Scripts used in Binaries RSP Analyses

Added by Richard Dubois, last edited by Richard Dubois on Nov 08, 2009

There are a handful of python scripts being used in the RSP analysis chain to date:

- select.py: run gtselect; fcopy to select CTBCLASSLEVEL and gtmktime to apply zenith cuts and update GTIs
- like.py: run exposure cube, map, diffuse response columns and likelihood
- bin.py: make light curves

The scripts currently live in:

/u/ey/richard/GLAST/mQ/pyMQ/

but will soon be in cvs.

The python scripts use the optparse interface to allow supplying arguments - it makes setting defaults quite easy, especially since most of the options are the same fromsource to source. Read the script to see what the defaults are.

Setup

```
noric11:richard> more ~/GLAST/mQ/pyMQ/setup.csh
setenv ST_INST /nfs/farm/g/glast/u30/builds/rh9_gcc32opt/ScienceTools/Science'
setenv PYTHONPATH
setenv PFILES /nfs/farm/g/glast/u33/richard/FlightData/firstLight/pfiles/
setenv CMTCONFIG rh9_gcc32opt
setenv CMTPATH ${ST_INST}
source $CMTPATH/likeGui/*/cmt/setup.csh
source $CMTPATH/pyExposure/*/cmt/setup.csh
setenv PYTHONPATH /u/ey/richard/GLAST/mQ/pyMQ/:$PYTHONPATH
source /afs/slac/g/glast/applications/astroTools/astrotools_setup.csh
```

A sample driver for an analysis is runLSI.py.

Spectral Fits

select.py

```
noric11:richard> ~/GLAST/mQ/pyMQ/select.py -h
Usage: select.py [options]

run gtmktime and gtselect

Options:
  --version             show program's version number and exit
  -h, --help            show this help message and exit
  -s SCFILE, --scfile=SCFILE
                        Filespec of spacecraft file
  -e EVFILE, --evfile=EVFILE
                        Filespec of events file
  -f FILTER, --filter=FILTER
                        filter string
  -o OFILE, --ofile=OFILE
                        output file name
  -r RA, --ra=RA        ra value of cut
  -d DEC, --dec=DEC     dec value of cut
  -q RADIUS, --radius=RADIUS
                        radius of cut
  --astroRadius=ASTRORADIUS
                        applied to astroserver ft1 file?
  -t TMIN, --tmin=TMIN  tmin value of cut
  -u TMAX, --tmax=TMAX  tmax value of cut
  -j EMIN, --emin=EMIN  emin value of cut
  -g EMAX, --emax=EMAX  emax value of cut
  -z ZMAX, --zmax=ZMAX  max value of zenith angle
  --class=CLASSLEVEL    CTBCLASSLEVEL
  --angsep=ANGSEP       apply zenith cut in gtmktime (yes/no)
  --rockCut=ROCKCUT     max value of rocking angle
  --classVar=CLASSVAR   var with event class: CTBCLASSLEVEL or not
```

like.py

```
noric11:richard> ~/GLAST/mQ/pyMQ/like.py -h
Usage: like.py [options]

run gtlike

Options:
  --version             show program's version number and exit
  -h, --help            show this help message and exit
  -s SCFILE, --scfile=SCFILE
                        Filespec of spacecraft file
  -e EVFILE, --evfile=EVFILE
                        Filespec of events file
  --expfile=EXPFILE     output file name
  --expCube=EXPCUBE     exposure cube output file
  --irfs=IRFS           irfs to use
  --sourceModel=SOURCEMODEL
                        source model file
  --optimizer=OPTIMIZER
                        select minimisation package
  --fluxmdl=FLUXMDL     output xml file of model
  --likeOnly=LIKEONLY   only run ilkelihood (yes/no)
  --diffrsp=DIFFRSP     force diffuse response (yes/no)
  --plot=PLOT           run the gui (yes/no)
  --writeCounts=WRITECOUNTS
                        file spec for counts_spectra fits file
  --writeResults=WRITERESULTS
                        file spec for results.txt
  --sourceName=SOURCENAME
                        name of target source in xml file
  --resultsLog=RESULTSLOG
                        summary results output file
  --emin=EMIN           min energy for energy integral
  --emax=EMAX           min energy for energy integral
  --tol=TOL             tolerance value
```

bin.py

```
noric11:richard> ~/GLAST/mQ/pyMQ/bin.py -h
Usage: bin.py [options]

run gtbin

Options:
  --version             show program's version number and exit
  -h, --help            show this help message and exit
  -s SCFILE, --scfile=SCFILE
                        Filespec of spacecraft file
  -e EVFILE, --evfile=EVFILE
                        Filespec of events file
  --alg=ALGORITHM       choice of algorithm
  --timeBinAlg=TIMEBINALG
                        algorithm for time binning
  -o OFILE, --ofile=OFILE
                        output file name
  --dtime=DTIME         time bin size
  --emin=EMIN           minimum energy
  --emax=EMAX           minimum energy
  --nxpix=NXPIX         number x pixels
  --nypix=NYPIX         number y pixels
  -c COORD, --coord=COORD
                        coordinate system
  --irfs=IRFS           irfs to use
  -r RA, --ra=RA        ra value of cut
  -d DEC, --dec=DEC     dec value of cut
  --index=INDEX         assumed power law spectral index
  --exposure=EXPOSURE   run gtexposure yes/no
```

psearch.py

```
ttanaka@ki-ls02 $ ./psearch.py -h
Usage: psearch.py [options]

Options:
  -h, --help            show this help message and exit
  -l LCFILE, --lc=LCFILE
                        LC file for input
  -r ROOTFILE, --root=ROOTFILE
                        ROOT file for output
  -g GIFFILE, --gif=GIFFILE
                        Gif file for output
```

makeCountsMap.py

```
Options:
  --version             show program's version number and exit
  -h, --help            show this help message and exit
  -s SCFILE, --scfile=SCFILE
                        Filespec of spacecraft file
  -e EVFILE, --evfile=EVFILE
                        Filespec of events file
  --alg=ALGORITHM       choice of algorithm
  --obfile=OBFILE       gtbin output file name
  --ebinalg=EBINALG     energy binning algorithm
  --enumbins=ENUMBINS   number of energy bins
  --emin=EMIN           minimum energy
  --emax=EMAX           minimum energy
  --binsz=BINSZ         angular bin size in image center (deg)
  --nxpix=NXPIX         number x pixels
  --nypix=NYPIX         number y pixels
  -c COORD, --coord=COORD
                        coordinate system
  --proj=PROJ           coordinate projection
  --ra=RA               right ascension
  --dec=DEC             declination
```

residMap.py

```
Options:
  --version             show program's version number and exit
  -h, --help            show this help message and exit
  -s SCFILE, --scfile=SCFILE
                        Filespec of spacecraft file
  -e EVFILE, --evfile=EVFILE
                        Filespec of events file
  --alg=ALGORITHM       choice of algorithm
  --obfile=OBFILE       gtbin output file name
  --ebinalg=EBINALG     energy binning algorithm
  --enumbins=ENUMBINS   number of energy bins
  --emin=EMIN           minimum energy
  --emax=EMAX           minimum energy
  --binsz=BINSZ         angular bin size in image center (deg)
  --nxpix=NXPIX         number x pixels
  --nypix=NYPIX         number y pixels
  -c COORD, --coord=COORD
                        coordinate system
  --proj=PROJ           coordinate projection
  --ra=RA               right ascension
  --dec=DEC             declination
  --expCube=EXPCUBE     exposure cube
  --sourceModel=SOURCEMODEL
                        source model xml file
  --bexpmap=BEXPMAP     binned exposure map file
  --osfile=OSFILE       gtsrcmaps output file name
  --irfs=IRFS           irfs
  --omfile=OMFILE       gtmodel output file name
  --modelOnly=MODONLY   gtmodel output file name
  --orfile=ORFILE       residuals output file name
```

plotLC.py

```
Options:
  --version             show program's version number and exit
  -h, --help            show this help message and exit
  -i INFILE, --infile=INFILE
                        Filespec of input binned LC fits file
  -e OUTFILE, --outfile=OUTFILE
                        Filespec of output graphics map file
```

plotFoldLC.py

```
Options:
  --version             show program's version number and exit
  -h, --help            show this help message and exit
  -i INFILE, --infile=INFILE
                        Filespec of input binned LC fits file
  -e OUTFILE, --outfile=OUTFILE
                        Filespec of output graphics map file
  --period=PERIOD
                        orbital period (days) to fold on
  --t0=TO
                        zero phase (MET) to subtract from times
```

plot_counts.py

```
Options:
  --version             show program's version number and exit
  -h, --help            show this help message and exit
  -i INFILE, --infile=INFILE
                        Filespec of input gtlike "counts" fits file
  -e OUTFILE, --outfile=OUTFILE
                        Filespec of output graphics plot file
```

Old Stuff

```
\#\!/usr/bin/env python
"""
Generate test data with dataSubselector.

@author J. Chiang <jchiang@slac.stanford.edu>
"""
\#
# $Header: /nfs/slac/g/glast/ground/cvs/sane/python/tests/dataSubselector_tes
\#

\##from setPaths import *
from gt_apps import filter, maketime
from optparse import OptionParser
import os

def run(clean=False):

os.system("/u/ey/richard/GLAST/mQ/pyMQ/select.py \--scfile FT2-20080923-merge

os.system("/u/ey/richard/GLAST/mQ/pyMQ/like.py \--scfile FT2-20080923-merged.

if __name__ == "\__main_\_":
run()
```

This shows you the minimum set of input parameters you can give to run the analysis. This assumes that the input

datasets have been assembled one way or another - currently via getRuns.py, but soon just as easily from the data servers. Another example driver script is runLSI.py: /nfs/farm/g/glast/u31/richard/FlightData/firstLight /templateScripts/runLSI.py.

Here is a snippet from a batch log running like.py by itself to just redo the fit, using existing ft1, ft2, cube and map files:

```
/u/ey/richard/GLAST/mQ/pyMQ/like.py \-s /nfs/farm/g/glast/u18//RSP-tasks/RSP
```

getRuns.py

```
noric11:richard> \~/GLAST/mQ/pyMQ/getRuns.py \-h
Usage: getRuns.py [options]

select runs and merge FT2 file

Options:
\--version             show program's version number and exit
\-h, \--help            show this help message and exit
\--FT1=FT1LIST         List of FT1 files
\--FT2=FT2LIST         List of FT2 files
\--FT1out=FT1OUT       merged FT1 file
\--FT2out=FT2OUT       merged FT2 file
\--minRun=MINRUN       minimum run number
\--maxRun=MAXRUN       maximum run number
\-f FOLDER, \--folder=FOLDER
source folder in catalogue
\--noFind=NOFIND       skip find step (yes/no)
\--doFT1=DOFT1         merge FT1 (yes/no)
\--doFT2=DOFT2         merge FT1 (yes/no)
```

Labels

mq