

# 1\_\_Obtain\_\_Seyfert\_\_Samples

May 31, 2021

## 1 Obtain Seyfert Samples

In this notebook we show the steps to obtain the Seyfert sample of galaxies.

We import only the `astropy` Table as most of the work for this step is done outside Jupyter notebooks (with [TOPCAT](#)).

```
[5]: from astropy.table import Table
```

### 1.1 Obtaining the list

We obtain the [SIMBAD](#) (SMB) sample of galaxies (SMB\_3Dec.fits) with the following SQL query.

**Be aware that here we refer to SQL to the queries. However, [SIMBAD-TAP](#) actually uses [ADQL](#) (Astronomical Data Query Language), which is based on SQL.**

```
SELECT main_id
      ,ra
      ,DEC
      ,coo_bibcode
      ,otype_txt
      ,rvz_bibcode
      ,rvz_radvel
      ,rvz_redshift
      ,rvz_type
      ,nbref
      ,alltypes.otypes
FROM basic
JOIN alltypes ON oid = oidref
WHERE basic.otype = 'Sy1'
      OR basic.otype = 'Sy2'
      OR basic.otype = 'SyG'
```

### 1.2 Using TOPCAT for the cross-match

We used [TOPCAT](#) in this step. First, we obtained the [Veron-Cetty M.P. , Veron P. 2010](#) (VCV) catalog from [VizieR](#). Then, we added the SMB file in TOPCAT. We cross-matched the two samples in terms of coordinates, assuming galaxies are the same if their coordinates differ by less than 2 arcsecs. The final file (SMB\_VCV.fits) is saved and then is transformed in a [VOTable](#) here. This step can also be done in TOPCAT.

```
[3]: Matched = Table.read('../Data/Raw/SMB_VCV.fits', format='fits')
Matched['main_id', 'otype_txt', 'otypes'].write('../Data/Raw/SMBVCV_VO',
↳format='votable')
```

### 1.3 Get the bibcodes for the classification type

We run the following SQL query to search for the bibcodes of the otypes in SMB with the SMB-VCV\_VO file. We save this information in a separated file (BibcodesOtypes.fits)

```
SELECT Sy.main_id
      ,otypes.otype_txt
      ,otypes.origin
FROM basic
JOIN ident ON oidref = oid
JOIN otypes USING (oidref)
JOIN TAP_UPLOAD.Seyf AS Sy ON Sy.main_id = basic.main_id
WHERE id = Sy.main_id
```

With this information, we can clean the files in 2\_Clean\_Sample.ipynb

### 1.4 Another solution

After checking that the two SQL queries can be united without uploading a new table, another solution to get the origin of the otype is run the following SQL query

```
SELECT DISTINCT main_id
      ,ra
      ,DEC
      ,otypes.otype_txt
      ,otypes.origin
FROM basic
JOIN ident ON oidref = oid
JOIN otypes USING (oidref)
WHERE basic.otype = 'Sy1'
      OR basic.otype = 'Sy2'
      OR basic.otype = 'SyG'
```

and then you could filter with the selected SMB\_VCV information.

#### Notebook info

```
[6]: %load_ext watermark
%watermark -a "Andres Ramos" -d -v -m
print('Specific Python packages')
%watermark -iv -w --packages astropy
```

The watermark extension is already loaded. To reload it, use:

```
%reload_ext watermark
```

Author: Andres Ramos

Python implementation: CPython  
Python version : 3.8.3  
IPython version : 7.16.1

Compiler : GCC 7.3.0  
OS : Linux  
Release : 3.10.0-1160.el7.x86\_64  
Machine : x86\_64  
Processor : x86\_64  
CPU cores : 8  
Architecture: 64bit

Specific Python packages

json : 2.0.9  
numpy : 1.19.5  
autopep8: 1.5.4

Watermark: 2.1.0