

example__downloading__data

October 30, 2025

1 SDSS Galaxy Spectra Download and Matching

This notebook/script demonstrates how to query the **SDSS DR18 SkyServer** to retrieve spectroscopic data for a given list of galaxies. The input catalog must contain the columns “**ra**”, “**dec**”, and “**z**”.

The workflow:

1. Cross-match the input sample with SDSS spectroscopic objects within 1 arcsec tolerance.
2. For each matched galaxy, construct the corresponding SDSS FITS spectrum filename.
3. Download the spectra (if not already present) and store them locally.
4. Save all metadata (coordinates, redshift, plate, fiberID, MJD, run2d, etc.) into a JSON file (`galaxies_info.json`) for easy reuse.

This ensures that every downloaded spectrum is accompanied by its full set of identifiers and observational parameters, making the dataset ready for further analysis or reproducibility.

```
[6]: import os
import json
import wget
import pandas as pd
from spectools import get_spec
```

In the first step, define the directories where you want to save your downloaded data.

```
[7]: SkyServer_DataRelease = "DR18"
URL_DR18 = 'https://dr18.sdss.org/sas/dr18/spectro/sdss/redux'
downloaded_data_directory = '../data/galaxies/'
galaxies_info_file = '../data/galaxies/galaxies_info.json'

os.makedirs(downloaded_data_directory, exist_ok=True)

data = pd.read_csv('../data/galaxies.csv')
data
```

```
[7]:
```

	ra	dec	z
0	229.525576	42.745854	0.040272
1	28.311913	0.050259	0.019350

2	21.536851	-1.292157	0.018429
3	21.383000	-1.591000	0.017816
4	21.588522	-1.314362	0.016772
5	21.392144	-1.207726	0.019097
6	21.510423	-1.225596	0.018016
7	21.281444	-1.258806	0.020224
8	21.223958	-1.500778	0.015868
9	21.348600	-1.194754	0.017585
10	21.448256	-1.179668	0.018402
11	21.295000	-1.612222	0.017629
12	21.226713	-1.257844	0.016497
13	21.197125	-1.525639	0.017866

The next step query the SDSS to retrieve the galaxies in our sample, and then perform a cross-match between the two tables to determine which entries correspond to the objects of interest.

```
[8]: obj = get_spec.search_skyserver(data, SkyServer_DataRelease)

matching_files = get_spec.match_galaxies(obj, data)
matching_files
```

```
[8]:
```

	ra	dec	z	specobjid	class	plate \
0	229.525576	42.745854	0.040272	1889376924388583424	GALAXY	1678
1	28.311913	0.050259	0.019350	1693505516698888192	GALAXY	1504
5	21.392144	-1.207726	0.019097	1746326611545843712	GALAXY	1551
6	21.510423	-1.225596	0.018016	449268178174896128	GALAXY	399
7	21.281444	-1.258806	0.020224	1690048390685026304	GALAXY	1501
9	21.348600	-1.194754	0.017585	449291817674893312	GALAXY	399
10	21.448256	-1.179668	0.018402	449269002808616960	GALAXY	399
12	21.226713	-1.257844	0.016497	783649736845453312	GALAXY	696

	mjd	fiberid	run2d	bestObjID
0	53433	425	26	1237662301903192106
1	52940	553	26	1237663784209809871
5	53327	203	26	1237663782596117236
6	51817	124	26	1237663782596182075
7	53740	264	26	1237663782596116676
9	51817	210	26	1237663782596116701
10	51817	127	26	1237663782596182178
12	52209	85	26	1237663782596051087

Here the data is downloaded!

```
[9]: data_info = {}
for _, row in matching_files.iterrows():

    run2d = row['run2d']
    plate = row['plate']
```

```

fiberid = row['fiberid']
mjd = row['mjd']

# This is the fits name format
spec = f'spec-{str(plate).zfill(4)}-{mjd}-{str(fiberid).zfill(4)}.fits'

url = f'{URL_DR18}/{run2d}/spectra/lite/{str(plate).zfill(4)}/{spec}'

if spec not in os.listdir(downloaded_data_directory):
    wget.download(url, out = downloaded_data_directory)

galaxy_name = f'spec-{str(plate).zfill(4)}-{mjd}-{str(fiberid).zfill(4)}'
data_info[galaxy_name] = row.to_dict()

```

```

[10]: with open(galaxies_info_file, 'w', encoding='utf-8') as f:
        json.dump(data_info, f)

```