

Gaia Archive Tutorial

a short introduction for the Nuclear Physics in Astrophysics XI School

Table of contents

- [Installation of the required packages](#)
- [GACS login](#)
- [Gaia table names](#)
- [Positional query with epoch propagation](#)
- [Inspect the nss_two_body_orbit table content](#)
- [Download all datalink info for one source](#)
- [RVS spectra for one star](#)
- [XP spectra with GaiaXPy for one star](#)
- [Learn more](#)

Installation of the required packages: astroquery and GaiaXPy

If not already done, install [astroquery](#)

```
In [1]: !pip install astroquery
```

```
Requirement already satisfied: astroquery in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (0.4.6)
Requirement already satisfied: beautifulsoup4>=4.3.2 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astroquery) (4.11.1)
Requirement already satisfied: requests>=2.4.3 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astroquery) (2.28.1)
Requirement already satisfied: keyring>=4.0 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astroquery) (23.4.0)
Requirement already satisfied: astropy>=4.0 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astroquery) (5.1)
Requirement already satisfied: numpy>=1.16 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astroquery) (1.21.5)
Requirement already satisfied: html5lib>=0.999 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astroquery) (1.1)
Requirement already satisfied: pyvo>=1.1 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astroquery) (1.5)
Requirement already satisfied: pyerfa>=2.0 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astropy>=4.0->astroquery) (2.0.0)
Requirement already satisfied: packaging>=19.0 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astropy>=4.0->astroquery) (21.3)
Requirement already satisfied: PyYAML>=3.13 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astropy>=4.0->astroquery) (6.0)
Requirement already satisfied: soupsieve>1.2 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from beautifulsoup4>=4.3.2->astroquery) (2.3.1)
Requirement already satisfied: webencodings in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from html5lib>=0.999->astroquery) (0.5.1)
Requirement already satisfied: six>=1.9 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from html5lib>=0.999->astroquery) (1.16.0)
Requirement already satisfied: importlib-metadata>=3.6 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from keyring>=4.0->astroquery) (4.11.3)
Requirement already satisfied: idna<4,>=2.5 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from requests>=2.4.3->astroquery) (3.3)
Requirement already satisfied: charset-normalizer<3,>=2 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from requests>=2.4.3->astroquery) (2.0.4)
Requirement already satisfied: certifi>=2017.4.17 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from requests>=2.4.3->astroquery) (2022.9.24)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from requests>=2.4.3->astroquery) (1.26.11)
Requirement already satisfied: zipp>=0.5 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from importlib-metadata>=3.6->keyring>=4.0->astroquery) (3.8.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from packaging>=19.0->astropy>=4.0->astroquery) (3.0.9)
```

If not already done, install [GaiaXPy](#).

See also <https://gaia-dpci.github.io/GaiaXPy-website/tutorials.html>

```
In [2]: !pip install GaiaXPy
```

```

Requirement already satisfied: GaiaXPy in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (2.1.0)
Requirement already satisfied: astroquery in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from GaiaXPy) (0.4.6)
Requirement already satisfied: aenum in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from GaiaXPy) (3.1.15)
Requirement already satisfied: numpy>=1.13 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from GaiaXPy) (1.21.5)
Requirement already satisfied: pandas>=1.0.0 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from GaiaXPy) (1.4.4)
Requirement already satisfied: scipy in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from GaiaXPy) (1.9.1)
Requirement already satisfied: packaging in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from GaiaXPy) (21.3)
Requirement already satisfied: tqdm>=4.64.0 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from GaiaXPy) (4.64.1)
Requirement already satisfied: fastavro in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from GaiaXPy) (1.9.3)
Requirement already satisfied: astropy in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from GaiaXPy) (5.1)
Requirement already satisfied: matplotlib in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from GaiaXPy) (3.5.2)
Requirement already satisfied: python-dateutil>=2.8.1 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from pandas>=1.0.0->GaiaXPy) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from pandas>=1.0.0->GaiaXPy) (2022.1)
Requirement already satisfied: PyYAML>=3.13 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astropy->GaiaXPy) (6.0)
Requirement already satisfied: pyerfa>=2.0 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astropy->GaiaXPy) (2.0.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from packaging->GaiaXPy) (3.0.9)
Requirement already satisfied: requests>=2.4.3 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astroquery->GaiaXPy) (2.28.1)
Requirement already satisfied: html5lib>=0.999 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astroquery->GaiaXPy) (1.1)
Requirement already satisfied: beautifulsoup4>=4.3.2 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astroquery->GaiaXPy) (4.11.1)
Requirement already satisfied: pyvo>=1.1 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astroquery->GaiaXPy) (1.5)
Requirement already satisfied: keyring>=4.0 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from astroquery->GaiaXPy) (23.4.0)
Requirement already satisfied: cycycler>=0.10 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from matplotlib->GaiaXPy) (0.11.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from matplotlib->GaiaXPy) (1.4.2)
Requirement already satisfied: fonttools>=4.22.0 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from matplotlib->GaiaXPy) (4.25.0)
Requirement already satisfied: pillow>=6.2.0 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from matplotlib->GaiaXPy) (9.2.0)
Requirement already satisfied: soupsieve>1.2 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from beautifulsoup4>=4.3.2->astroquery->GaiaXPy) (2.3.1)
Requirement already satisfied: webencodings in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from html5lib>=0.999->astroquery->GaiaXPy) (0.5.1)
Requirement already satisfied: six>=1.9 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from html5lib>=0.999->astroquery->GaiaXPy) (1.16.0)
Requirement already satisfied: importlib-metadata>=3.6 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from keyring>=4.0->astroquery->GaiaXPy) (4.11.3)
Requirement already satisfied: idna<4,>=2.5 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from requests>=2.4.3->astroquery->GaiaXPy) (3.3)
Requirement already satisfied: charset-normalizer<3,>=2 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from requests>=2.4.3->astroquery->GaiaXPy) (2.0.4)
Requirement already satisfied: certifi>=2017.4.17 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from requests>=2.4.3->astroquery->GaiaXPy) (2022.9.24)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from requests>=2.4.3->astroquery->GaiaXPy) (1.26.11)
Requirement already satisfied: zipp>=0.5 in /Users/sklioner/opt/anaconda3/lib/python3.9/site-packages (from importlib-metadata>=3.6->keyring>=4.0->astroquery->GaiaXPy) (3.8.0)

```

The following packages will be used for the project:

```

In [3]: import numpy as np
import matplotlib.pyplot as plt
from astroquery.gaia import Gaia

```

```

In [4]: from gaiaxpy import calibrate
from gaiaxpy import plot_spectra

```

GACS login

Not needed for the following. A prompt will ask for the user name and password:

```

In [ ]: Gaia.login()

```

You can also use a credential file, see <https://astroquery.readthedocs.io/en/latest/gaia/gaia.html#login-logout>

Once you are logged in, you can [upload a file to your user space](#) and for example run a [positional cross-match](#) with the Gaia tables. Note that you need to ensure that you do not have user tables with the same names as the ones you will be using here, else you would need to [delete your old tables](#). Once logged in, you will see also your tables and the ones shared with you with the `Gaia.load_tables` command.

Gaia table names

In this way we download the names of all tables avails from the Gaia archive

```
In [5]: tables = Gaia.load_tables(only_names=True)
INFO: Retrieving tables... [astroquery.utils.tap.core]
INFO: Parsing tables... [astroquery.utils.tap.core]
INFO: Done. [astroquery.utils.tap.core]
```

```
In [6]: for table in (tables):
        print (table.get_qualified_name())
```

external.external.apassdr9
external.external.catwise2020
external.external.gaiadr2_astrophysical_parameters
external.external.gaiadr2_geometric_distance
external.external.gaiaedr3_distance
external.external.gaiaedr3_gcns_main_1
external.external.gaiaedr3_gcns_rejected_1
external.external.gaiaedr3_spurious
external.external.gaia_eso_survey
external.external.galex_ais
external.external.ravedr5_com
external.external.ravedr5_dr5
external.external.ravedr5_gra
external.external.ravedr5_on
external.external.ravedr6
external.external.sdssdr13_photoprimary
external.external.skymapperdr1_master
external.external.skymapperdr2_master
external.external.tmass_xsc
external.external.xgboost_table1
external.external.xgboost_table2
gaiadr1.gaiadr1.aux_qso_icrf2_match
gaiadr1.gaiadr1.ext_phot_zero_point
gaiadr1.gaiadr1.allwise_best_neighbour
gaiadr1.gaiadr1.allwise_neighbourhood
gaiadr1.gaiadr1.gsc23_best_neighbour
gaiadr1.gaiadr1.gsc23_neighbourhood
gaiadr1.gaiadr1.ppmxl_best_neighbour
gaiadr1.gaiadr1.ppmxl_neighbourhood
gaiadr1.gaiadr1.sdss_dr9_best_neighbour
gaiadr1.gaiadr1.sdss_dr9_neighbourhood
gaiadr1.gaiadr1.tmass_best_neighbour
gaiadr1.gaiadr1.tmass_neighbourhood
gaiadr1.gaiadr1.ucac4_best_neighbour
gaiadr1.gaiadr1.ucac4_neighbourhood
gaiadr1.gaiadr1.urat1_best_neighbour
gaiadr1.gaiadr1.urat1_neighbourhood
gaiadr1.gaiadr1.cepheid
gaiadr1.gaiadr1.phot_variable_time_series_gfov
gaiadr1.gaiadr1.phot_variable_time_series_gfov_statistical_parameters
gaiadr1.gaiadr1.rrl_yrae
gaiadr1.gaiadr1.variable_summary
gaiadr1.gaiadr1.allwise_original_valid
gaiadr1.gaiadr1.gsc23_original_valid
gaiadr1.gaiadr1.ppmxl_original_valid
gaiadr1.gaiadr1.sdssdr9_original_valid
gaiadr1.gaiadr1.tmass_original_valid
gaiadr1.gaiadr1.ucac4_original_valid
gaiadr1.gaiadr1.urat1_original_valid
gaiadr1.gaiadr1.gaia_source
gaiadr1.gaiadr1.tgas_source
gaiadr2.gaiadr2.aux_allwise_agn_gdr2_cross_id
gaiadr2.gaiadr2.aux_iers_gdr2_cross_id
gaiadr2.gaiadr2.aux_sso_orbit_residuals
gaiadr2.gaiadr2.aux_sso_orbits
gaiadr2.gaiadr2.dr1_neighbourhood
gaiadr2.gaiadr2.allwise_best_neighbour
gaiadr2.gaiadr2.allwise_neighbourhood
gaiadr2.gaiadr2.apassdr9_best_neighbour
gaiadr2.gaiadr2.apassdr9_neighbourhood
gaiadr2.gaiadr2.gsc23_best_neighbour
gaiadr2.gaiadr2.gsc23_neighbourhood
gaiadr2.gaiadr2.hipparcos2_best_neighbour

gaiadr2.gaiadr2.hipparcos2_neighbourhood
gaiadr2.gaiadr2.panstarrs1_best_neighbour
gaiadr2.gaiadr2.panstarrs1_neighbourhood
gaiadr2.gaiadr2.ppmxl_best_neighbour
gaiadr2.gaiadr2.ppmxl_neighbourhood
gaiadr2.gaiadr2.ravedr5_best_neighbour
gaiadr2.gaiadr2.ravedr5_neighbourhood
gaiadr2.gaiadr2.sdssdr9_best_neighbour
gaiadr2.gaiadr2.sdssdr9_neighbourhood
gaiadr2.gaiadr2.tmass_best_neighbour
gaiadr2.gaiadr2.tmass_neighbourhood
gaiadr2.gaiadr2.tycho2_best_neighbour
gaiadr2.gaiadr2.tycho2_neighbourhood
gaiadr2.gaiadr2.urat1_best_neighbour
gaiadr2.gaiadr2.urat1_neighbourhood
gaiadr2.gaiadr2.sso_observation
gaiadr2.gaiadr2.sso_source
gaiadr2.gaiadr2.vari_cepheid
gaiadr2.gaiadr2.vari_classifier_class_definition
gaiadr2.gaiadr2.vari_classifier_definition
gaiadr2.gaiadr2.vari_classifier_result
gaiadr2.gaiadr2.vari_long_period_variable
gaiadr2.gaiadr2.vari_rotation_modulation
gaiadr2.gaiadr2.vari_rrlyrae
gaiadr2.gaiadr2.vari_short_timescale
gaiadr2.gaiadr2.vari_time_series_statistics
gaiadr2.gaiadr2.panstarrs1_original_valid
gaiadr2.gaiadr2.gaia_source
gaiadr2.gaiadr2.ruwe
gaiadr3.gaiadr3.gaia_source
gaiadr3.gaiadr3.gaia_source_lite
gaiadr3.gaiadr3.astrophysical_parameters
gaiadr3.gaiadr3.astrophysical_parameters_supp
gaiadr3.gaiadr3.oa_neuron_information
gaiadr3.gaiadr3.oa_neuron_xp_spectra
gaiadr3.gaiadr3.total_galactic_extinction_map
gaiadr3.gaiadr3.total_galactic_extinction_map_opt
gaiadr3.gaiadr3.commanded_scan_low
gaiadr3.gaiadr3.allwise_best_neighbour
gaiadr3.gaiadr3.allwise_neighbourhood
gaiadr3.gaiadr3.apassdr9_best_neighbour
gaiadr3.gaiadr3.apassdr9_join
gaiadr3.gaiadr3.apassdr9_neighbourhood
gaiadr3.gaiadr3.dr2_neighbourhood
gaiadr3.gaiadr3.gsc23_best_neighbour
gaiadr3.gaiadr3.gsc23_join
gaiadr3.gaiadr3.gsc23_neighbourhood
gaiadr3.gaiadr3.hipparcos2_best_neighbour
gaiadr3.gaiadr3.hipparcos2_neighbourhood
gaiadr3.gaiadr3.panstarrs1_best_neighbour
gaiadr3.gaiadr3.panstarrs1_join
gaiadr3.gaiadr3.panstarrs1_neighbourhood
gaiadr3.gaiadr3.ravedr5_best_neighbour
gaiadr3.gaiadr3.ravedr5_join
gaiadr3.gaiadr3.ravedr5_neighbourhood
gaiadr3.gaiadr3.ravedr6_best_neighbour
gaiadr3.gaiadr3.ravedr6_join
gaiadr3.gaiadr3.ravedr6_neighbourhood
gaiadr3.gaiadr3.sdssdr13_best_neighbour
gaiadr3.gaiadr3.sdssdr13_join
gaiadr3.gaiadr3.sdssdr13_neighbourhood
gaiadr3.gaiadr3.skymapperdr2_best_neighbour
gaiadr3.gaiadr3.skymapperdr2_join

gaiadr3.gaiadr3.skymapperdr2_neighbourhood
gaiadr3.gaiadr3.tmass_psc_xsc_best_neighbour
gaiadr3.gaiadr3.tmass_psc_xsc_join
gaiadr3.gaiadr3.tmass_psc_xsc_neighbourhood
gaiadr3.gaiadr3.tycho2tdsc_merge_best_neighbour
gaiadr3.gaiadr3.tycho2tdsc_merge_neighbourhood
gaiadr3.gaiadr3.urat1_best_neighbour
gaiadr3.gaiadr3.urat1_neighbourhood
gaiadr3.gaiadr3.galaxy_candidates
gaiadr3.gaiadr3.galaxy_catalogue_name
gaiadr3.gaiadr3.qso_candidates
gaiadr3.gaiadr3.qso_catalogue_name
gaiadr3.gaiadr3.nss_acceleration_astro
gaiadr3.gaiadr3.nss_non_linear_spectro
gaiadr3.gaiadr3.nss_two_body_orbit
gaiadr3.gaiadr3.nss_vim_fl
gaiadr3.gaiadr3.binary_masses
gaiadr3.gaiadr3.chemical_cartography
gaiadr3.gaiadr3.gold_sample_carbon_stars
gaiadr3.gaiadr3.gold_sample_fgkm_stars
gaiadr3.gaiadr3.gold_sample_obas_stars
gaiadr3.gaiadr3.gold_sample_solar_analogues
gaiadr3.gaiadr3.gold_sample_spss
gaiadr3.gaiadr3.gold_sample_ucd
gaiadr3.gaiadr3.sso_orbits
gaiadr3.gaiadr3.synthetic_photometry_gspc
gaiadr3.gaiadr3.vari_spurious_signals
gaiadr3.gaiadr3.agn_cross_id
gaiadr3.gaiadr3.frame_rotator_source
gaiadr3.gaiadr3.gaia_crf3_xm
gaiadr3.gaiadr3.alerts_mixedin_sourceids
gaiadr3.gaiadr3.science_alerts
gaiadr3.gaiadr3.gaia_source_simulation
gaiadr3.gaiadr3.gaia_universe_model
gaiadr3.gaiadr3.sso_observation
gaiadr3.gaiadr3.sso_reflectance_spectrum
gaiadr3.gaiadr3.sso_source
gaiadr3.gaiadr3.xp_summary
gaiadr3.gaiadr3.vari_agn
gaiadr3.gaiadr3.vari_cepheid
gaiadr3.gaiadr3.vari_classifier_class_definition
gaiadr3.gaiadr3.vari_classifier_definition
gaiadr3.gaiadr3.vari_classifier_result
gaiadr3.gaiadr3.vari_compact_companion
gaiadr3.gaiadr3.vari_eclipsing_binary
gaiadr3.gaiadr3.vari_epoch_radial_velocity
gaiadr3.gaiadr3.vari_long_period_variable
gaiadr3.gaiadr3.vari_microlensing
gaiadr3.gaiadr3.vari_ms_oscillator
gaiadr3.gaiadr3.vari_planetary_transit
gaiadr3.gaiadr3.vari_planetary_transit_13june2022
gaiadr3.gaiadr3.vari_rad_vel_statistics
gaiadr3.gaiadr3.vari_rotation_modulation
gaiadr3.gaiadr3.vari_rrlyrae
gaiadr3.gaiadr3.vari_short_timescale
gaiadr3.gaiadr3.vari_summary
gaiadr3.gaiadr3.tycho2tdsc_merge
gaiaedr3.gaiaedr3.gaia_source
gaiaedr3.gaiaedr3.agn_cross_id
gaiaedr3.gaiaedr3.commanded_scan_low
gaiaedr3.gaiaedr3.dr2_neighbourhood
gaiaedr3.gaiaedr3.frame_rotator_source
gaiaedr3.gaiaedr3.allwise_best_neighbour

gaiaedr3.gaiaedr3.allwise_neighbourhood
gaiaedr3.gaiaedr3.apassdr9_best_neighbour
gaiaedr3.gaiaedr3.apassdr9_join
gaiaedr3.gaiaedr3.apassdr9_neighbourhood
gaiaedr3.gaiaedr3.gsc23_best_neighbour
gaiaedr3.gaiaedr3.gsc23_join
gaiaedr3.gaiaedr3.gsc23_neighbourhood
gaiaedr3.gaiaedr3.hipparcos2_best_neighbour
gaiaedr3.gaiaedr3.hipparcos2_neighbourhood
gaiaedr3.gaiaedr3.panstarrs1_best_neighbour
gaiaedr3.gaiaedr3.panstarrs1_join
gaiaedr3.gaiaedr3.panstarrs1_neighbourhood
gaiaedr3.gaiaedr3.ravedr5_best_neighbour
gaiaedr3.gaiaedr3.ravedr5_join
gaiaedr3.gaiaedr3.ravedr5_neighbourhood
gaiaedr3.gaiaedr3.sdssdr13_best_neighbour
gaiaedr3.gaiaedr3.sdssdr13_join
gaiaedr3.gaiaedr3.sdssdr13_neighbourhood
gaiaedr3.gaiaedr3.skymapperdr2_best_neighbour
gaiaedr3.gaiaedr3.skymapperdr2_join
gaiaedr3.gaiaedr3.skymapperdr2_neighbourhood
gaiaedr3.gaiaedr3.tmass_psc_xsc_best_neighbour
gaiaedr3.gaiaedr3.tmass_psc_xsc_join
gaiaedr3.gaiaedr3.tmass_psc_xsc_neighbourhood
gaiaedr3.gaiaedr3.tycho2tdsc_merge_best_neighbour
gaiaedr3.gaiaedr3.tycho2tdsc_merge_neighbourhood
gaiaedr3.gaiaedr3.urat1_best_neighbour
gaiaedr3.gaiaedr3.urat1_neighbourhood
gaiaedr3.gaiaedr3.gaia_source_simulation
gaiaedr3.gaiaedr3.gaia_universe_model
gaiaedr3.gaiaedr3.tycho2tdsc_merge
gaiafpr.gaiafpr.crowded_field_source
gaiafpr.gaiafpr.lens_candidates
gaiafpr.gaiafpr.lens_catalogue_name
gaiafpr.gaiafpr.lens_observation
gaiafpr.gaiafpr.lens_outlier
gaiafpr.gaiafpr.sso_observation
gaiafpr.gaiafpr.sso_source
gaiafpr.gaiafpr.interstellar_medium_params
gaiafpr.gaiafpr.interstellar_medium_spectra
gaiafpr.gaiafpr.vari_epoch_radial_velocity
gaiafpr.gaiafpr.vari_long_period_variable
gaiafpr.gaiafpr.vari_rad_vel_statistics
public.public.hipparcos
public.public.hipparcos_newreduction
public.public.hubble_sc
public.public.igsl_source
public.public.igsl_source_catalog_ids
public.public.tycho2
public.public.dual
tap_config.tap_config.coord_sys
tap_config.tap_config.properties
tap_schema.tap_schema.columns
tap_schema.tap_schema.key_columns
tap_schema.tap_schema.keys
tap_schema.tap_schema.schemas
tap_schema.tap_schema.tables

Positional query with epoch propagation

Look if a binary star GJ473 is resolved by Gaia. It's Simbad position (<https://simbad.u-strasbg.fr/simbad/sim-id?Ident=GJ473&NbIdent=1&Radius=2&Radius.unit=arcmin&submit=submit+id>) need to be propagated to the Gaia epoch using EPOCH_PROP_POS, see https://www.cosmos.esa.int/web/gaia-users/archive/writing-queries/#epoch_prop. The following query selects sources within 5" from GJ473 according to its Simbad J2000 information, adapted from the query used by the "Basic Search" GUI:

```
In [7]: queryGJ473 = """SELECT source_id, phot_g_mean_mag, parallax, pmra, pmdec, ruwe
FROM gaiadr3.gaia_source
WHERE
CONTAINS(
    POINT('ICRS',ra,dec),
    CIRCLE(
        'ICRS',
        COORD1(EPOCH_PROP_POS(188.3223395,9.0210534,227.9000,-1730.0000,203.0000,1.0000,2000,2016.0)),
        COORD2(EPOCH_PROP_POS(188.3223395,9.0210534,227.9000,-1730.0000,203.0000,1.0000,2000,2016.0)),
        5./3600.)
)=1"""
```

```
In [8]: job = Gaia.launch_job(queryGJ473)
r = job.get_results()
r
```

Out [8]: Table length=2

SOURCE_ID	phot_g_mean_mag	parallax	pmra	pmdec	ruwe
	mag	mas	mas / yr	mas / yr	
int64	float32	float64	float64	float64	float32
3902874650601954816	11.240162	223.47747382423316	-1710.4676540646703	203.09780864044555	13.949631
3902874650602495232	11.235116	231.11849780261377	-1795.6609024600048	217.78856656191624	6.116347

Inspect the nss_two_body_orbit table content

The field nss_solution_type gives the type of two body orbit solved for. Find the number of sources of each nss_solution_type: https://gea.esac.esa.int/archive/documentation/GDR3/Gaia_archive/chap_datamodel/sec_dm_non--single_stars_tables/ssec_dm_nss_two_body_orbit.html#nss_two_body_orbit-nss_solution_type

```
In [9]: job = Gaia.launch_job("SELECT nss_solution_type, count(*) FROM gaiadr3.nss_two_body_orbit GROUP BY nss_solution_type")
job.get_results()
```

Out [9]: Table length=12

nss_solution_type	COUNT_ALL
object	int64
AstroSpectroSB1	33467
EclipsingBinary	86918
EclipsingSpectro	155
Orbital	134598
OrbitalAlternative	619
OrbitalAlternativeValidated	10
OrbitalTargetedSearch	345
OrbitalTargetedSearchValidated	188
SB1	181327
SB1C	202
SB2	4630
SB2C	746

Download all datalink info for one source

We take here the Cepheid BF Oph (Gaia DR3 4111834567779557376) which is the "Single Object" default example of the GUI.

```
In [10]: datalink = Gaia.load_data(ids="4111834567779557376", retrieval_type='ALL', output_file = None)
datalink.keys()
```

```
Out[10]: dict_keys(['MCMC_GSPPHOT-Gaia DR3 4111834567779557376.xml', 'XP_SAMPLED-Gaia DR3 4111834567779557376.xml', 'EPOCH_PHOTOMETRY-Gaia DR3 4111834567779557376.xml', 'MCMC_MSC-Gaia D
R3 4111834567779557376.xml', 'XP_CONTINUOUS-Gaia DR3 4111834567779557376.xml', 'RVS-Gaia DR3 4111834567779557376.xml'])
```

The XP sampled and RVS products consist in a table that is serialised following the [IVOA Spectrum Data Model](#) (see for details the [DataLink Products serialisation tutorial](#)). As a result, a number of parameters (including the source_id) associated to these files is stored in the table metadata. The cells below show, using RVS Spectra as an example, how to extract these parameters, and how to export the table content to an Astropy Table object.

```
In [11]: dl_key = next((s for s in list(datalink.keys()) if "RVS" in s), None)

dl_product = datalink[dl_key][0]
print("\nContent:")
items = [item for item in dl_product.iter_fields_and_params()]
for item in items:
    print(item)
```

Content:

```
<PARAM ID="source_id" datatype="long" name="source_id" ucd="meta.id;meta.main" value="4111834567779557376"/>
<PARAM ID="solution_id" datatype="long" name="solution_id" ucd="meta.version" value="5950420259779346465"/>
<PARAM ID="combined_transits" datatype="int" name="combined_transits" ucd="meta.number" value="31"/>
<PARAM ID="combined_ccds" datatype="int" name="combined_ccds" ucd="meta.number" value="92"/>
<PARAM ID="deblended_ccds" datatype="int" name="deblended_ccds" ucd="meta.number" value="0"/>
<PARAM ID="spatialLocation" arraysize="2" datatype="double" name="spatialLocation" ucd="pos.eq" unit="deg" utype="spec:Char.SpatialAxis.Coverage.Location.Value" value="[256.522
9102 -26.58056513]"/>
<PARAM ID="TimeAxisCoverageLocation" datatype="double" name="TimeAxisCoverageLocation" ucd="time.epoch" unit="yr" utype="spec:Char.TimeAxis.Coverage.Location.Value" value="201
6.0"/>
<PARAM ID="TimeAxisCoverageBoundsExtent" datatype="double" name="TimeAxisCoverageBoundsExtent" ucd="time.duration" unit="yr" utype="spec:Char.TimeAxis.Coverage.Bounds.Extent" v
alue="2.83"/>
<PARAM ID="spectralAccuracyStatError" datatype="double" name="spectralAccuracyStatError" ucd="stat.error;em.wl" unit="nm" utype="spec:Char.SpectralAxis.Accuracy.StatError" valu
e="0.0"/>
<PARAM ID="spectralLocation" datatype="double" name="spectralLocation" ucd="em.wl;instr.bandpass" unit="nm" utype="spec:Char.SpectralAxis.Coverage.Location.Value" value="0.0"/>
<PARAM ID="spectralCoverageBoundsExtent" datatype="double" name="spectralCoverageBoundsExtent" ucd="instr.bandwidth" unit="nm" utype="spec:Char.SpectralAxis.Coverage.Bounds.Ext
ent" value="24.0"/>
<PARAM ID="spectralCoverageBoundsStart" datatype="double" name="spectralCoverageBoundsStart" ucd="em.wl;stat.min" unit="nm" utype="spec:Char.SpectralAxis.Coverage.Bounds.Start"
value="846.0"/>
<PARAM ID="spectralCoverageBoundsStop" datatype="double" name="spectralCoverageBoundsStop" ucd="em.wl;stat.max" unit="nm" utype="spec:Char.SpectralAxis.Coverage.Bounds.Stop" va
lue="870.0"/>
<PARAM ID="SpatialExtent" datatype="double" name="SpatialExtent" ucd="instr.fov" unit="deg" utype="spec:Spectrum.Char.SpatialAxis.Coverage.Bounds.Extent" value="0.000491106"/>
<PARAM ID="DataModel" arraysize="13" datatype="char" name="DataModel" utype="spec:Spectrum.DataModel" value="Spectrum 1.01"/>
<PARAM ID="Publisher" arraysize="13" datatype="char" name="Publisher" ucd="meta.curation" utype="spec:Curation.Publisher" value="ESA/Gaia/DPAC"/>
<PARAM ID="Title" arraysize="8" datatype="char" name="Title" ucd="meta.title;meta.dataset" utype="spec:DataID.Title" value="Spectrum"/>
<PARAM ID="SpectralAxisUcd" arraysize="5" datatype="char" name="SpectralAxisUcd" ucd="meta.ucd" utype="spec:Spectrum.Char.SpectralAxis.Ucd" value="em.wl"/>
<PARAM ID="SpectralAxisUnit" arraysize="2" datatype="char" name="SpectralAxisUnit" ucd="meta.unit" unit="nm" utype="spec:Spectrum.Char.SpectralAxis.Unit" value="nm"/>
<PARAM ID="FluxAxisUcd" arraysize="29" datatype="char" name="FluxAxisUcd" ucd="meta.ucd" utype="spec:Spectrum.Char.FluxAxis.Ucd" value="phot.flux.density;arith.ratio"/>
<PARAM ID="FluxAxisUnit" arraysize="10" datatype="char" name="FluxAxisUnit" ucd="meta.unit" utype="spec:Spectrum.Char.FluxAxis.Unit" value="Flux value"/>
<FIELD ID="wavelength" datatype="double" name="wavelength" ucd="em.wl" unit="nm" utype="spec:Data.SpectralAxis.Value"/>
<FIELD ID="flux" datatype="float" name="flux" ucd="phot.flux;em.opt.I"/>
<FIELD ID="flux_error" datatype="float" name="flux_error" ucd="stat.error;phot.flux;em.opt.I"/>
```

```
In [12]: rvs_table=dl_product.to_table()
print("\nFirst 5 lines:")
print(rvs_table[0:5])
```

First 5 lines:

wavelength nm	flux	flux_error
846.0	1.001296	0.0036515018
846.01	1.0005543	0.0021563233
846.02	0.9992022	0.0024518226
846.03	1.0013205	0.0024563277
846.04	1.0021714	0.0020916252

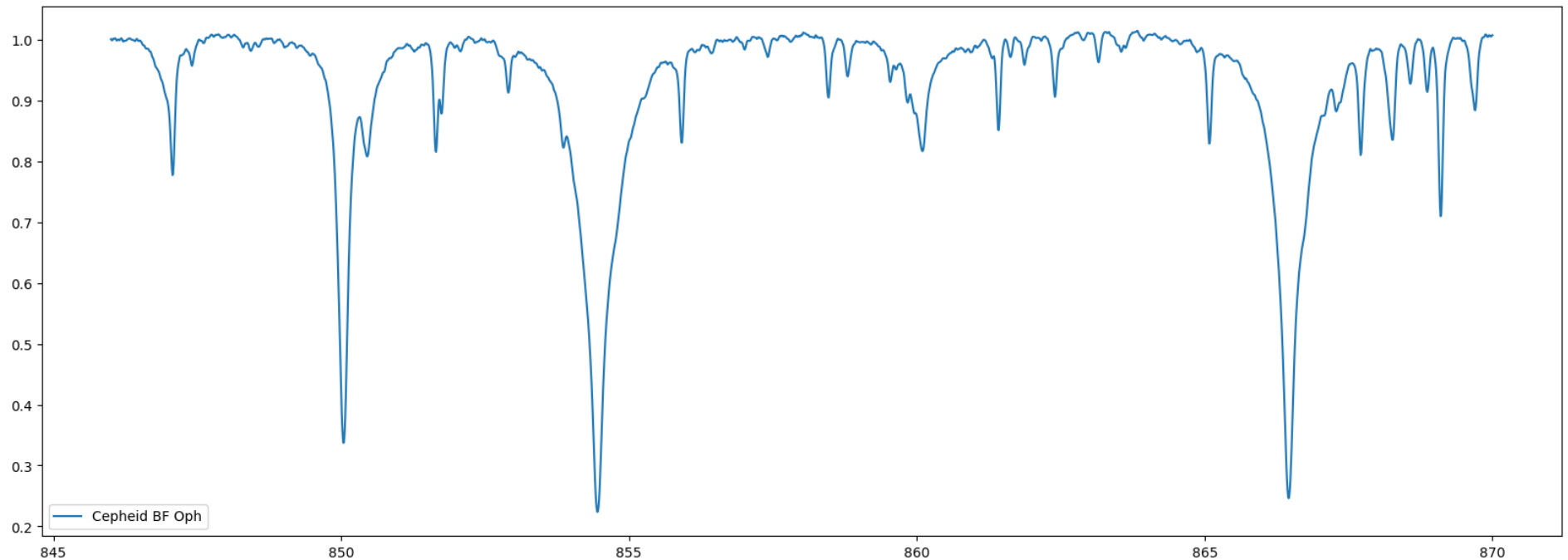
RVS Spectra

```
In [13]: RVSSourceIds = ["4111834567779557376"]
RVSSourceLabels = ["Cepheid BF 0ph"]
```

```
In [14]: rvsdatalink = Gaia.load_data(ids=RVSSourceIds, retrieval_type='RVS', data_structure = 'COMBINED', output_file = None)
rvsdatalink.keys()
```

```
Out[14]: dict_keys(['RVS-Gaia DR3 4111834567779557376.xml'])
```

```
In [15]: dl_key      = list(rvsdatalink.keys())[0]
tables    = [product.to_table() for product in rvsdatalink[dl_key]]
plt.figure(figsize=[20,7])
for rvsspectra in tables:
    plt.plot(rvsspectra['wavelength'], rvsspectra['flux'], '-')
    plt.gca().legend(RVSsourceLabels)
plt.show()
```



XP spectra

GaiaXPY calibrate function can both download the datalink XP_CONTINUOUS spectra and calibrate it to get the equivalent of a XP_SAMPLED spectra directly:

```
In [16]: XPsourceIds = ["4111834567779557376"]
calibrated_spectra, sampling = calibrate(XPsourceIds)
calibrated_spectra
```

Reading input DataFrame...

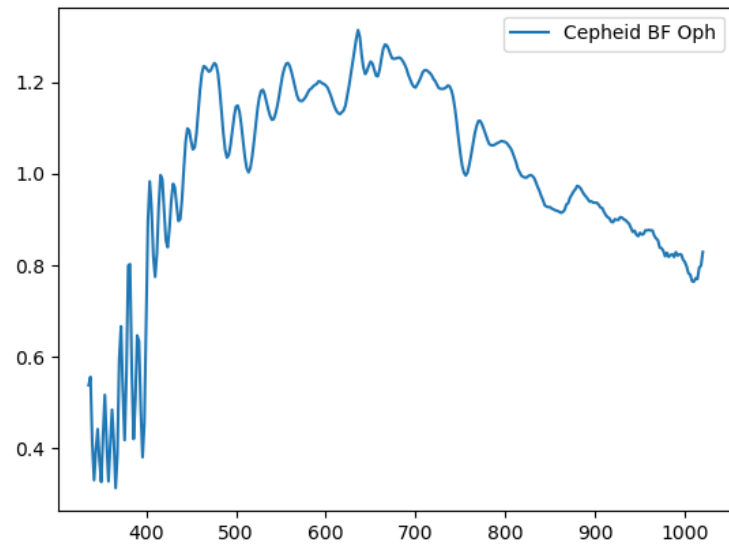
0/1 [00:00<?, ?spec/s]

Done! Output saved to path: ./output_spectra.csv

```
Out[16]:
```

	source_id	flux	flux_error
0	4111834567779557376	[1.80360298496842e-14, 1.8649371946396767e-14, ...]	[4.878989666861508e-15, 3.903498037530731e-15, ...]

```
In [17]: plt.plot(sampling,calibrated_spectra.loc[0,"flux"]/np.mean(calibrated_spectra.loc[0,"flux"]))
plt.gca().legend(["Cepheid BF Oph"])
plt.show()
```



Learn more

astroquery.gaia documentation page: <https://astroquery.readthedocs.io/en/latest/gaia/gaia.html>

GACS demo and tutorials: <https://www.cosmos.esa.int/web/gaia-users/archive/help>. In particular:

- ADQL Query Examples: <https://www.cosmos.esa.int/web/gaia-users/archive/writing-queries#ADQLQueryExamples>
- A DataLink tutorial: https://www.cosmos.esa.int/web/gaia-users/archive/use-cases#datalink_dr3_pleiades
- Tutorial to retrieve large amounts (data for more than 5000 sources) of DataLink products : https://www.cosmos.esa.int/web/gaia-users/archive/datalink-products#datalink_jntb_get_above_lim

GaiaXPY tutorials: <https://gaia-dpci.github.io/GaiaXPY-website/tutorials/>