PRAC 1

Daniel López Ramírez
M2.951 - Tipologia i cicle de vida de les dades

Contingut

Context

El context per aquest dataset s'ha basat en intentar recollir estadístiques de jugadors en cada partit d'una competició per a poder utilitzar aquesta informació estadística en la preparació de nous partits o d'equips per a intentar guanyar o millorar en una competició. Avaluant les diverses plataformes d'estadístiques de basket, es va optar per utilitzar la de l'Eurolliga, ja que és una competició continental, i dintre de la seva web és pot recollir tota la informació de totes les temporades que s'han jugat.

Titol Dataset

Estadístiques de Basket

Descripció

El conjunt de dades generades pel procés de web scraping recull les estadístiques de cada partit d'eurolliga (d'un any o de tots els anys), amb les dades de cada jugador de cada equip. El procés de web scraping es pot generalitzar per recollir la informació tant de la NBA com de l'ACB o de les lligues de Basket de les que es vulguin recollir informació.

Imatge dataset

Com a imatge del dataset, he seleccionat una imatge que reflecteixi l'esport del que es volen recollir les dades, per a després poder treballar-les.

Contingut del dataset

El nostre dataset conté tots els partits de l'eurolliga 2018-2019 per a poder realitzar el seu estudi. El programari realitzat, permet extreure els partits de qualsevol temporada o de totes les temporades, però crec que per a poder treballar amb ell, és millor poder treballar amb una sola temporada.

Per a cada partit de la temporada de la lliga seleccionada, en el nostre cas, l'eurolliga, es recullen dades del partit i els equips, i les estadístiques de cada jugador. Aquestes dades es porten recollint des de l'any 2000. La URL on realitzar el web scraping és: https://www.euroleague.net/main/results/showgame?gamecode=259&seasoncode=E2017#!boxscore

On podem anar iterant el número de la variable *"gamecode"* que seleccionan el número de partit de la temporada (normalment de 1 a 260). Per a poder saber fins a quin codi de partit hem d'arribar, podem consultar si el resultat que retorna el servidor quan fem el get és un codi 200 o un altre codi. La segona variable que hem de tenir en compte és la de *"seasoncode"* on seleccionarem la temporada d'eurolliga que volem recuperar.

Un cop amb la pàgina del partit recuperada, recollim les diverses variables que ens interessen consultant les diverses classes on es troben:

- Dades jugadors → classe de tipus td "PlayerContainer"
- Equips → classe de tipus div "eu-team-stats-teamname"
- Entrenadors → classe de tipus span "title"
- Ronda → classe de tipus div "round-header"
- Resultat → classe de tipus div "game-score"
- Data → classe de tipus div "dates"

Amb aquestes classes, hem recollit la següent informació:

- Data partit: data i hora del partit jugat
- Camp: camp on es juga el partit
- Temporada: temporada a la que pertany el partit
- Ronda: ronda de la temporada
- Fase: fase de la temporada
- Equip: equip del partit
- Punts: punts de l'equip
- Entrenador: nom de l'entrenador
- Jugador: nom del jugador
- Minuts: minuts jugats pel jugador
- Punts: punts del jugador
- Intents2: tirs de 2 intentats pel jugador
- Intents3: tirs de 3 intentats pel jugador
- Tirs lliures: tirs lliures intentats pel jugador
- Rebots ofensius: rebots ofensius del jugador
- Rebots defensius: rebots defensius del jugador
- Rebots totals: rebots totals del jugador
- Assistències: assistències del jugador
- Robatoris: robatoris del jugador
- Pèrdues: pèrdues del jugador
- Bloquejos fets: taps fets pel jugador
- Bloquejos rebuts: taps rebuts pel jugador
- Faltes realitzades: faltes realitzades pel jugador
- Faltes rebudes: faltes rebudes pel jugador
- PIR (Performance Index Rating): índex de rendiment del jugador generat per una fórmula matemàtica per determinar el jugador més valuós de la jornada.

Agraïments

Les dades s'han pogut recuperar gràcies a la informació publicada a la web de l'Eurolliga per a cada partit de cada temporada. Aquesta informació està publicada en format HTML que ens ha permès recuperar-lo, seguint les dades recuperades amb l'anàlisi inicial per a no ser bloquejats, com per exemple el delay entre peticions de 15 segons. Donem les

gràcies a l'Eurolliga per publicar aquesta informació i que pugui ser recollida per al seu estudi posterior.

Inspiració

Aquest conjunt de dades és interessant, ja que ens permet recollir la informació de tota una temporada d'Eurolliga en un sol fitxer, per així poder analitzar-la. A part de les dades estadístiques que podem extreure, com els millors jugadors per posició o quins jugadors augmenten el seu rendiment quan s'arriba a les fases finals, també podem intentar elaborar models predictius per veure quin equip podria guanyar la competició, o bé si els fitxatges que vol realitzar un equip els hi permetria "teòricament" guanyar la competició.

Llicència

La llicència pel dataset resultant hauria de permetre el següent:

- El dataset pot ser copiat i redistribuït en qualsevol medi o format.
- El dataset pot ser reutilitzat, transformat o creat per altres tant amb propòsits comercials com no comercials.
- S'ha de fer referència al creador del dataset quan es vulgui utilitzar.
- S'ha de llicenciar la nova obra sobre els mateixos termes.

Per tant la llicència que compleix aquestes regles seria: CC BY-SA 4.0

Codi font i dataset

Tant el codi font com el dataset es troba en el següent enllaç de github.

En la carpeta "src" es pot trobar el codi font, que està composat per 3 fitxers en python:

- Basketscrapper.py: funció principal per a realitzar el web scraping. Primer de tot es realitza una avaluació inicial del site, revisant el fitxer robots.txt, el sitemap, la grandària del site, la tecnologia emprada i el propietari del lloc. Aquesta informació s'extreu amb l'ajut de la llibreria "Robots.py", i a més es guarda en un fitxer pdf (en el nostre cas euroleague.pdf) amb l'ajut de la llibreria "Export.py". La informació recollida ens permet realitzar el web scraping de la web d'estadístiques ajustant els diversos valors recollits, i generant el dataset amb l'ajut de la llibreria "euroleague.py".
- Robots.py: llibreria que ens permet realitzar l'avaluació inicial d'un site.
- Export.py: llibreria que ens permet generar un fitxer pdf amb la informació que se li passa.
- Euroleague.py: llibreria que ens permet realitzar el web scraping a la web d'estadístiques de l'eurolliga.

Finalment, adjunto la taula amb les contribucions realitzades a cada part de la pràctica.

Contribucions	Signa
Recerca prèvia	Daniel López
Redacció de les respostes	Daniel López
Desenvolupament codi	Daniel López

Recursos

Recursos relacionats amb web scraping de Basket

- https://rdrr.io/cran/BAwiR/
- https://nycdatascience.com/blog/student-works/web-scraping/web-scraping-nba-stats/
- https://towardsdatascience.com/web-scraping-nba-stats-4b4f8c525994
- https://stackoverflow.com/questions/54994665/scraping-nba-advanced-stats-with-python-beautifulsoup
- https://www.r-bloggers.com/scraping-nba-game-data-from-basketball-reference-com/
- http://kevincsong.com/Scraping-stats.nba.com-with-python/
- https://github.com/ccagrawal/nbaTools

Recursos relacionats amb les dades de l'Eurolliga

- www.euroleague.net https://www.euroleague.net/main/results/showgame?gamecode=65&seasoncode=E2018
- view-source:https://www.euroleague.net/main/results/showgame?gamecode=65&seasoncode=E2018
- https://www.euroleague.net/main/results/showgame?gamecode=259&seasoncode=E2017#!boxscore
- view-source:https://www.euroleague.net/main/results/showgame?gamecode=257&seasoncode=E2017

Recursos relacionats amb la programació en python

- https://docs.python.org/3/library/urllib.robotparser.html
- https://github.com/c4software/python-sitemap
- https://pypi.org/project/ultimate-sitemap-parser/
- https://pyfpdf.readthedocs.io/en/latest/index.html
- https://www.pythonforbeginners.com/system/python-sys-argv
- https://stackoverflow.com/questions/21570780/using-python-and-beautifulsoup-saved-webpage-source-codes-into-a-local-file
- https://stackoverflow.com/questions/32938575/grabbing-a-certain-td-class-with-beautifulsoup
- https://www.daniweb.com/programming/software-development/threads/405662/beautifulsoup-to-extract-multiple-td-tags-within-tr
- https://datascience.stackexchange.com/questions/10857/how-to-scrape-a-table-from-a-webpage
- https://stackoverflow.com/questions/41687476/using-beautiful-soup-to-find-specific-class
- https://stackoverflow.com/questions/22217713/how-to-select-a-class-of-div-inside-of-a-div-with-beautiful-soup?rq=1
- https://linuxhint.com/python-beautifulsoup-tutorial-for-beginners/
- https://medium.com/@epicshane/using-beautifulsoup4-to-find-class-exact-match-3e263a95e330
- https://www.crummy.com/software/BeautifulSoup/bs4/doc/#next-element-and-previous-element
- https://www.w3schools.com/python/python_ref_string.asp
- https://docs.python.org/3/library/datetime.html#strftime-and-strptime-behavior
- https://www.crummy.com/software/BeautifulSoup/bs4/doc/#navigablestring

- https://stackoverflow.com/questions/275018/how-can-i-remove-a-trailing-newline
- https://pythontic.com/datetime/datetime/attributes
- https://www.w3schools.com/python/python_file_write.asp
- https://tutorial.eyehunts.com/python/python-create-file-empty-text-file-not-exist/
- https://stackoverflow.com/questions/30559214/get-date-and-time-of-installation-for-packages-installed-via-pip?lq=1
- https://stackoverflow.com/questions/24736316/see-when-packages-were-installed-updated-using-pip/24736563#24736563
- https://www.pythonforbeginners.com/argv/more-fun-with-sys-argv

Recursos relacionats amb llicències

- https://creativecommons.org/publicdomain/zero/1.0/
- https://creativecommons.org/faq/#what-are-creative-commons-licenses
- https://creativecommons.org/licenses/
- https://creativecommons.org/licenses/by-nc-sa/4.0/
- https://creativecommons.org/licenses/by-sa/4.0/
- https://opendatacommons.org/licenses/odbl/
- https://creativecommons.org/licenses/by-sa/3.0/deed.es
- https://es.wikipedia.org/wiki/Licencias_Creative_Commons
- https://upload.wikimedia.org/wikipedia/commons/2/25/Nerdson216es.png