

Introduction to SciPy

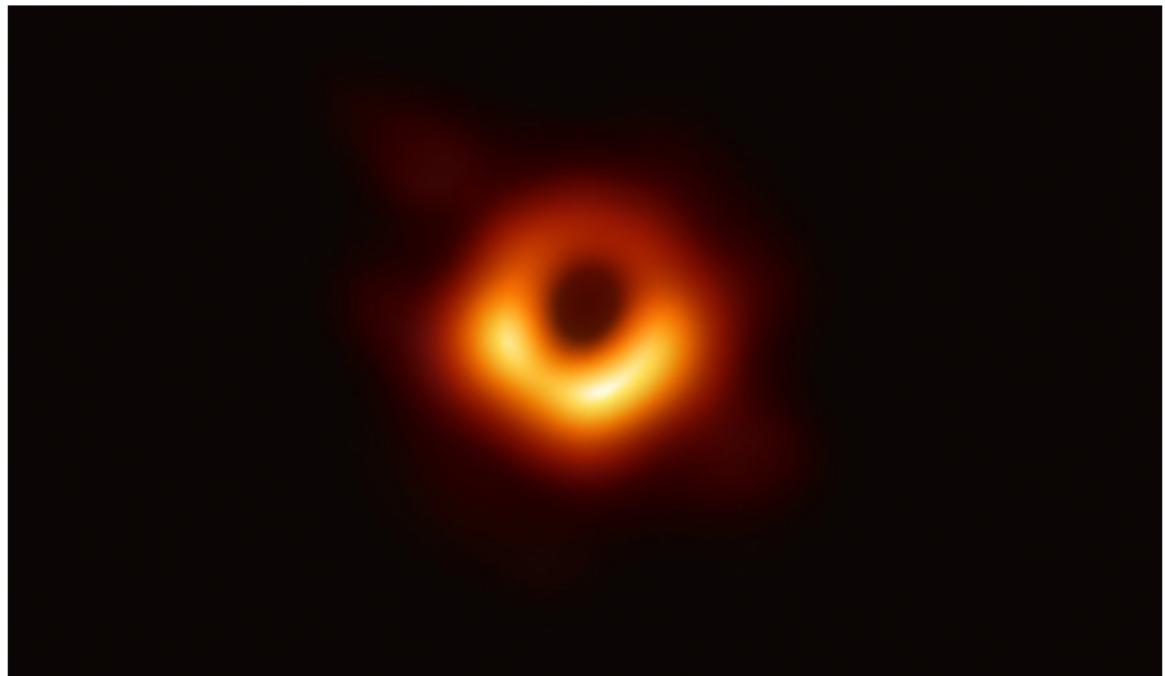
Gert-Ludwig Ingold



<https://github.com/gertingold/euroscipy-scipy-tutorial.git>

Event Horizon Telescope

supermassive black hole in M87



credit: EHT Collaboration

Event Horizon Telescope

THE ASTROPHYSICAL JOURNAL LETTERS, 875:L3 (32pp), 2019 April 10

© 2019. The American Astronomical Society.

OPEN ACCESS

<https://doi.org/10.3847/2041-8213/ab0c57>



First M87 Event Horizon Telescope Results. III. Data Processing and Calibration

The Event Horizon Telescope Collaboration

(See the end matter for the full list of authors.)

Received 2019 February 11; revised 2019 March 3; accepted 2019 March 3; published 2019 April 10

Abstract

We present the calibration and reduction of Event Horizon Telescope (EHT) 1.3 mm radio wavelength observations of the supermassive black hole candidate at the center of the radio galaxy M87 and the quasar 3C 279, taken during the 2017 April 5–11 observing campaign. These global very long baseline interferometric

ACKNOWLEDGEMENTS. We acknowledge the significance that Maunakea, where the SMA and JCMT EHT stations are located, has for the indigenous Hawaiian people.

Facilities: EHT, ALMA, APEX, IRAM:30 m, JCMT, LMT, SMA, ARO:SMT, SPT.

Software: DiFX (Deller et al. 2011), CALC, PolConvert (Martí-Vidal et al. 2016), HOPS (Whitney et al. 2004), CASA (McMullin et al. 2007), AIPS (Greisen 2003), ParselTongue (Kettenis et al. 2006), GNU Parallel (Tange 2011), GILDAS, eht-imaging (Chael et al. 2016, 2018), Numpy (van der Walt

et al. 2011), Scipy (Jones et al. 2001), Pandas (McKinney 2010), Astropy (The Astropy Collaboration et al. 2013, 2018), Jupyter (Kluyver et al. 2016), Matplotlib (Hunter 2007).

Appendix Site and Data Issues

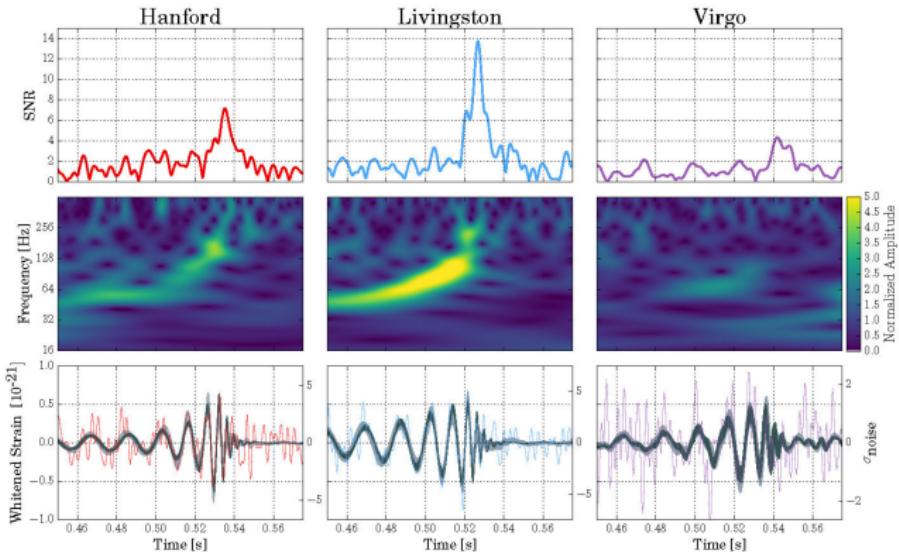
A.1. Issues Requiring Mitigation

The JCMT and SMA are located within hundreds of meters of each other on Maunakea. The small natural seeing due to



keynote on Thursday by Sara Issaoun

Gravitational waves



credit: LIGO/Caltech/MIT/LSC

☞ Leo Singer, SciPy 2018 keynote

Role of Python in Recent Gravitational Wave Astronomy Breakthroughs
with LIGO and Virgo

<https://www.youtube.com/watch?v=PiZ0gxAiGuU>

Gravitational waves

PHYSICAL REVIEW D 93, 122003 (2016)



GW150914: First results from the search for binary black hole coalescence with Advanced LIGO

B. P. Abbott *et al.*^{*}

(LIGO Scientific Collaboration and Virgo Collaboration)

(Received 9 March 2016; published 7 June 2016)

On September 14, 2015, at 09:50:45 UTC the two detectors of the Laser Interferometer Gravitational-Wave Observatory (LIGO) simultaneously observed the binary black hole merger GW150914. We report the results of a matched-filter search using relativistic models of compact-object binaries that recovered

masses from $1 M_{\odot}$ to $99 M_{\odot}$, total mass less than $100 M_{\odot}$ and dimensionless spins up to 0.99. The search was performed using two independently implemented analyses, referred to as PyCBC [3–5] and GstLAL [6–8]. These analyses use a common set of template waveforms [9–11], but differ in their implementations of matched filtering

```
requires: @gstreamername -> csp_server >= @MIN_GSTREAMER_VERSION@
Requires: lal >= @MIN_LAL_VERSION@
Requires: lalburst >= @MIN_LALBURST_VERSION@
Requires: lalmetaio >= @MIN_LALMETAIO_VERSION@
Requires: lalinpiral >= @MIN_LALINPIRAL_VERSION@
Requires: lalsimulation >= @MIN_LALSIMULATION_VERSION@
Requires: numpy > @MIN_NUMPY_VERSION@
Requires: orc >= @MIN_ORC_VERSION@
Requires: python >= @MIN_PYTHON_VERSION@
Requires: python-%{gstreamername}
Requires: python-gobject >= @MIN_PYGOBJECT_VERSION@
Requires: python2-lal >= @MIN_LAL_VERSION@
Requires: python-ligo-lw >= @MIN_LIGO_LW_VERSION@
Requires: python2-ligo-segments >= @MIN_LIGO_SEGMENTS_VERSION@
Requires: scipy
Requires: zlib
BuildRequires: doxygen >= @MIN_DOXYGEN_VERSION@
```

The screenshot shows a GitHub repository page for `gwastro / pycbc`. The repository has 80 issues and 23 pull requests. The current branch is `master`, and the file being viewed is `pycbc / requirements.txt`. The file contains 43 lines (38 sloc) and is 1020 Bytes. The code listing shows dependencies for various Python packages:

```
1 # requirements for most basic library use
2 astropy>=2.0.3,<3.0.0; python_version <= '2.7'
3 astropy>=2.0.3; python_version >= '3.0'
4 Makro>=1.0.1
5 decorator>=3.4.2
6 scipy>=0.16.0; python_version >= '3.5'
7 scipy>=0.16.0,<1.3.0; python_version <= '3.4'
8 matplotlib>=2.0.0
9 numpy>=1.13.0,<1.15.3; python_version <= '2.7'
10 numpy>=1.13.0; python_version >= '3.0'
11 nillow
```

Tutorial on the scientific Python ecosystem



Gaël Varoquaux • Emmanuelle Gouillart • Olav Vahters • Pierre de Buyl
Christopher Burns • Adrian Chauve • Robert Cimrman • Christophe Combelles
Ralf Gommers • André Espaze • Zbigniew Jędrzejewski-Szmcık
Valentin Haenel • Gert-Ludwig Ingold • Fabian Pedregosa • Didrik Pinte
Nicolas P. Rougier • Joris Van den Bossche • Pauli Virtanen

and many others...



NumPy/SciPy Documentation

docs.scipy.org



SciPy

Numpy and Scipy Documentation

Welcome! This is the documentation for NumPy and Scipy.

For contributors:

- [NumPy developer guide](#)
- [Scipy developer guide](#)

Latest releases:

[Complete NumPy Manual](#)
[HTML|zip]

[NumPy Reference Guide](#)
[PDF]

[NumPy User Guide](#)
[PDF]

[F2Py Guide](#)

[Scipy Reference Guide](#)
[HTML|zip], [PDF]

Others:

For newer versions see
<https://www.numpy.org/doc/>

[NumPy \(development version\) Reference Guide](#)
[PDF]

[NumPy \(development version\) User Guide](#)
[PDF]

[NumPy 1.17.0 Reference Guide](#), [HTML|zip], [PDF]

[NumPy 1.17.0 User Guide](#), [PDF]

NumPy v. 1.18.0 Reference Guide, [HTML|zip] | NumPy v. 1.17.0 User Guide, [HTML|zip]

SEE ALSO:
[Mailing Lists](#) ↗
[Q & A](#) ↗
[stackoverflow.com](#) ↗
[Scipy Lecture Notes](#) ↗



[SciPy.org](#) | [docs](#)

[index](#) | [modules](#) | [next](#)

Table of Contents

- [SciPy](#)
 - [Tutorial](#)
 - [Developer's Guide](#)
 - [API References](#)

Next topic

[Installing and upgrading](#)

Quick search

search

Release: 1.3.0
Date: May 17, 2019

SciPy (pronounced "Sigh Pie") is open-source software for mathematics, science, and engineering.

- [Installing and upgrading](#)
- [API: importing from SciPy](#)
- [Release Notes](#)

Tutorial

Tutorials with worked examples and background information for most SciPy submodules.

- [SciPy Tutorial](#)
 - [Introduction](#)
 - [Basic functions](#)
 - [Special functions \(scipy.special\)](#)
 - [Integration \(scipy.integrate\)](#)
 - [Optimization \(scipy.optimize\)](#)
 - [Interpolation \(scipy.interpolate\)](#)
 - [Fourier transforms \(scipy.fftpack\)](#)
 - [Signal processing \(scipy.signal\)](#)
 - [Linear Algebra \(scipy.linalg\)](#)
 - [Sparse Eigenvalue Problems with ARPACK](#)
 - [Compressed Sparse Graph Routines \(scipy.sparse.csgraph\)](#)
 - [Spatial data structures and algorithms \(scipy.spatial\)](#)
 - [Statistics \(scipy.stats\)](#)
 - [Multidimensional image processing \(scipy.ndimage\)](#)
 - [File I/O \(scipy.io\)](#)

Developer's Guide

Explanations of how to start contributing to SciPy, and descriptions of maintenance activities and policies.

- [SciPy Code of Conduct](#)
- [Code of Ethics](#)
- [Building from sources](#)
- [SciPy Developer Guide](#)
- [SciPy project governance](#)

To get an overview of where help or new features are desired or planned, see the roadmap:

- [SciPy Roadmap](#)
- [Detailed SciPy Roadmap](#)
- [Toolchain Roadmap](#)

API Reference

The exact API of all functions and classes, as given by the doctests. The API documents expected types and allowed features for all functions, and all parameters available for the algorithms.

- [Clustering package \(scipy.cluster\)](#)
- [Constants \(scipy.constants\)](#)
- [Miscellaneous constants \(scipy.constants\)](#)
- [Integration and ODEs \(scipy.integrate\)](#)
- [Interpolation \(scipy.interpolate\)](#)
- [Input and output \(scipy.io\)](#)
- [Linear algebra \(scipy.linalg\)](#)
- [Miscellaneous routines \(scipy.misc\)](#)
- [Multidimensional image processing \(scipy.ndimage\)](#)
- [Orthogonal distance regression \(scipy.odr\)](#)
- [Optimization and Root Finding \(scipy.optimize\)](#)
- [Signal processing \(scipy.signal\)](#)
- [Sparse matrices \(scipy.sparse\)](#)
- [Sparse linear algebra \(scipy.sparse.linalg\)](#)
- [Spatial data structures and algorithms \(scipy.spatial\)](#)
- [Special functions \(scipy.special\)](#)
- [Statistical functions \(scipy.stats\)](#)
- [Statistical functions for masked arrays \(scipy.stats.mstats\)](#)
- [Low-level callback functions](#)

A brief history of SciPy

SciPy 1.0—Fundamental Algorithms for Scientific Computing in Python

Pauli Virtanen¹, Ralf Gommers^{2,*}, Travis E. Oliphant^{3,4,5,6,2}, Matt Haberland^{7,8,*}, Tyler Reddy^{9,*}, David Cournapeau¹⁰, Evgeni Burovski¹¹, Pearu Peterson^{12,13}, Warren Weckesser¹⁰, Jonathan Bright¹⁴, Stéfan J. van der Walt¹⁵, Matthew Brett¹⁶, Joshua Wilson¹⁰, K. Jarrod Millman^{15,17}, Nikolay Mayorov¹⁸, Andrew R. J. Nelson¹⁹, Eric Jones⁵, Robert Kern⁵, Eric Larson²⁰, CJ Carey²¹, İlhan Polat¹⁰, Yu Feng²², Eric W. Moore²³, Jake VanderPlas²⁴, Denis Laxalde¹⁰, Josef Perktold¹⁰, Robert Cimrman²⁵, Ian Henriksen²⁶, E. A. Quintero¹⁰, Charles R Harris¹⁰, Anne M. Archibald²⁷, Antônio H. Ribeiro²⁸, Fabian Pedregosa²⁹, Paul van Mulbregt³⁰, and SciPy 1.0 Contributors

<https://arxiv.org/abs/1907.10121>

SciPy packages

cluster	clustering package
constants	constants
fftpack	discrete Fourier transforms
integrate	integration and ordinary differential equations
interpolate	interpolation
io	input and output
linalg	linear algebra
misc	miscellaneous routines
ndimage	multi-dimensional image processing
odr	orthogonal distance regression
optimize	optimization and root finding
signal	signal processing
sparse	sparse matrices
spatial	spatial algorithms and data structures
special	special functions
stats	statistical functions

SciPy packages

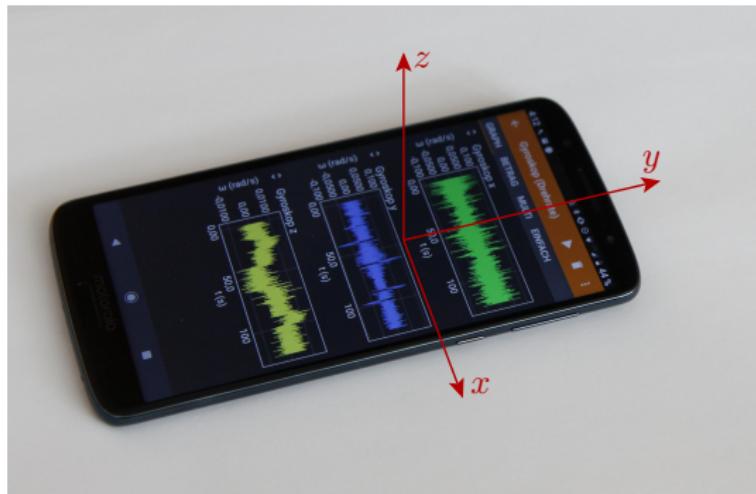
cluster	clustering package
constants	constants
fftpack	discrete Fourier transforms
integrate	integration and ordinary differential equations
interpolate	interpolation
io	input and output
linalg	linear algebra
misc	miscellaneous routines
ndimage	multi-dimensional image processing
odr	orthogonal distance regression
optimize	optimization and root finding
signal	signal processing
sparse	sparse matrices
spatial	spatial algorithms and data structures
special	special functions
stats	statistical functions

Taking data with the smartphone



smartphone app developed by the
2nd Institute of Physics of the RWTH Aachen University
see phyphox.org for more information

The screenshot shows the phyphox app's main menu. Under 'Raw Sensors', there are icons and descriptions for Acceleration (without g), Acceleration with g, Gyroscope (rotation rate), Light, Location (GPS), Magnetometer, and Pressure. Under 'Saved experiment states', there are five entries: 'Messung 04.07.19 10:05 nac...', 'Messung 04.07.19 6:51 nach...', 'Messung 10.07.19 6:57 vorm.', 'Messung 14.06.19 8:28 nach...', and 'Messung 22.06.19 2:03 nach...'. A yellow '+' button is at the bottom right.



TGV Duplex

