

Planet Data Collection

Using the Open Exoplanet Catalogue database:

https://github.com/OpenExoplanetCatalogue/open_exoplanet_catalogue/
(https://github.com/OpenExoplanetCatalogue/open_exoplanet_catalogue/).

Data License

Copyright (C) 2012 Hanno Rein

Permission is hereby granted, free of charge, to any person obtaining a copy of this database and associated scripts (the "Database"), to deal in the Database without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Database, and to permit persons to whom the Database is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Database. A reference to the Database shall be included in all scientific publications that make use of the Database.

THE DATABASE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE DATABASE OR THE USE OR OTHER DEALINGS IN THE DATABASE.

Follow instructions to get the xml file

```
In [1]: import xml.etree.ElementTree as ET, urllib.request, gzip, io
url = "https://github.com/OpenExoplanetCatalogue/oec_gzip/raw/master/systems.xml.gz"
oec = ET.parse(gzip.GzipFile(fileobj=io.BytesIO(urllib.request.urlopen(url).read())))
```

Parse into Pandas DataFrame

Information on what each field means can be found [here](#)

(https://github.com/OpenExoplanetCatalogue/open_exoplanet_catalogue/#data-structure).

```
In [3]: import pandas as pd

def parse(base):
    db = oec.findall(f".://{base}")

    exclude = ['star', 'videolink', 'binary'] if base in ['system', 'binary']
    else ['planet']

    columns = set([attribute.tag for attribute in db[0] if attribute.tag not i
n exclude])
    results = pd.DataFrame(columns=columns)

    for entry in db:
        data = {col : entry.findtext(col) for col in columns}
        # count binary and star items in each
        if base in ['system', 'binary']:
            data['binaries'] = len(entry.findall('.//binary'))
            data['stars'] = len(entry.findall('.//star'))
        # count planet items in each
        if base in ['system', 'star', 'binary']:
            data['planets'] = len(entry.findall('.//planet'))
        results = results.append(data, ignore_index=True)

    return results
```

Parse planet data

```
In [4]: planets = parse('planet')
planets.head()
```

Out[4]:

	discoverymethod	description	periastrontime	discoveryyear	eccentricity	semimajoraxis	period
0	RV	11 Com b is a brown dwarf-mass companion to th...	2452899.6	2008	0.231	1.29	326.6
1	RV	11 Ursae Minoris is a star located in the cons...	2452861.04	2009	0.08	1.54	516.2
2	RV	14 Andromedae is an evolved star in the conste...	2452861.4	2008	0	0.83	185.8
3	RV	The star 14 Herculis is only 59 light years aw...	None	2002	0.359	2.864	176.8
4	RV	14 Her c is the second companion in the system...	None	2006	0.184	9.037	988.8

Parse system data

```
In [5]: systems = parse('system')
systems.head()
```

Out[5]:

	distance	rightascension	name	constellation	declination	binaries	planets	stars
0	88.9	12 20 43.0255	11 Com	Coma Berenices	+17 47 34.3392	0.0	1.0	1.0
1	122.1	15 17 05.88899	11 UMi	Ursa Minor	+71 49 26.0466	0.0	1.0	1.0
2	79.2	23 31 17.41346	14 And	Andromeda	+39 14 10.3092	0.0	1.0	1.0
3	18.1	16 10 24.3152	14 Her	Hercules	+43 49 03.4987	0.0	2.0	1.0
4	21.146	19 41 48.95343	16 Cygni	Cygnus	+50 31 30.2153	2.0	1.0	3.0

Parse binary data

```
In [6]: binaries = parse('binary')
        binaries.head()
```

Out[6]:

	name	positionangle	separation	binaries	planets	stars
0	16 Cygni	133.30	39.56	1.0	1.0	3.0
1	16 Cygni AC	209	3.4	0.0	0.0	2.0
2	2M0441+2301	237.3	12.37	1.0	1.0	3.0
3	2M 044145	79.61	0.2323	0.0	0.0	2.0
4	2M 1938+4603	None	None	0.0	1.0	2.0

Parse star data

```
In [7]: stars = parse('star')
        stars.head()
```

Out[7]:

	magK	magB	metallicity	magH	name	mass	magV	spectraltype	radius	magJ	temperature
0	2.282	5.74	-0.35	2.484	11 Com	2.7	4.74	G8 III	19	2.943	4742
1	1.939	6.415	0.04	2.091	11 UMi	1.80	5.024	K4III	24.08	2.876	4340
2	2.331	6.24	-0.24	2.608	14 And	2.2	5.22	K0III	11	3.019	4813
3	4.714	7.57	0.43	4.803	14 Her	1.0	6.67	K0 V	0.708	5.158	5311.0
4	4.43	6.59	0.096	4.72	16 Cygni A	1.11	5.95	G2V	1.243	5.09	5825

Save to CSVs

```
In [8]: planets.to_csv('data/planets.csv', index=False)
        binaries.to_csv('data/binaries.csv', index=False)
        stars.to_csv('data/stars.csv', index=False)
        systems.to_csv('data/systems.csv', index=False)
```

[← Chapter 8](#) ([../../ch_08/anomaly_detection.ipynb](#))</div>[Next Notebook →](#) ([../planets_ml.ipynb](#))

In []: