

Example_LLH_setup_fixed_dirs

March 8, 2022

```
[1]: import numpy as np
from astropy.io import fits
from astropy.table import Table, vstack
from astropy.wcs import WCS
import os
from scipy import optimize, stats, interpolate
from scipy.integrate import quad
import argparse
import time
import multiprocessing as mp
import matplotlib.pyplot as plt
from matplotlib.colors import LogNorm
from matplotlib import cm
import sys
import pandas as pd
pd.options.display.max_columns = 250
pd.options.display.max_rows = 250
import healpy as hp
from copy import copy, deepcopy
# sys.path.append('BatML/')
import logging, traceback
import sys
#logging.basicConfig(stream=sys.stdout, level=logging.DEBUG)

[2]: NITRATES_path = '/Users/tparsota/Documents/BAT_SCRIPTS/NITRATES/NITRATES/'
      #%cd {NITRATES_path}

[3]: # if the response directories are all put in the same directory
      # and given as an env variable (like in this cell), then config.py will
      # be able to find them correctly as long as they keep the original names
      # (Ex: ray trace directotry is still called 'ray_traces_detapp_npy')
      # These paths can also just be hardcoded into config.py
      # most of these paths can also be given directly as an argument to
      # some analysis objects, like Source_Model_InOutFoV()
os.environ['NITRATES_RESP_DIR'] = '/Users/tparsota/Documents/BAT_SCRIPTS/
↳NITRATES_BAT_RSP_FILES/'
```

```
[4]: from config import rt_dir
from ray_trace_funcs import RayTraces
from event2dpi_funcs import det2dpi, mask_detxy
from flux_models import Cutoff_Plaw_Flux, Plaw_Flux, get_eflux_from_model
from models import Source_Model_InOutFoV, Bkg_Model_wFlatA,\
    CompoundModel, Point_Source_Model_Binned_Rates, im_dist
from ray_trace_funcs import RayTraces
from sqlite_funcs import get_conn
from dbread_funcs import get_info_tab, get_twinds_tab
from hp_funcs import ang_sep
from coord_conv_funcs import theta_phi2imxy, imxy2theta_phi,\
    ↪convert_imxy2radec,\
    ↪convert_radec2thetaphi, convert_radec2imxy
from do_llh_inFoV4realtime2 import parse_bkg_csv
from LLH import LLH_webins
from minimizers import NLLH_ScipyMinimize_Wjacob
```

```
[5]: %matplotlib inline
```

```
[6]: ebins0 = np.array([15.0, 24.0, 35.0, 48.0, 64.0])
ebins0 = np.append(ebins0, np.logspace(np.log10(84.0), np.log10(500.0), 5+1))[:
    ↪-1]
ebins0 = np.round(ebins0, decimals=1)[: -1]
ebins1 = np.append(ebins0[1:], [350.0])
nebins = len(ebins0)
print("Nebins: ", nebins)
print("ebins0: ", ebins0)
print("ebins1: ", ebins1)

# work_dir = '/storage/work/jjd330/local/bat_data/realtime_workdir/F646018360/'
work_dir = os.path.join(NITRATES_path, 'F646018360')
conn = get_conn(os.path.join(work_dir, 'results.db'))
info_tab = get_info_tab(conn)
trigger_time = info_tab['trigtimeMET'][0]
print('trigger_time: ', trigger_time)

evfname = os.path.join(work_dir, 'filter_evdata.fits')
ev_data = fits.open(evfname)[1].data
GTI_PNT = Table.read(evfname, hdu='GTI_POINTING')
GTI_SLEW = Table.read(evfname, hdu='GTI_SLEW')
dmask = fits.open(os.path.join(work_dir, 'detmask.fits'))[0].data
attfile = fits.open(os.path.join(work_dir, 'attitude.fits'))[1].data

att_ind = np.argmin(np.abs(attfile['TIME'] - trigger_time))
att_quat = attfile['QPARAM'][att_ind]
print('attitude quaternion: ', att_quat)
```

```

ndets = np.sum(dmask==0)
print("Ndets: ", np.sum(dmask==0))

t_end = trigger_time + 1e3
t_start = trigger_time - 1e3
mask_vals = mask_detxy(dmask, ev_data)
bl_dmask = (dmask==0.)

bl_ev = (ev_data['EVENT_FLAGS']<1)&\
        (ev_data['ENERGY']<=500.)&(ev_data['ENERGY']>=14.)&\
        (mask_vals==0.)&(ev_data['TIME']<=t_end)&\
        (ev_data['TIME']>=t_start)

print("Nevents: ", np.sum(bl_ev))
ev_data0 = ev_data[bl_ev]

```

```

('Nebins: ', 9)
('ebins0: ', array([ 15. , 24. , 35. , 48. , 64. , 84. , 120. , 171.5, 245.
]))
('ebins1: ', array([ 24. , 35. , 48. , 64. , 84. , 120. , 171.5, 245. , 350.
]))
('trigger_time: ', 646018383.1787)
('attitude quaternion: ', array([-0.03597053,  0.2345147 , -0.64420835,
0.72712074]))
('Ndets: ', 14932)
('Nevents: ', 1367885)

```

```

[7]: ra, dec = 233.117, -26.213
theta, phi = convert_radec2thetaphi(ra, dec, att_quat)
print(theta, phi)
imx, imy = convert_radec2imxy(ra, dec, att_quat)
print(imx, imy)

```

```

(38.54132137017975, 137.65241966813443)
(-0.5887551341212709, -0.5366203642198198)

```

```

[8]: flux_params = {'A':1.0, 'gamma':0.5, 'Epeak':1e2}
flux_mod = Cutoff_Plaw_Flux(E0=100.0)

```

```

[9]: # use rt_dir from config.py or
# rt_dir = '/path/to/ray_traces_detapp_npy/'
rt_obj = RayTraces(rt_dir)

```

```

[10]: rt = rt_obj.get_intp_rt(imx, imy)
print(np.shape(rt))

```

```

(173, 286)

```

```
[11]: %%time
# will by default use the resp directories from config.py
sig_mod = Source_Model_InOutFoV(flux_mod, [ebins0,ebins1], bl_dmask,\
                                rt_obj, use_deriv=True)

# or the paths can be given
# resp_tab_dname = '/path/to/resp_tabs_ebins/'
# hp_flor_resp_dname = '/path/to/hp_flor_resps/'
# comp_flor_resp_dname = '/path/to/comp_flor_resps/'
# sig_mod = Source_Model_InOutFoV(flux_mod, [ebins0,ebins1], bl_dmask,\
#                                 rt_obj, use_deriv=True,\
#                                 resp_tab_dname=resp_tab_dname,\
#                                 hp_flor_resp_dname=hp_flor_resp_dname,\
#                                 comp_flor_resp_dname=comp_flor_resp_dname)
```

CPU times: user 705 μ s, sys: 390 μ s, total: 1.1 ms
Wall time: 506 μ s

```
[12]: %%time
sig_mod.set_theta_phi(theta, phi)
```

WARNING: hdu= was not specified but multiple tables are present, reading in first available table (hdu=1) [astropy.io.fits.connect]
WARNING:astropy:hdu= was not specified but multiple tables are present, reading in first available table (hdu=1)

(0.0, 30.48, -14.117)
(0.0, 30.48, -14.117)

response.py:1499: RuntimeWarning: invalid value encountered in divide
(self.orig_photonEs[self.Einds1] - self.orig_photonEs[self.Einds0])

(36.0, 45.0)
2.652419668134428
42.34758033186557
max rt: 0.8604
initing ResponseDPI, with fname
/Users/tparsota/Documents/BAT_SCRIPTS/NITRATES_BAT_RSP_FILES/resp_tabs_ebins/drm_theta_36.0_phi_30.0.fits
initing ResponseDPI, with fname
/Users/tparsota/Documents/BAT_SCRIPTS/NITRATES_BAT_RSP_FILES/resp_tabs_ebins/drm_theta_36.0_phi_45.0.fits
initing ResponseDPI, with fname
/Users/tparsota/Documents/BAT_SCRIPTS/NITRATES_BAT_RSP_FILES/resp_tabs_ebins/drm_theta_45.0_phi_30.0.fits
initing ResponseDPI, with fname
/Users/tparsota/Documents/BAT_SCRIPTS/NITRATES_BAT_RSP_FILES/resp_tabs_ebins/drm_theta_45.0_phi_45.0.fits

CPU times: user 4.47 s, sys: 2.11 s, total: 6.59 s
Wall time: 6.42 s

```
[13]: from config import solid_angle_dpi_fname
# or give the path directly
# solid_angle_dpi_fname = '/path/to/solid_angle_dpi.npy'
bkg_fname = os.path.join(work_dir, 'bkg_estimation.csv')
solid_ang_dpi = np.load(solid_angle_dpi_fname)
bkg_df, bkg_name, PSnames, bkg_mod, ps_mods = parse_bkg_csv(bkg_fname, \
↳solid_ang_dpi, \
                ebins0, ebins1, bl_dmask, rt_dir)
bkg_mod.has_deriv = False
bkg_mod_list = [bkg_mod]
Nsracs = len(ps_mods)
if Nsracs > 0:
    bkg_mod_list += ps_mods
    for ps_mod in ps_mods:
        ps_mod.has_deriv = False
    bkg_mod = CompoundModel(bkg_mod_list)
tmid = trigger_time
bkg_row = bkg_df.iloc[np.argmin(np.abs(tmid - bkg_df['time']))]
bkg_params = {pname:bkg_row[pname] for pname in \
                bkg_mod.param_names}
bkg_name = bkg_mod.name
bkg_params
```

```
['4U 1700-377', 'GRO J1655-40', 'GX 339-4', 'Sco X-1']
```

```
/Users/tparsota/opt/anaconda3/envs/test_py2/lib/python2.7/site-
packages/numpy/core/fromnumeric.py:56: FutureWarning:
The current behaviour of 'Series.argmax' is deprecated, use 'idxmin'
instead.
```

```
The behavior of 'argmin' will be corrected to return the positional
minimum in the future. For now, use 'series.values.argmax' or
'np.argmax(np.array(values))' to get the position of the minimum
row.
```

```
return getattr(obj, method)(*args, **kws)
```

```
[13]: {'4U 1700-377_imx': -0.09814853057709713,
'4U 1700-377_imy': -0.4742076074486664,
'4U 1700-377_rate_0': 0.012490135921897544,
'4U 1700-377_rate_1': 0.00724252648862388,
'4U 1700-377_rate_2': 0.004529564586462892,
'4U 1700-377_rate_3': 0.0016876913514016756,
'4U 1700-377_rate_4': 0.0013255091059552946,
'4U 1700-377_rate_5': 0.0012956105576278066,
'4U 1700-377_rate_6': 0.0005834459169539205,
'4U 1700-377_rate_7': 5e-08,
'4U 1700-377_rate_8': 0.00014013200316638987,
'Background_bkg_rate_0': 0.09137032207011836,
'Background_bkg_rate_1': 0.0661578002239374,
```

'Background_bkg_rate_2': 0.040089856902610514,
'Background_bkg_rate_3': 0.039491993420780136,
'Background_bkg_rate_4': 0.03467802668352813,
'Background_bkg_rate_5': 0.0351076584048365,
'Background_bkg_rate_6': 0.03360249200395507,
'Background_bkg_rate_7': 0.024861082684787414,
'Background_bkg_rate_8': 0.01763994761450761,
'Background_flat_0': 0.0,
'Background_flat_1': 0.0,
'Background_flat_2': 0.18596326494289767,
'Background_flat_3': 0.07663442445332364,
'Background_flat_4': 0.3020855890115875,
'Background_flat_5': 0.7611510180823338,
'Background_flat_6': 0.8091430254293075,
'Background_flat_7': 1.0,
'Background_flat_8': 1.0,
'GRO J1655-40_imx': -0.11341596786535295,
'GRO J1655-40_imy': -0.5359750413588411,
'GRO J1655-40_rate_0': 0.005277185138471884,
'GRO J1655-40_rate_1': 0.002203663426192921,
'GRO J1655-40_rate_2': 0.00031497302758220266,
'GRO J1655-40_rate_3': 0.001764064431127689,
'GRO J1655-40_rate_4': 5e-08,
'GRO J1655-40_rate_5': 9.114980149845126e-05,
'GRO J1655-40_rate_6': 3.886335048265329e-05,
'GRO J1655-40_rate_7': 5e-08,
'GRO J1655-40_rate_8': 0.0007082358100819457,
'GX 339-4_imx': 0.008729300861405498,
'GX 339-4_imy': -0.6947927093434689,
'GX 339-4_rate_0': 0.0037893354510156046,
'GX 339-4_rate_1': 0.003214889430859694,
'GX 339-4_rate_2': 0.0017605414459212125,
'GX 339-4_rate_3': 0.0015331679372228933,
'GX 339-4_rate_4': 0.0015777295548874018,
'GX 339-4_rate_5': 0.0014172765864478645,
'GX 339-4_rate_6': 0.0008599988841135245,
'GX 339-4_rate_7': 0.0013329459515836204,
'GX 339-4_rate_8': 0.0005777085477427549,
'Sco X-1_imx': -0.4801793393603647,
'Sco X-1_imy': -0.20015662298079426,
'Sco X-1_rate_0': 0.04972196231874512,
'Sco X-1_rate_1': 0.005921355615060677,
'Sco X-1_rate_2': 5e-08,
'Sco X-1_rate_3': 0.0003592599287820135,
'Sco X-1_rate_4': 0.00040493931447181734,
'Sco X-1_rate_5': 5e-08,
'Sco X-1_rate_6': 5e-08,

```
'Sco X-1_rate_7': 5e-08,  
'Sco X-1_rate_8': 6.897789976904059e-05}
```

```
[14]: pars_ = {}  
pars_['Signal_theta'] = theta  
pars_['Signal_phi'] = phi  
for pname,val in bkg_params.iteritems():  
    pars_[bkg_name+'_'+pname] = val  
for pname,val in flux_params.iteritems():  
    pars_['Signal_'+pname] = val  
pars_
```

```
[14]: {'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_4U 1700-377_imx':  
-0.09814853057709713,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_4U 1700-377_imy':  
-0.4742076074486664,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_4U 1700-377_rate_0':  
0.012490135921897544,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_4U 1700-377_rate_1':  
0.00724252648862388,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_4U 1700-377_rate_2':  
0.004529564586462892,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_4U 1700-377_rate_3':  
0.0016876913514016756,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_4U 1700-377_rate_4':  
0.0013255091059552946,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_4U 1700-377_rate_5':  
0.0012956105576278066,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_4U 1700-377_rate_6':  
0.0005834459169539205,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_4U 1700-377_rate_7':  
5e-08,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_4U 1700-377_rate_8':  
0.00014013200316638987,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_bkg_rate_0':  
0.09137032207011836,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_bkg_rate_1':  
0.0661578002239374,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_bkg_rate_2':  
0.040089856902610514,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_bkg_rate_3':  
0.039491993420780136,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_bkg_rate_4':  
0.03467802668352813,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_bkg_rate_5':  
0.0351076584048365,  
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_bkg_rate_6':
```

0.03360249200395507,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_bkg_rate_7':
 0.024861082684787414,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_bkg_rate_8':
 0.01763994761450761,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_flat_0': 0.0,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_flat_1': 0.0,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_flat_2':
 0.18596326494289767,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_flat_3':
 0.07663442445332364,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_flat_4':
 0.3020855890115875,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_flat_5':
 0.7611510180823338,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_flat_6':
 0.8091430254293075,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_flat_7': 1.0,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Background_flat_8': 1.0,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GR0 J1655-40_imx':
 -0.11341596786535295,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GR0 J1655-40_imy':
 -0.5359750413588411,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GR0 J1655-40_rate_0':
 0.005277185138471884,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GR0 J1655-40_rate_1':
 0.002203663426192921,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GR0 J1655-40_rate_2':
 0.00031497302758220266,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GR0 J1655-40_rate_3':
 0.001764064431127689,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GR0 J1655-40_rate_4':
 5e-08,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GR0 J1655-40_rate_5':
 9.114980149845126e-05,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GR0 J1655-40_rate_6':
 3.886335048265329e-05,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GR0 J1655-40_rate_7':
 5e-08,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GR0 J1655-40_rate_8':
 0.0007082358100819457,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GX 339-4_imx':
 0.008729300861405498,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GX 339-4_imy':
 -0.6947927093434689,
 'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GX 339-4_rate_0':
 0.0037893354510156046,

```

'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GX 339-4_rate_1':
0.003214889430859694,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GX 339-4_rate_2':
0.0017605414459212125,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GX 339-4_rate_3':
0.0015331679372228933,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GX 339-4_rate_4':
0.0015777295548874018,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GX 339-4_rate_5':
0.0014172765864478645,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GX 339-4_rate_6':
0.0008599988841135245,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GX 339-4_rate_7':
0.0013329459515836204,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_GX 339-4_rate_8':
0.0005777085477427549,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Sco X-1_imx':
-0.4801793393603647,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Sco X-1_imy':
-0.20015662298079426,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Sco X-1_rate_0':
0.04972196231874512,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Sco X-1_rate_1':
0.005921355615060677,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Sco X-1_rate_2': 5e-08,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Sco X-1_rate_3':
0.0003592599287820135,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Sco X-1_rate_4':
0.00040493931447181734,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Sco X-1_rate_5': 5e-08,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Sco X-1_rate_6': 5e-08,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Sco X-1_rate_7': 5e-08,
'Background+4U 1700-377+GR0 J1655-40+GX 339-4+Sco X-1_Sco X-1_rate_8':
6.897789976904059e-05,
'Signal_A': 1.0,
'Signal_Epeak': 100.0,
'Signal_gamma': 0.5,
'Signal_phi': 137.65241966813443,
'Signal_theta': 38.54132137017975}

```

```

[15]: comp_mod = CompoundModel([bkg_mod, sig_mod])

sig_miner = NLLH_ScipyMinimize_Wjacob('')

sig_llh_obj = LLH_webins(ev_data0, ebins0, ebins1, bl_dmask, has_err=True)

sig_llh_obj.set_model(comp_mod)

```

```
sig_miner.set_llh(sig_llh_obj)
```

```
[16]: fixed_pnames = pars_.keys()
fixed_vals = pars_.values()
trans = [None for i in range(len(fixed_pnames))]
sig_miner.set_trans(fixed_pnames, trans)
sig_miner.set_fixed_params(fixed_pnames, values=fixed_vals)
sig_miner.set_fixed_params(['Signal_A'], fixed=False)
```

```
[17]: %%time
flux_params['gamma'] = 0.8
flux_params['Epeak'] = 350.0
sig_mod.set_flux_params(flux_params)
```

CPU times: user 86.5 ms, sys: 3.88 ms, total: 90.4 ms
Wall time: 88.3 ms

```
[18]: t0 = trigger_time - 0.512
t1 = t0 + 2.048
sig_llh_obj.set_time(t0, t1)
```

```
[19]: %%time
pars, nllh, res = sig_miner.minimize()
```

CPU times: user 65.4 ms, sys: 23.7 ms, total: 89.1 ms
Wall time: 89.3 ms

```
[20]: print(res)
print(nllh)
print(pars)
```

```
[      fun: 45894.07068100062
 hess_inv: <1x1 LbfgsInvHessProduct with dtype=float64>
   jac: array([0.00239975])
 message: 'CONVERGENCE: REL_REDUCTION_OF_F_<= _FACTR*EPSMCH'
   nfev: 8
    nit: 5
  status: 0
 success: True
      x: array([0.006875])]
[45894.07068100062]
[[0.006875003655224621]]
```

```
[21]: %%time
pars_['Signal_A'] = 1e-10
bkg_nllh = -sig_llh_obj.get_logprob(pars_)
print bkg_nllh
sqrtTS = np.sqrt(2.*(bkg_nllh - nllh[0]))
```

```
print sqrtTS
```

```
46038.7433272
```

```
17.0101526275675
```

```
CPU times: user 6.16 ms, sys: 1.22 ms, total: 7.37 ms
```

```
Wall time: 6.48 ms
```

```
[ ]:
```