Descripción de la práctica

1. Contexto

En un contexto donde existe una gran demanda de vivienda en alquiler, estamos interesados en conocer la evolución de su renta. Para ello vamos a proceder a la recogida diaria de los datos relativos a determinados parámetros de interés que aparecen en los anuncios publicados en el portal inmobiliario *fotocasa*[1]. Establecemos como área piloto del estudio la ciudad de Barcelona, pudiendo ser elegida cualquier otra ciudad, región o país que pueda ser objeto de análisis. La extracción de los datos la vamos a realizar con técnicas de web scraping con la finalidad de construir un *dataset*, éste podrá ser utilizado para la posterior creación de un índice de precios coyuntural que tendrá una periodicidad de 3 meses. Este índice permitirá tener un conocimiento de cómo es la evolución de los precios de alquiler para una ciudad, región o país determinado (en nuestro caso la ciudad de Barcelona). El resto de datos del *dataset* se podrán utilizar para crear un perfil de vivienda tipo del área elegida.

[1] www.fotocasa.es

2. Título

HouseScraper

3. Descripción del dataset

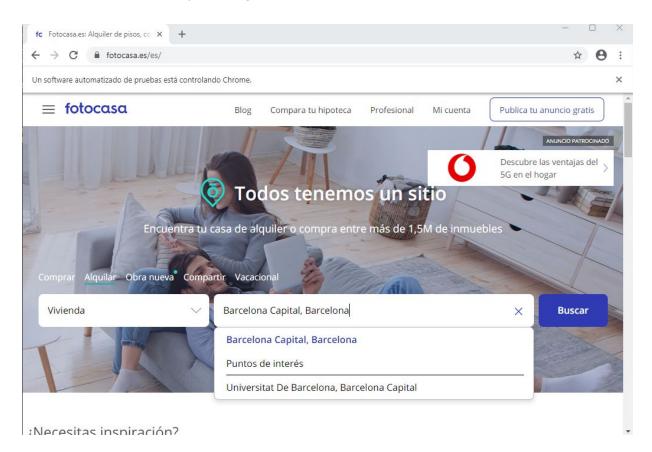
El dataset está formado por los datos capturados del portal fotocasa. Éstos se capturan haciendo peticiones GET a la url de la búsqueda de alquiler de viviendas, obteniendo el código HTML y recorriendolo para buscar los datos de los parámetros de interés. Para extraer el código utilizamos la aplicación SELENIUM, con la que imitamos el comportamiento humano recorriendo la página hasta el final para que se cargue todo el código y poder extraer todos los datos. En el siguiente apartado mostramos las distintas pantallas de donde extraemos el código.

4. Representación gráfica

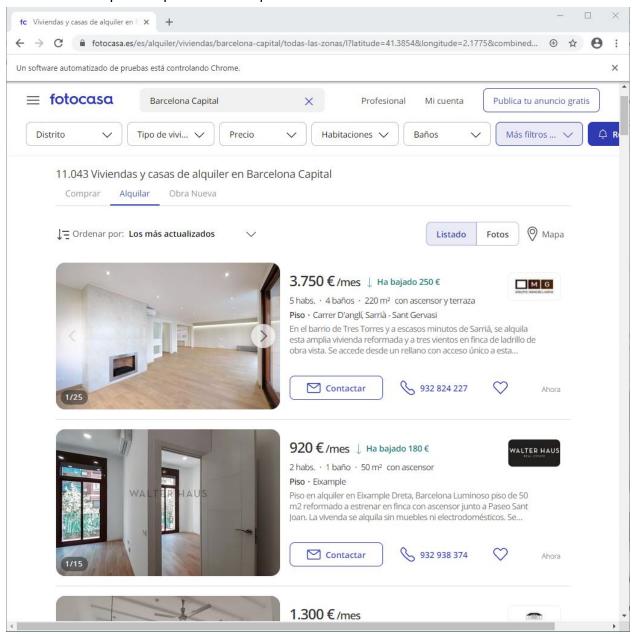
La imagen siguiente corresponde a la pantalla inicial, donde nos aparece el primer obstáculo que es la pantalla emergente de las cookies, tendremos que decirle a SELENIUM que haga clic en "Aceptar y cerrar".



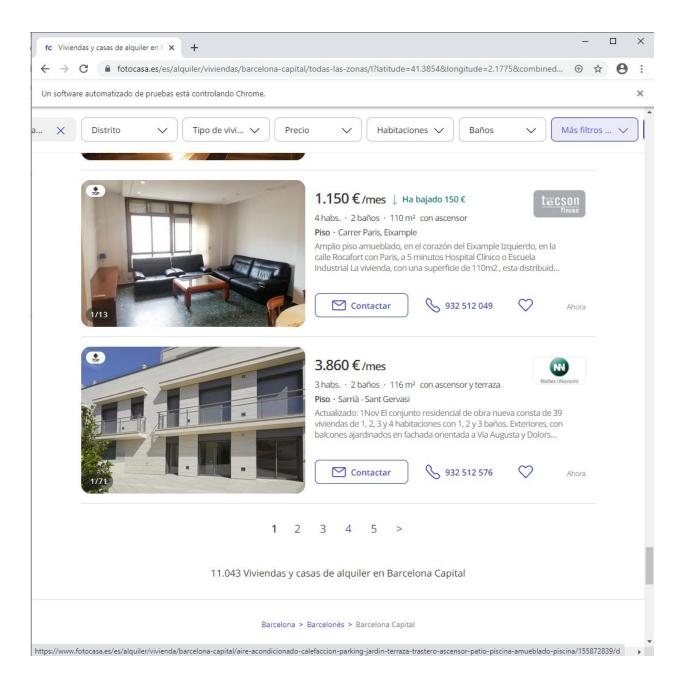
Ahora le decimos a SELENIUM que seleccione "Alquilar", que introduzca la zona a buscar, en nuestro caso "Barcelona" y que haga clic en "Buscar".



Y obtenemos la primera pantalla de la que se obtendrán los datos.



Utilizando SELENIUM ordenamos por los más recientes y hacemos scrolling hasta el final de la página para que se cargue todo el código HTML. Una vez llegado al final de la página, pinchamos en ">" y avanzamos a la página siguiente reiterando el proceso hasta el número de páginas indicado en "constant.py".



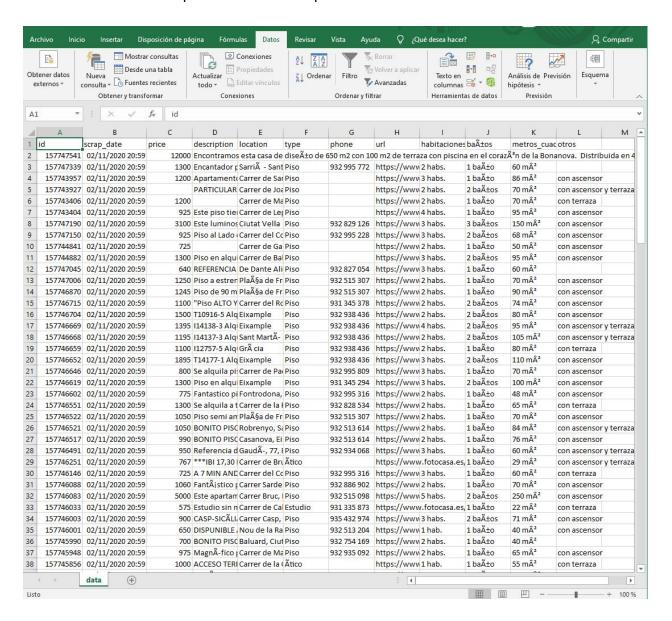
5. Contenido del dataset

En las imágenes anteriores se muestra la forma que tienen los anuncios de los cuales vamos a extraer los datos. Como ejemplo, para el último anuncio de la imagen anterior obtenemos:

- id del anuncio = 157747541
- scrap_date = 02/11/2020 20:59
- price = 3.860
- description = El conjunto residencial de obra nueva consta de...
- location = Sarriá Sant Gervasi

- type = Piso
- phone = 932512576
- url=https://www.fotocasa.es/es/alquiler/vivienda/barcelona-capital/aire-acondicionado-cal efaccion-parking-jardin-terraza-trastero-ascensor-patio-piscina-amueblado-piscina/1558 72839/d
- habitaciones = 3
- baños = 2
- metros_cuadrados = 116
- otros = con ascensor y terraza

Pasamos a mostrar el aspecto del dataset separado en columnas.



Presentamos de una forma más formal la descripción de los campos de dataset:

Nombre	Tipo	Tamaño	Descripción
Identificador	Integer		Identificador del anuncio
Fecha de consulta	Date		Fecha en la que se ha obtenido la información
Precio	Integer		Precio de la vivienda
Dirección	Text	50	Dirección de la vivienda anunciada
Tipo	Text	10	Tipo de vivienda (Piso, Casa, Chalet,)
Télefono	Text	9	Teléfono del anunciante
Url	Text	100	Enlace del anuncio
Habitaciones	Integer		Número de habitaciones
Baños	Integer		Número de baños
Superficie	Real		Metros cuadrados
Otros	String		Otras características como, si tiene terraza, garaje, etc.
Descripción	Text	250	Descripción del anuncio

Los datos se guardan con la fecha en la que se obtuvieron ya que van variando con el tiempo.

6. Agradecimientos

Los datos originales pertenecen a *Fotocasa*, portal inmobiliario líder de España fundado en 1999. Sin la existencia de la actual web no sería posible este proyecto.

En cuanto a estudios similares, el propio portal de *Fotocasa* realiza sus propios estudios e informes, vease; LÓPEZ GARCÍA, ANAÏS. *La vivienda en alquiler en 2019* [en línea]. [Fecha de consulta: 9 de octubre de 2020]. Disponible en: https://prensa.fotocasa.es/informes/.

Estos informes han servido de inspiración para realizar este proyecto.

7. Inspiración

En el portal inmobiliario fotocasa existe una sección donde se calcula un índice de precios de alquiler. Este índice solo nos aporta información del precio por metro cuadrado, sin aportar más información sobre las características de las viviendas que pertenecen a una zona determinada. Por otro lado, las zonas suelen coincidir con distritos censales, por lo que no son zonas homogéneas y un precio medio puede ser engañoso.

Con nuestro dataset, se puede mejorar mucho esta información, ya que podemos crear nuestras propias zonas a partir de la localización de las viviendas. Este conjunto de datos es interesante porque, como ya anunciamos en el apartado 1, nos permite tener un conocimiento detallado de cómo es la evolución de la renta de alquiler residencial en un área, ciudad o país determinado, creando índices y/o predicciones de esta evolución.

Por la naturaleza de los datos recogidos, podemos establecer zonas o áreas tensionadas dentro de las grandes ciudades, ayudando de este modo, a la regulación del mercado residencial de alquiler.

También este dataset nos puede servir para obtener un perfil de la vivienda tipo en un área determinada, así como crear modelos para predecir la renta de ésta.

8. Licencia

Debido a que los datos de origen se obtienen de una página con derechos de autor (licencia Copyright) y a que no está pensada para la descarga automática de sus datos, este dataset no puede tener una licencia que permita cualquier clase de modificación de los datos como las Creative Commons.

La licencia elegida sería **Unknown License** puesto que el objetivo es que un usuario a nivel particular pueda usar los datos para tener un índice de precios y ver la evolución, pero sin usar esos datos para lucrarse ya sea directamente o tratándolos para obtener otros.

9. Otras observaciones

La página de la que se obtiene la información original no dispone de API que permita consultarlos de forma sencilla y leerlos a través de un XML o JSON, así que es necesario el uso de web scraping a las páginas HTML.

Además, tampoco permite el uso de peticiones GET a través de medios externos al navegador como la librería *requests* de Python. El resultado que se obtiene variando el user-agent y otras cabeceras es un contenido que requiere de verificación humana a través de captchas, concretamente de seleccionar una serie de imágenes que estén relacionados con lo indicado.

Aunque existen algunas librerías gratuitas para tratar los captchas, sólo funcionan para versiones simples en las que hay que introducir las letras que se muestran en una imagen. Las versiones más complejas como en este caso requerirían un fuerte trabajo adicional que van más allá de la asignatura y la práctica.

Por ese motivo el medio elegido para capturar el contenido es Selenium.

Otra observación que se quiere hacer llegar al lector de este documento es acerca del uso de comentarios en el código.

Se han seguido las buenas prácticas de forma que todo el código está dividido en funciones de pocas líneas usando tanto en éstas como en las variables nombres que describen lo que se está haciendo, de forma que el código se puede entender simplemente leyéndolo.

Bajo estas prácticas, los comentarios se deberían reducir al mínimo ya que probablemente si hay que explicar algo es que no tiene el nombre correcto o ciertas partes del código se pueden partir más.

No obstante se han hecho aclaraciones adicionales para evitar cualquier tipo de duda.

10. Funcionamiento general

El funcionamiento del programa es el siguiente:

- 1) Se detectan los parámetros de entrada, se inicializan los valores que sean necesarios y se pasan los datos al manejador de Selenium.
- 2) El manejador de Selenium se inicializa y con los parámetros de entrada empieza el proceso de web scraper.
- 3) Se abre una conexión con la web de fotocasa, se selecciona alquiler, se elige la ciudad y se realiza la búsqueda.
- 4) Se cambia el criterio de orden, para cada página devuelta por la búsqueda:
 - a) Se hace scroll de toda la página hasta llegar hacia abajo para que se cargue toda la información.
 - b) Se hace scroll hasta la parte del paginador.
 - c) Se captura el código HTML que hay hasta ese momento y se le pasa la información a BeautifulSoup.
 - d) Para cada anuncio que no sea de publicidad se extrae la información y se realizan transformaciones para eliminar retornos de carro, convertir monedas a números, convertir fechas, etc. guardándose el resultado en una lista de python.
 - e) Se vuelca la información guardada a un fichero csv, añadiéndose al final del mismo.
 - f) Se cambia a la siguiente página.
- 5) El número máximo de páginas a recorrer es indicado a través de un parámetro de entrada.

11. Buenas prácticas

Se han seguido las siguientes buenas prácticas:

- Comprobación de existencia de API: No existe ninguna API gratuita que permita el acceso a la información.
- Parseo de HTML: Se ha usado BeautifulSoup para el parseo del HTML
- Peticiones al servidor web: De ninguna manera se satura al servidor web de peticiones ya que el uso de Selenium simula la interacción de un humano. Además, después de cada acción se ha añadido un tiempo de espera.
- Modificación del user agent: El user agent es aleatorio en cada ejecución.
- Funcionamiento del web scraper: Desde el comienzo de la práctica se ha ido verificando y modificando el código en función de los cambios que se han ido haciendo en la web origen. Se garantiza el correcto funcionamiento en el momento de la última revisión de este documento.
- Calidez y robustez de los datos obtenidos: Se han estudiado los datos obtenidos y se han realizado transformaciones teniendo en cuenta su calidad.

12. Contribuciones

Contribuciones	Firma
Investigación previa	JMGR, JMA
Redacción de las respuestas	JMGR, JMA
Desarrollo del código	JMGR, JMA

13. Dataset Zenodo

El enlace donde se encuentra el dataset de Zenodo es el siguiente: https://doi.org/10.5281/zenodo.4242841

El DOI es: 10.5281/zenodo.4242840