PRAC 1: Tipologia i cicle de vida de les dades: Web Scraping  
Autor: Ivan Puig

Data: 22/10/2018

Descripció de la pràctica:

L’objectiu d’aquesta activitat serà la creació d’un dataset a partir de les dades contingudes al web. Heu d’indicar les següents característiques del dataset general:

1. Títol del dataset. Cal que poseu un títol que sigui descriptiu.

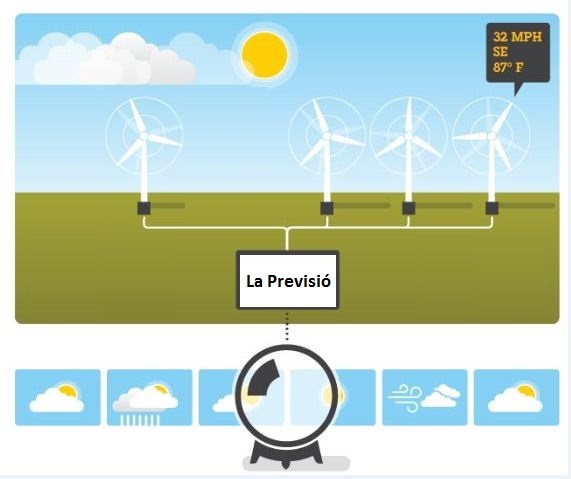
Recollida d’informació de la previsió meteorològica dels propers dies als parc eòlics repartits per Estats Units.

2. Subtítol del dataset. Agregueu una descripció àgil del vostre conjunt de dades pel vostre subtítol.

En el nostre data set volem recollir informació sobre les previsions meteorològiques dels propers dies en els diferents parcs eòlics dels estats units. Amb aquestes dades es vol disposar d’una informació que pugui ser consultada, compartida amb d’altres aplicacions de tercers o ser emmagatzemades per a futures previsions basades en models predictius.

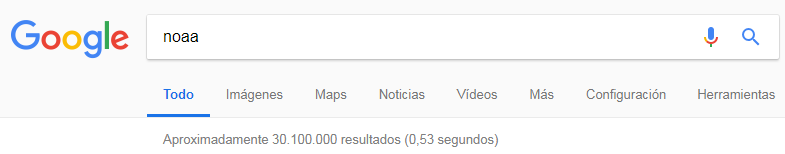
*\*****Nota de l’autor****: Actualment treballo en una companyia americana que gestiona i monitoritza parcs eòlics als Estats Units. De forma periòdica s’accedeix a les previsions meteorològiques del NOAA per planificar intervencions o tasques de manteniment en els parcs. Aquesta consulta es fa completament manual seleccionant les coordenades del parc sobre un mapa i obtenint la previsió de forma visual (icones). L’objectiu de la pràctica es recollir tota la informació (previsions per als propers dies) de tots els parcs de forma automàtica i emmagatzemar-la en format CSV per al seu posterior ús.*

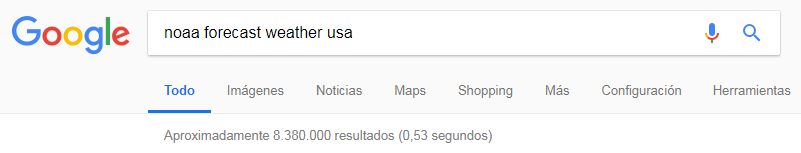
3. Imatge. Agregueu una imatge que identifiqui el vostre dataset visualment.



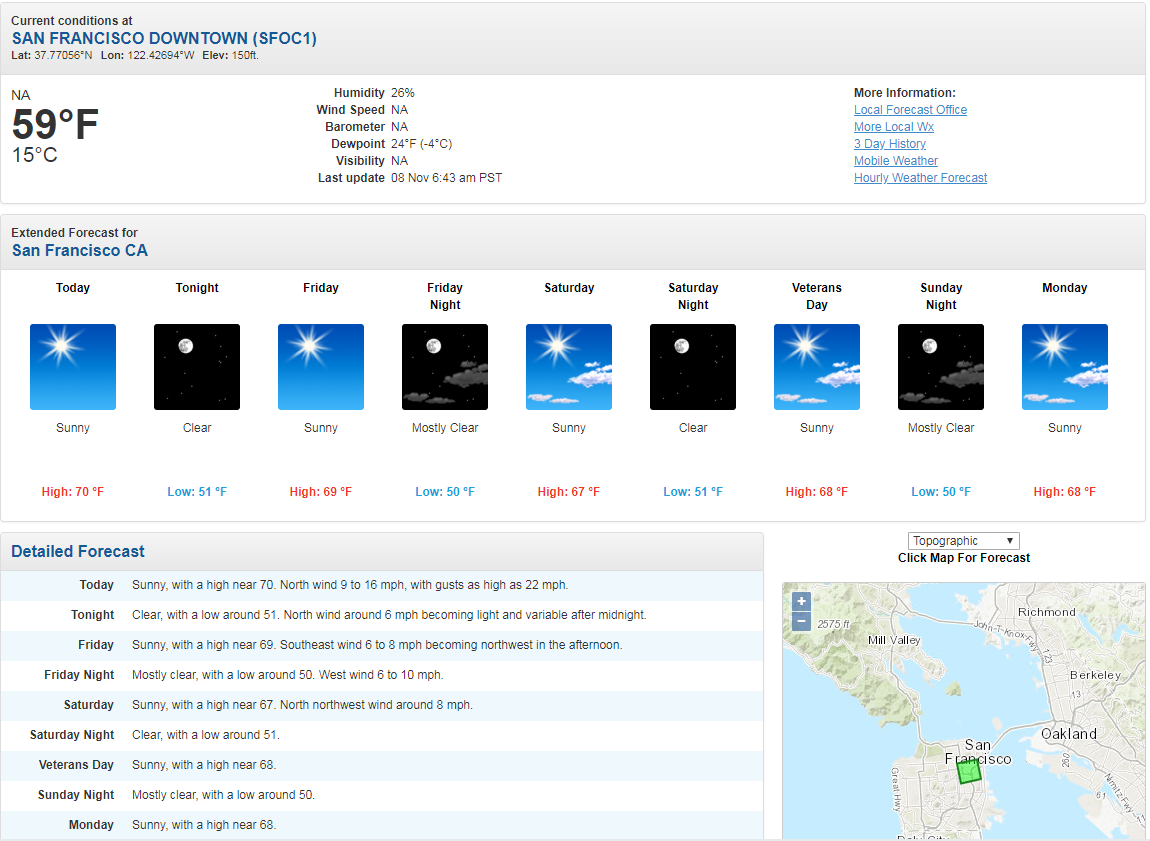
4. Context. Quina és la matèria del conjunt de dades?

Servei meteorològic dels Estats Units, adjunt a l’institut Nacional Oceanogràfic i Atmosfèric (NOAA). Aparentment recull més de 30 milions d’entrades a la web, però nosaltres accedirem en concret al servei nacional de meteorològic, concretament a les previsions meteorològiques. (http://forecast.weather.gov).





Aquestes previsions meteorològiques recullen informació del pronòstic del temps (basant-nos en coordenades de geo-localització) per als propers dies. Dades que incorporen petites descripcions, temperatura i d’altres paràmetres. En el cas concret de la pràctica no recollirem totes les dades per no generar grans volums d’informació.



5. Contingut. Quins camps inclou? Quin és el període de temps de les dades i com s'ha recollit?

Els camps recollits són:

* Dia de la previsió (periode)
* Descripció curta de la previsió metereorologica (descripcio\_curta)
* Sensació tèrmica i temperatura (temp)
* Descripció detallada de la previsió (descripcio)

També s’ha afegit un camp identificador per a fer més senzilla la seva gestió en el fitxer CSV.

Les dades són recollides setmanalment (previsió setmanal) i emmagatzemades en fitxers CSV. Aquests fitxers són identificats per localització del parc eòlic (nom) i data (data de la descàrrega), per exemple “localització\_Phoenix\_08-11-2018.csv”.

Per a obtenir la localització dels parc eòlics i per automatitzar al màxim el procés, les coordenades seran proporcionades per un fitxer XML que llegirem en el moment de fer la descàrrega. D’aquesta manera en una sola acció descarregarem la previsió meteorològica de tots els parcs dels Estats Units per aquella setmana.

Si haguéssim d’afegir un nou parc eòlic a la llista, tan sols ens caldria afegir la entrada al fitxer XML amb el nom i les coordenades.

6. Agraïments. Qui és propietari del conjunt de dades? Inclou cites de recerca o anàlisi anteriors.

El conjunt original de dades correspon al servei meteorològic nacional depenent del govern dels Estats Units (Dept de Comerce).

National Weather Service

El propi NOAA dedica un apartat específic per al desenvolupament de projectes basats en les seves dades.

<https://www.noaa.gov/research>

També disposa d’un repositori de dades accessible per a tothom qui necessiti accedir a les seves dades:

<https://www.esrl.noaa.gov/csd/datasets.html>

7. Inspiració. Per què és interessant aquest conjunt de dades? Quines preguntes li agradaria respondre la comunitat?

Es evident que estem davant d’una de les informacions més demandades a Internet. La previsió del temps juga un paper important en diferents escenaris socioeconòmics, com per exemple, el turisme, l’agricultura, etc. Per tant la previsió del temps té una valor afegit innegable.

En el nostre cas en particular és crucial per a la planificació d’activitats d’Operació i Manteniment dels parc eòlics.

Es per aquest motiu, i tal i com he comentat en apartats anteriors, que he decidit recollir aquesta informació, no tant sols amb l’objectiu de facilitar l’accés a la informació d’una forma més automàtica, sinó d’analitzar-la més endavant per avaluar fins a quin punt les previsions son ajustades o no i generar, de cara al futur, possibles models de previsió que millorin els actuals i permetin una planificació de les activitats (O&M) a parc molt més acurada.

8. Llicència. Cal que seleccioneu una d’aquestes llicències i cal dir perquè l’heu seleccionada:

En el meu cas em decanto per una llicencia CC0: Public Domain licence. El per què d’aquesta elecció? Primer veiem que es el CC, per més endavant delimitar a CC0.

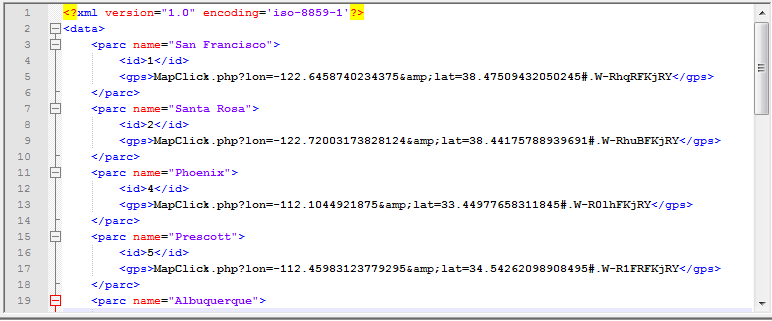
Tal com diu la Wikipedia: CC és la abreviació de Creative Commons és una organització sense ànim de lucre dedicada a promoure l’accés i l’intercanvi de cultura. Desenvolupa un conjunt d’instruments jurídics de caràcter gratuït que faciliten usar i compartir la creativitat i el coneixement.

Aquests instruments jurídics consisteixen en un conjunt de models de contractes de llicenciament o llicencies de drets d’autor que ofereixen a l’autor una manera simple i estandarditzada de cedir permisos al públic per compartir i utilitzar el seu treball creatiu sota els termes i condicions de la seva elecció.

En el nostre cas en particular crec que les dades referents a previsions metrològiques són obertes i de lliure accés, i poden ser distribuïdes i reutilitzades sense cap mena de limitació.

9. Codi: Cal adjuntar el codi amb el que heu generat el dataset, preferiblement amb R o Python.

*########################################################################################################################  
#Importem les llibreries necessaries per al nostre codi***import** requests  
**import** pandas **as** pd  
**import** csv  
**import** time  
**import** sys  
**import** datetime  
**from** xml.dom **import** minidom  
**from** bs4 **import** BeautifulSoup  
  
*#Utilitzarem una funcio a on el parametre sera la URL a fer web scraping  
#En aquesta funcio treballarem els conceptes vistos al materia dodent com son:  
#el tractament de error, espaiat en la descarrega, modificar user agent***def** obtenir\_dades(nom\_parc,url\_pagina):  
  
 *#Modifiquem les capceleres i l'user agent per evitar ser bloqueixats.  
 # Hi ha la possibilitat en Python de generar user agent de forma aleatoria amb:  
 #ua=UserAgent()  
 #requests.get("http://forecast.weather.gov/MapClick.php?lat=37.7772&lon=-122.4168", headers={'user-agent': ua.random})* capcelera = {**"Accept"**: **"text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,\\*/\*;q=0.8"**, **"Accept-Encoding"**: **"gzip, deflate, sdch, br"**, **"Accept-Language"**: **"en-US,en;q=0.8"**, **"Cache-Control"**: **"no-cache"**, **"dnt"**: **"1"**, **"Pragma"**: **"no-cache"**, **"Upgrade-Insecure-Requests"**: **"1"**, **"User-Agent"**: **"Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.77 Safari/537.36"**}  
  
 *#Configurem l espaiat de descarrega per a no collapsar la pagina* t0 = time.time()  
  
 *#Codi per al tractament dels errors que es derivin de la execucio del codi (Timeouts, pagina inaccesible)* **try**:  
  
 *#Descarrega de la pagina web amb les previsions per a una localitat; atencio als parametres de capcalera i timeouts* pagina = requests.get(url\_pagina, headers=capcelera, timeout=10)  
  
 *#Obtenim el temps de resposta en la descarrega de la pagina* temps\_resposta = time.time() - t0  
 *#Ara ja podem espaiar en el temps les succesives descarregues que poguem fer, per exemple, aplicant un delay de  
 #2 vegades el temps de descarrega* time.sleep(2 \* temps\_resposta)  
 *#o un valor fixe de 5 segons tal i com veiem en la linea seguent  
 #time.sleep(5)  
  
 # Verifiquem que la descarrega ha tingut exit* **if** pagina.status\_code == 200:  
 **print**(**"Descarrega de la web satidfactoria ("**,pagina.status\_code,**")"**)  
  
 *#Inicialitzem l'objecte BeautifulSoup per al seu analisis posterior* soup = BeautifulSoup(pagina.content, **'html.parser'**)  
 *#Localitzem l'identificador que conte les dades que necessitem recuperar (previssio setmana)* propera\_setmana = soup.find(id=**"seven-day-forecast"**)  
 previsio\_dies = propera\_setmana.find\_all(class\_=**"tombstone-container"**)  
  
 *#Cerquem la previssio de tota la setmana dins del HTML descarregat* etiq\_period = propera\_setmana.select(**".tombstone-container .period-name"**)  
 periodes = [pt.get\_text() **for** pt **in** etiq\_period]  
 desc\_curtes = [sd.get\_text() **for** sd **in** propera\_setmana.select(**".tombstone-container .short-desc"**)]  
 temperatures = [t.get\_text() **for** t **in** propera\_setmana.select(**".tombstone-container .temp"**)]  
 descripcions = [d[**"title"**] **for** d **in** propera\_setmana.select(**".tombstone-container img"**)]  
  
 *#i preparem el data set amb les dades recollides* data\_set\_temps = pd.DataFrame({  
 **"periode"**: periodes,  
 **"descripcio\_curta"**: desc\_curtes,  
 **"temperatura"**: temperatures,  
 **"descripcio"**:descripcions  
 })  
  
 *#Donem un cop d'ull a com queda el data set* **print**(data\_set\_temps)  
  
 *#Finalment copiem el data set al fitxer CSV. El fitxer l'anomenarem amb el nom del parc i la data d'extraccio* ara = datetime.datetime.now()  
 data\_set\_temps.to\_csv(**'Previsio\_'** + nom\_parc + **"\_"** + ara.strftime(**"%Y-%m-%d"**) + **'.csv'**, encoding=**'utf-8'**, index=False)  
 **print**(**"Descarrega de les dades del parc eolic "**, nom\_parc, **" realitzat amb exit!"**)  
  
 **else**:  
 **print**(**"Descarrega de la web erronia ("**, pagina.status\_code, **") \n"**)  
  
 *#Realitzem el tractament dels errors per TimeOut o per caiguda de la pagina* **except** requests.exceptions.Timeout:  
 **print**(**"Atenció!!! s'ha produit un error per Timeout"**)  
 **except** requests.exceptions.RequestException:  
 **print**(**"Atenció!!! s'ha produit un error per caiguda de la web"**)

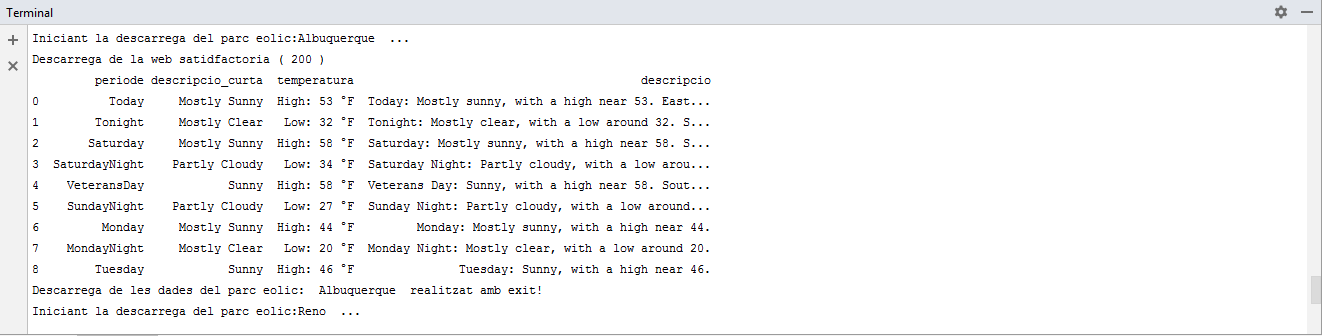


*#Programa principal  
#Lleguir l'XML amb les localitzacions (parcs eolics)*doc = minidom.parse(**"localitzacions\_parcs.xml"**)  
  
*#obtenim les dades parc per parc del XML*parcs = doc.getElementsByTagName(**"parc"**)  
  
*#Per cada parc del fitxer fem la crida a la funcio que*

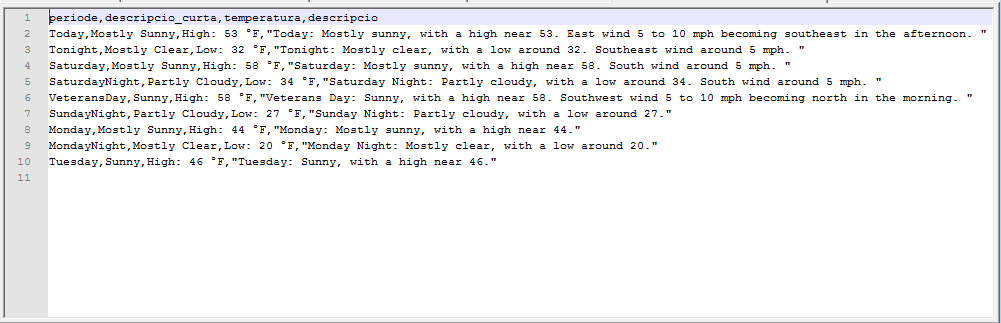
*#extreura les dades i les copiara al fitxer csv***for** parc **in** parcs:  
 *#obtenim els parametres de la funcio* nom = parc.getAttribute(**"name"**)  
 gps = parc.getElementsByTagName(**"gps"**)[0]  
  
 *#Cridem a la funcio* **print**(**"Iniciant la descarrega del parc eolic: %s "** % nom, **"..."**)  
 obtenir\_dades(nom, **"http://forecast.weather.gov/"** + gps.firstChild.data)  
  
*#Una vegada tenim tots els fitxers donem per finalitzada la extraccio***print**(**"Fi de la extraccio de dades!!!"**)  
  
*##################Fi Codi########################################################################################*

10. Dataset: Dataset en format CSV

Representació del data set:



Representació del fitxer CSV:



I finalment resultat de la descarrega del temps per a tots els parc eòlics (un fitxer per parc).

Després amb eines ETL podríem unir-los o tractar-los per separat.

