barcelona/ son propiedad de la empresa ENGEL & VÖLKERS GMBH. Todos los datos de la web, incluida la información de los usuarios son propiedad de esta empresa, y están protegidos por la ley de protección de datos.

Los pasos que hemos seguido para actuar de acuerdo a los principios éticos y legales en el contexto del proyecto elegido son los siguientes:

1. Hemos verificado si la página web de Engels & Völkers no permite explícitamente el web scraping. Para ello hemos consultado si existe el archivo robots.txt. En ausencia de este archivo hemos verificado en los

apartados de “Aviso legal” o “Política de privacidad” si se establece algo al respecto. No hemos encontrado nada que haga referencia al web scraping.
2. Hemos verificado que los datos recopilados en nuestro proyecto no estén protegidos por el RGPD. En este caso, al ser datos que no identifican a una persona física se pueden recopilar.
3. Hemos seleccionado datos que no están protegidos ni por contraseña ni por autenticación.
4. Mencionamos en el proyecto al propietario de los datos y la página web desde donde hemos obtenido nuestro dataset.

El hecho de que la adquisición o alquiler de una vivienda, sea una necesidad humana, lo ha convertido en un negocio que mueve grandes cantidades de dinero y en una de las principales fuentes de especulación. Por esta razón, existen muchos trabajos previos a éste donde, a partir de los datos de viviendas de una zona determinada, se realizan análisis predictivos del precio de venta de una propiedad con unas características concretas, empleando técnicas basadas en machine learning. Estos análisis permiten determinar qué factores, entre los estudiados, afectan al precio de las viviendas en esa zona.

Como ejemplo, a continuación, citamos algunos de estos análisis anteriores a este proyecto:

1. Subhradeep. (2019). House Price Predict: Decision Tree, Random Forest. Kaggle. Recuperado el 9 de abril de 2023, de https://www.kaggle.com/code/subhradeep88/house-price-predict-decision-tree-random-forest

En este análisis se aplica regresión múltiple, árboles de regresión y random forest regresión para predecir el valor venta de las viviendas en King Country, de un dataset que se encuentra también en la plataforma llamado “Houses Sales in King County, USA”. Se comparan los resultados obtenidos de estos tres algoritmos.

Otro análisis sobre predicción de precios de venta de viviendas es:

2. Grajales álzate, Y. V. (2019). Modelo de predicción de precios de viviendas en el municipio de Rionegro para apoyar la toma de decisiones de compra y venta de propiedad raíz. Trabajo de final de carrera, Universidad Pontificia Bolivariana, Medellin, Colombia, 2019. Recuperado el 8 de abril de 2023: https://repository.upb.edu.co/bitstream/handle/20.500.11912/5285/Modelo%20predicci%C3%B3n%20precios%20viviendas.pdf?sequence=1&isAllowed=y

La autora aplica varios modelos de machine learning para predecir la relación entre las viviendas y el precio de venta.

Al ser un caso de uso típico de estudio, también podemos encontrar papers donde se analizan las mejores técnicas a aplicar para predecir el precio de las viviendas o donde se analiza el estado del arte acerca de este tema:

1. Madhuri, C.R., Anuradha, G., & Pujitha, M.V. (2019). House Price Prediction Using Regression Techniques: A Comparative Study. En IEEE 6th International Conference on Smart Structures and Systems (ICSSS 2019). VR Siddhartha Engineering College, Vijayawada, India. Recuperado de: https://ieeexplore.ieee.org/abstract/document/8882834

2. Mohd, T., Jamil, N.S., Johari, N., Abdullah, L., Masrom, S. (2020). An Overview of Real Estate Modelling Techniques for House Price Prediction. In: Kaur, N., Ahmad, M. (eds) Charting a Sustainable Future of ASEAN in Business and Social Sciences. Springer, Singapore. https://doi.org/10.1007/978-981-15-3859-9_28

7. Inspiración

Este conjunto de datos puede ser interesante, ya que se trata de un conjunto muy segmentado de viviendas, al incluir los precios y características de las viviendas

catalogadas como “lujosas” en la provincia de Barcelona. Por ello, puede ayudar a entender que características de este tipo de viviendas influyen más en su precio de venta, que suele ser mas alto que el resto de viviendas.

Realizar un análisis predictivo de precios de este conjunto de datos nos permitirá responder las siguientes preguntas:

1. ¿Qué características son las que más influyen en el precio de venta de un inmueble de lujo?
2. ¿Qué propiedades se encuentran por encima o por debajo del precio predicho por el algoritmo? ¿Son buenas oportunidades de inversión?
3. Sabiendo que, en propiedades de calidad media, baja, el tamaño es una de las características que más influye en el precio, en las viviendas de lujo, ¿Cómo está relacionada la superficie de la vivienda con su precio?
4. ¿Influye el municipio o la región de Barcelona en el precio de las propiedades de lujo?

Como hemos visto en el punto anterior, en el estudio “Modelo de predicción de precios de viviendas en el municipio de Rionegro para apoyar la toma de decisiones de compra y venta de propiedad raíz” el análisis predictivo de precios de inmuebles es una herramienta que puede ayudar en la toma de decisiones de compra y venta de inmuebles.

Para decidir el algoritmo adecuado, probaremos regresión múltiple, árboles de regresión y random forest regresión, que son tres técnicas que se emplean normalmente en este caso de uso, como hemos visto en los estudios anteriormente citados. Como solución, escogeremos el algoritmo que mejor estimación de precio proporcione.

8. Licencia

Se ha seleccionado la licencia GNU General Public License v3.0.

La razón principal para elegir esta licencia es que se trata de una licencia de software libre y código abierto y permite que los usuarios de este, puedan libremente usar, estudiar y compartir este dataset. Al tratarse de una licencia tipo copyleft, cualquier distribución que se haga de los datos deberá llevar esta misma licencia.

9. Código

El código implementado para la obtención del dataset se ha realizado en Python, y a continuación comentaremos los aspectos más relevantes de como se realiza el proceso de recolección de datos.

Se ha empleado la biblioteca de Python Selenium, aunque se ha elegido el web driver de la biblioteca undetected_chromedriver en vez del que tiene por defecto. Esto permitirá evitar que la página web nos detecte como bot.

```
# Usamos 'undetected_chromedriver' en vez del webdriver de selenium por defecto
# from selenium import webdriver
import undetected_chromedriver as uc
from selenium.webdriver.common.by import By
from selenium.common.exceptions import NoSuchElementException
```

Se ha definido una clase EngelAndVolkersScraper para encapsular los métodos que permiten extraer la información de la página de Engel & Völkers:

- **_get_page(url):** carga las páginas de navegación y a las que deseamos aplicar scraping y te esperamos un tiempo aleatorio entre 1 y 3 segundos para evitar que nos detecte como bot.

```
"""
def _get_page(self, url):
    time.sleep(randint(1,3))
    self.driver.get(url)
```

- **_get_item_links():** Utiliza XPath para obtener los enlaces a las viviendas en la página actual. Get_attribute() permite obtener el enlace como texto.

```
def _get_item_links(self):
    return [link.get_attribute('href') for link in self.driver.find_elements(By.XPATH, "//a[@class='ev-property-container']")]
```

- **_get_next_page():** obtiene el enlace a la siguiente página de resultados (si existe) y lo devuelve como una cadena. Si no hay ninguna página más, se retorna None.

```
def _get_next_page(self):
    try:
        return driver.find_element(By.CLASS_NAME, 'ev-pager-next').get_attribute('href')
    except NoSuchElementException:
        return None
```

- **_generate_string(item, columna_name, head=False):** obtiene el valor de un elemento texto sea ‘value\nkey’ (datos en cabecera con salto de línea entre valor y key) o ‘key value’ (datos en detalles).

```
def _generate_string(self, item, column_name, head=False):
    if (head):
        return item[0:item.index(column_name)]
    else:
        return item[item.index(column_name)+len(column_name)+1:]
```

- **_built_house():** inicio del proceso de scraping de la vivienda extrayendo los atributos necesarios para generar el csv (adjuntamos una parte del código).

```
def _build_house(self):
    house = {
        "eav_id": None,
        "title": None,
        "subtitle": None,
        "n_rooms": None,
        "n_bedrooms": None,
        "n_bathrooms": None,
        "useful_area": None,
        "built_area": None,
```

- **_built_house():** crea un json, y si falla una página o un link, posteriormente te crea un csv que te servirá para saber que viviendas no se han podido extraer.

```
Crea una entrada de error como JSON
@Params index : <int> (número de página donde ha fallado)
@Param2 level : <String> (es página o sub-página)
@Param3 link : <String> (enlace fallido)
@Return: dict
"""
def _build_error(self, index, level, link, exceptionType):
    return {"page_index": index+1, "level": level, "link": link, "exceptionType": exceptionType}
```

- **_built_house():** es necesario aceptar o declinar las cookies para poder extraer los detalles de las viviendas.

```
Declina el uso de cookies de la página web
Utiliza XPATH para localizar el componente de botón que declina
@Return: void
"""
def _decline_cookies(self):
    try:
        disagreeProcessingDataBtn = self.driver.find_element(By.XPATH, "//button[@id='didomi-notice-disagree-button']")
        disagreeProcessingDataBtn.click()
    except NoSuchElementException:
        pass
```

- **_get_data(self):** navega por la página, extrae los links de las viviendas y hace el scraping. (se adjunta un trozo del código)

```
Navega por cada página de la busqueda de viviendas, en cada página se extraen los links de las viviendas
En la primera iteración se ha de aceptar/declinar las cookies
@Return: void
"""
def get_data(self):
    i = 0
    nextPage = self.mainUrl
    errors, houses = [], []
    while nextPage != None:
        try:
            self._get_page(nextPage)
        except Exception as e:
            errors.append(self._build_error(i, "page", nextPage, type(e)))
            print("-> ERROR on page {pagelink} with exception {exceptionType}".format(pagelink = nextPage, exceptionType = type(e)))
            break
```

- **main:** puerta de entrada al programa de scraping.

```
Programa principal
Se indica la URL de EngelAndVolkers con el municipio
Se carga el webdriver, se imprime el User-Agent y se inicia el wescraping
"""
if __name__ == '__main__':
    target_url = "https://www.engelvoelkers.com/es/search/?q=&startIndex=0&businessArea=residential&sortOrder=DESC&sortField=sortPrice&pageSize=1
    driver = uc.Chrome(use_subprocess=True)
    userAgent = driver.execute_script("return navigator.userAgent")
    print("User-Agent:",userAgent)
    eav = EngelAndVolkersScraper(target_url, driver)
    eav.get_data()
```

Uno de los problemas que presenta la web es que los datos están segmentados, teniendo etiquetas clave-valor. Para solucionarlo, se han creado dos listas de claves-valores para la cabecera de la casa y para los detalles, y mediante contains, se detecta la clave y se hace un substring porque al tener un formato diferente, pero tener el mismo XPath dificulta la extracción puesto que hay elementos que se solapan.

Otra de las dificultades encontradas ha sido que es necesario aceptar o declinar las cookies para poder extraer correctamente los datos. Si no se realiza este paso, los datos de los detalles se extraen como None.

10. Dataset

El enlace al dataset en Zenodo es:

https://zenodo.org/record/7827625#.ZDkAvS9j6_I

DOI: 10.5281/zenodo.7827625

11. Video

El enlace al Video del proyecto es:

https://cv.uoc.edu/app/blogaula222/222_m2_851_01_448590/2023/04/15/practica-1-web-scraping-2/

12. Contribuciones

Contribuciones	Firma
Investigación previa	M.E.M.M, V.M.G

Redacción de las respuestas	M.E.M.M, V.M.G
Desarrollo del código	M.E.M.M, V.M.G
Participación en el vídeo	M.E.M.M, V.M.G