Tutorial: Exploring Gaia data with TOPCAT and STILTS

Mark Taylor, University of Bristol, UK
`m.b.taylor@bristol.ac.uk`

TOPCAT: http://www.starlink.ac.uk/topcat/ *(version 4.10-2 or later recommended)*

STILTS: http://www.starlink.ac.uk/stilts/ *(version 3.4-1 or later recommended)*

Mailing list: `topcat-user@jiscmail.ac.uk`

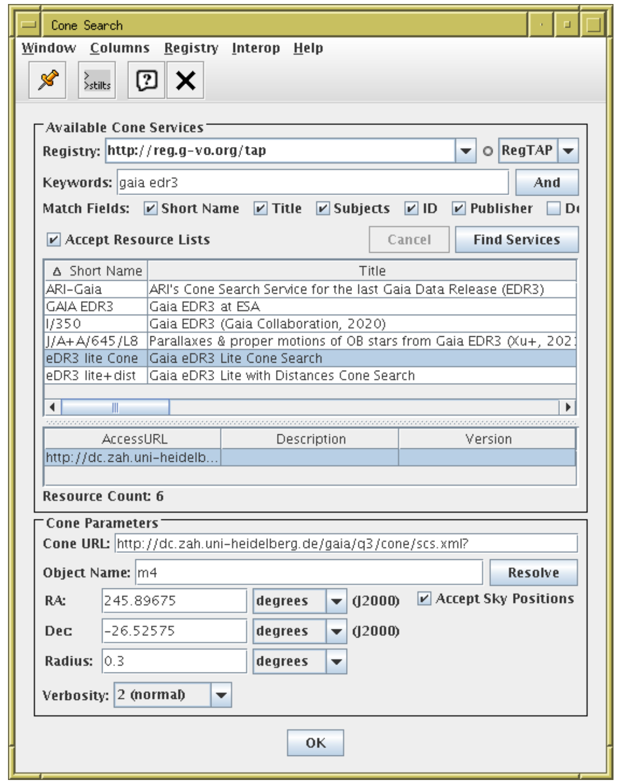
Version: 2025-12-05 `Revision efea3e4 (https://github.com/mbtaylor/tctuto)`

Contents

This tutorial uses data from Gaia Data Release 3 (DR3) [1] to lead you through some of the capabilities of TOPCAT and STILTS. It assumes you have TOPCAT, and hence STILTS, installed. For best results, you should have the manuals to hand: http://www.starlink.ac.uk/topcat/sun253/ and http://www.starlink.ac.uk/stilts/sun256/.

1 Cluster identification #1: M4 in proper motion space using Cone Search

In this example we will determine the mean parallax of the stars in the globular cluster Messier 4 (M4, or NGC 6121).



1.1 Acquire Gaia data in the M4 region

1. Start TOPCAT.

2. Open the **VO|Cone Search** window (i.e. use the **Cone Search** submenu of the **VO** menu in the main topcat window).

3. Fill in **Keywords**: "`gaia dr3`", and hit **Find Services**.

4. There are a few options, that should mostly give similar results. The one with Short Name "`DR3 lite Cone`" is a good choice. Select it by clicking on its row, and the partial URL of the service appears in the **Cone URL** field.

5. Fill in **Object Name**: "`M4`", then hit **Resolve** to fill in sky position fields.

6. **Radius**: "`0.3`" (degrees)

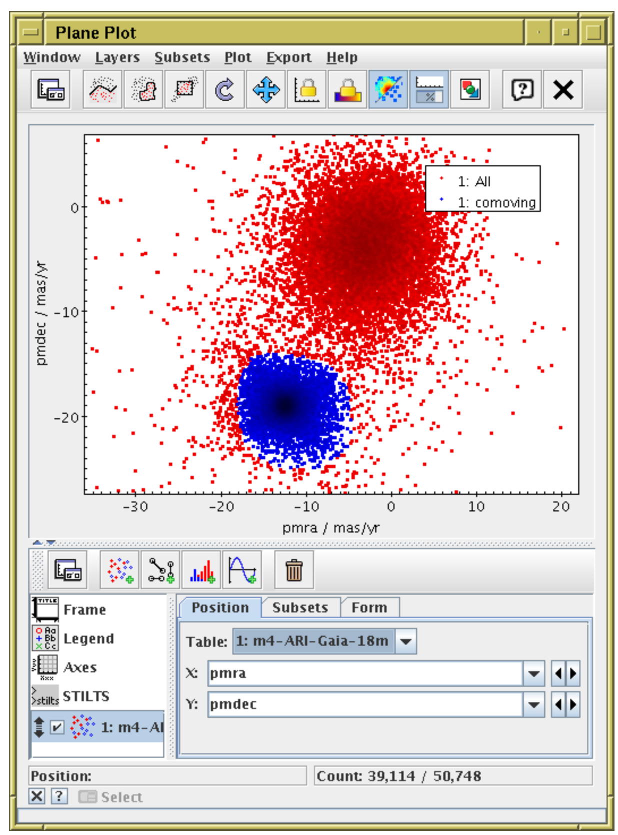
7. Hit **OK**; new table is loaded into topcat main control window with about 50 000 rows. If the download is too slow, cancel and try with **Radius**: "`0.1`", which retrieves about 20 000 rows, and maybe **Verbosity**: "`1`", which reduces the number of columns requested. Or try it with another of the listed services.

8. Examine the loaded table: look at the **Views|Table Data** and **Views|Column Info** windows (you can either go through the menus or find the right button in the toolbar)

9. Use the **Graphics|Sky Plot** window to see the positions on the sky.

10. Play with the plot. Note overdense regions are coloured darker. Use the options in the **Form** tab to change marker size, colour and shading. Practice navigation: use mouse drag and wheel (or CTRL-drag). Click the little ? button at bottom left for navigation help; note navigation details are different for different plot types. The **Plot|Rescale** button is useful to reset the scaling.

1.2 Identify comoving cluster



1. Plot sources in proper motion space: **Graphics|Plane Plot** window, **X**: "`pmra`", **Y**: "`pmdec`"

2. Note overdensity far from (0,0); use mouse to navigate

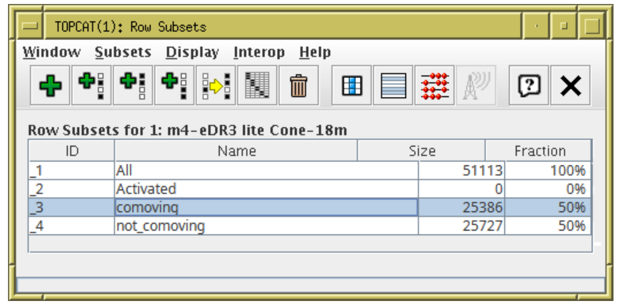
3. Graphically select this comoving cluster as new Subset: **Subsets|Draw Subset Region** button, drag mouse around cluster, hit button again

4. A **New Subset** dialogue pops up: fill in **New Subset Name**: "`comoving`", **Add Subset**

5. Look in **Subsets** tab of plot window; turn **All** and **comoving** subsets off/on

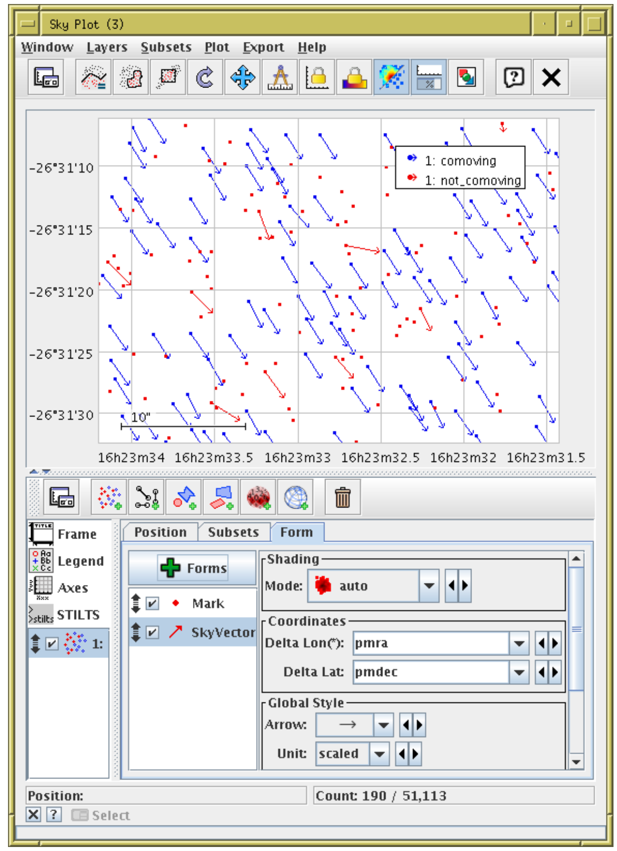
1.3 Manage Subsets

1. Open the **Views|Row Subsets** window
2. See the new **comoving** subset
3. Create an additional subset that contains the background objects, i.e. those not in the comoving set: click to select the **comoving** row, then invoke the **Subsets|Invert Subset** action.
4. A new subset **not_comoving** will appear; if you like you can rename it "**background**" by double-clicking in the **Name** field



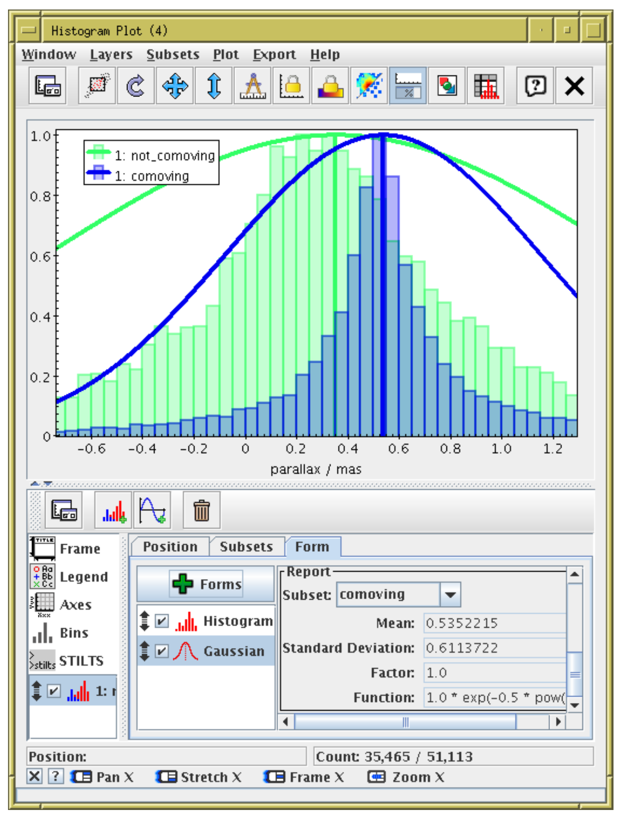
1.4 Examine cluster members

1. Go back to the **Graphics|Sky Plot** from section 1.1 (or open a new one)
2. In **Subsets** tab turn different subsets on/off using checkboxes
3. Plot the proper motions. In the **Form** tab, use the **Forms** menu and select the **Add SkyVector** item, with **Delta Lon(*)**: "`pmra`", **Delta Lat**: "`pmdec`". The arrows will initially be much too long (units of degrees); you will have to set **Unit**: "`scaled`" (auto-scaling), and optionally adjust to taste with the **Scale** slider. Zoom in so you can see some individual objects. All the cluster objects have similar proper motions, non-cluster ones have various directions, or none (no measured P.M.).



1.5 Determine parallax

1. Plot histogram of parallaxes:
   Graphics|Histogram Plot,
   X: "`parallax`"
2. In **Subsets** tab, make sure only subsets **comoving** and **not_comoving**, not **All**, are plotted
3. Normalise histograms to the same height: click on the **Bins** control in the left-hand panel, select the **General** tab, and set **Normalise** to **maximum**. Return to **histogram layer** control
4. Zoom in horizontally (mouse below X axis, use wheel/CTRL-drag) to read off the comoving peak/average parallax. Note the comoving (cluster) distribution is different from the background sample.
5. For a more accurate result, fit Gaussian to data: **Forms** menu in **Form** tab, **Gaussian** option. Then scroll to bottom of Gaussian layer description in **Form** tab, select **Subset**: "`comoving`", and read off parallax **Mean** and **S.D.**
6. Invert mean parallax get distance to cluster (1000/parallax in mas = distance in parsec).
7. To do this without plotting, you can read off the mean and S.D. for parallax in the **Views|Column Statistics** window (from the main window) for the **comoving** subset.

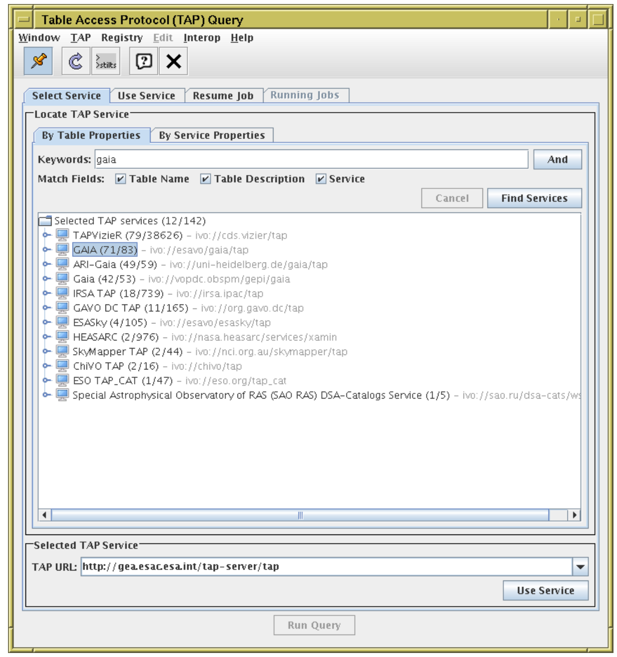


Note: careful when inverting parallaxes!
In general $r = 1/\varpi$ is *not reliable* because of errors. It's OK here because we are averaging over many measurements with smallish errors. Rule of thumb for single measurements: if $\varpi/\sigma_\varpi > 5$ it's probably OK. See Luri et al. 2018 [2] for fuller discussion.

2 Cluster identification #2: Hyades in 3-D velocity space using TAP

This example locates the Hyades open cluster in 3-dimensional velocity space, using Gaia's proper motion and radial velocity observations. We can't start this time by making a positional query (cone search), since the Hyades is very delocalised on the sky, because it's so close, so a cone would contain way too many sources. So we need to make a more sophisticated query using TAP.

2.1 Locate Gaia TAP service

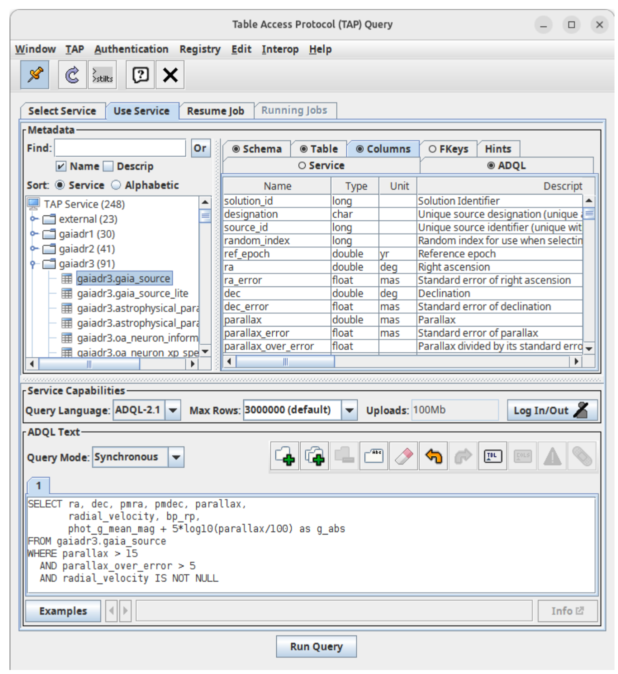


1. Open the TAP window:
   VO|Table Access Protocol (TAP) Query
2. Fill in **Keywords**: "gaia" and hit **Find Services** button
3. There are several services with Gaia data in various forms; **GAIA** (ESA) or **ARI-Gaia** (Heidelberg) are good choices. The service URL appears in the field at the bottom of the window.
4. Hit the **Use Service** button at the bottom

2.2 Explore the TAP service

Use the TAP window to explore the tables that are present and their metadata.

1. Browse the table list on the left. Click on the "handles" to expand the branches (schemas). The tables in the `gaiadr3` schema are the ones with Gaia DR3 data.
2. Select table `gaiadr3.gaia_source` and look at **Table** and **Columns** tabs, to see information about available columns.
3. Look in the **Hints** tab for a very basic ADQL cheat sheet
4. Type in to the bottom panel some very simple ADQL: "`SELECT TOP 10 ra, dec FROM gaiadr3.gaia_source`"
5. Note that syntax errors (including partial or misspelt tables/columns) get highlighted in red.
6. Hit **Run Query** to run the query; if successful a new table will be loaded



2.3 Acquire astrometric data

In the TAP window, open a new query tab using the **Add Tab** button above the ADQL entry panel and run this query:

```
SELECT ra, dec, pmra, pmdec, parallax,
       radial_velocity, bp_rp,
       phot_g_mean_mag + 5*log10(parallax/100) as g_abs
FROM gaiadr3.gaia_source
WHERE parallax > 15
AND parallax_over_error > 5
AND radial_velocity IS NOT NULL
```

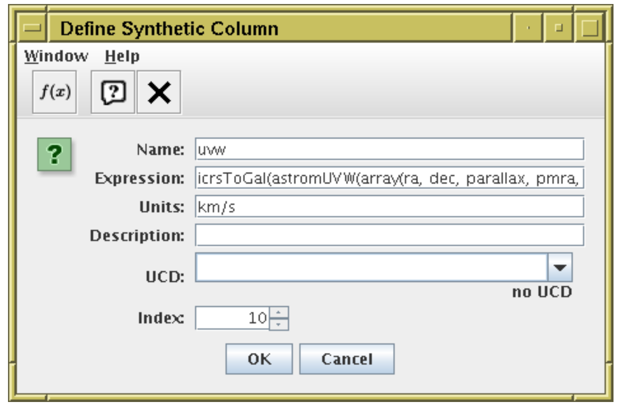
The result should contain about 60 000 rows.

The query is for all the nearby sources (nominally within $1000/15 \approx 66$ parsec) with observed radial velocities (only about 34 million out of 2 billion DR3 sources have RV) and good determinations of parallax. The fact that parallax error is $\leq$20% means that it's OK to invert parallax to calculate distance and absolute magnitude. We are retrieving all the basic astrometric parameters (enough for the 6-d phase space) and some photometry.

The Hyades should be in there somewhere. Can we find them?

2.4 Calculate 3-d velocity components

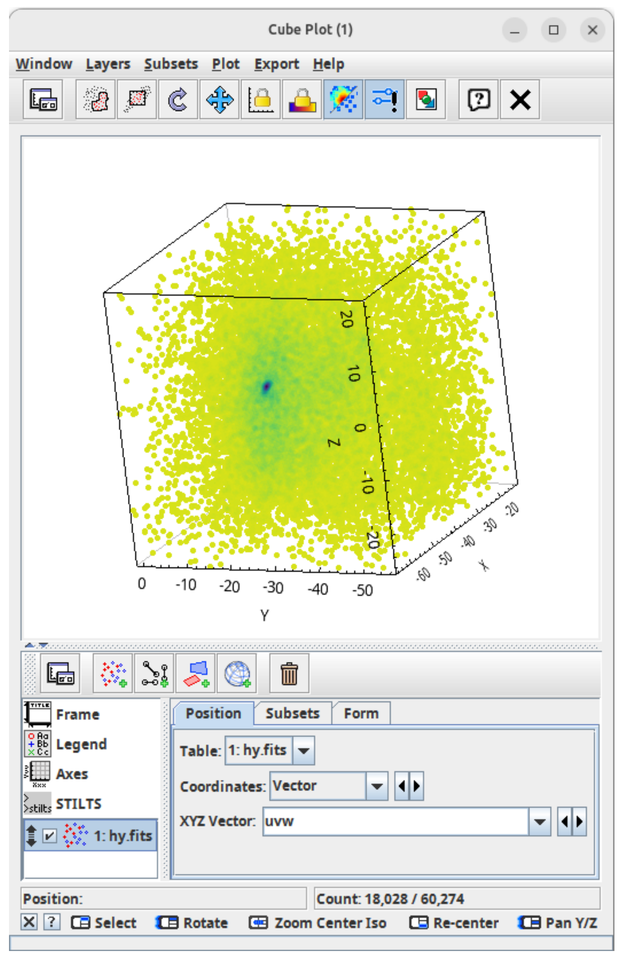
We have the astrometric quantities measured by Gaia, which contain full phase space information, but need transformations to yield Cartesian position/velocity coordinates. TOPCAT's *expression language* can help.



1. Open the Help|Available Functions browser to see what functions TOPCAT provides (they are listed in the manual too).
2. Look under the **Gaia** entry to see astrometry-specific items
3. We will use the `astromUVW(astrom6)` and maybe `icrsToGal` and `astromXYZ` functions. The **Examples** items in the function documentation are useful; for use with the `gaia_source` catalogue, you can often just copy and paste
4. Open the **Views|Column Info** window and choose the **Columns|New Synthetic Column** action
5. Create a new column giving Cartesian velocity components: **Name**: "`uvw`", **Units**: "`km/s`" and for **Expression**:

   ```
   astromUVW(array(ra, dec, parallax, pmra, pmdec, radial_velocity))
   ```

   That calculates velocities along ICRS axes. If you want it in Galactic coordinates, wrap the whole expression in the `icrsToGal(...)` function.
6. Look at the new column in the **Views|Table Data** window (scroll all the way to the right). It is a 3-element array; you can access the array elements using expressions `uvw[0]`, `uvw[1]`, `uvw[2]`

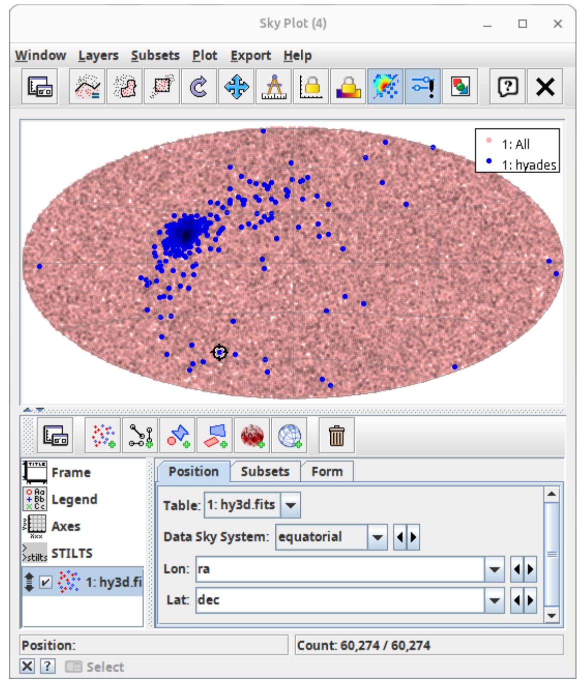


2.5 Identify Hyades graphically in 3-d velocity space

1. Plot points in 3-d space: **Graphics|Cube Plot**, **Coordinates**: "`Vector`", **XYZ Vector**: "`uvw`".
2. Select **Mode**: "`density`" in the **Form** tab. You can change the colour map to taste using the **Shader** selector.
3. Now, navigate through the cube to find an overdensity. This takes a bit of practice, but it's fun once you work out how. Click the little ? button at bottom left for navigation help; the most useful actions are drag to rotate, mouse wheel (2-fingered drag on some trackpads) to zoom, and right click (or CTRL-click) on a dense region to recenter.
4. Navigate so only the objects in the overdense region (about 200 of them? see the count at the bottom right of the window) are visible inside the wireframe — these should be the Hyades, which all have similar 3D velocities.
5. Use **Subsets|Subset From Visible** action, **Name**: "`hyades`", **Add Subset**.

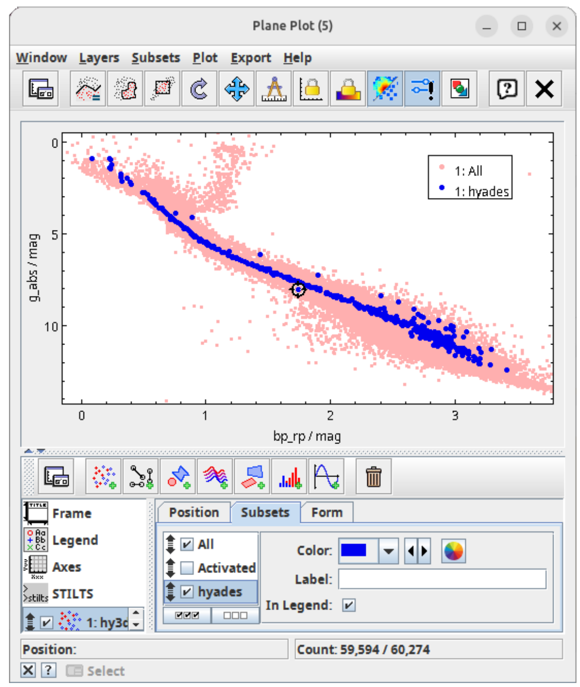
2.6 View positions

1. Go back and plot this subset on the sky (**Graphics|Sky Plot**)
2. Use the **Subsets** tab to make sure that **hyades** and background objects (**All**) are plotted in different colours. You might want to set **Shading Mode**: "`flat`" and fiddle with marker size and shape for clarity (in the **Form** tab).
3. To get an all-sky view, use the **Axes** control, **Projection** tab, and change the **Projection** selector from "**sin**" to "**aitoff**"
4. If you like, you can plot it in 3-d space as well: **Graphics|Sphere Plot**, **Lon**: "`ra`", **Lat**: "`dec`", **Radius**: "`1000./parallax`"
5. Leave any plot windows open for later.



2.7 Colour-magnitude diagram

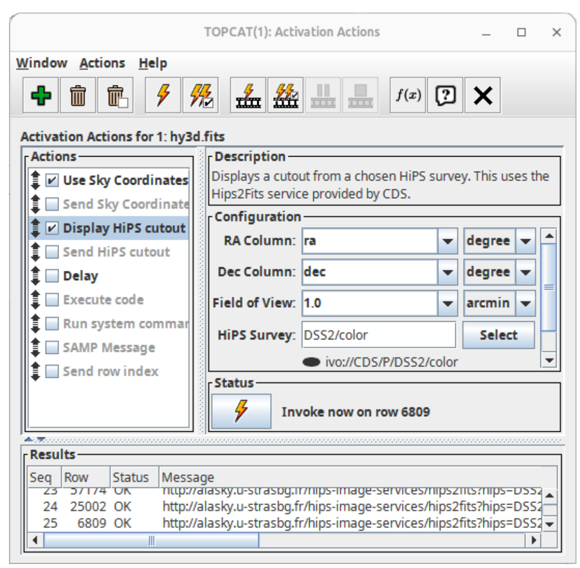
1. Plot a colour-magnitude diagram: **Graphics|Plane Plot**, **X**: "`bp_rp`", **Y**: "`g_abs`".
2. Use the **Axes** control, **Coords** tab, **Y Flip** checkbox to flip it the right way round.
3. Use the **Subsets** tab to make sure that **hyades** and background objects (**All**) are plotted in different colours. Hyades should (mostly) sit on a nice tight main sequence!



2.8 Investigate outliers

There are some sources sitting off the main sequence. We will try to find more information about these.

1. In the **Subsets** tab hide the **All** subset leaving just the **hyades** visible.
2. Now click on an outlier. A little round cursor will appear on it. The same cursor will identify that source in any other plots it appears in, e.g. the sky plot from section 2.6. In some cases you can see by sky position that the source is a non-member.
3. Open the **Views|Data Window** and see that the relevant row is highlighted when you click on a plotted point. This works the other way round too, as long as the point is visible.
4. You can make all sorts of other things happen when you click on a plotted point (or row in the table window) as well. Open the **Views|Activation Actions** window to see how these are controlled.
5. Check the checkbox for **Display HiPS Cutout** and click again on a plotted point — a little window should pop up with a sky image centered on that source. Click on the **Display HiPS Cutout** item to configure it better, e.g. **Field of View**: "`1 arcmin`", or select a different **HiPS Survey** to supply the imagery, then click again on a plotted point to see it working.
6. If Aladin Desktop is running, try the **Send Sky Coordinates** option too which uses a communication protocol called SAMP; Aladin should re-center on the source you've clicked on. Aladin Lite can do the same thing too, but a bit of extra setup is required to make it listen to SAMP.



3 Match Gaia and HST observations

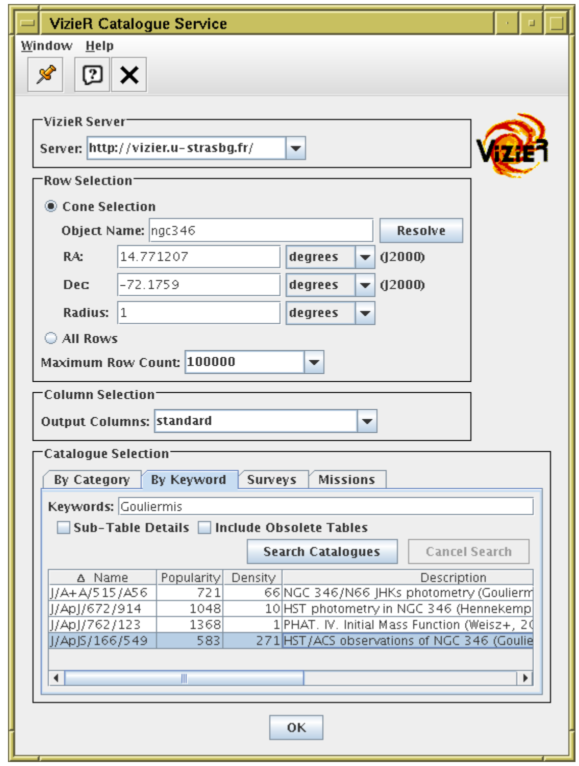
In this example we have a local catalogue from a publication by Gouliermis et al. 2006 [4], available in VizieR as J/ApJS/166/549. This contains about 100 000 sources observed by the ACS instrument on the Hubble Space Telescope at epoch ≈ J2004.6 of stars in NGC346, a cluster in the Small Magellanic Cloud. We match these positions with positions in the main Gaia DR3 catalogue at J2016.0.

3.1 Acquire HST observations

There are various ways to do this, but here we will use TOPCAT's VizieR dialogue window, which talks directly to the VizieR catalogue service.

1. Open **VO|VizieR Catalogue Service** window
2. **Object Name**: "`ngc346`", and **Resolve** to fill in **RA** and **Dec**
3. **Radius**: "`1`" (degrees)
4. **Maximum Row Count**: "`100000`" (or some large number)
5. Catalogue selection panel: **By Keyword** tab
6. Enter in **Keywords**: "`Gouliermis`"
7. Select "`J/ApJS/166/549`"
8. Hit the **OK** button at the bottom. A new table with 99 079 rows should be loaded

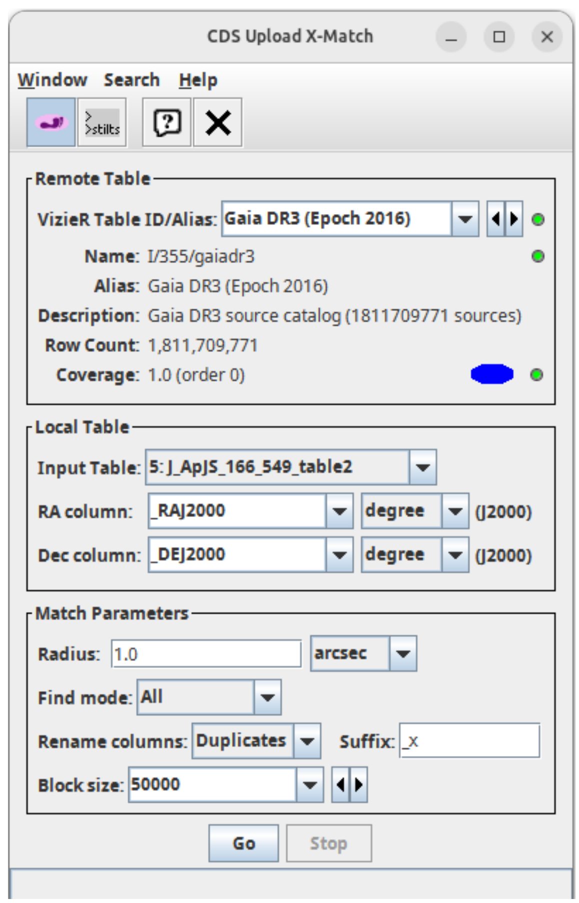
Alternatively, you could download the table from the VizieR web page.



3.2 Crossmatch with Gaia

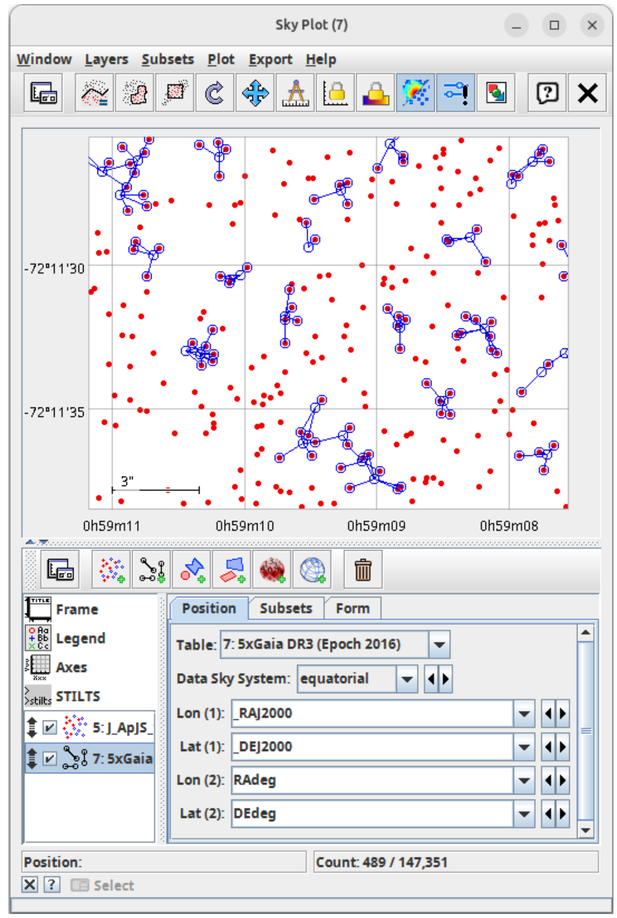
Now we want to find associations of the HST objects with sources from Gaia DR3. Use the CDS X-Match service from TOPCAT. This uploads a local table to the CDS X-Match service, where the match is made against the Gaia DR3 catalogue (or any other catalogue in VizieR). The resulting matched catalogue is then received as a new table in TOPCAT.

1. Open the **VO|CDS Upload X-Match** window
2. Fill in the fields:
   VizieR Table ID/Alias: "`Gaia DR3 (Epoch 2016)`"
   Input Table: "`J_ApJS_166_549_table2`"
   (or whatever the HST table is called)
   RA column: "`_RAJ000`", **Dec column**: "`_DEJ2000`"
   (should be filled in automatically)
   Radius: "`1`" (arcsec)
   Find Mode: "`All`" ⟵ *Important!*
3. Hit **Go**; within a few seconds, it should inform you that a new table has been loaded, with about 24 000 rows.
4. Look at the columns of the new table (all HST followed by all Gaia) in the **Views|Column Info** window.



3.3 Visualise the crossmatch

1. Open a **Graphics|Sky Plot** window
2. Plot the HST observations: **Table**: "J_ApJS_166_549_table2", **Lon**: "_RAJ2000", **Lat**: "_DEJ2000"
3. Overplot the actual matches. Add a new Pair layer: **Layers|Add Pair Control** and fill in:
   Table: "4xGAIA DR3" (or whatever the xmatch result table is called) and both sets of coordinates:
   Lon(1): "_RAJ2000", **Lat(1)**: "_DEJ2000" (HST)
   Lon(2): "RAdeg", **Lat(2)**: "DEdeg" (Gaia)
4. Zoom in to look at the associations. There are too many! What is this plot telling you?
5. You can fiddle around with the **Form** tab to make the plot clearer, e.g. add a **Mark2** layer; change marker size, shape or colour.

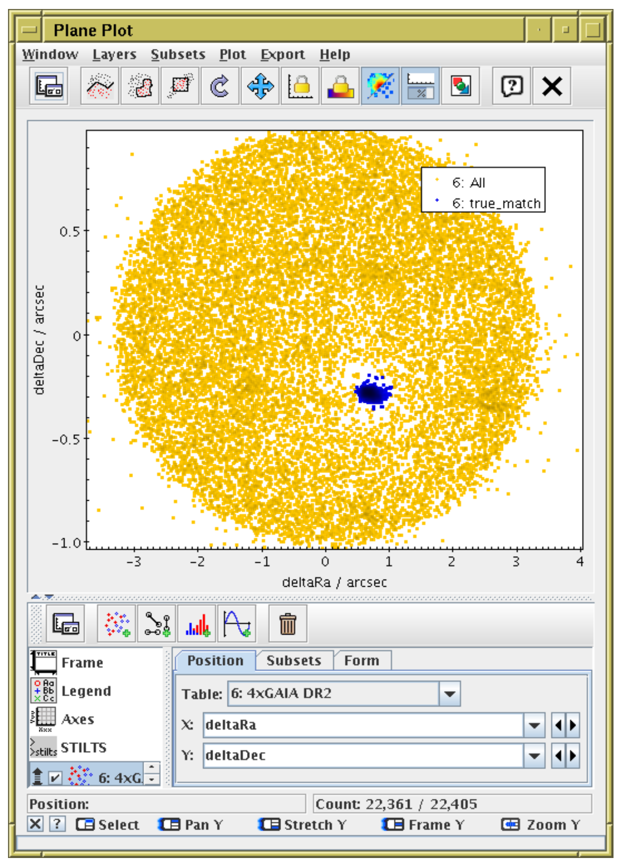


Visualising the results of a crossmatch is very often a good idea, unless you're pretty sure what you're going to get. Here, you can see it was crucial to understand the results: most of these matches are spurious, because there is a high density of HST sources.

Leave this plot open, we will come back to it after the next section.

3.4 Investigate and identify matches

1. Add new columns giving RA/Dec discrepancies between HST and Gaia positions:
   Open **Views|Column Info** window,
   then define new columns using **Columns|New Synthetic Column**:
   Name: "deltaRa", **Expression**:
   "3600*(RAdeg - _RAJ2000)", **Units**: "arcsec"
   Name: "deltaDec", **Expression**:
   "3600*(DEdeg - _DEJ2000)", **Units**: "arcsec"
2. Use **Graphics|Plane Plot** window,
   plot **X**: "deltaRa", **Y**: "deltaDec"
3. Identify overdense region, select by drawing a blob as in section 1.2, define new subset **true_match**.
4. Go back to the sky associations plot from the previous section, and use the **Subsets** tab to visualise which are the true matches. Why do you think this offset is not zero?
5. Make a colour-colour diagram combining HST and Gaia photometry:
   Use **Graphics|Plane Plot** window, plot **X**: "Vmag-Imag", **Y**: "BPmag-RPmag", display **true_match** subset only.



3.5 Alternative crossmatch: use local files

The crossmatch in section 3.2 was done by sending a local file to an external service. This is often an efficient way to do it, but there are other options. Here, we will do the same crossmatch by operating on two local files with positions covering the same sky region.

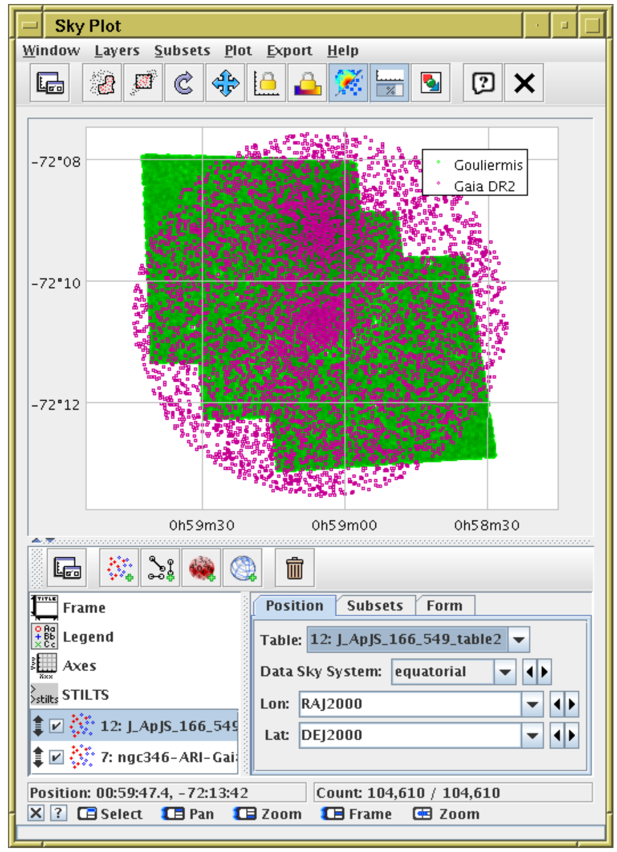
1. First, retrieve Gaia data in the region of interest. Use the **VO|Cone Search** window to a Gaia service as in section 1.1, but this time fill in **Object Name**: "`ngc346`", **Radius**: "`0.05`" (degrees).

2. Plot the HST (from VizieR) and Gaia (from Cone Search) datasets on the sky: open the **Graphics|Sky Plot** window, fill in RA and Dec as **Lat** and **Lon** for one of the tables, then use the **Layers|Add Position Control** action to overplot the same thing for the other dataset.

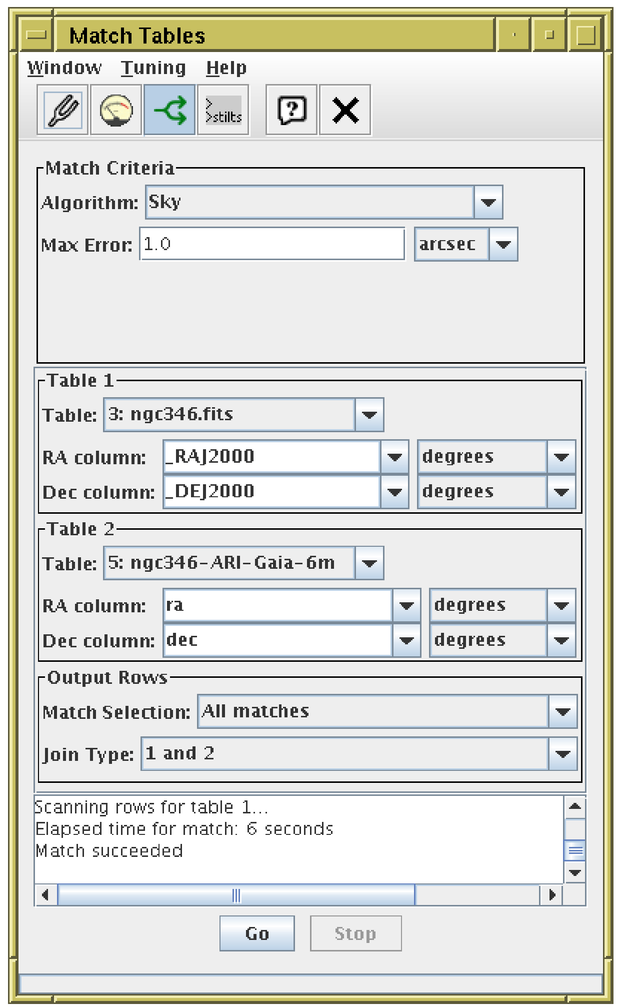
3. Open the **Joins|Pair Match** window from the main control window. Default **Match Criteria** (Sky, 1 arcsec) are OK in this case. Fill in the **Table 1** and **Table 2** details for the HST and Gaia tables. Entry **Match Selection**: "`All matches`".

4. Hit **Go** and wait a few seconds for the match to complete.

5. When complete, a popup window will tell you, and offer to **Plot Result**. If you select this option, you will see a plot like the one from section 3.3, except that the unmatched Gaia sources are also plotted (which can sometimes be useful information).



Other matching options are available in the local match windows, including identifying objects that don't match, matching internally within a table, matching between three or more tables, etc. Note that unlike most things in TOPCAT, crossmatching can take up significant amounts of memory, so matching multi-million-row tables can sometimes grind to a halt or fail.



Bonus

- Use the **Graphics|Histogram Plot** (as in section 1.5) to find the mean values of δ RA, δ Dec for the true matches. What do these values tell you?

- The matches done here are between Gaia positions at J2016.0 and HST positions taken at approximately J2004.6. Use the `epochProp` function in the expression language to do the match with the positions as Gaia proper motions predict for J2004.6. Does it make much difference?

4 M4 in proper motion space using STILTS

In this section we will re-do the determination of the distance to M4 from section 1, but this time from the command line, though we will use some of the results of the earlier GUI activity. Using TOPCAT interactively to work out how to write STILTS scripts for later batch use like this is a common pattern of work.

4.1 Get STILTS running

1. Unlike TOPCAT, you can't bluff your way through STILTS by pointing'n'clicking; you'll need some documentation. Find the manual `http://www.starlink.ac.uk/stilts/sun256/` or maybe just google for "stilts" and go to the **Documentation** section.

2. Make sure you have it installed. Running "`stilts`" on its own may work; if not then "`java -jar stilts.jar ...`" should do. If you have topcat, you can run "`java -jar topcat-full.jar -stilts ...`" instead. More convenient (on Un*x), download the `stilts` script into the same directory as `topcat-full.jar` (or `stilts.jar`). Then the directory needs to be on your path, or you can use the full pathname or set up an alias. We will write just "`stilts`" from now on.

3. Run a command:

   ```
   stilts calc expression="1+2"
   ```

   should print out "3"

4. Find the documentation for the `calc` command in the manual Appendix B. Look at the **Usage** and **Examples** subsections.

5. Get command-line help by running "`stilts calc help`" and "`stilts calc help=expression`".

4.2 Acquire data from Cone Search service

1. Investigate the STILTS `cone` command; see the `cone` documentation in Appendix B of the manual or run

   ```
   stilts cone help
   ```

2. Find out the cone search parameters we used in 1.1. This includes the service URL (from the **Cone URL** field) and the central RA (longitude) and Dec (latitude) parameters of M4, as well as the search radius.

3. Putting those together, we need to run

   ```
   stilts cone serviceurl="http://dc.g-vo.org/gaia/q3/cone/scs.xml?"
               lon=245.89675 lat=-26.52575 radius=0.3 out=m4.fits
   ```

   to give the output file `m4.fits` — the output format has been determined by file extension (you can write e.g. `m4.vot` if you prefer a different format).

Note: some values containing whitespace or other strange characters need to be quoted to prevent the shell expanding them. This quoting can get a bit messy. A good tip is to avoid spaces where you don't need them, e.g. in expressions. I *believe* the commands as written in this section will work in common Un*x shells, also Windows command prompt and PowerShell.

4.3 Manipulate the downloaded table using `tpipe`

We will perform some processing on the downloaded table using the `tpipe` command. This reads an input table, optionally performs operations on it, and writes it to output, either to a file (in the same or different format) or some other destination. It works like a Unix pipeline.

The operations are defined by adding *filters* such as "`select`" (retain only some rows) or "`addcol`" (add a new column). Filters are specified by adding using zero or more "`cmd=`" parameters on the command line, and documented in Section 6.1: "Processing Filters" of the STILTS manual.

The default *output mode* is write to a file, but other options such as "`meta`" (display column metadata), "`count`" (count rows) and "`stats`" (calculate mean, st.dev etc) are also available. Select non-default output modes using the "`omode=`" parameter on the command line, documented in Section 6.4: "Output Modes" of the manual.

1. Count the rows in the query result table:

   ```
   stilts tpipe in=m4.fits omode=count
   ```

2. See what columns the query result table has:

```
stilts tpipe in=m4.fits omode=meta
```

or for a more manageable output (customised metadata; the “meta” filter turns the table into a table with one row per input table column):

```
stilts tpipe in=m4.fits cmd="meta name units class"
```

3. For convenience, write a new table with only a few of the columns. The `keepcols` filter produces a table with only the listed columns (the list is space-separated and needs to be quoted). The `clearparams` filter removes per-table metadata which tends to clutter up the screen output.

```
stilts tpipe in=m4.fits cmd="keepcols 'ra dec pmra pmdec parallax'" cmd='clearparams *'
             out=m4mini.fits
```

You can then examine this new file `m4mini.fits` by loading it into TOPCAT (from the Load Window or by running “`topcat m4mini.fits`” from the command line), or by using other STILTS commands, for instance:

```
stilts tpipe in=m4mini.fits
stilts tpipe in=m4mini.fits omode=gui
```

4. Calculate statistics using the `stats` output mode:

```
stilts tpipe in=m4mini.fits omode=stats
```

5. View a few rows of the table. Then using the `head` filter keeps only the first few rows:

```
stilts tpipe in=m4mini.fits cmd="head 10"
```

You can select rows using the `select` filter, and combine filters by stringing them together on the command line. Try both:

```
stilts tpipe in=m4mini.fits cmd="head 10" cmd="select parallax>0"
stilts tpipe in=m4mini.fits cmd="select parallax>0" cmd="head 10"
```

Why are they different?

6. Add a column calculating estimated distance:

```
stilts tpipe in=m4mini.fits cmd="addcol dist_est 1000/parallax" cmd="head 10"
```

7. Write a new table containing only those rows in the comoving region of proper motion space. Looking at the proper motion plot from section 1.2 you can estimate the expression by eye as being approximately a circle radius 2 centered on (-12.6,-18.9). The `select` filter includes only rows for which the given expression is true, and the `hypot(a,b)` function is short for `sqrt(x*x,y*y)`.

```
stilts tpipe in=m4mini.fits cmd="select 'hypot(pmra+12.6,pmdec+18.9)<2'" out=cluster.fits
```

In fact TOPCAT can produce these geometric expressions for you: you can use the **Subsets|Draw Algebraic Subset** (Ellipse) action in the plot window to generate them graphically.

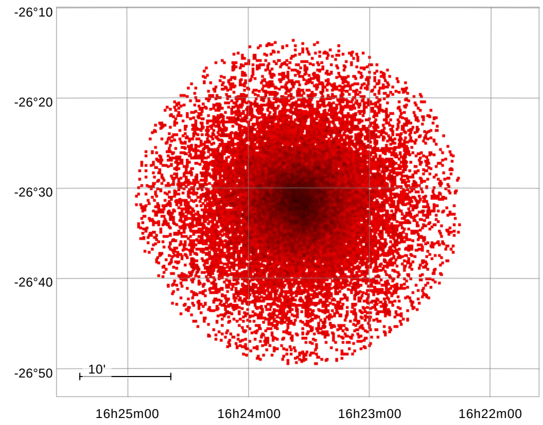
8. Calculate the mean and S.D. of cluster parallax using the previous selection:

```
stilts tpipe in=cluster.fits cmd="keepcols 'parallax'" omode=stats
```

or do it all in one go:

```
stilts tpipe in=m4.fits cmd="select 'hypot(pmra+12.6,pmdec+18.9)<2'"
                        cmd="keepcols 'parallax'"
                        cmd="stats name mean stdev"
```

4.4 Plot using STILTS



1. Plot the cluster on the screen:

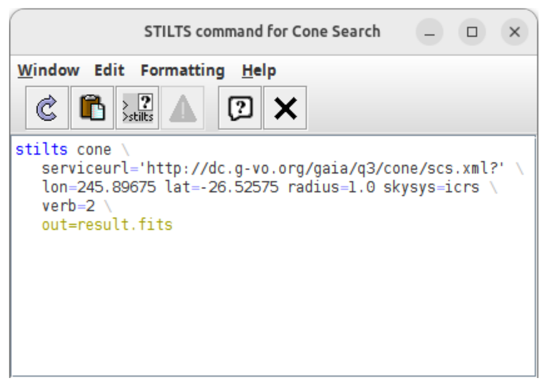
   ```
   stilts plot2sky in=cluster.fits layer1=mark lon=ra lat=dec
   ```

   Note that this plot is interactive — you can zoom and drag it around as in a TOPCAT plot window.

2. Create a plot graphics file. Just add an output parameter like "`out=plot.png`" or "`out=plot.pdf`" to the above `plot2sky` command to write static output files.

4.5 Get help from TOPCAT

STILTS has a steeper learning curve than TOPCAT, and some of the commands, especially plotting commands, can get quite complicated. In many cases, TOPCAT can tell you the STILTS command that is equivalent to what you see in the GUI. Look out for the **STILTS** control in the control stack bottom left of the plot windows, and in the tool bars for crossmatch and VO-related windows.



1. Set up the Cone Search window like in Section 1.1.
2. Click the **STILTS** button in the toolbar. This will pop up a window with the STILTS command that will perform that cone search.
3. Note that changing selections and controls in the Cone Search window changes the settings of the displayed STILTS command.
4. Copy and paste the text into a terminal and execute it (you can use the **Edit|Copy** button or do it by hand) and it should create a new file with the cone search result.
5. Click the **Help|Help for STILTS Command** button, and it should open the relevant page from the manual in your web browser.
6. Use the information to fiddle with the parameters of the command — change the sky position or search radius and execute it again.

4.6 … and more

There is *lots* more that `tpipe`, and STILTS in general, can do. Explore the manual! Or try to replicate the other exercises using STILTS.

5 Local Herzsprung-Russell Diagram

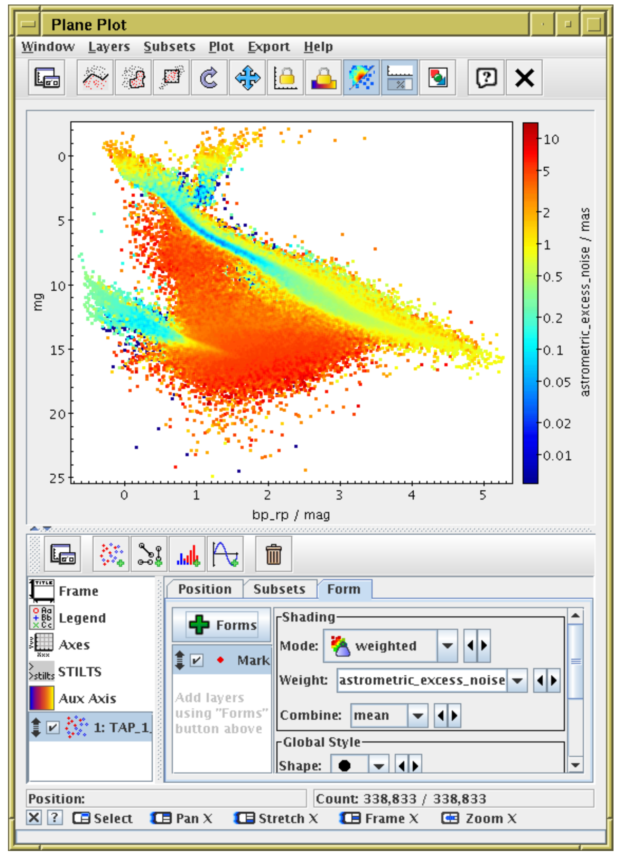
In this example we will use a TAP query to download all the nearby Gaia sources with good astrometry and photometry, and calculate their absolute magnitudes to construct an HR diagram, performing a couple of cleaning operations to improve the data. This procedure loosely follows Appendix C of the Gaia DR2 astrometry paper Lindegren et al. 2018 [3].

5.1 Acquire data from TAP service

1. Open the TAP window **VO|Table Access Protocol (TAP) Query**
2. Select one of the Gaia services (probably the ESA one) and **Use Service**
3. Choose **Mode**: "`Asynchronous`" (just above the ADQL text entry panel). This query may take a minute or two, so a synchronous query might time out.
4. Execute the following query:

```
SELECT ra, dec, parallax, phot_g_mean_mag, bp_rp,
       astrometric_excess_noise,
       phot_bp_rp_excess_factor
FROM gaiadr3.gaia_source
WHERE parallax > 10
  AND parallax_over_error > 10
  AND phot_bp_mean_flux_over_error > 10
  AND phot_rp_mean_flux_over_error > 10
```

   You should get a table with 306 517 sources; they are nominally within 100 pc, and have ϖ (parallax), BP and RP with small errors. In particular, the parallax error is small enough that ϖ^{-1} is a reasonable estimate of distance.



5.2 Plot HRD

1. Add a new column calculating absolute G magnitude, using parallax: **Columns|New Synthetic Column** in **Column Info** window:
   Name: "`g_abs`",
   Expression: "`phot_g_mean_mag + 5*log10(parallax/100)`",
   Units: "`mag`"
2. Make a **Graphics|Plane Plot**, with **X**: "`bp_rp`" ($BP - RP$ colour), **Y**: "`g_abs`" (absolute G magnitude). Use the **Axes** control, **Coords** tab, **Y Flip** checkbox to flip it the right way round. Structure visible, but lots of interlopers.
3. Play around with different **Shading Modes** in **Form** tab.
4. Colour points using the other columns using modes **Aux**, **Weighted**. What's the difference between the two?

5.3 Exclude astrometrically suspect sources

1. Try different weighting/aux columns to see which one can be used to exclude points in the unwanted region between the main sequence and white dwarf branch.

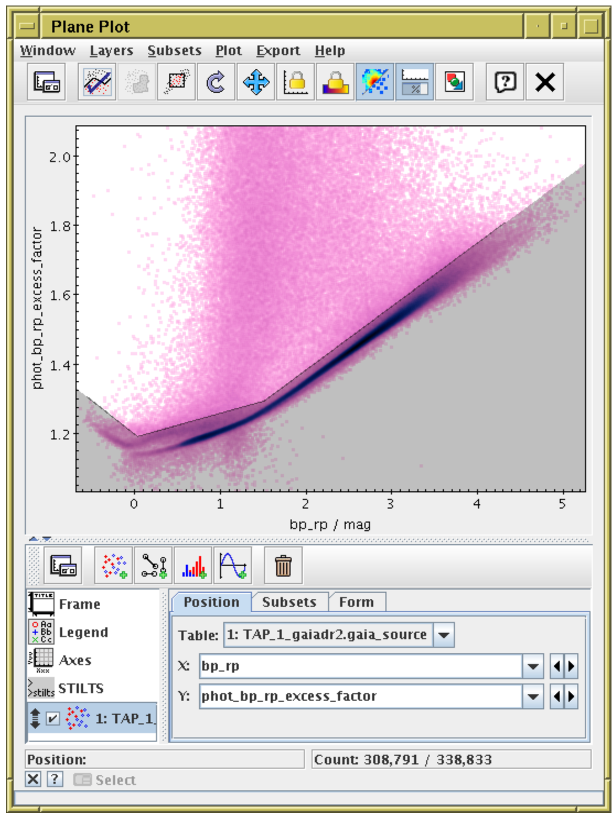
2. Experiment with the colour map settings (**Aux Axis** control) to find a suitable threshold.

3. Create a subset using an algebraic expression that excludes the spurious points: go to the **Views|Row Subsets** window and use the **Subsets|New Subset** action: **Subset Name**: "astrom_ok", **Expression**: "astrometric_excess_noise < 1"

4. Go back to the plot, and make sure only the astrom_ok subset is plotted (**Subsets** tab). Now there are fewer spurious sources.

The astrometric_excess_noise column characterises the goodness of fit of the astrometric solution to the observations. But you didn't need to know that to improve the data like this.

5.4 Exclude photometrically suspect sources



1. Open a new **Graphics|Plane Plot**, with **X**: "bp_rp" ($BP - RP$ colour), **Y**: "phot_bp_rp_excess_factor".

The quantity on the Y axis is some measure of photometric reliability. High values are bad, but how high is colour-dependent. We will define a region in this space to exclude the unusually high values. This time we will interactively draw a polygon rather than a blob.

2. Click the **Subsets|Draw Subset Polygon** action

3. Select **Point inclusion mode**: "BELOW" in the popup

4. Click on a few points above the overdense region until the shaded area roughly covers it.

5. When you're done, click on the button again.

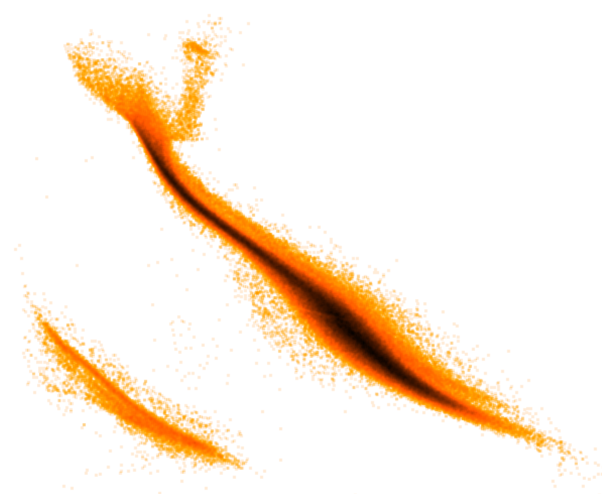
6. Fill in the **Subset Name** field in the popup (e.g. photom_ok) and hit **OK**

7. Go to the **Views|Row Subsets** window, where you can see the new subset alongside astrom_ok.

8. Create a new subset combining the two: **Subsets|New Subset** action: **Subset Name**: "ok" **Expression**: "astrom_ok && photom_ok"

9. Go back to the plot, and make sure only the ok subset is plotted (**Subsets** tab). Now there are fewer spurious sources.

5.5 Explore the HRD



The photometry and astrometry in Gaia DR2 is so good that plotting a Herzsprung-Russell Diagram by following the steps above gives a lot of astrophysical information. Identify the different populations with the help of the zoom and shading options in TOPCAT: main sequence, giant branches, the double-stranded white dwarfs sequence representing the split between helium and hydrogen burning, and if you look closely a notch in the main sequence near absolute G magnitude of 10 (Jao et al 2018 [5]).

Bonus

Go back and reproduce all the other exercises using STILTS!

References

[1] Gaia Collaboration et al., "Gaia Data Release 3: Summary of the contents and survey properties", Astronomy and Astrophysics `https://doi.org/10.1051/0004-6361/202243940` (2023)

[2] X.Luri et al., "Gaia DR2: Using Gaia parallaxes", Astronomy and Astrophysics *616*, A9 (2018), `2018A&A...616A...9L`

[3] L.Lindegren et al., "Gaia DR2: The astrometric solution", Astronomy and Astrophysics *616*, A2 (2018), `2018A&A...616A...2L`

[4] D.A.Gouliermis, A.E.Dolphin, W.Brandner and Th.Henning, "The Star-forming Region NGC 346 in the Small Magellanic Cloud with Hubble Space Telescope ACS Observations. I. Photometry", ApJS, 166 p.549 `2006ApJS..166..549G`

[5] W.-C.Jao, T.J.Henry, D.R.Gies, N.C.Hambly, "A Gap in the Lower Main Sequence Revealed by Gaia Data Release 2", ApJ Letts, 861, L11 (2018), `2018ApJ...861L..11J`

[6] M.B.Taylor, "TOPCAT & STIL: Starlink Table/VOTable Processing Software", ASP Conf. Ser. 347, 29 (2005), `2005ASPC..347...29T`

[7] M.B.Taylor, "STILTS — A Package for Command-Line Processing of Tabular Data", ASP Conf.Ser. 351, 666 (2006), `2006ASPC..351..666T`

Acknowledgements

This work has made use of data from the European Space Agency (ESA) mission *Gaia* (`https://www.cosmos.esa.int/gaia`), processed by the *Gaia* Data Processing and Analysis Consortium (DPAC, `https://www.cosmos.esa.int/web/gaia/dpac/consortium`). Funding for the DPAC has been provided by national institutions, in particular the institutions participating in the *Gaia* Multilateral Agreement.

If you use TOPCAT or STILTS in your work, please consider citing [6] or [7].