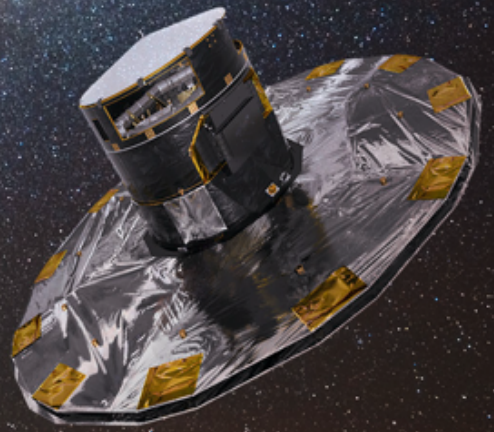
Gaia Early Data Release 3
UK Gaia science platform

Dave Morris
Gaia DPAC: Edinburgh / Cambridge / Barcelona / ESAC



National Astronomy Meeting
19th July 2021

The problem

- Existing archive systems are great for
  - Examination of (relatively) small subsets of data (e.g. spatially limited)
  - Performing very limited server-side manipulation of primitive column types
  - …
- Large scale analysis limited to
  - Pre-defined, pre-baked aspects (e.g. visualisations) OR
  - Download the whole catalogue (~ 1TB at EDR3)
- Data volume set to grow significantly over the next releases
  - DR3: 10s of TB
  - DR4+: 100s of TB+
- Complexity will increase too
  - Time-resolved data, spectra, more advances data products

Large-scale analyses through relational systems are impossible, and wholesale download will be no longer an option ...

A simple example: mean angular motions over the sky

```
SELECT (source_id / 1407374883553280) AS hpx6, COUNT(*) AS n,
       AVG(pmra) AS avg_pm_ra, AVG(pmdec) AS avg_pm_dec
FROM gaiaedr3.gaia_source
GROUP BY hpx6
```

- Existing TAP services built on relational technology
  - Limited set of aggregate functions, e.g. no robust estimation
  - No provision for user-defined functions
  - Execution time > 1hr 30min
- In reality wish to do much more than basic statistical aggregates, e.g.
  - User-defined statistical aggregates
  - Higher order statistics
  - Machine Learning
  - …

Gaia science platforms

Gaia DPAC have been investigating Apache Spark for a “big data” platform-as-a-service to provide

- Scale-out data handling
- High performance distributed computing
- Familiar web notebook environment
- Familiar APIs in Python

in an end-user “code-to-data” service: the Gaia Data Analysis Framework

UK Gaia science platform

UK Gaia project is deploying the Gaia Data Analysis Framework on STFC's computing cloud e-infrastructure "IRIS"

- Prototype working now (see following demo slides)
- Aiming to have end-user service in place by DR3 (end Q2 next year)
- Looking for interested parties to beta-test the system based on EDR3

Simple example : mean angular motions over the sky

Zeppelin Notebook Job Search nch

…AglaisPublicExamples/Mean proper moti Head default

Set HEALPix resolution FINISHED

```
%pyspark

# set the required HEALPixelisation level here:
healpix_level = 6
# HEALPix level : no. of pixels
# 4 : 3072
# 5 : 12288
# 6 : 49152 ~ 1 square degree pixels
# 7 : 196608
```

Took 0 sec. Last updated by nch at June 25 2021, 12:09:39 PM.

Define a data frame by SQL query FINISHED

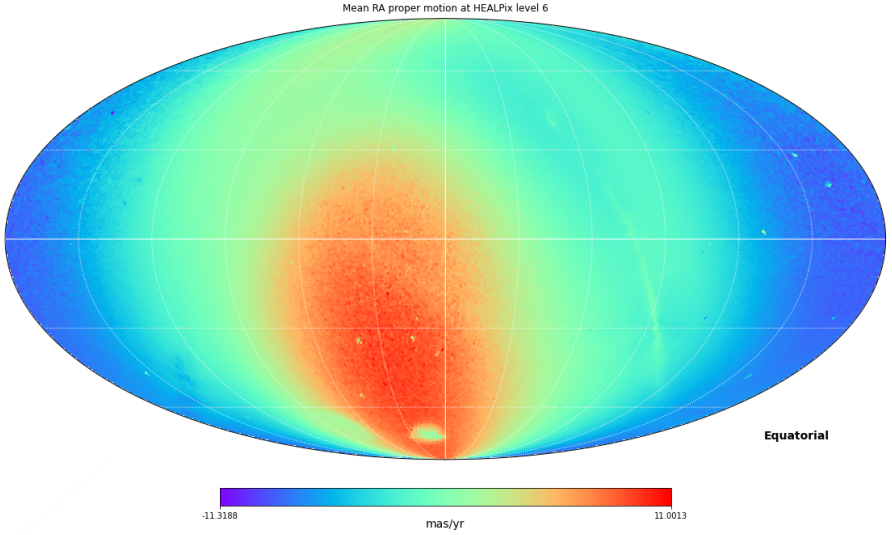
```
%pyspark
import math

# compute relevant pixelisation quantities
nside = int(math.pow(2, healpix_level))
powers_of_2 = 35 + (12 - healpix_level)*2
divisor = int(math.pow(2, powers_of_2))

# formulate SQL query
query = "SELECT floor(source_id /  %d"%(divisor) + ") AS hpx_id, COUNT(*) AS n, AVG(pmra) AS avg_pmra, AVG(pmdec) AS avg_pmdec FROM gaia_source GROUP BY hpx_id"

# define a data frame aggregation of the relevant quantities (note this is cached for use in two subsequent cells)
df = spark.sql(query).cache()
```

Took 2 sec. Last updated by nch at June 25 2021, 12:09:41 PM.

Mean RA proper motion plot

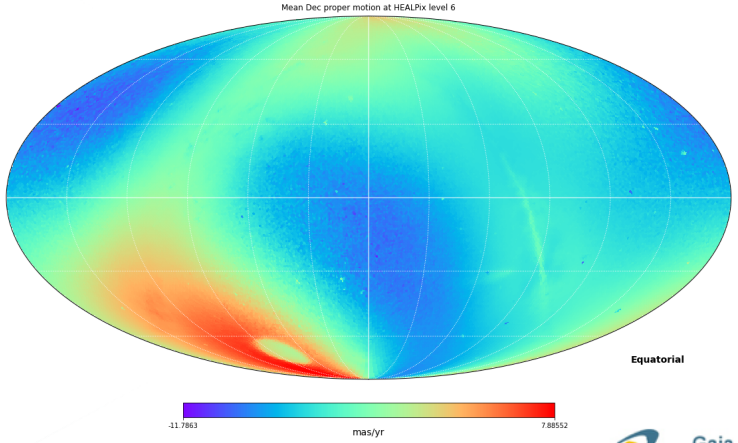
FINISHED

```
%pyspark

# plot up the sky counts
import matplotlib.pyplot as plot
import numpy as np
import healpy as hp

# set a figure to use along with a plot size (landscape, golden ratio)
plot.figure(1, figsize = (16.18, 10.0))

# healpy constants appropriate to the HEALPix indexing encoded in Gaia source IDs
npix = hp.nside2npix(nside)

# do the visualisation
array_data = np.empty(npix)
for item in df.rdd.collect():  array_data[item[0]] = item[2]
hp.mollview(array_data, fig = 1, coord='C', unit='mas/yr', nest=True, title='Mean RA proper motion at HEALPix level %d'%(healpix_level), cmap='rainbow')
hp.graticule(coord='C', color='white')
```



Took 1 min 23 sec. Last updated by nch at June 25 2021, 12:11:04 PM.

1min 28sec to run the query and plot the graph



Took 1 sec. Last updated by nch at June 25 2021, 12:11:05 PM.

Gaia DPAC

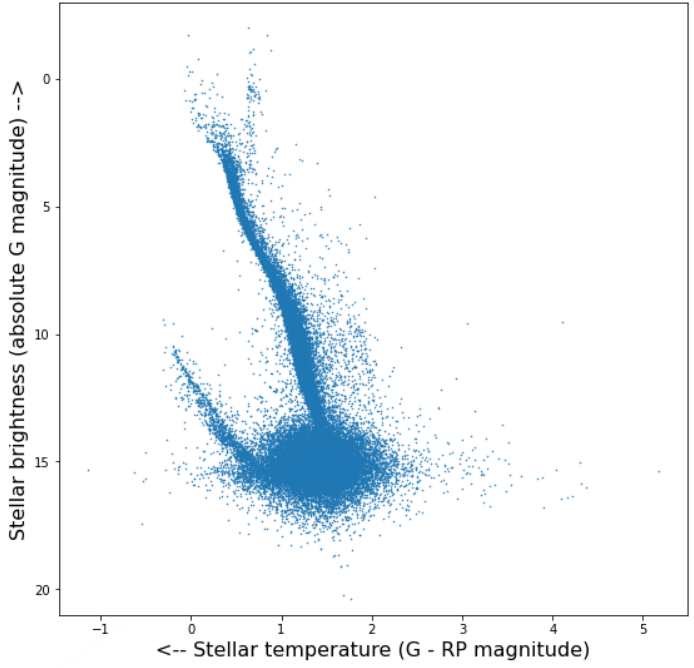
Complex example : Random Forest ML classifier

Using ML to define an astrometrically clean sample of stars

FINISHED

Follows the Gaia EDR3 performance verification "The Gaia Catalogue of Nearby Stars" (Smart et al. 2021) in classifying astrometric solutions as good or bad via supervised ML. Employs a Random Forrest classifier plus appropriately defined training sets - see https://arxiv.org/abs/2012.02061 for further details. The work flow implemented here follows closely that described in Section 2, "GCNS Generation" (GCNS = Gaia Catalogue of Nearby Stars) and is designed to clean up a 100pc (= nearby) sample.

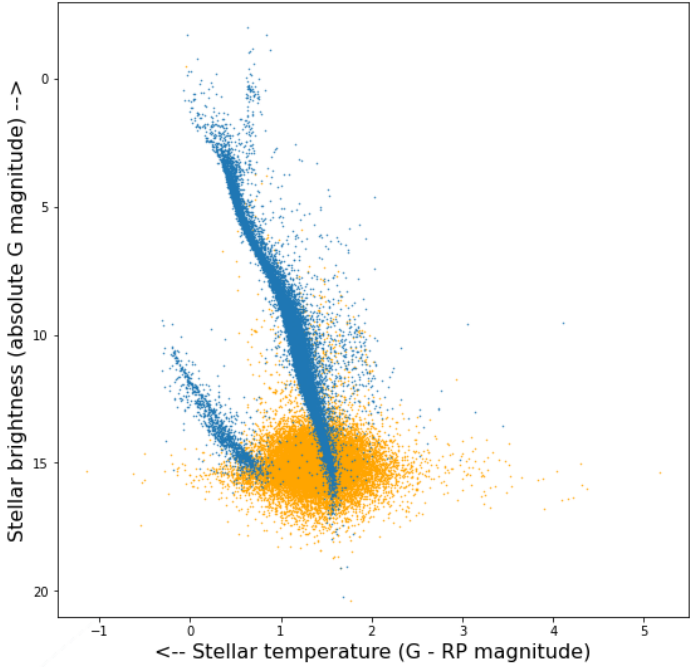
Took 0 sec. Last updated by nch at June 25 2021, 1:43:39 PM.



Train a RandomForest decision tree to classify good and bad astrometric solutions

6min to create the training data and train the classifier

25sec to classify 1,724,028 sources and plot the results



Beta testing

Call for people interested in being beta testers

Booked sessions with tech support

Q4 2021

Outline of the science case you would like to explore

Outline of the technology or algorithm you are interested in using

Contact a member of the development team

- Nigel Hambly <nch@roe.ac.uk>
- Dave Morris <dmr@roe.ac.uk>
- Stelios Voutsinas <stv@roe.ac.uk>

Interested in the most likely to break the system