Predicting IAAF World Half Marathon Championship & Training Strategy for Athlete

Ruofan Lyu
001222480

Introduction

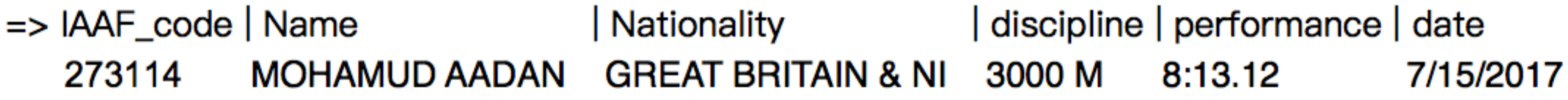
IAAF(International Association of Athletics Federations) World Half Marathon Championship is a well-known sports event in the world. As a possible pre-qualification of the most famous six World Marathon Major, including Boston Marathon, it is a highly participated marathon event.

Predicting athletes’ performance is significant. It helps athletes to determine their further training strategy. For sport enthusiasts, predicting can bring more professional guidance to them and attract more public attention.

Approach

- Web data scraping

- Build a regression model based on predictor variables: age, discipline, performance, date



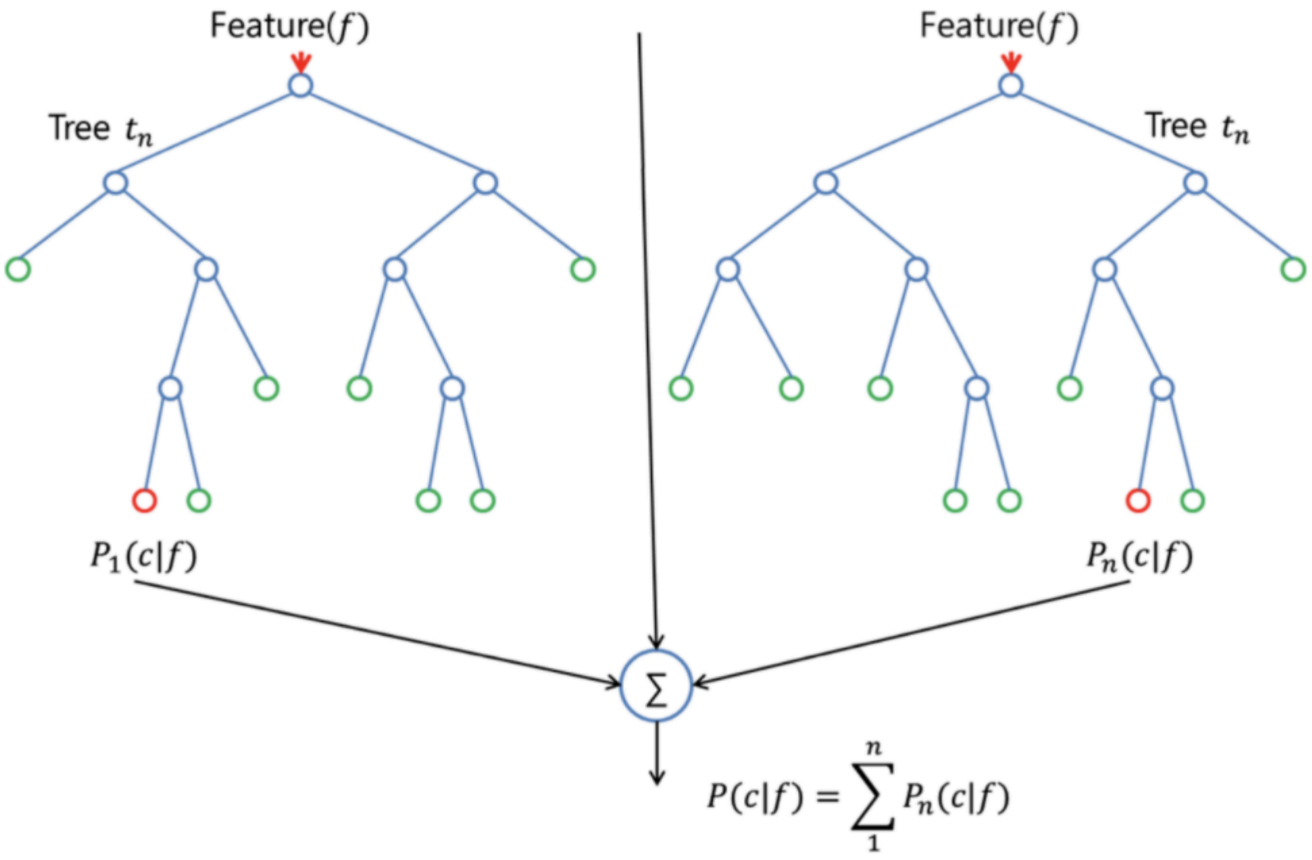
```
=> IAAF_code | Name              | Nationality          | discipline | performance | date
     273114     MOHAMUD AADAN    GREAT BRITAIN & NI    3000 M      8:13.12          7/15/2017
```

- Provide a training strategy

Multiple regression

In the multivariate case, when there is more than one independent variable, the regression line cannot be visualized in the two dimensional space. We could construct a linear equation containing all those variables. In general then, multiple regression procedures will estimate a linear equation of the form:

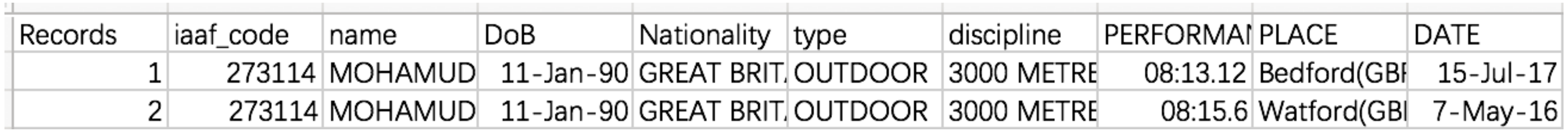
$$Y = a + b_1*X_1 + b_2*X_2 + ... + b_p*X_p$$



Random Forest

Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction.

One big advantage of random forest is, that it can be used for both classification and regression problems, which form the majority of current machine learning systems.



Records	iaaf_code	name	DoB	Nationality	type	discipline	PERFORMAI	PLACE	DATE
1	273114	MOHAMUD	11-Jan-90	GREAT BRIT.	OUTDOOR	3000 METRE	08:13.12	Bedford(GBI	15-Jul-17
2	273114	MOHAMUD	11-Jan-90	GREAT BRIT.	OUTDOOR	3000 METRE	08:15.6	Watford(GBI	7-May-16

- Age when attending competition => decision tree1: best age to get better performance?
- Skilled discipline => decision tree2: expert in longer distance race get better performance in half marathon?
- Date influence => decision tree3: athletes have different performance status in specific time?

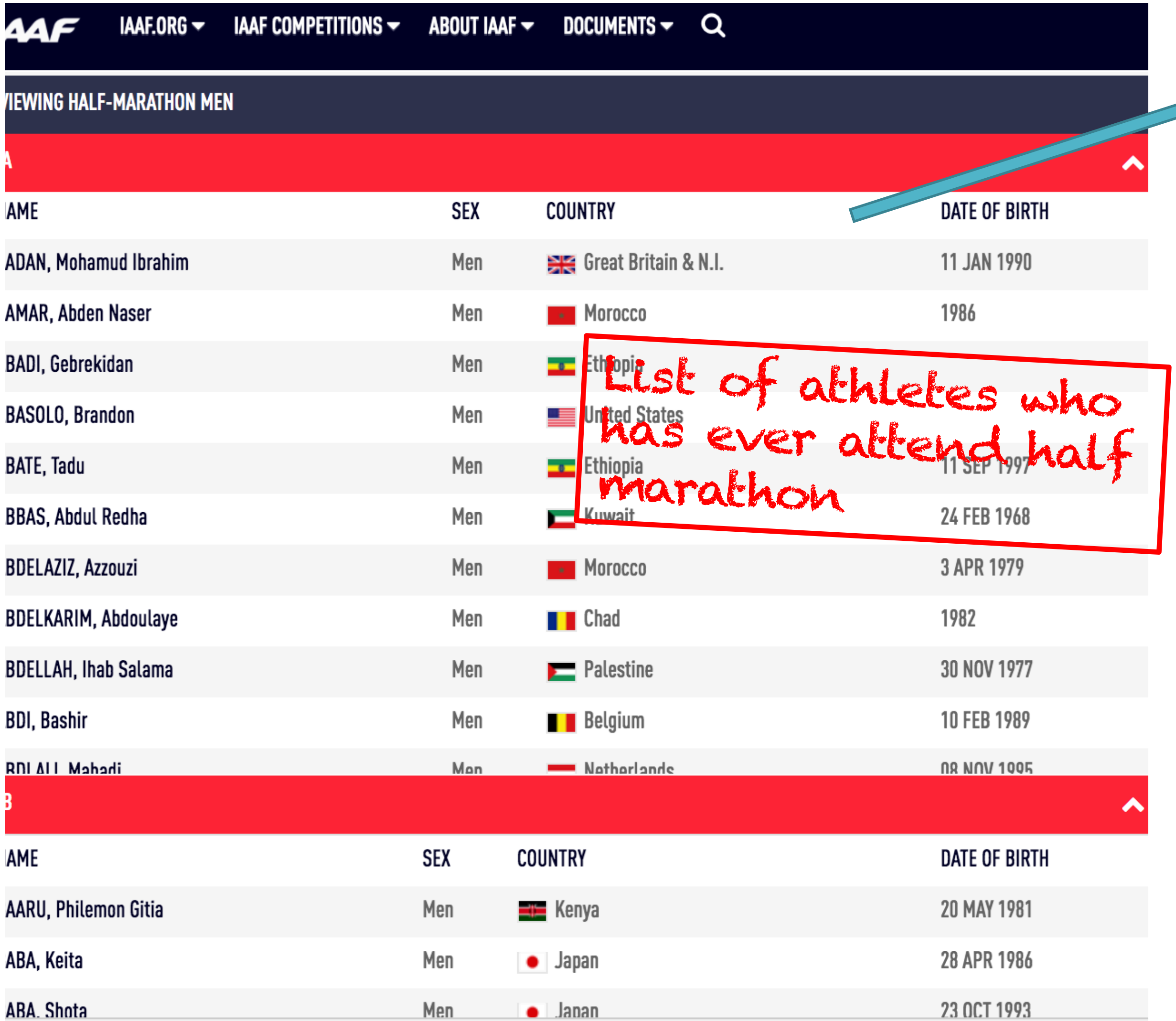
Naive Bayes classifier

In machine learning, naive Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with independence assumptions between the features.

$$P(C_k|x)\,\frac{P(C_k)P(x|C_k)}{P(x)}$$

Knowing the probability of x|Ck, calculate P(Ck|x): to determine the result.

Data Source



IEWING HALF-MARATHON MEN

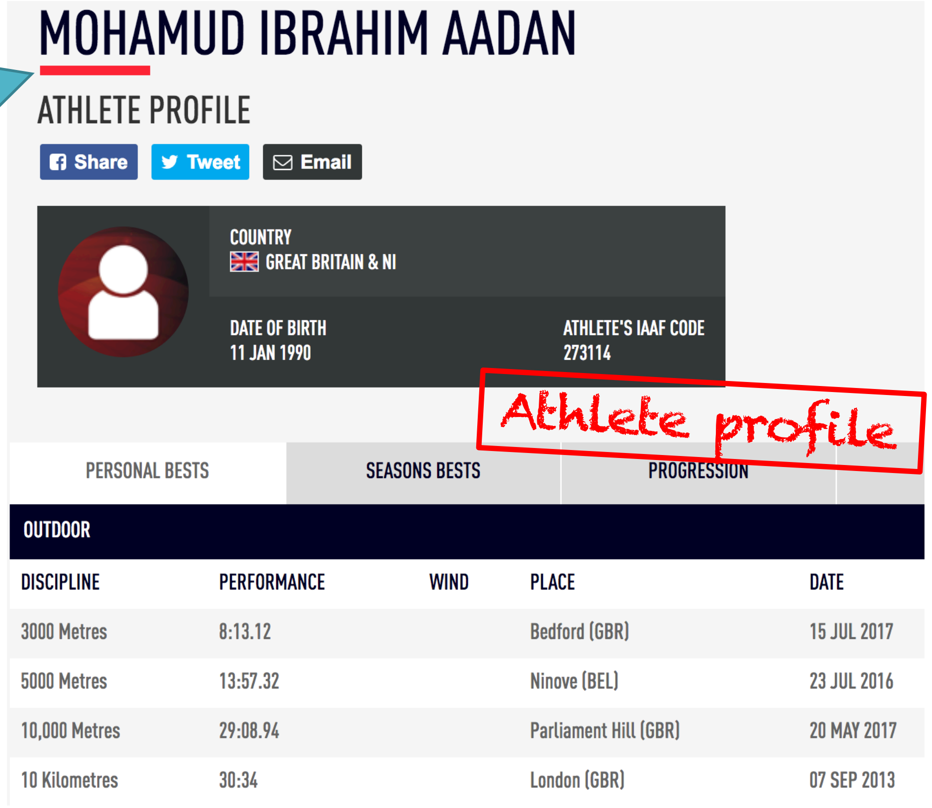
A

AME	SEX	COUNTRY	DATE OF BIRTH
ADAN, Mohamud Ibrahim	Men	Great Britain & N.I.	11 JAN 1990
AMAR, Abden Naser	Men	Morocco	1986
BADI, Gebrekidan	Men	Ethiopia	
BASOLO, Brandon	Men	United States	
BATE, Tadu	Men	Ethiopia	11 SEP 1997
BBAS, Abdul Redha	Men	Kuwait	24 FEB 1968
BDELAZIZ, Azzouzi	Men	Morocco	3 APR 1979
BDELKARIM, Abdoulaye	Men	Chad	1982
BDELLAH, Ihab Salama	Men	Palestine	30 NOV 1977
BDI, Bashir	Men	Belgium	10 FEB 1989
BDI ALI, Mahadi	Men	Netherlands	08 NOV 1995

B

AME	SEX	COUNTRY	DATE OF BIRTH
AARU, Philemon Gitia	Men	Kenya	20 MAY 1981
ABA, Keita	Men	Japan	28 APR 1986
ABA, Shota	Men	Japan	23 OCT 1993

List of athletes who has ever attend half marathon



MOHAMUD IBRAHIM AADAN

ATHLETE PROFILE

COUNTRY
GREAT BRITAIN & NI

DATE OF BIRTH
11 JAN 1990

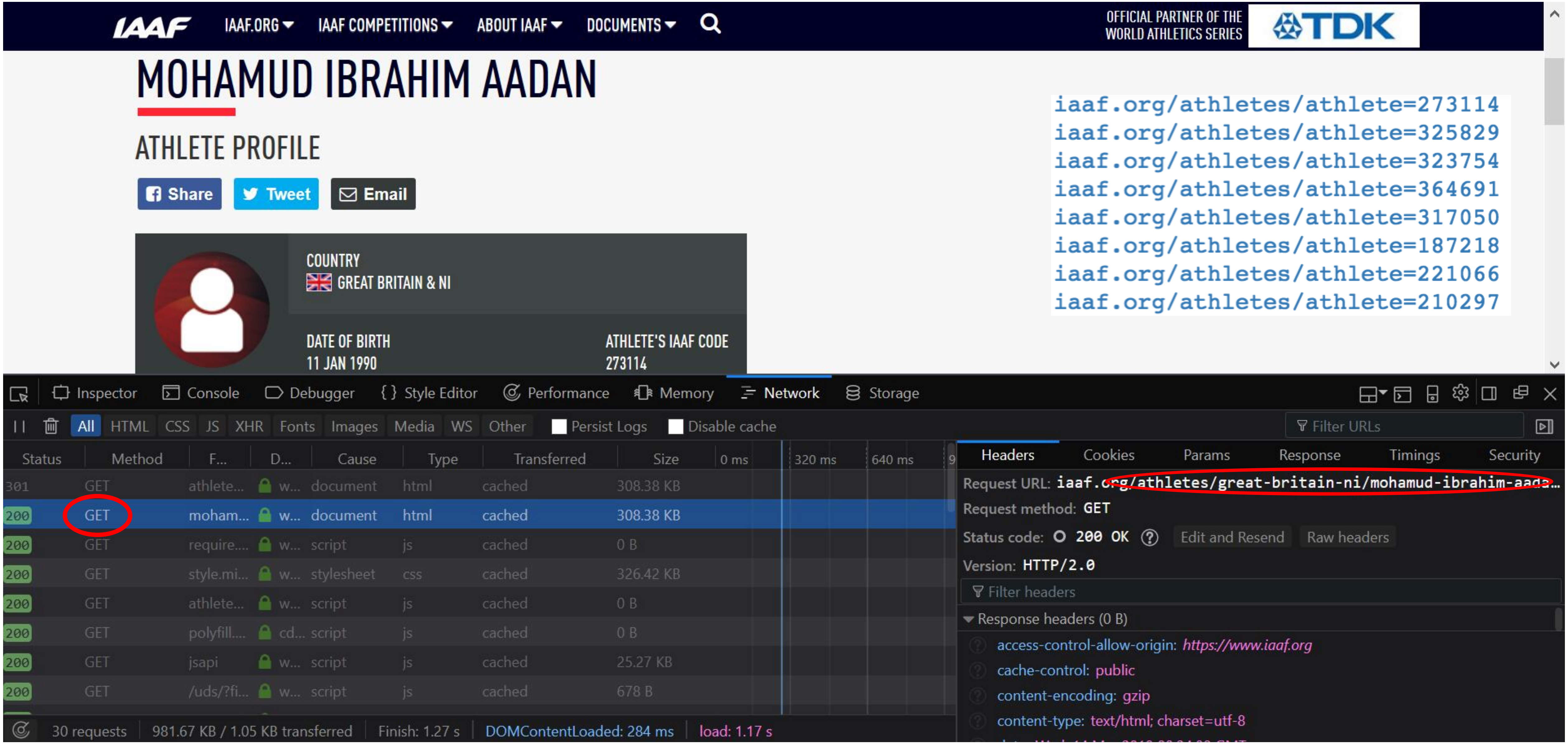
ATHLETE'S IAAF CODE
273114

OUTDOOR

DISCIPLINE	PERFORMANCE	WIND	PLACE	DATE
3000 Metres	8:13.12		Bedford (GBR)	15 JUL 2017
5000 Metres	13:57.32		Ninove (BEL)	23 JUL 2016
10,000 Metres	29:08.94		Parliament Hill (GBR)	20 MAY 2017
10 Kilometres	30:34		London (GBR)	07 SEP 2013

Athlete profile

Data scraping



MOHAMUD IBRAHIM AADAN

ATHLETE PROFILE

COUNTRY
GREAT BRITAIN & NI

DATE OF BIRTH
11 JAN 1990

ATHLETE'S IAAF CODE
273114

```
iaaf.org/athletes/athlete=273114
iaaf.org/athletes/athlete=325829
iaaf.org/athletes/athlete=323754
iaaf.org/athletes/athlete=364691
iaaf.org/athletes/athlete=317050
iaaf.org/athletes/athlete=187218
iaaf.org/athletes/athlete=221066
iaaf.org/athletes/athlete=210297
```

□ Persist Logs □ Disable cache

Status	Method	F...	D...	Cause	Type	Transferred	Size
301	GET	athlete...	w...	document	html	cached	308.38 KB
200	GET	moham...	w...	document	html	cached	308.38 KB
200	GET	require...	w...	script	js	cached	0 B
200	GET	style.mi...	w...	stylesheet	css	cached	326.42 KB
200	GET	athlete...	w...	script	js	cached	0 B
200	GET	polyfill...	cd...	script	js	cached	0 B
200	GET	jsapi	w...	script	js	cached	25.27 KB
200	GET	/uds/?fi...	w...	script	js	cached	678 B

0 ms 320 ms 640 ms

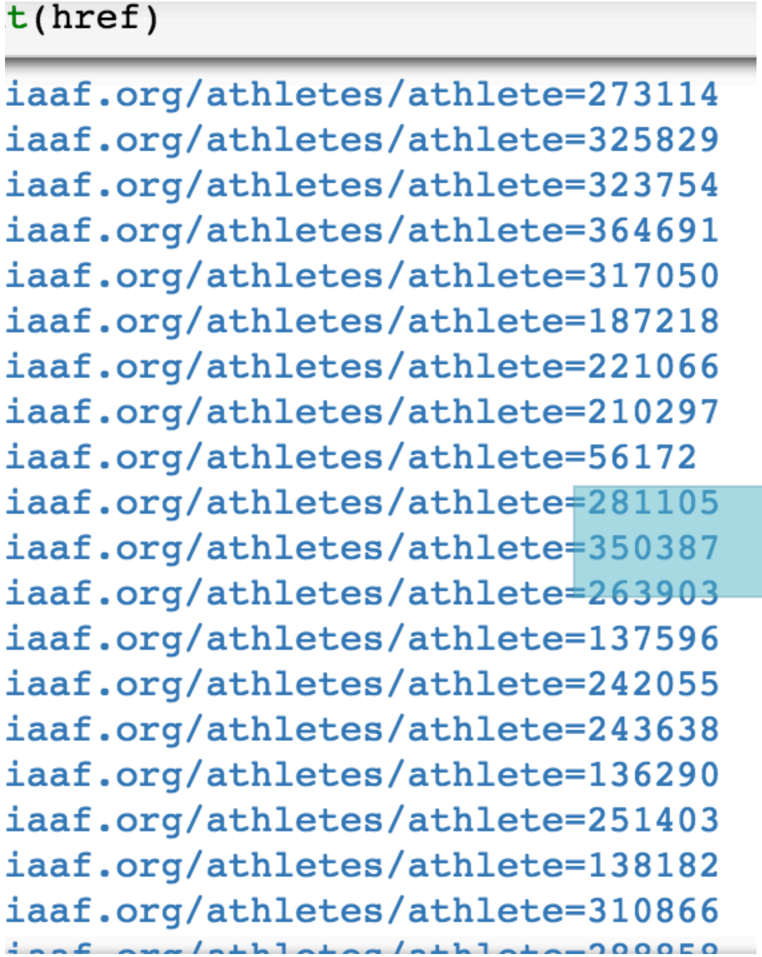
30 requests | 981.67 KB / 1.05 KB transferred | Finish: 1.27 s | DOMContentLoaded: 284 ms | load: 1.17 s

Request URL: iaaf.org/athletes/great-britain-ni/mohamud-ibrahim-aada…
Request method: GET
Status code: 200 OK
Version: HTTP/2.0

Response headers (0 B)
access-control-allow-origin: https://www.iaaf.org
cache-control: public
content-encoding: gzip
content-type: text/html; charset=utf-8

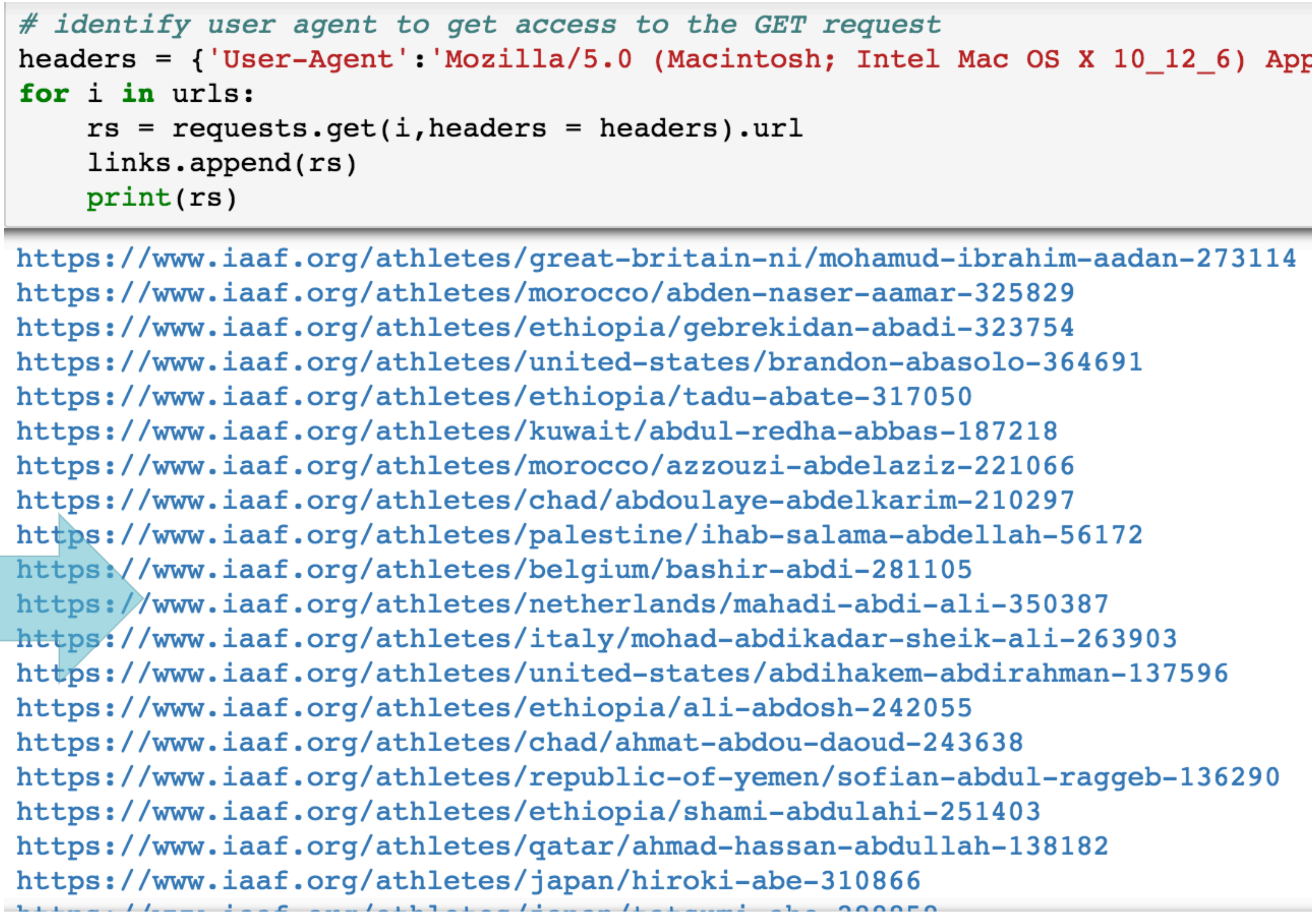
Redirected URL →

Original URL
↓



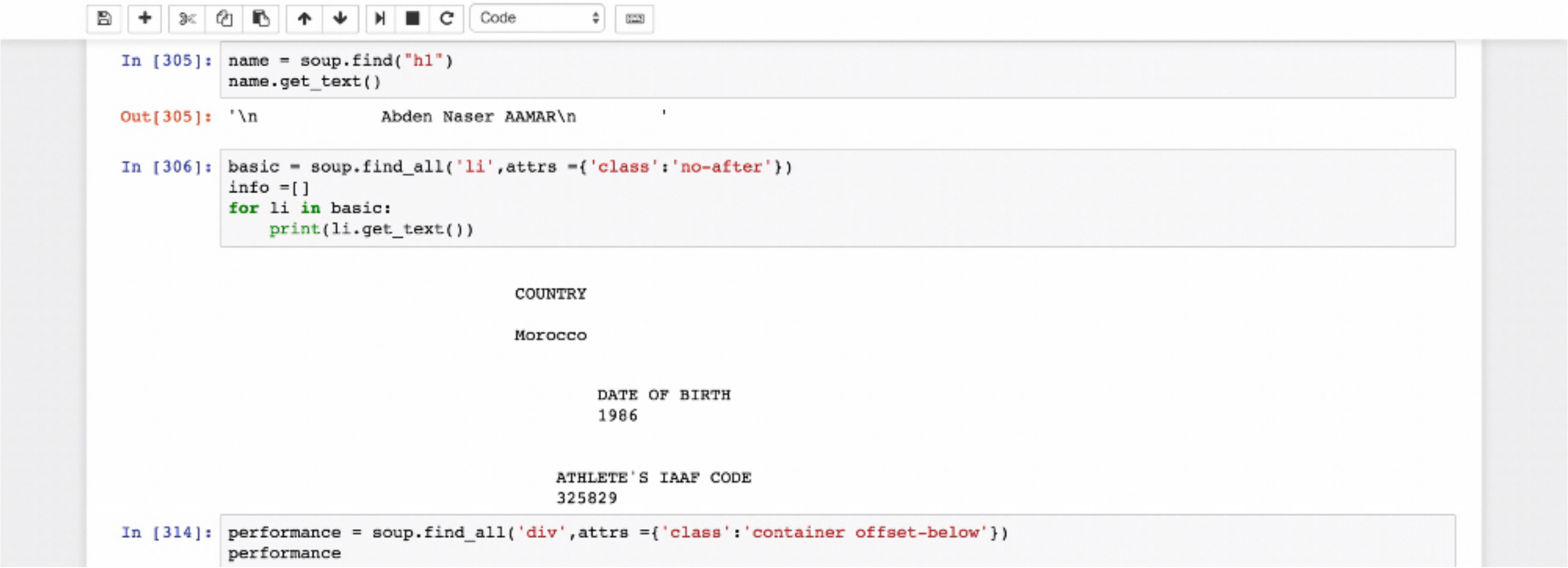
```
t(href)
```

```
iaaf.org/athletes/athlete=273114
iaaf.org/athletes/athlete=325829
iaaf.org/athletes/athlete=323754
iaaf.org/athletes/athlete=364691
iaaf.org/athletes/athlete=317050
iaaf.org/athletes/athlete=187218
iaaf.org/athletes/athlete=221066
iaaf.org/athletes/athlete=210297
iaaf.org/athletes/athlete=56172
iaaf.org/athletes/athlete=281105
iaaf.org/athletes/athlete=350387
iaaf.org/athletes/athlete=263903
iaaf.org/athletes/athlete=137596
iaaf.org/athletes/athlete=242055
iaaf.org/athletes/athlete=243638
iaaf.org/athletes/athlete=136290
iaaf.org/athletes/athlete=251403
iaaf.org/athletes/athlete=138182
iaaf.org/athletes/athlete=310866
iaaf.org/athletes/athlete=288858
```

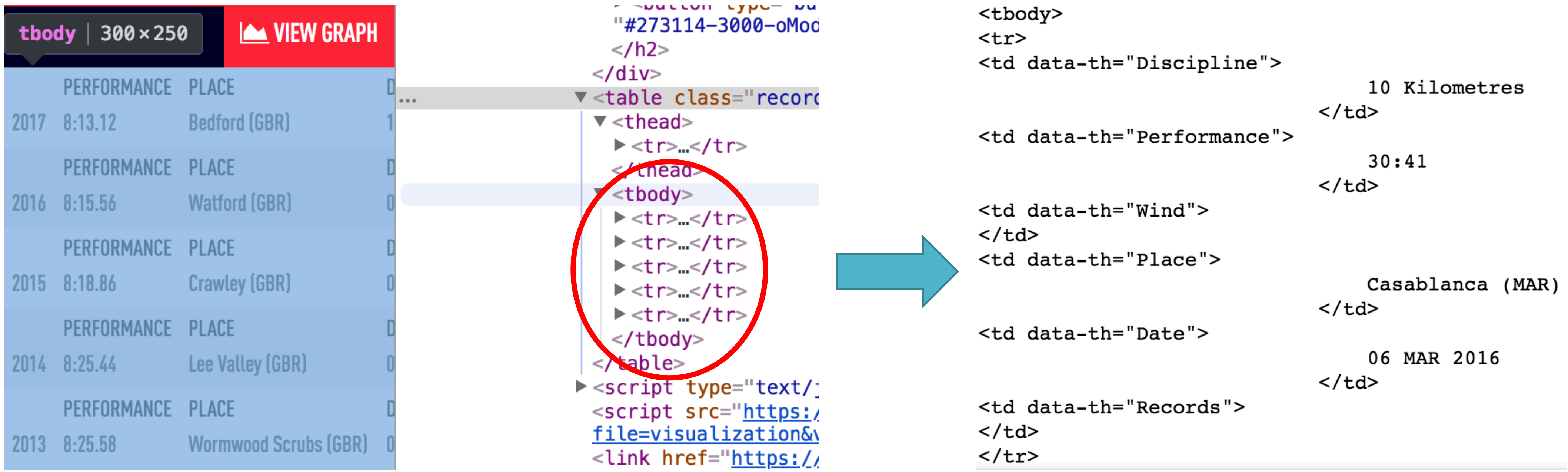


```
# identify user agent to get access to the GET request
headers = {'User-Agent':'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) App
for i in urls:
    rs = requests.get(i,headers = headers).url
    links.append(rs)
    print(rs)
```

```
https://www.iaaf.org/athletes/great-britain-ni/mohamud-ibrahim-aadan-273114
https://www.iaaf.org/athletes/morocco/abden-naser-aamar-325829
https://www.iaaf.org/athletes/ethiopia/gebrekidan-abadi-323754
https://www.iaaf.org/athletes/united-states/brandon-abasolo-364691
https://www.iaaf.org/athletes/ethiopia/tadu-abate-317050
https://www.iaaf.org/athletes/kuwait/abdul-redha-abbas-187218
https://www.iaaf.org/athletes/morocco/azzouzi-abdelaziz-221066
https://www.iaaf.org/athletes/chad/abdoulaye-abdelkarim-210297
https://www.iaaf.org/athletes/palestine/ihab-salama-abdellah-56172
https://www.iaaf.org/athletes/belgium/bashir-abdi-281105
https://www.iaaf.org/athletes/netherlands/mahadi-abdi-ali-350387
https://www.iaaf.org/athletes/italy/mohad-abdikadar-sheik-ali-263903
https://www.iaaf.org/athletes/united-states/abdihakem-abdirahman-137596
https://www.iaaf.org/athletes/ethiopia/ali-abdosh-242055
https://www.iaaf.org/athletes/chad/ahmat-abdou-daoud-243638
https://www.iaaf.org/athletes/republic-of-yemen/sofian-abdul-raggeb-136290
https://www.iaaf.org/athletes/ethiopia/shami-abdulahi-251403
https://www.iaaf.org/athletes/qatar/ahmad-hassan-abdullah-138182
https://www.iaaf.org/athletes/japan/hiroki-abe-310866
https://www.iaaf.org/athletes/japan/tatsumi-abe-288858
```



```
In [305]: name = soup.find("h1")
          name.get_text()

Out[305]: '\n            Abden Naser AAMAR\n        '

In [306]: basic = soup.find_all('li',attrs ={'class':'no-after'})
          info =[]
          for li in basic:
              print(li.get_text())


                            COUNTRY

                            Morocco


                                    DATE OF BIRTH
                                    1986


                                ATHLETE'S IAAF CODE
                                325829

In [314]: performance = soup.find_all('div',attrs ={'class':'container offset-below'})
          performance
```

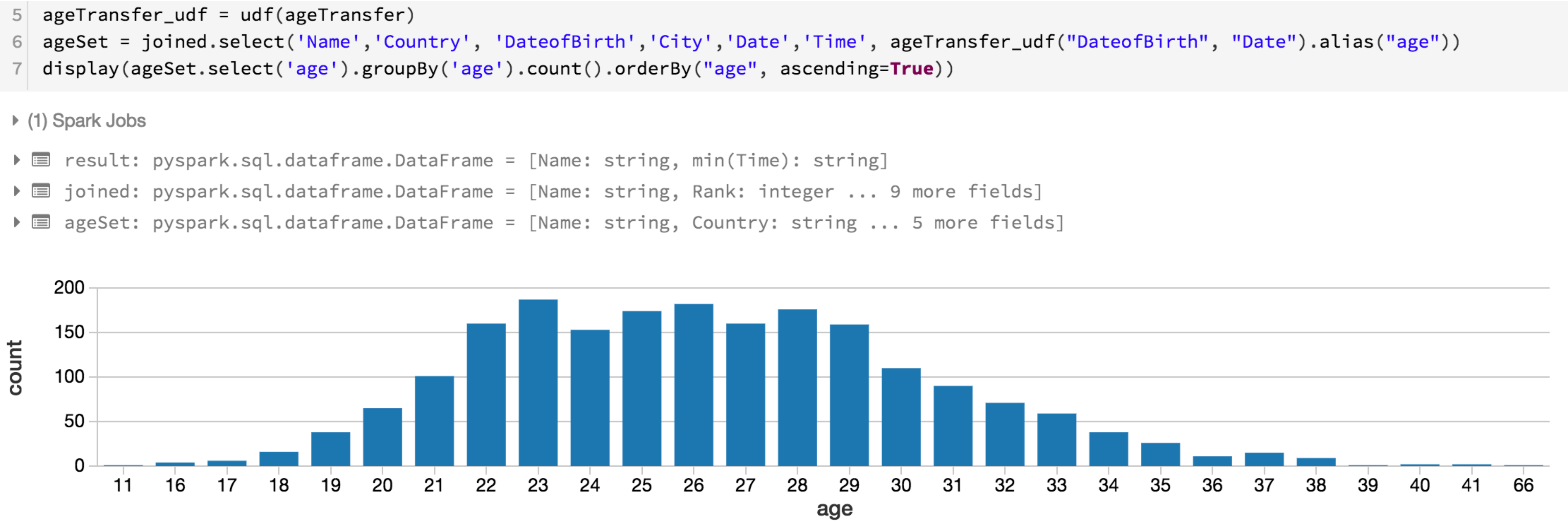


	PERFORMANCE	PLACE	D
2017	8:13.12	Bedford (GBR)	1
	PERFORMANCE	PLACE	D
2016	8:15.56	Watford (GBR)	0
	PERFORMANCE	PLACE	D
2015	8:18.86	Crawley (GBR)	0
	PERFORMANCE	PLACE	D
2014	8:25.44	Lee Valley (GBR)	0
	PERFORMANCE	PLACE	D
2013	8:25.58	Wormwood Scrubs (GBR)	0

```
  "#273114-3000-oMod
  </h2>
</div>
▼<table class="recorc
  ▼<thead>
    ▶<tr>…</tr>
  </thead>
  ▼<tbody>
    ▶<tr>…</tr>
    ▶<tr>…</tr>
    ▶<tr>…</tr>
    ▶<tr>…</tr>
    ▶<tr>…</tr>
  </tbody>
</table>
▶<script type="text/:
<script src="https:/
file=visualization&v
<link href="https://
```

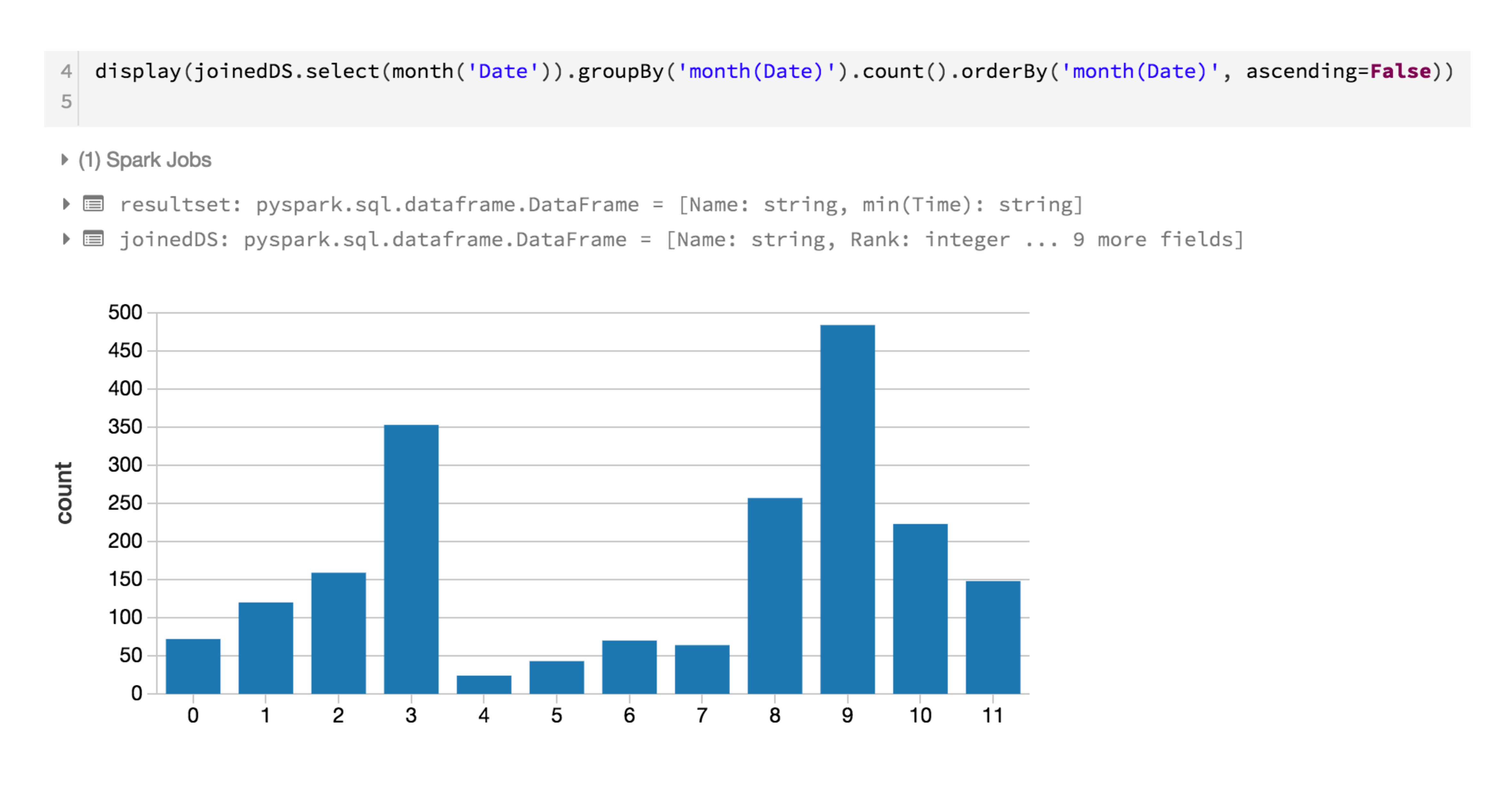
```
<tbody>
<tr>
<td data-th="Discipline">
                10 Kilometres
            </td>

<td data-th="Performance">
                30:41
            </td>

<td data-th="Wind">
</td>
<td data-th="Place">

                Casablanca (MAR)
            </td>

<td data-th="Date">
                06 MAR 2016
            </td>

<td data-th="Records">
</td>
</tr>
```

Data Analysis



```
ageTransfer_udf = udf(ageTransfer)
ageSet = joined.select('Name','Country', 'DateofBirth','City','Date','Time', ageTransfer_udf("DateofBirth", "Date").alias("age"))
display(ageSet.select('age').groupBy('age').count().orderBy("age", ascending=True))
```

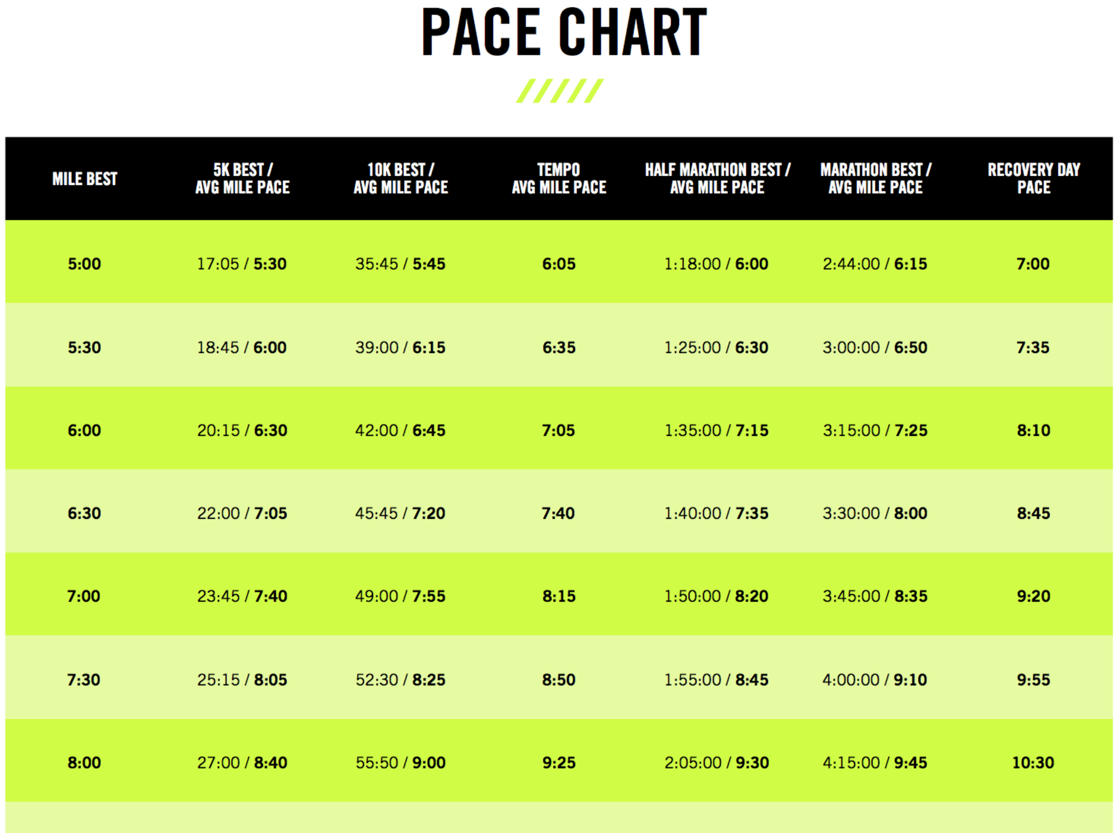
```
▸ (1) Spark Jobs
▸ result: pyspark.sql.dataframe.DataFrame = [Name: string, min(Time): string]
▸ joined: pyspark.sql.dataframe.DataFrame = [Name: string, Rank: integer ... 9 more fields]
▸ ageSet: pyspark.sql.dataframe.DataFrame = [Name: string, Country: string ... 5 more fields]
```



```
display(joinedDS.select(month('Date')).groupBy('month(Date)').count().orderBy('month(Date)', ascending=False))
```

▸ (1) Spark Jobs

```
▸ resultset: pyspark.sql.dataframe.DataFrame = [Name: string, min(Time): string]
▸ joinedDS: pyspark.sql.dataframe.DataFrame = [Name: string, Rank: integer ... 9 more fields]
```



PACE CHART

MILE BEST	5K BEST / AVG MILE PACE	10K BEST / AVG MILE PACE	TEMPO AVG MILE PACE	HALF MARATHON BEST / AVG MILE PACE	MARATHON BEST / AVG MILE PACE	RECOVERY DAY PACE
5:00	17:05 / **5:30**	35:45 / **5:45**	**6:05**	1:18:00 / **6:00**	2:44:00 / **6:15**	**7:00**
5:30	18:45 / **6:00**	39:00 / **6:15**	**6:35**	1:25:00 / **6:30**	3:00:00 / **6:50**	**7:35**
6:00	20:15 / **6:30**	42:00 / **6:45**	**7:05**	1:35:00 / **7:15**	3:15:00 / **7:25**	**8:10**
6:30	22:00 / **7:05**	45:45 / **7:20**	**7:40**	1:40:00 / **7:35**	3:30:00 / **8:00**	**8:45**
7:00	23:45 / **7:40**	49:00 / **7:55**	**8:15**	1:50:00 / **8:20**	3:45:00 / **8:35**	**9:20**
7:30	25:15 / **8:05**	52:30 / **8:25**	**8:50**	1:55:00 / **8:45**	4:00:00 / **9:10**	**9:55**
8:00	27:00 / **8:40**	55:50 / **9:00**	**9:25**	2:05:00 / **9:30**	4:15:00 / **9:45**	**10:30**

Next Steps

- Build prediction model
- Apply athletes’ data to the training plan

The training strategy basically refer to the Half Marathon Pace Chart

References

- Thomas A. Severini. (2014). Analytic Methods in Sports: Using Mathematics and Statistics to Understand Data from Baseball, Football, Basketball, and Other Sports. Chapter 2.
- S Tufféry. (2011). Data mining and statistics for decision making. 534 -538
- Michael Bowles. (2015). Machine Learning in Python: Essential Techniques for Predictive Analysis. 122-124
- Brian Hanley. Pacing profiles and pack running at the IAAF World Half Marathon Championships. (2015). https://pdfs.semanticscholar.org/819e/8db906d3673e46b6f609d1d77838f34ae1f1.pdf
- https://en.wikipedia.org/wiki/Naive_Bayes_classifier
- https://towardsdatascience.com/the-random-forest-algorithm-d457d499ffcd