



TÉCNICAS DE WEB SCRAPING

JOSÉ MANUEL GARCÍA RODES

TÉCNICAS DE WEB SCRAPING

SUMARIO

- 1.- ¿A qué nos referimos cuando hablamos de trabajar con Big Data?
- 2.- ¿Cuáles son las ventajas de trabajar con lenguajes como Python o R en vez de usar programas como el SPSS?
- 3.- ¿Qué es el web scraping?
- 4.- Ejemplos de uso
 - 4.1.- Descarga de informaciones de los medios de comunicación
 - 4.2.- Descarga de actas parlamentarias de los BOCG
 - 4.3.- Descarga de twits
- 5.- Uso del web scraping en los institutos de estadística de las Comunidades Autónomas
- 6.- Repositorio GitHub

TÉCNICAS DE WEB SCRAPING

I.- ¿A QUÉ NOS REFERIMOS CUANDO HABLAMOS DE TRABAJAR CON BIG DATA?

DEFINICIÓN

- Cuando hablamos de Big Data nos referimos a conjuntos de datos o combinaciones de conjuntos de datos cuyo tamaño (**volumen**), complejidad (**variabilidad**) y velocidad de crecimiento (**velocidad**) dificultan su captura, gestión, procesamiento o análisis mediante tecnologías y herramientas convencionales, tales como bases de datos relacionales y estadísticas convencionales o paquetes de visualización, dentro del tiempo necesario para que sean útiles.

TÉCNICAS DE WEB SCRAPING

I.- ¿A QUÉ NOS REFERIMOS CUANDO HABLAMOS DE TRABAJAR CON BIG DATA?

- Hoy en día disponemos de la capacidad de almacenar datos que derivan de las actuaciones y de las actividades de la gente en su quehacer ordinario. Antes para poder determinar pautas de conducta y de actuación de las personas se acudía; o bien, a encuestas o a elementos de carácter estadístico que no siempre estaban disponibles, ahora, tenemos **nuevas fuentes de datos** de las que se nutre el Big Data para aplicarlo a las Ciencias Sociales son entre otras; sensores móviles; datos de ubicación y comportamiento; feeds de Twitter; mapas satelitales; minería de textos.
- Para poder utilizar estas fuentes lo principal es tener acceso a ellas. A día de hoy existen una gran cantidad de **repositorios de datos públicos**, cabe destacar los aparecidos con la digitalización de las administraciones públicas, por poner algún ejemplo, podemos encontrar la iniciativa de datos abiertos del Gobierno de España que contiene más de 50.000 conjuntos de datos abiertos.
- El problema viene cuando los **datos no** están **accesibles directamente** y es ahí donde entran técnicas como el Web Scapring, que explicaremos con más detalle más adelante, para poder acceder a ellos.

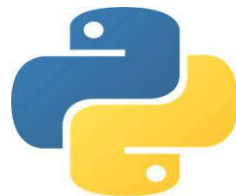
TÉCNICAS DE WEB SCRAPING

2.- ¿CUÁLES SON LAS VENTAJAS DE TRABAJAR CON LENGUAJES COMO PYTHON O R EN VEZ DE USAR PROGRAMAS COMO EL SPSS?

- Tanto **R** como **Python** son unos lenguajes de programación de código abierto y libre. Actualmente estos lenguajes han tenido un enorme incremento en personas y/o empresas que lo usan. Estos lenguajes se vienen aplicando a diversas áreas como economía, finanzas, ciencias sociales, ingeniería, biología, machine learning, física, etc .
- Por otro lado, *Statistical Package for the Social Sciences* o **SPSS**, es un software con licencia que tiene gran uso en las ciencias sociales. Posee una interfaz amigable, la cual mediante botones puedes aplicar métodos estadísticos a tus datos, en contrapartida de **R** y **Python** que se manejan mediante línea de comandos.
- En los últimos años estos lenguajes han madurado mucho, actualmente existen unos IDE's llamados **RStudio** para **R** y **Spayder** para **Python** que los hacen más fáciles.
- También podemos utilizar estos dos lenguajes con los conocidos cuadernos **Jupyter**, una forma sencilla de crear nuestros scrips y de poder ejecutar el código paso a paso comentando los resultados.
- Para descargar todo el entorno completo se recomienda instalarse **Anaconda Navigator** en la edición individual.



Studio®



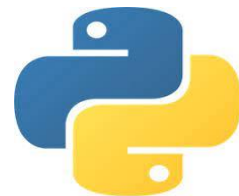
TÉCNICAS DE WEB SCRAPING

2.- ¿CUÁLES SON LAS VENTAJAS DE TRABAJAR CON LENGUAJES COMO PYTHON O R ENVEZ DE USAR PROGRAMAS COMO EL SPSS?

- **R** y **Python** son softwares muchísimo más amplios que **SPSS**, pues poseen una gran variedad de funcionalidades para el tratamiento y el análisis de los datos, pero para todo esto necesitas aprender un poco de programación en R y Python que aunque al principio parece muy complicado, son dos lenguajes con una curva de aprendizaje muy rápida.
- La biblioteca estándar de Python es muy amplia, y ofrece una gran cantidad de recursos. Además de la biblioteca estándar, existe una colección creciente de varios miles de componentes (abarcando módulos o programas individuales, paquetes o *frameworks* completos de desarrollo de aplicaciones), disponibles en el Python Package Index.
- Actualmente, el repositorio oficial CRAN de R recoge cerca de 10.000 paquetes publicados y, además existen muchos más publicados en Internet.



Studio®



TÉCNICAS DE WEB SCRAPING

3.- ¿QUÉ ES EL WEB SCRAPING?

- Una de las aplicaciones tanto de **R** como de **Python** es el raspado de páginas web, en inglés “**web scraping**” o “**scrapear**”. Estas técnicas son un proceso dentro de la Ciencia de Datos que se utiliza para la extracción de datos de sitios web, normalmente en el lenguaje de programación Python y mediante machine learning, simulando cómo navegaría un ser humano en determinadas páginas web.
- El objetivo puede ser transformar contenidos, almacenar datos de la web, reconocer estructuras de código HTML único, recopilar información, hacer minería y análisis de datos, automatizar la creación de enlaces, price mapping, caza de tendencias, monitorización de la competencia, optimización de precios... Son muchos sus usos, y en general son beneficiosos para cualquier proyecto digital.



TÉCNICAS DE WEB SCRAPING

3.- ¿QUÉ ES EL WEB SCRAPING?

- Para el web scraping con R tenemos la librería rvest ayuda a extraer (o recolectar) datos de páginas web.
- Para el web scraping con Python tenemos la librería Requests junto con Beautifulsoup, entre otras. La primera extrae el código de la página web y la segunda crea un árbol con todos los elementos del documento y puede ser utilizado para extraer información.
- Tanto para R como para Python existe la librería Selenium que se encarga de imitar el comportamiento humano cuando estamos navegando por una web y va accediendo a las distintas partes que nos interesa extraer la información.



TÉCNICAS DE WEB SCRAPING

3.- ¿QUÉ ES EL WEB SCRAPING?

Directrices de la política de raspado web de Eurostat

Prácticas (pautas de implementación)

- Respete el protocolo de exclusión de robots.txt y solo siga los enlaces en la medida necesaria para mantener la calidad de las estadísticas;
- Respetar los deseos de los propietarios de sitios web según lo establecido en los términos y condiciones en la medida de lo posible para verificar esos términos y el raspado no es esencial para mantener la calidad de las estadísticas como lo implica la ley estadística;
- Identificarse en la cadena de usuario-agente y proporcionar canales de contacto. Esto podría incluir un enlace a una página web que explique el propósito del raspador y qué datos recopila, los detalles de contacto del equipo responsable e información sobre cómo optar por no participar y solicitar que se eliminen los datos extraídos;
- Siga las convenciones estándar de Internet para el scraping, como los estándares establecidos por el consorcio W3C;
- Sea transparente acerca de sus actividades de raspado web, posiblemente proporcionando información en el sitio web asociado;

TÉCNICAS DE WEB SCRAPING

3.- ¿QUÉ ES EL WEB SCRAPING?

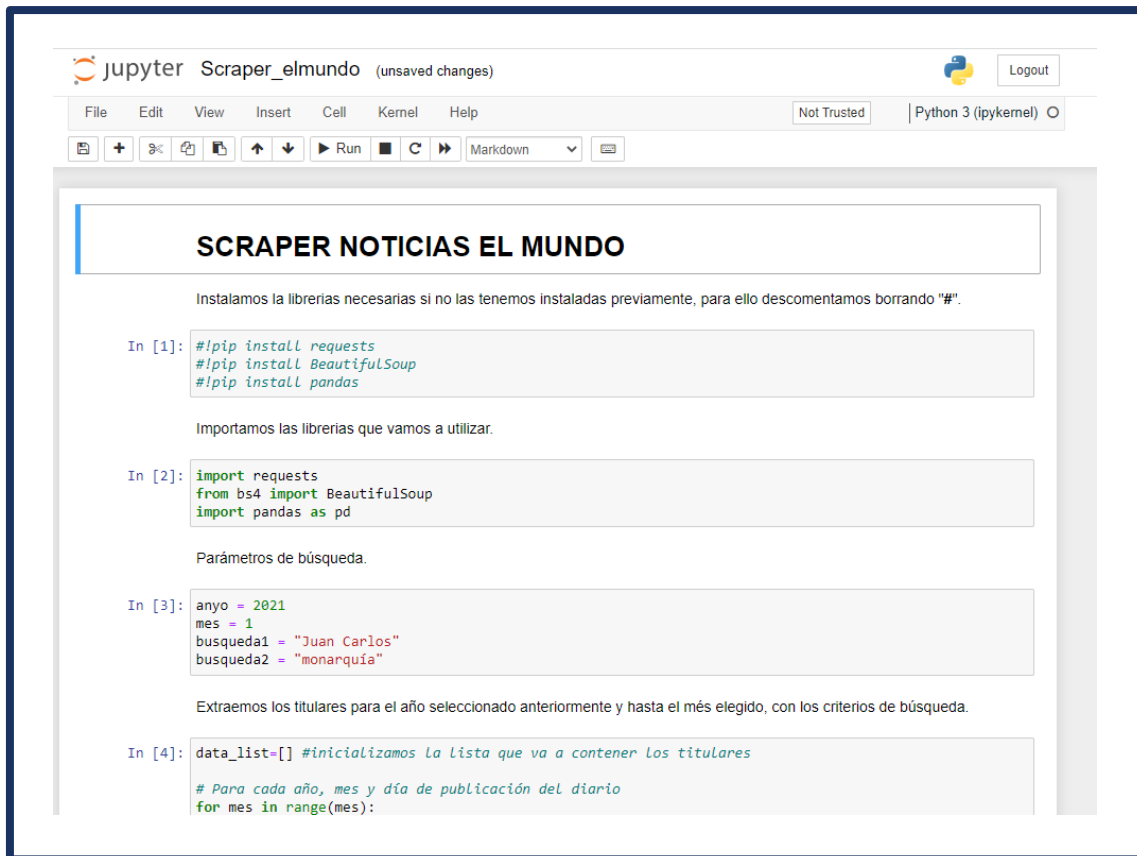
Directrices de la política de raspado web de Eurostat

Prácticas (pautas de implementación)

- Informe a los propietarios de sitios web si se extrae una cantidad considerable de datos de forma regular. Este no sería el caso si un sitio web se raspa con una frecuencia baja y no se raspa en profundidad;
- Trate de minimizar la carga en los servidores web mediante:
 - agregar tiempo de inactividad entre solicitudes,
 - raspado en un momento del día durante el cual no se espera que el servidor web esté bajo una gran carga,
 - optimizar la estrategia de raspado para minimizar el número de solicitudes a un dominio;
- Solo extraiga los datos dentro del alcance del mandato legal de la oficina de estadística y no reutilice ni distribuya los datos sin procesar;
- Maneje los datos extraídos de la web de forma segura; Evite el web scraping al usar API (Interfaz de programación de aplicaciones) públicas u otras opciones de provisión de datos cuando estén disponibles.

TÉCNICAS DE WEB SCRAPING

4.- EJEMPLOS DE USO



The screenshot shows a Jupyter Notebook interface with the title 'Scraper_elmundo' and a status of '(unsaved changes)'. The notebook contains four code cells with the following content:

```
SCRAPER NOTICIAS EL MUNDO
```

Instalamos la librerías necesarias si no las tenemos instaladas previamente, para ello descomentamos borrando "#".

```
In [1]: #!pip install requests
#!pip install BeautifulSoup
#!pip install pandas
```

Importamos las librerías que vamos a utilizar.

```
In [2]: import requests
from bs4 import BeautifulSoup
import pandas as pd
```

Parámetros de búsqueda.

```
In [3]: anyo = 2021
mes = 1
busqueda1 = "Juan Carlos"
busqueda2 = "monarquía"
```

Extraemos los titulares para el año seleccionado anteriormente y hasta el mes elegido, con los criterios de búsqueda.

```
In [4]: data_list=[] #inicializamos La lista que va a contener los titulares

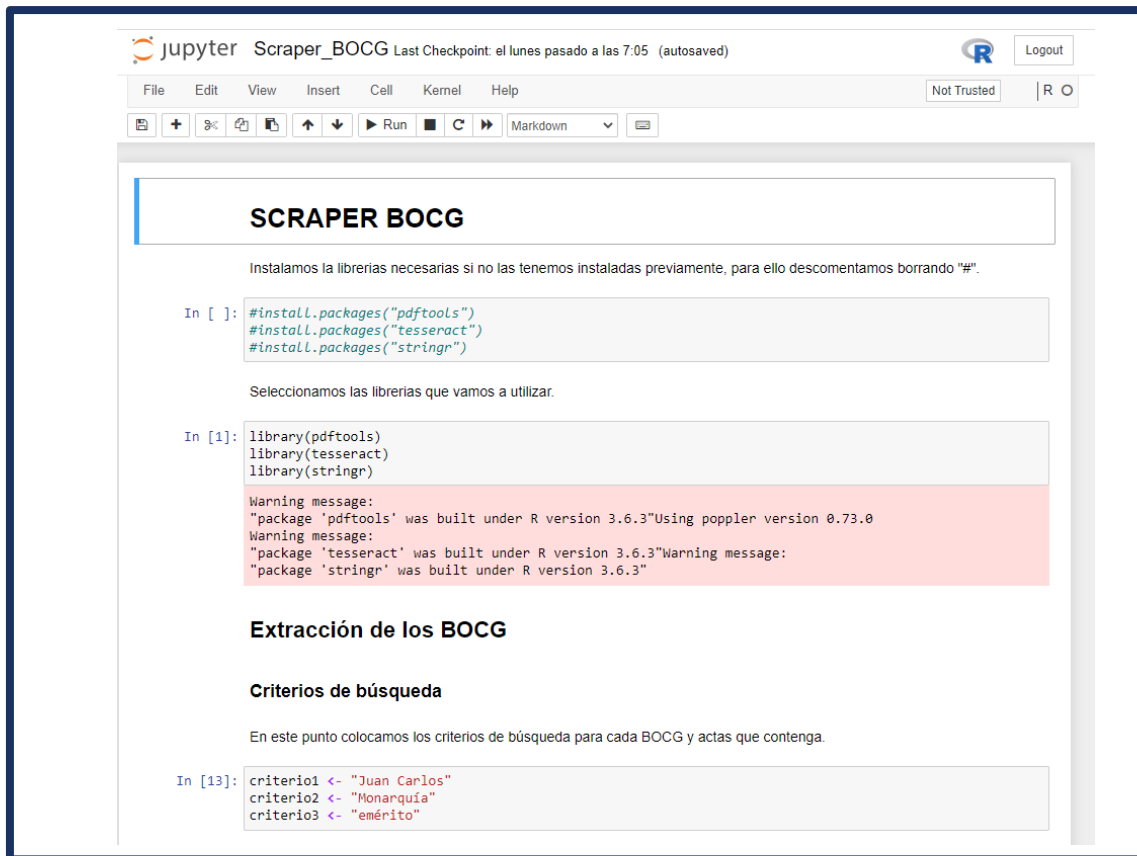
# Para cada año, mes y día de publicación del diario
for mes in range(mes):
```

4.1.- Descarga de informaciones de los medios de comunicación

- En esta diapositiva podemos ver un cuaderno **Jupyter** en el que hemos creado el código para descargarnos los titulares y los enlaces a las noticias aparecidas en el diario **El Mundo** en un periodo de tiempo determinado.
- Para más información sobre los cuadernos Jupyter pueden visitar la página en el enlace anterior.
- En el siguiente enlace podemos ver en detalle una imagen del cuaderno y como quedaría el código.

TÉCNICAS DE WEB SCRAPING

4.- EJEMPLOS DE USO

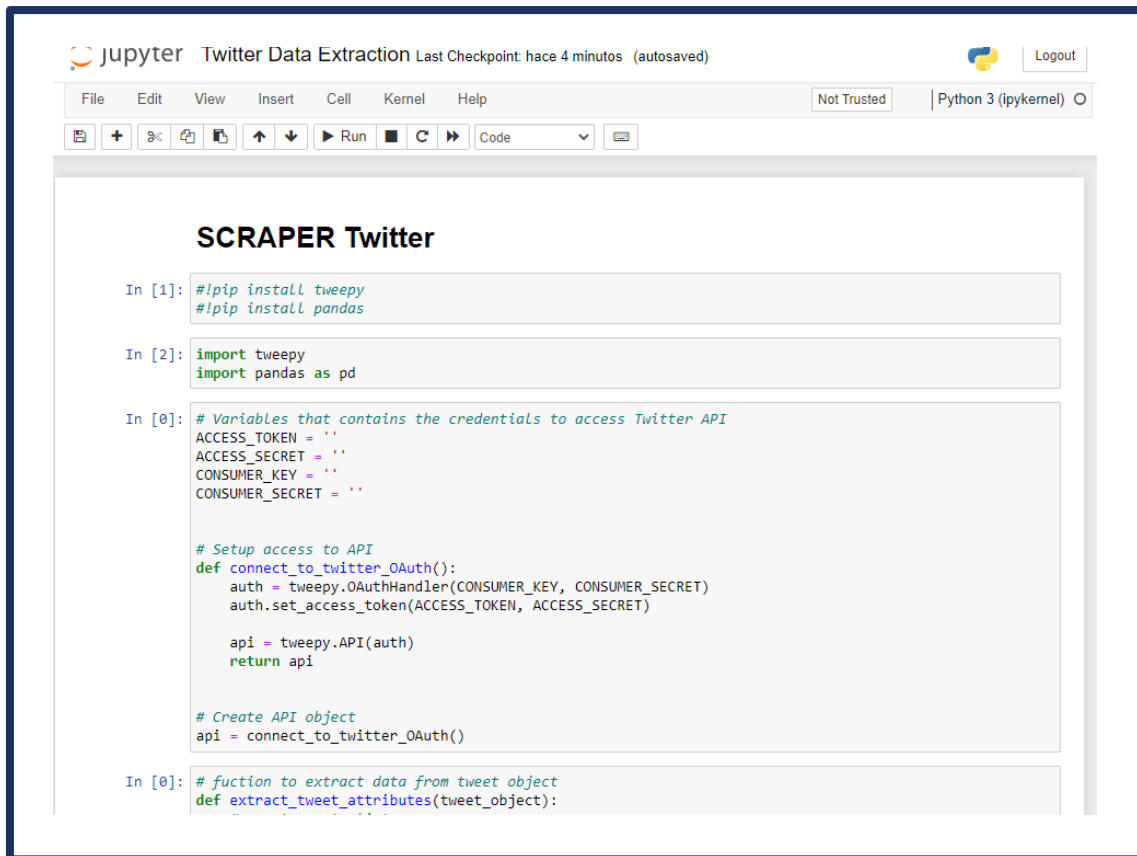


4.2.- Descarga de actas parlamentarias de los BOCG

- En esta diapositiva podemos ver un cuaderno **Jupyter** en el que hemos creado el código para descargarnos las actas parlamentarias contenidas en los BOCG's para los criterios de búsqueda y periodos concretos.
- Para más información sobre los cuadernos [Jupyter](#) pueden visitar la página en el enlace anterior.
- En el siguiente enlace podemos ver en detalle una [imagen del cuaderno](#) y como quedaría el código.

TÉCNICAS DE WEB SCRAPING

4.- EJEMPLOS DE USO



The screenshot shows a Jupyter Notebook interface with the title 'Twitter Data Extraction'. The top bar indicates 'Last Checkpoint: hace 4 minutos (autosaved)' and a 'Logout' button. The menu bar includes 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', and 'Help'. The toolbar shows icons for file operations, a 'Run' button, and a 'Code' dropdown. The notebook content is titled 'SCRAPER Twitter' and contains the following code:

```
In [1]: #!/pip install tweepy
#!/pip install pandas
```

```
In [2]: import tweepy
import pandas as pd
```

```
In [0]: # Variables that contains the credentials to access Twitter API
ACCESS_TOKEN = ''
ACCESS_SECRET = ''
CONSUMER_KEY = ''
CONSUMER_SECRET = ''

# Setup access to API
def connect_to_twitter_OAuth():
    auth = tweepy.OAuthHandler(CONSUMER_KEY, CONSUMER_SECRET)
    auth.set_access_token(ACCESS_TOKEN, ACCESS_SECRET)

    api = tweepy.API(auth)
    return api

# Create API object
api = connect_to_twitter_OAuth()
```

```
In [0]: # fuction to extract data from tweet object
def extract_tweet_attributes(tweet_object):
```

4.3.- Descarga de twits

- En esta diapositiva podemos ver un cuaderno **Jupyter** en el que hemos creado el código para descargarnos twits directamente con la API Twitter para los criterios de búsqueda concretos.
- Para más información sobre los cuadernos Jupyter pueden visitar la página en el enlace anterior.
- En el siguiente enlace podemos ver en detalle una imagen del cuaderno y como quedaría el código.

TÉCNICAS DE WEB SCRAPING

5.- USO DEL WEB SCRAPING EN LOS INSTITUTOS DE ESTADÍSTICA DE LAS COMUNIDADES AUTÓNOMAS



Islas Canarias
Del 15 al 19 de noviembre de 2021

Organización ▾ Programa Ponencias y pósteres ▾ Conferenciantes

Programa de las XXI Jornadas de Estadística de las Comunidades Autónomas



La pasada semana del 15 al 19 de noviembre, tuvieron lugar las XXI Jornadas de Estadística de las Comunidades Autónomas, donde los distintos institutos de estadística de las diferentes comunidades autónomas que participaron, expusieron sus avances en la recogida y análisis de nuevas fuentes de datos y los indicadores utilizados para medir las ODS.

Entre las ponencias más destacadas respecto al tema que nos ocupa cabe destacar las siguientes:

- Análisis de redes orientado a la generación de información estadística para el reto demográfico. Causas y propuestas a partir del estudio de la población susceptible de éxodo rural y su acceso a los equipamientos básicos
- El caso de uso de R aplicado al web scraping en estadísticas de inserción laboral
- Tratamiento de grandes volúmenes de datos con tidyverse. Ejemplos a partir de MCVL

TÉCNICAS DE WEB SCRAPING

5.- USO DEL WEB SCRAPING EN LOS INSTITUTOS DE ESTADÍSTICA DE LAS COMUNIDADES AUTÓNOMAS



Islas Canarias
Del 15 al 19 de noviembre de 2021

Organización ▾ Programa Ponencias y pósteres ▾ Conferenciantes

Programa de las XXI Jornadas de Estadística de las Comunidades Autónomas



- Librerías R y Python para acceder a la API del ISTAC
- Informes automatizados con R. Ejemplos de uso en el seguimiento de la pandemia
- R y Shiny para la difusión de datos e indicadores
- Comunicación interactiva de información municipal usando Markdown y Shiny
- Caracterización de la población andaluza mediante Python y QGIS
- Web scraping para características de empresas
- Técnicas de web scraping aplicadas a las estadísticas de inserción laboral
- Sistema de georreferenciación para fines estadísticos

TÉCNICAS DE WEB SCRAPING

6.- REPOSITORIO GITHUB



6.- Repositorio GitHub

- El código utilizado para el Web Scraping y esta presentación se puede descargar en el [repositorio](#) creado en GitHub



GRACIAS

[JMG.RODES@GMAIL.COM](mailto:jmg.rodas@gmail.com)