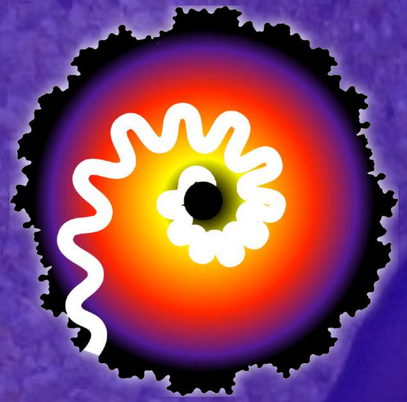


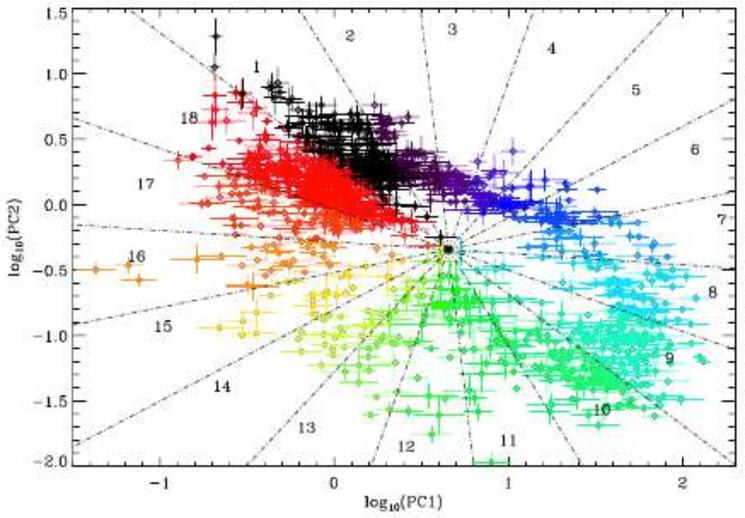
Stingray, HENDRICS, and DAVE

Spectral timing for all

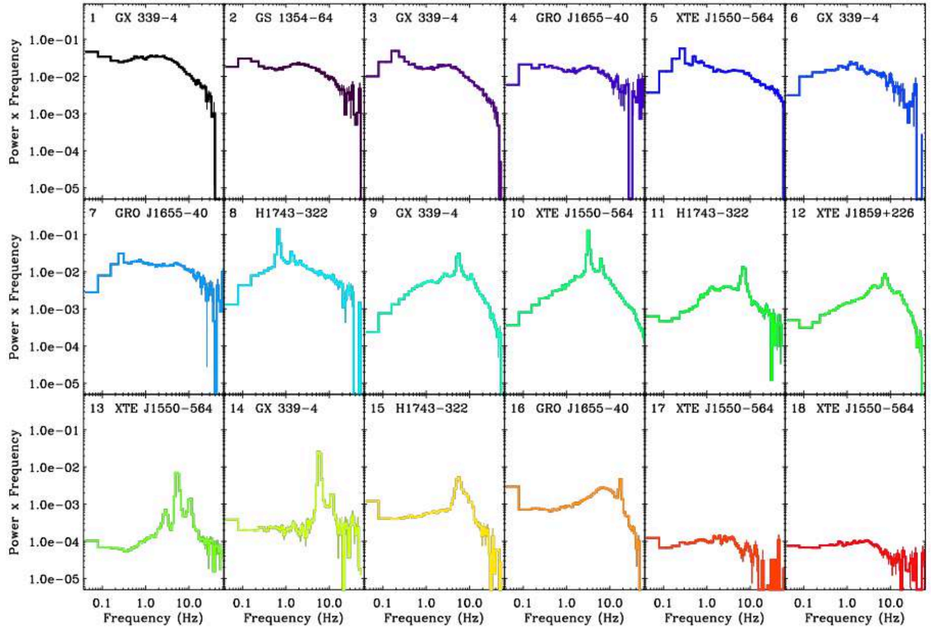
Matteo Bachetti

INAF-Osservatorio Astronomico di Cagliari

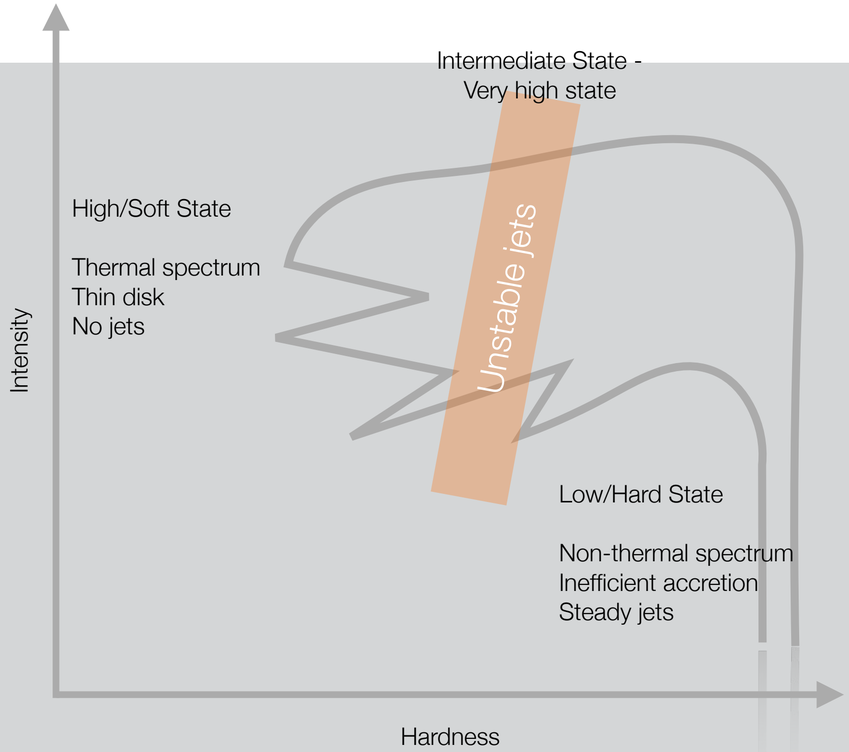
with Daniela Huppenkothen, Abigail Stevens, Paul Balm, Simone Migliari,
Ricardo Valles, Evandro Ribeiro, Himanshu Misra, Usman Khan, (...)

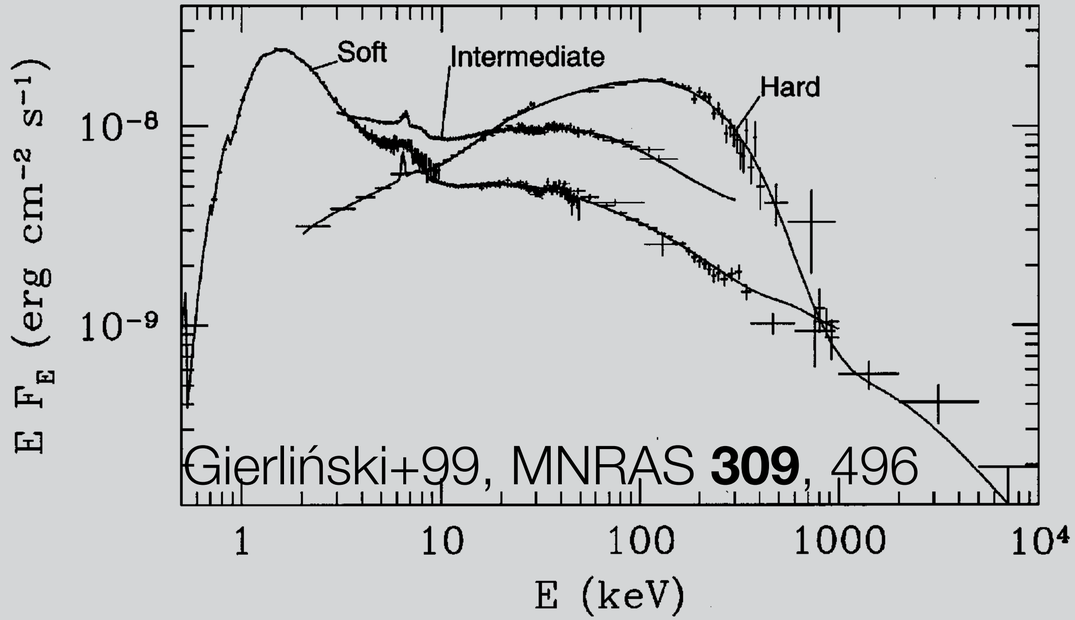


Heil, Uttley, & Klein-Wolt, *MNRAS* **448**, 3339–3347, 2015.



VARIABILITY





SPECTRA

Timing analysis

Periodograms
Power density spectra
Lomb-Scargle
Epoch folding search
Z-search, H-search
Cross spectra
Coherence spectra
(…)

Cross-correlation
Autocorrelation

Wavelets
Dynamical power spectra
Phaseogram

Spectral analysis

Continuum modeling
Color-color, Hardness-intensity
Absorption-emission features
High-resolution spectroscopy

Timing analysis

Periodograms
Power density spectra
Lomb-Scargle
Epoch folding search
Z-search, H-search
Cross spectra
Coherence spectra
(...)

Cross-correlation
Autocorrelation

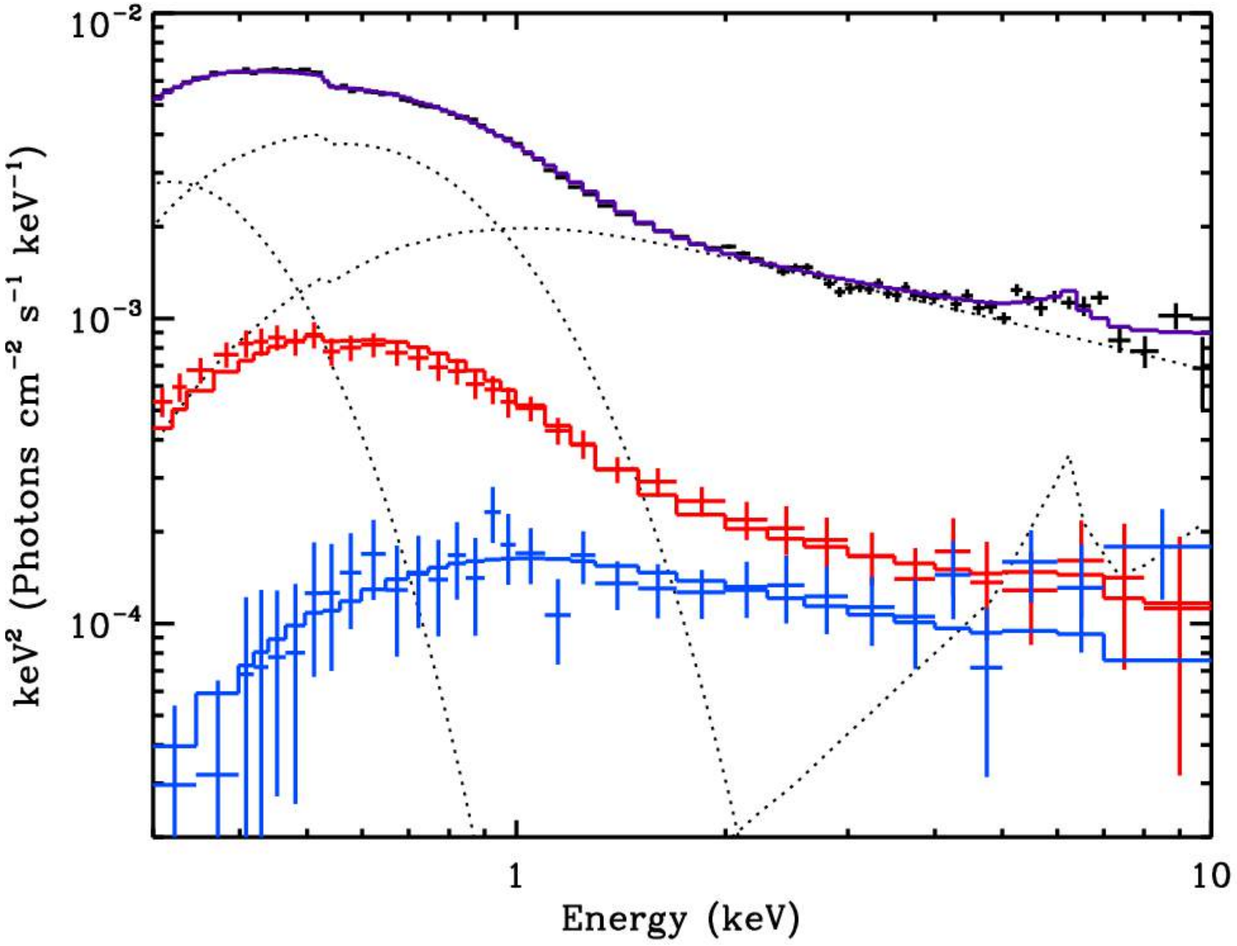
Wavelets
Dynamical power spectra
Phaseogram

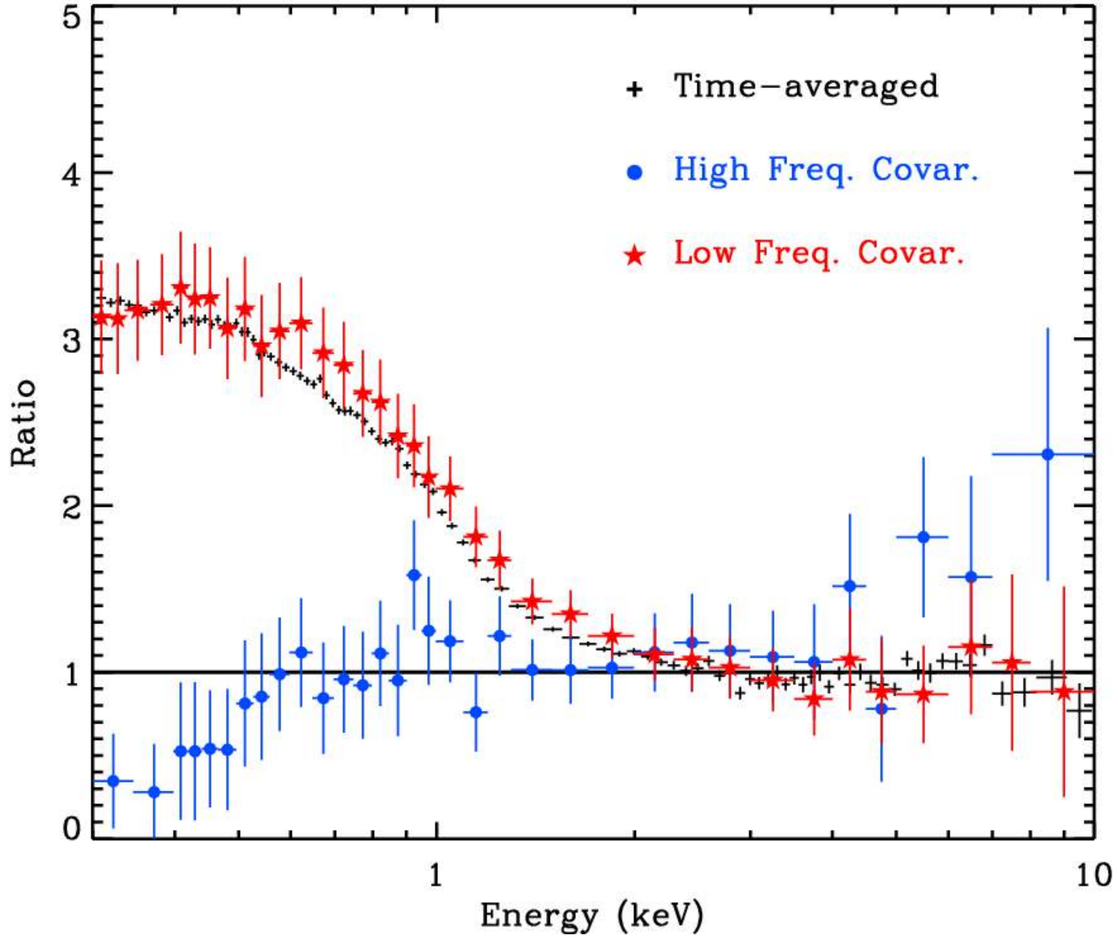
Spectral timing

phase-resolved spectra
rms-intensity diagrams
Time lags vs Energy
rms vs Energy
Covariance spectrum
(...)

Spectral analysis

Continuum modeling
Color-color, Hardness-intensity
Absorption-emission features
High-resolution spectroscopy



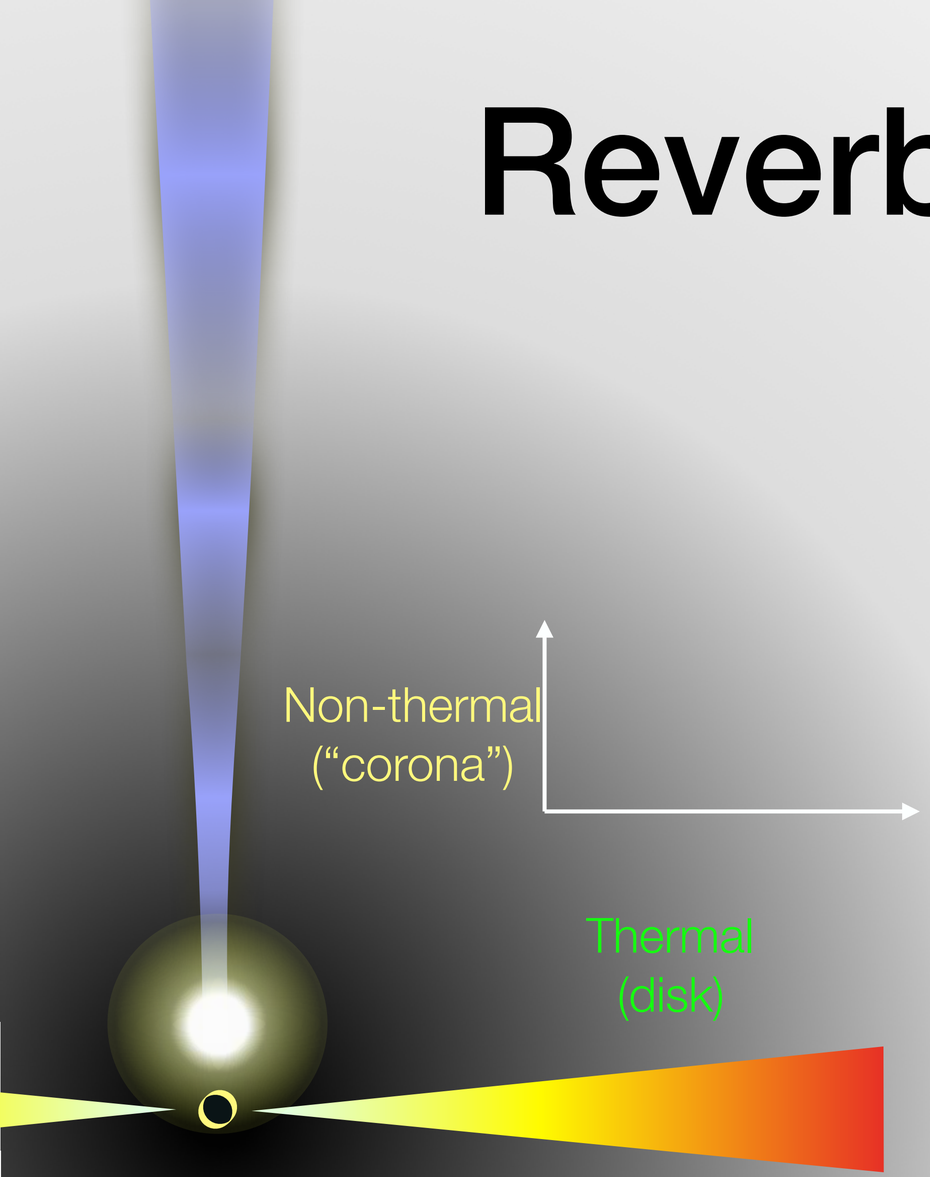


Covariance spectrum

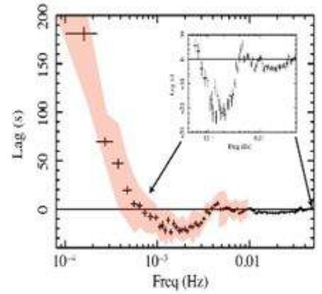
Look what spectral component is dominating the *variability*

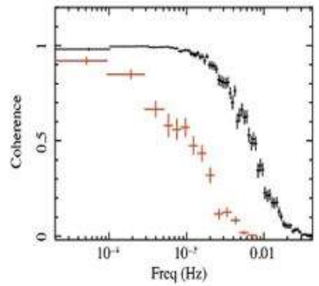
e.g. Chichuan,et al. *MNRAS* **436**, 3173-3185, 2015.

Reverberation mapping

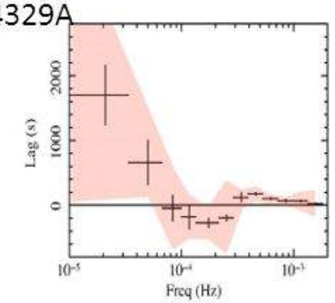


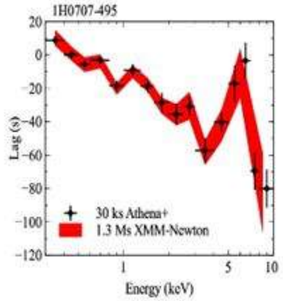
1H0707-495





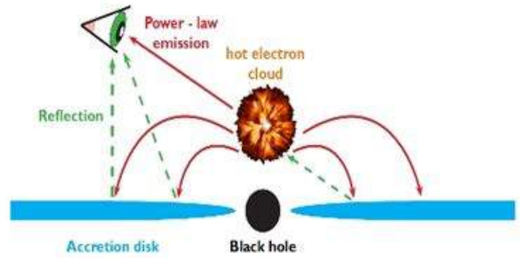
IC4329A

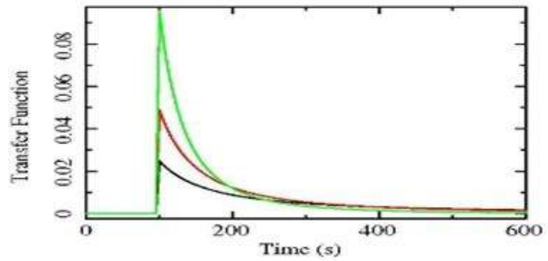


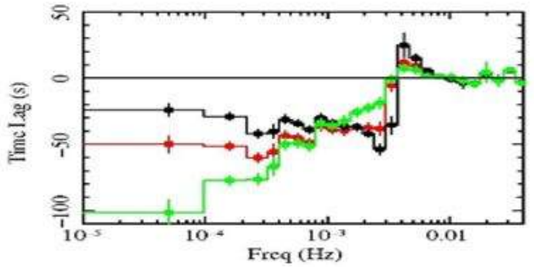


Corona/disk structure of (Reverberation mapping)

Time lag between 1-4keV and 0.3-1keV, Comparison between Athena and XMM

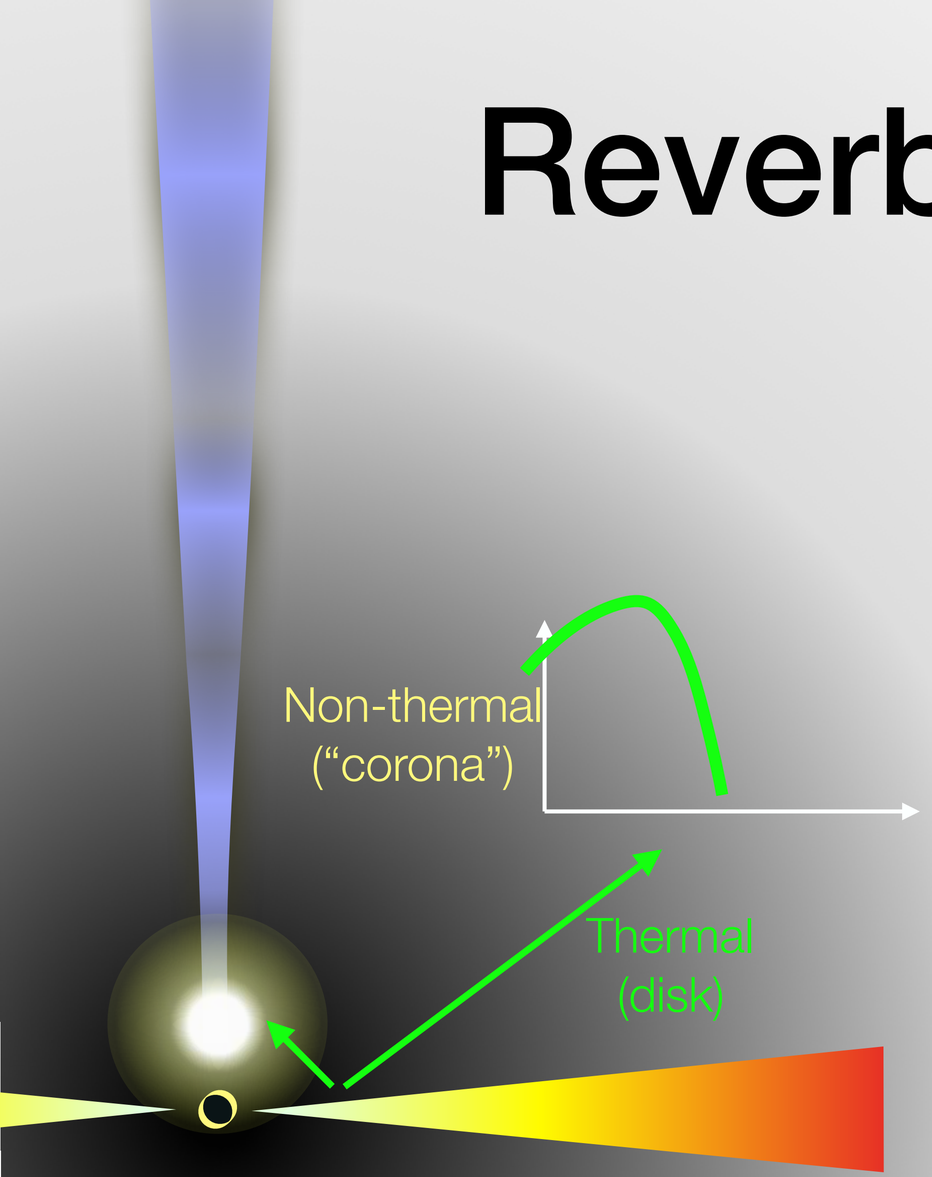




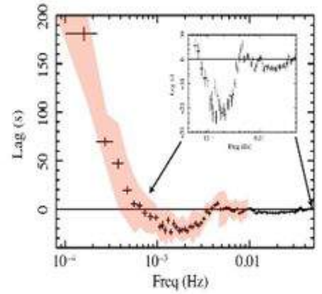


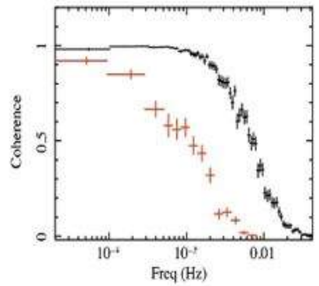
1H0707-495: time lag to probe the transfer structure of corona

Reverberation mapping

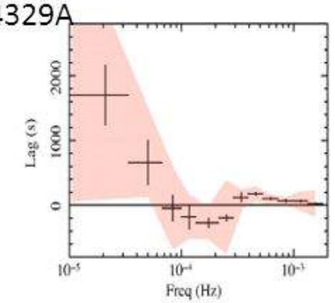


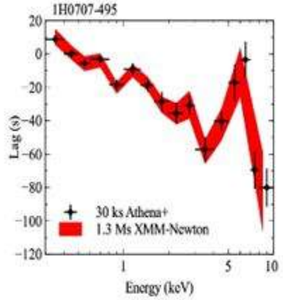
1H0707-495





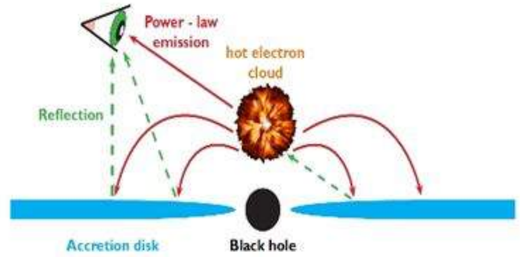
IC4329A

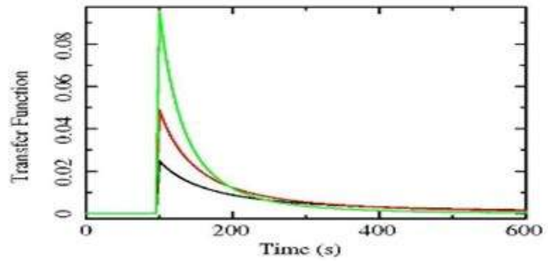


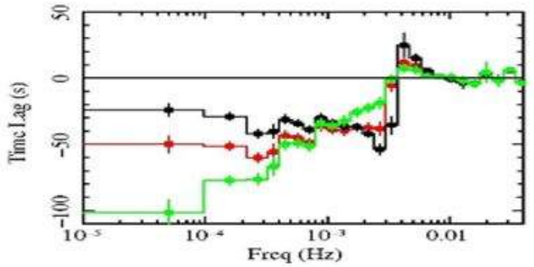


Corona/disk structure of (Reverberation mapping)

Time lag between 1-4keV and 0.3-1keV, Comparison between Athena and XMM

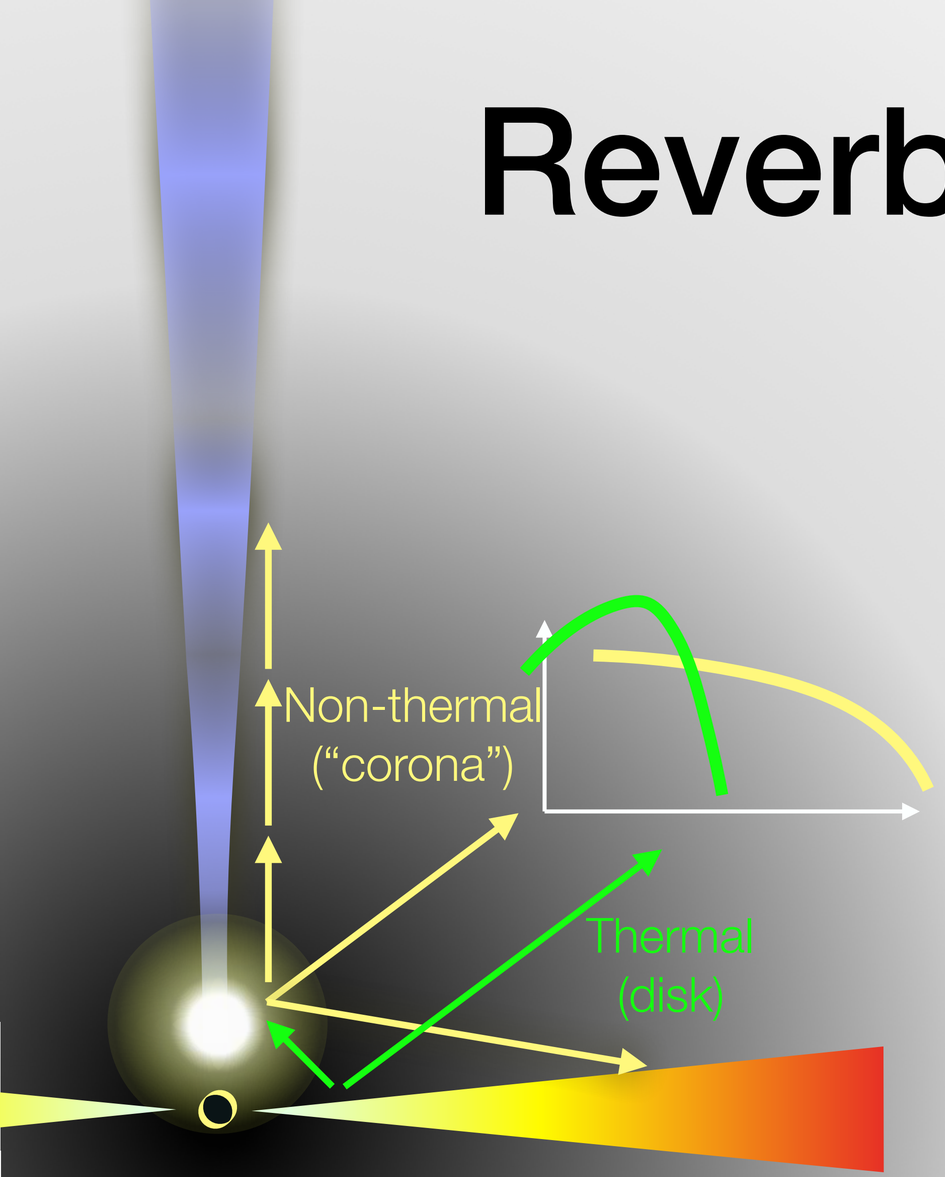




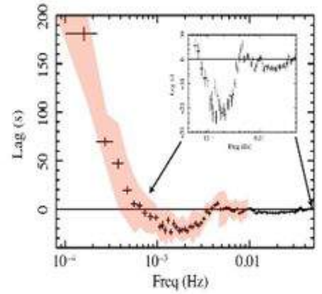


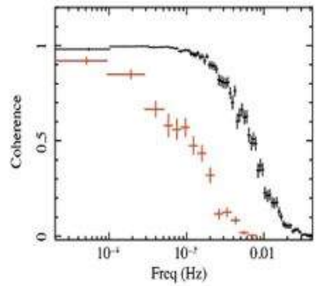
1H0707-495: time lag to probe the transfer structure of corona

Reverberation mapping

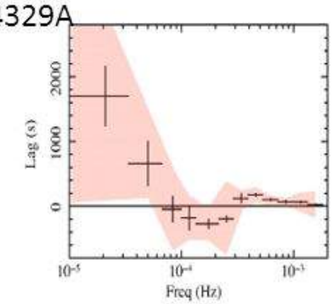


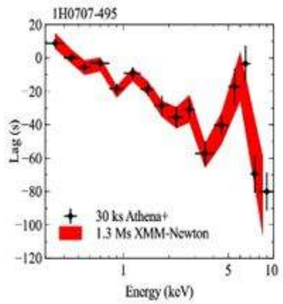
1H0707-495





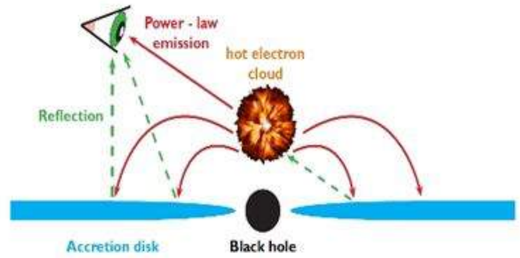
IC4329A

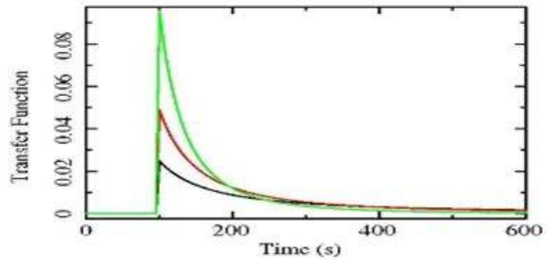


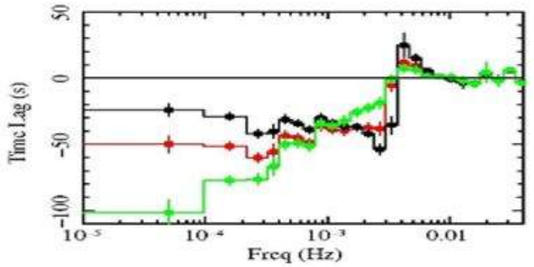


Corona/disk structure of (Reverberation mapping)

Time lag between 1-4keV and 0.3-1keV, Comparison between Athena and XMM

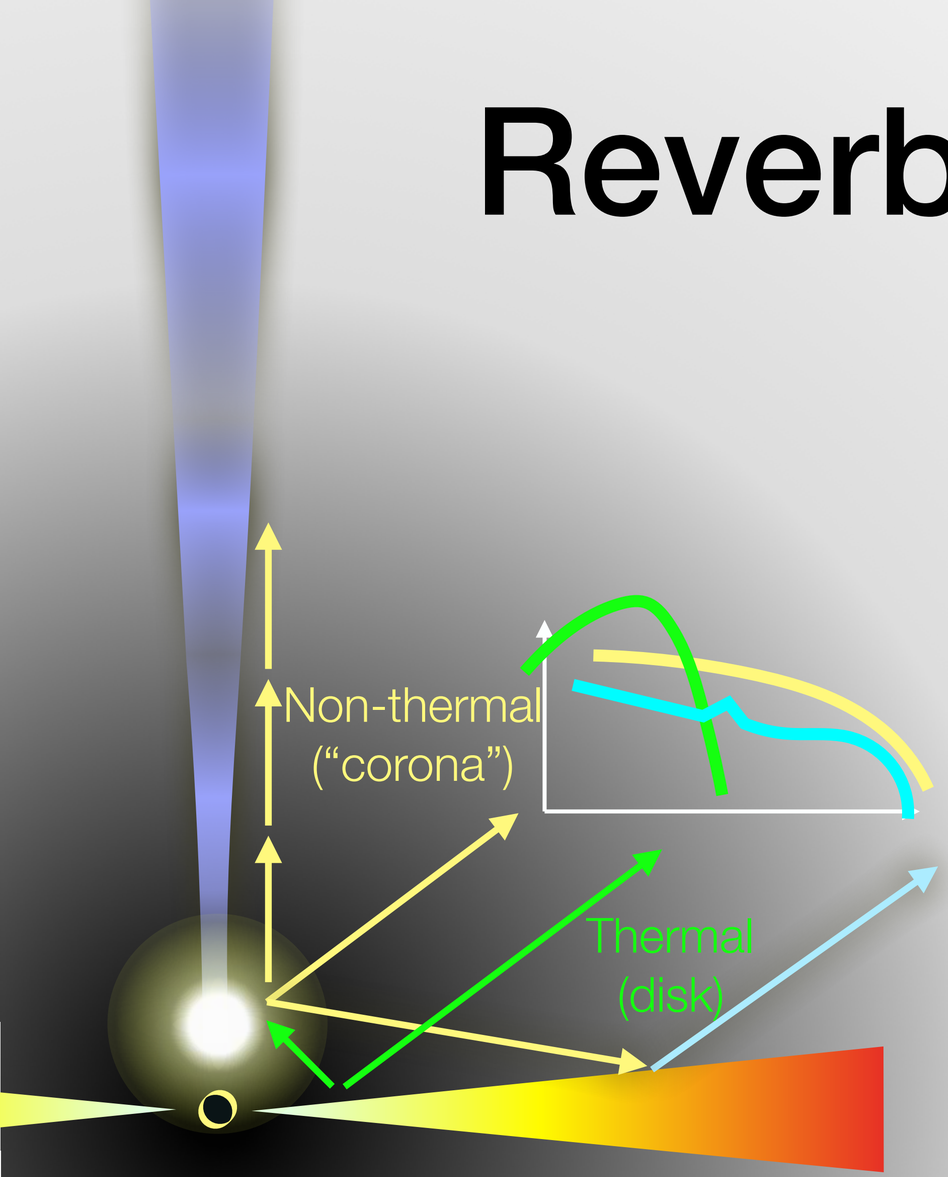




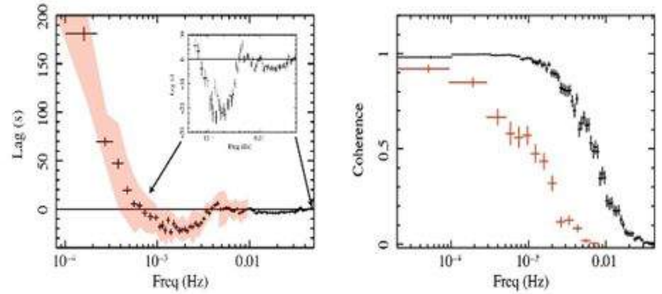


1H0707-495: time lag to probe the transfer structure of corona

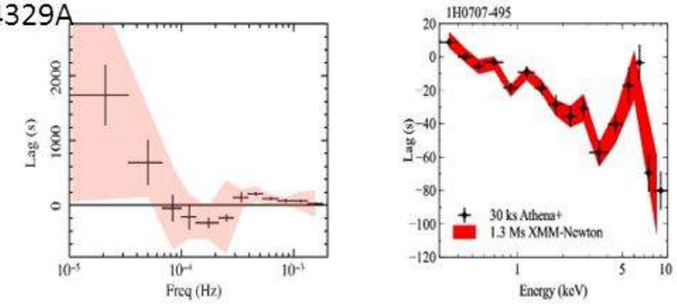
Reverberation mapping



1H0707-495

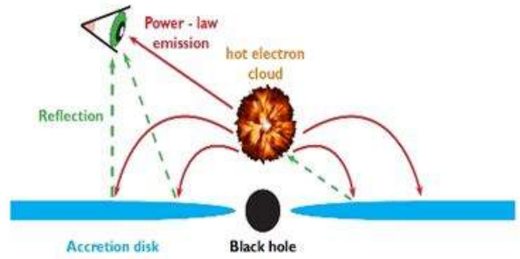


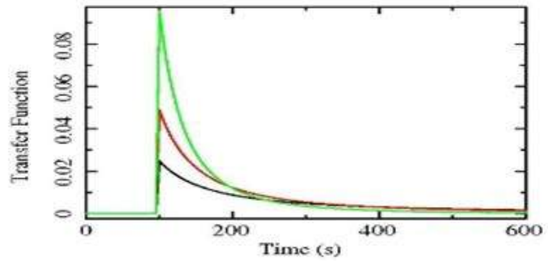
IC4329A

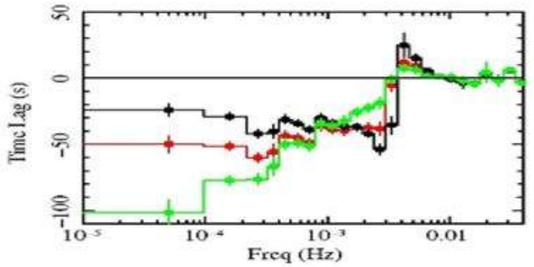


Corona/disk structure of (Reverberation mapping)

Time lag between 1-4keV and 0.3-1keV, Comparison between Athena and XMM







1H0707-495: time lag to probe the transfer structure of corona

Existing “public” software
(i.e. you can look at the code and read docs)

Spectral analysis	Timing analysis (+ lags)	Spectral timing

- Xspec
- Sherpa
- ISIS
- (…)

Existing “public” software

(i.e. you can look at the code or at least read docs)

Spectral analysis	Timing analysis (+ lags)	Spectral timing
• Xspec • Sherpa • ISIS • (…)	• (XRONOS) POWSPEC • SITAR, lsisscripts.sl	

Existing “public” software

(i.e. you can look at the code or at least read docs)

Spectral analysis	Timing analysis (+ lags)	Spectral timing
• Xspec • Sherpa • ISIS • (…)	• (XRONOS) POWSPEC ⚠ • SITAR, lsisscripts.sl	

Existing “public” software
(i.e. you can look at the code and read docs)

Spectral analysis	Timing analysis (+ lags)	Spectral timing

- Xspec
- Sherpa
- ISIS
- (…)

- (XRONOS) POWSPEC
- SITAR, lsisscripts.sl

Existing “public” software
(i.e. you can look at the code and read docs)

Spectral analysis	Timing analysis (+ lags)	Spectral timing

- Xspec
- Sherpa
- ISIS
- (…)

- (XRONOS) POWSPEC
- SITAR, lsisscripts.sl

Lorentz center

The X-ray Spectral-Timing Revolution

Workshop: 1 - 5 February 2016 Leiden, the Netherlands

Scientific Organizers

- Ed Cackett, Wayne State U
- Chris Done, Durham U
- Andy Fabian, U Cambridge
- Barbara De Marco, MPE Garching
- Phil Uttley, U Amsterdam

Stingray
Daniela Huppenkothen,
Abigail Stevens

MaLTPyNT
(in-development
Astropy-affiliated)
Matteo Bachetti

GUI for timing
Simone Migliari, Paul Balm

Stingray
Daniela Huppenkothen,
Abigail Stevens

MaLTPyNT
(in-development
Astropy-affiliated)
Matteo Bachetti

GUI for timing
Simone Migliari, Paul Balm

Stingray
Python API

HENDRICS
CLI to Stingray

DAVE
GUI to Stingray

Stingray
Daniela Huppenkothen,
Abigail Stevens

MaLTPyNT
(in-development
Astropy-affiliated)
Matteo Bachetti

GUI for timing
Simone Migliari, Paul Balm

Stingray
Python API

astropy
affiliated

HENDRICS
CLI to Stingray

astropy
affiliated

DAVE
GUI to Stingray

README.rst

Stingray

Master	build passing	docs latest	slack 4/54	coverage 97%

X-Ray Spectral Timing Made Easy

Stingray is an in-development spectral-timing software package for astrophysical X-ray (and more) data. Stingray merges existing efforts for a (spectral-)timing package in Python, and is structured with the best guidelines for modern open-source programming, following the example of Astropy .

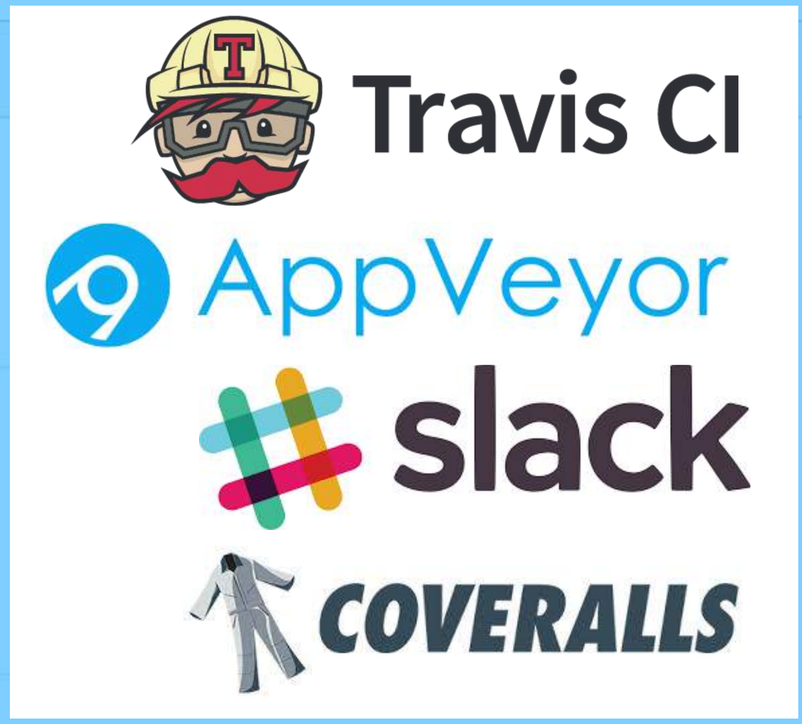
It is composed of:

1. a library of time series methods, including power spectra, cross spectra, covariance spectra, lags, and so on;
2. a set of scripts to load FITS data files from different missions;
3. a simulator of light curves and event lists, that includes different kinds of variability and more complicated phenomena based on the impulse response of given physical events (e.g. reverberation);
4. finally, an in-development GUI to ease the learning curve for new users.

README.rst

Stingray

Master	build passing	docs latest	slack 4/54	coverage 97%

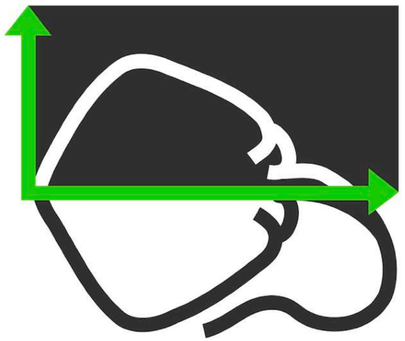


X-Ray Spectral Timing Made Easy

Stingray is an in-development spectral-timing software package for astrophysical X-ray (and more) data. Stingray merges existing efforts for a (spectral-)timing package in Python, and is structured with the best guidelines for modern open-source programming, following the example of Astropy .

It is composed of:

1. a library of time series methods, including power spectra, cross spectra, covariance spectra, lags, and so on;
2. a set of scripts to load FITS data files from different missions;
3. a simulator of light curves and event lists, that includes different kinds of variability and more complicated phenomena based on the impulse response of given physical events (e.g. reverberation);
4. finally, an in-development GUI to ease the learning curve for new users.



Stingray API

Stingray

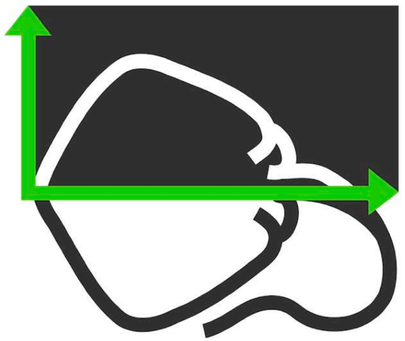
EventList

Lightcurve

Powerspectrum
Averagedpowerspectrum
Crossspectrum
Averagedcrossspectrum

Covariancespectrum
Crosscorrelation
RmsEnergySpectrum
LagEnergySpectrum

Phase-resolved spectra



Stingray API

modeling

simulator

pulse

Stingray

EventList

Lightcurve

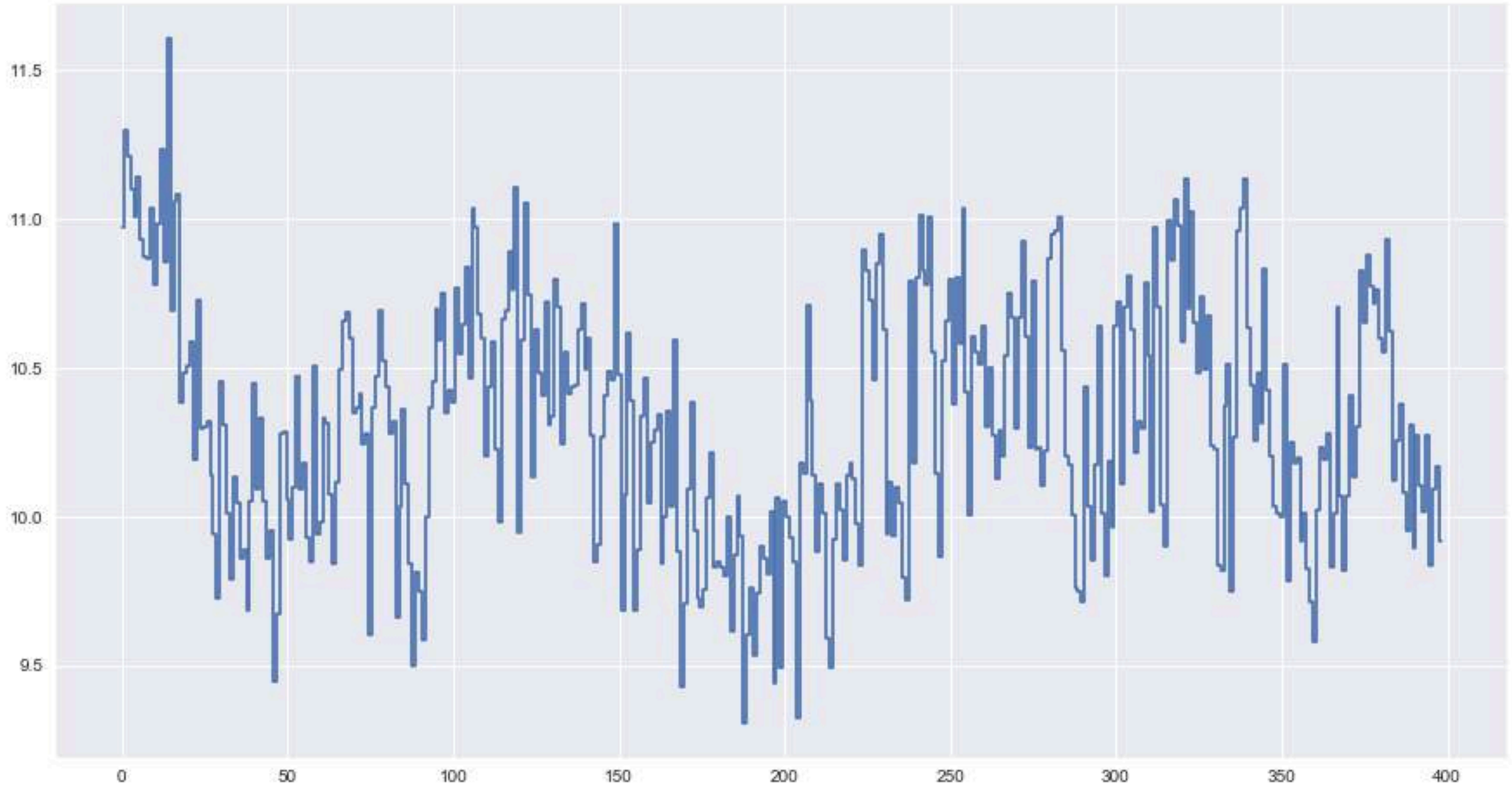
Powerspectrum
Averagedpowerspectrum
Crossspectrum
Averagedcrossspectrum

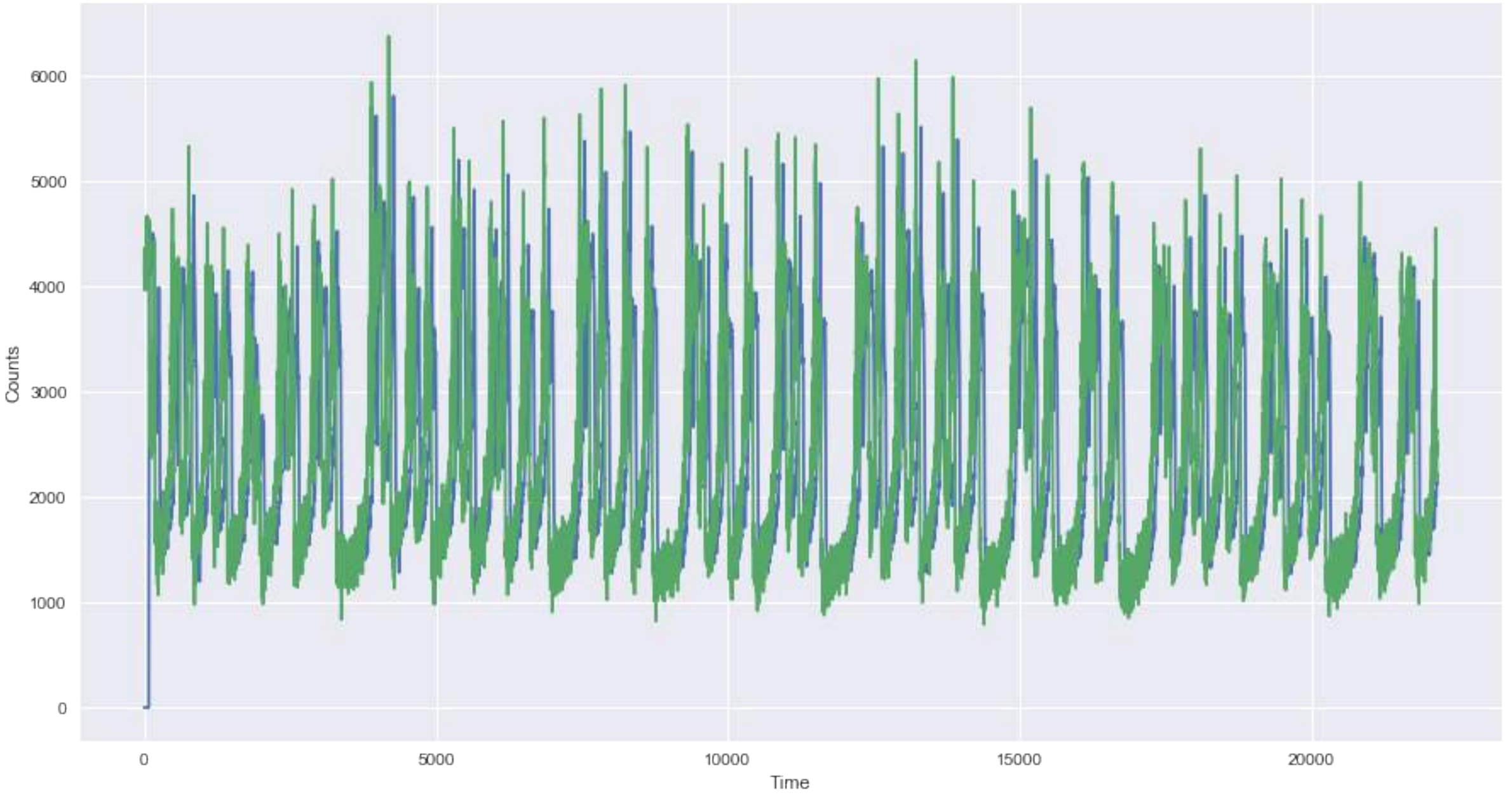
Covariancespectrum
Crosscorrelation
RmsEnergySpectrum
LagEnergySpectrum

Phase-resolved spectra

In [8]:
```
lc = sim.simulate('smoothbknpo', [.6, 0.9, .2, 4])
plt.plot(lc.counts[1:400], drawstyle='steps-mid')
```

Out[8]:
```
[<matplotlib.lines.Line2D at 0x11d260630>]
```




6000
5000
4000
3000
2000
1000
0
Counts
0
5000
10000
15000
20000
Time

usmanwardag Adds real 2-d plot — a3ebf31 on Aug 22, 2016

1 contributor

515 lines (514 sloc) | 79.5 KB | Raw | Blame | History

Contents

This notebook covers the basics of creating TransferFunction object, obtaining time and energy resolved responses, plotting them and using IO methods available. Finally, artificial responses are introduced which provide a way for quick testing.

Setup

Set up some useful libraries.

In [39]:
```
import numpy as np
from matplotlib import pyplot as plt
%matplotlib inline
```

Import relevant stingray libraries.

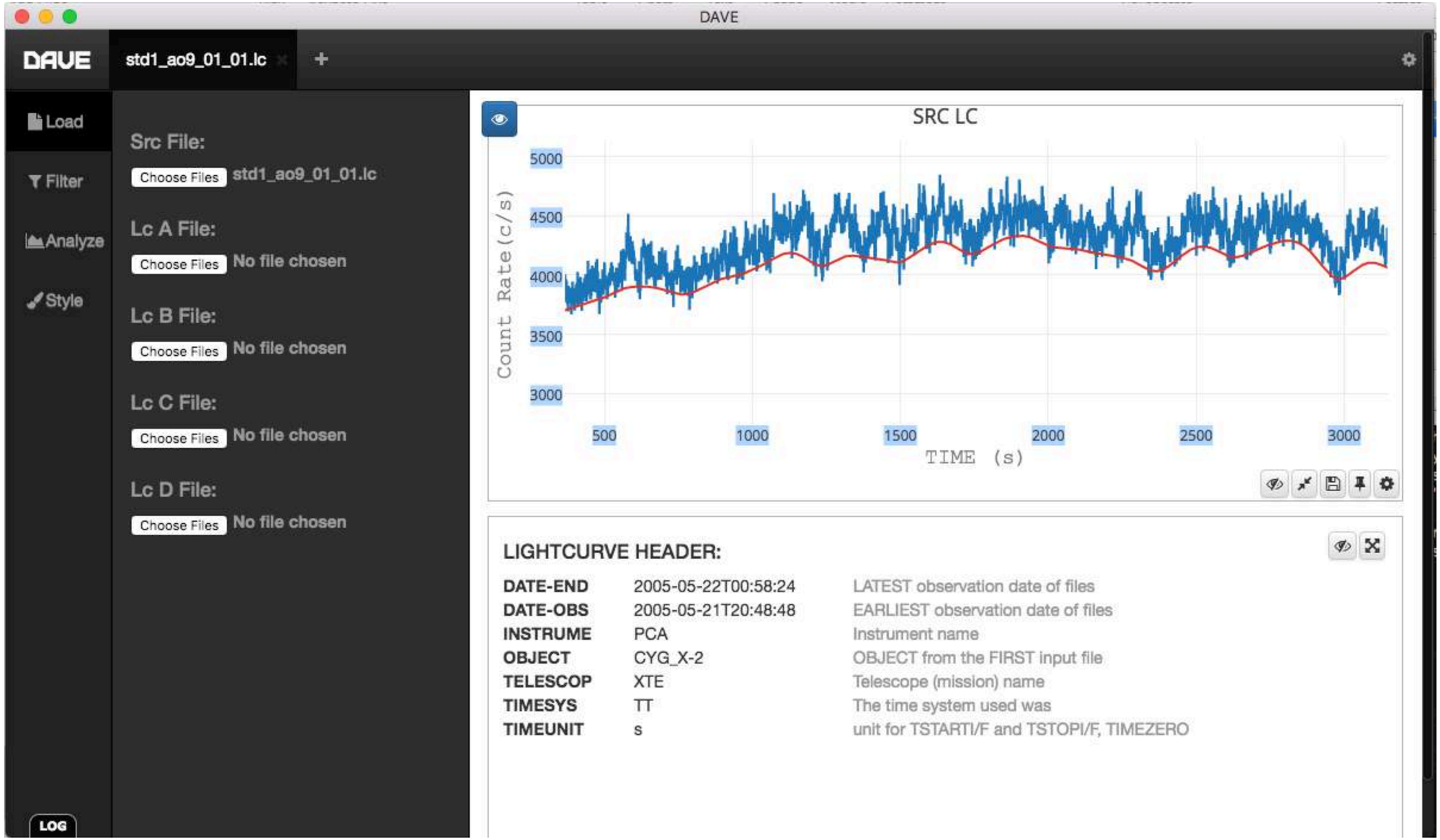
In [40]:
```
from stingray.simulator.transfer import TransferFunction
from stingray.simulator.transfer import simple_ir, relativistic_ir
```

Creating TransferFunction

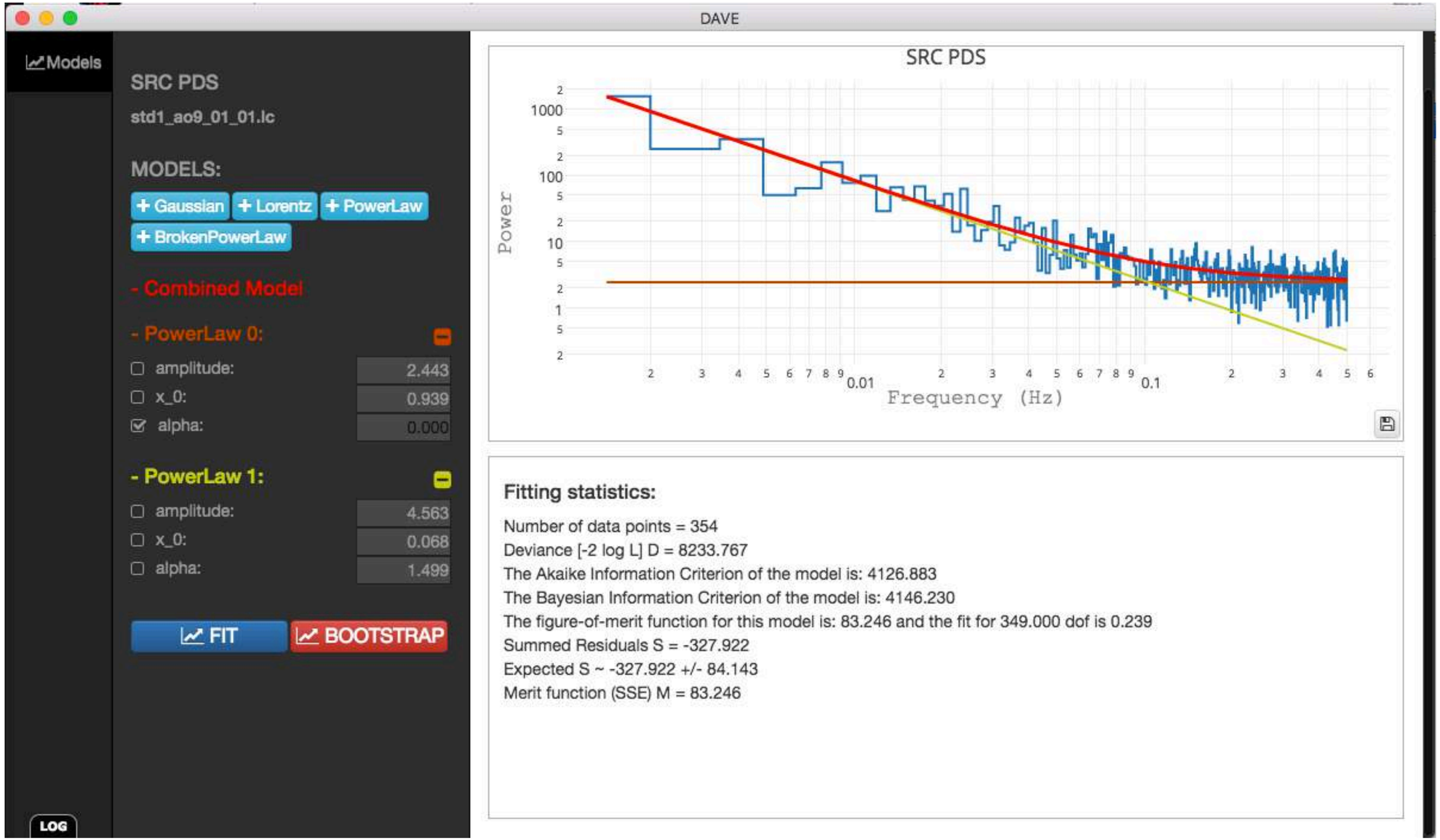

N_importe_quoi
UC1 — IPython: Dropbox/cospectra_data_share — bash — 101×28
s.py:60: UserWarning: SIMON says: Bin sizes in input time array aren't equal throughout! This could c
ause problems with Fourier transforms. Please make the input time evenly sampled.
warnings.warn("SIMON says: {0}".format(message), **kwargs)
(py3) portabile2:UC1 $ HENcolors mistery_source_1_00_nustar_fpma_ev_calib.nc -e 10 15 15 50
/Users/meo/anaconda/envs/py3/lib/python3.6/site-packages/stingray-0.1.dev1244-py3.6.egg/stingray/util
s.py:60: UserWarning: SIMON says: Bin sizes in input time array aren't equal throughout! This could c
ause problems with Fourier transforms. Please make the input time evenly sampled.
warnings.warn("SIMON says: {0}".format(message), **kwargs)
(py3) portabile2:UC1 $ ls
HENcalibrate.log
HENcolors.log
HENlcurve.log
HENreadevents.log
mistery_source_1_00.evt
mistery_source_1_00_nustar_fpma_E3-50_lc.nc
mistery_source_1_00_nustar_fpma_E_10-5_over_5-3.nc
mistery_source_1_00_nustar_fpma_E_50-15_over_15-10.nc
mistery_source_1_00_nustar_fpma_ev.nc
mistery_source_1_00_nustar_fpma_ev_calib.nc
mistery_source_1_00_nustar_fpma_std_lc.nc
(py3) portabile2:UC1 $ HENplot --CCD mistery_source_1_00_nustar_fpma_E_10-5_over_5-3.nc mistery_sourc
e_1_00_nustar_fpma_E_50-15_over_15-10.nc
/Users/meo/anaconda/envs/py3/lib/python3.6/site-packages/stingray-0.1.dev1244-py3.6.egg/stingray/util
s.py:60: UserWarning: SIMON says: Bin sizes in input time array aren't equal throughout! This could c
ause problems with Fourier transforms. Please make the input time evenly sampled.
warnings.warn("SIMON says: {0}".format(message), **kwargs)
(py3) portabile2:UC1 $ HENcolors mistery_source_1_00_nustar_fpma_ev_calib.nc -e 10 15 15 50 -b 25
HENDRICS


N_importe_quoi
UC1 — IPython: Dropbox/cospectra_data_share — bash — 101×28
s.py:60: UserWarning: SIMON says: Bin sizes in input time array aren't equal throughout! This could c
ause problems with Fourier transforms. Please make the input time evenly sampled.
warnings.warn("SIMON says: {0}".format(message), **kwargs)
(py3) portabile2:UC1 $ HENcolors mistery_source_1_00_nustar_fpma_ev_calib.nc -e 10 15 15 50
/Users/meo/anaconda/envs/py3/lib/python3.6/site-packages/stingray-0.1.dev1244-py3.6.egg/stingray/util
s.py:60: UserWarning: SIMON says: Bin sizes in input time array aren't equal throughout! This could c
ause problems with Fourier transforms. Please make the input time evenly sampled.
warnings.warn("SIMON says: {0}".format(message), **kwargs)
(py3) portabile2:UC1 $ ls
HENcalibrate.log
HENcolors.log
HENlcurve.log
HENreadevents.log
mistery_source_1_00.evt
mistery_source_1_00_nustar_fpma_E3-50_lc.nc
mistery_source_1_00_nustar_fpma_E_10-5_over_5-3.nc
mistery_source_1_00_nustar_fpma_E_50-15_over_15-10.nc
mistery_source_1_00_nustar_fpma_ev.nc
mistery_source_1_00_nustar_fpma_ev_calib.nc
mistery_source_1_00_nustar_fpma_std_lc.nc
(py3) portabile2:UC1 $ HENplot --CCD mistery_source_1_00_nustar_fpma_E_10-5_over_5-3.nc mistery_sourc
e_1_00_nustar_fpma_E_50-15_over_15-10.nc
/Users/meo/anaconda/envs/py3/lib/python3.6/site-packages/stingray-0.1.dev1244-py3.6.egg/stingray/util
s.py:60: UserWarning: SIMON says: Bin sizes in input time array aren't equal throughout! This could c
ause problems with Fourier transforms. Please make the input time evenly sampled.
warnings.warn("SIMON says: {0}".format(message), **kwargs)
(py3) portabile2:UC1 $ HENcolors mistery_source_1_00_nustar_fpma_ev_calib.nc -e 10 15 15 50 -b 25
HENDRICS

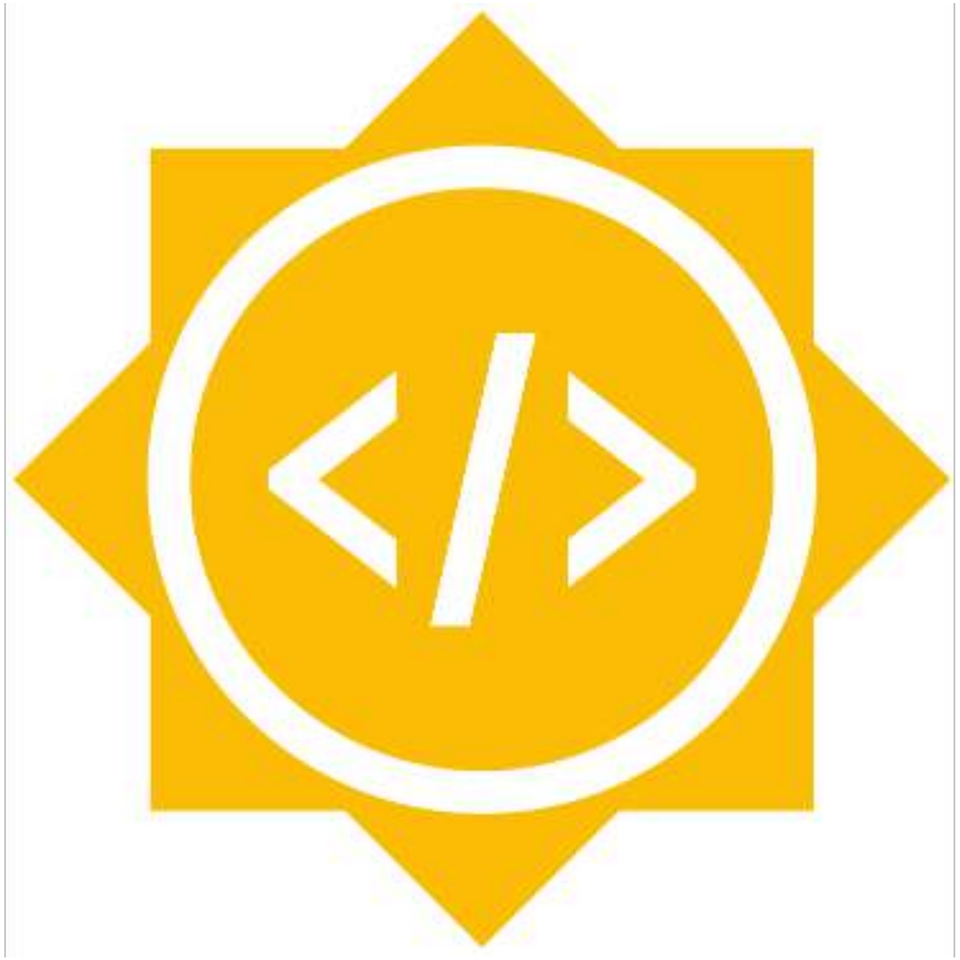
DAVE


DAVE
DAVE
std1_ao9_01_01.lc
Load
Filter
Analyze
Style
LOG
Src File:
Choose Files
std1_ao9_01_01.lc
Lc A File:
Choose Files
No file chosen
Lc B File:
Choose Files
No file chosen
Lc C File:
Choose Files
No file chosen
Lc D File:
Choose Files
No file chosen
SRC LC
Count Rate(c/s)
5000
4500
4000
3500
3000
500
1000
1500
2000
2500
3000
TIME (s)
LIGHTCURVE HEADER:
DATE-END 2005-05-22T00:58:24 LATEST observation date of files
DATE-OBS 2005-05-21T20:48:48 EARLIEST observation date of files
INSTRUME PCA Instrument name
OBJECT CYG_X-2 OBJECT from the FIRST input file
TELESCOP XTE Telescope (mission) name
TIMESYS TT The time system used was
TIMEUNIT s unit for TSTARTI/F and TSTOPI/F, TIMEZERO

DAVE


DAVE
Models
SRC PDS
std1_ao9_01_01.lc
MODELS:
+ Gaussian
+ Lorentz
+ PowerLaw
+ BrokenPowerLaw
- Combined Model
- PowerLaw 0:
amplitude: 2.443
x_0: 0.939
alpha: 0.000
- PowerLaw 1:
amplitude: 4.563
x_0: 0.068
alpha: 1.499
FIT
BOOTSTRAP
LOG
SRC PDS
Power
Frequency (Hz)
0.01
0.1
Fitting statistics:
Number of data points = 354
Deviance [-2 log L] D = 8233.767
The Akaike Information Criterion of the model is: 4126.883
The Bayesian Information Criterion of the model is: 4146.230
The figure-of-merit function for this model is: 83.246 and the fit for 349.000 dof is 0.239
Summed Residuals S = -327.922
Expected S ~ -327.922 +/- 84.143
Merit function (SSE) M = 83.246

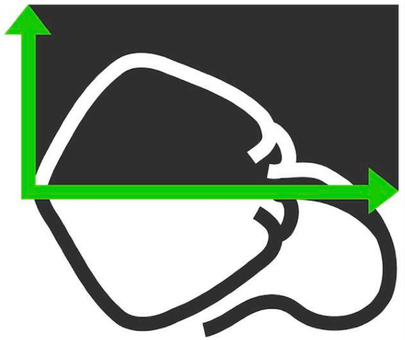
Google Summer of Code

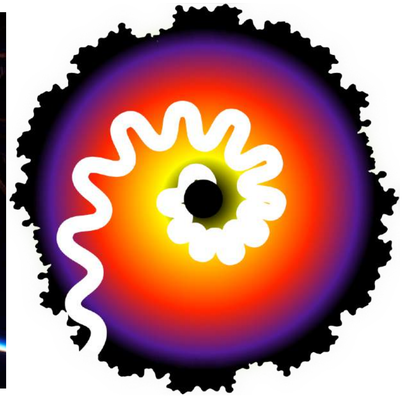


- **2018 (Openastronomy)**
  Selection in progress

- 2017 (Python Software Foundation)
  Omar Hammad
  Haroon Rashid

- 2016 (as Timelab)
  Himanshu Misra
  Usman Khan
  Danish Sodhi





- **Stingray** + **DAVE** + “**HENDRICS**”:
  Spectral timing, for everyone:
  1. Python **API**: for the brave
  2. **GUI**: shallow learning curve
  2. **Shell Scripts**: batch processing

- Open source, **BSD** and **Apache 2**

-