

iminuit and MINUIT2 Standalone

Hans Dembinski¹ for the **iminuit** authors Henry Schreiner²

ROOT Users Workshop 2018
Sarajevo, Bosnia and Herzegovina
March 22, 2018

¹MPIK Heidelberg

²University of Cincinnati

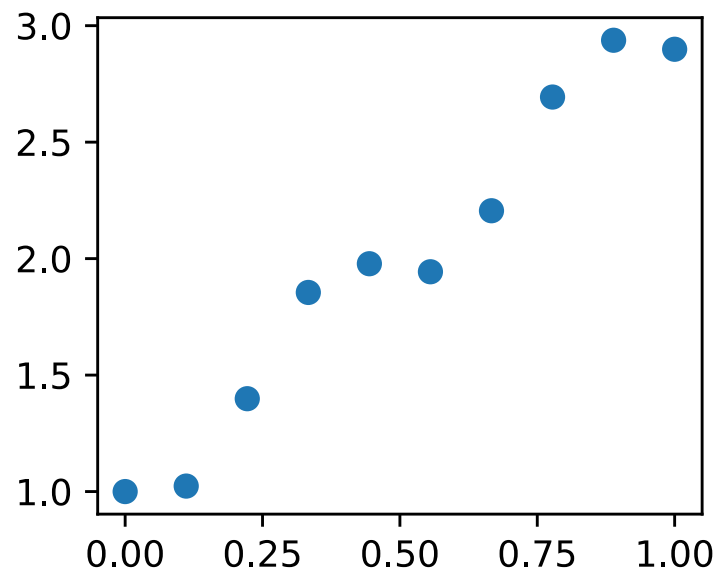


iminuit

- Interactive wrapper around C++ MINUIT2
 - ▶ Uses latest C++ code from ROOT-6.12
 - ▶ Features of MINUIT2 without ROOT dependency
 - ▶ Official successor of pyminuit & pyminuit2
- Easy to install: `pip install iminuit`
- Enhanced for interactive use and Cython compatibility
 - ▶ Nice print out in Jupyter notebooks and console
 - ▶ Simple plots built-in (using matplotlib)
- Docs: <http://iminuit.readthedocs.io>
- Issue tracker on Github (PRs welcome):
<https://github.com/iminuit/iminuit>
- Citable paper coming soon

Example: line fit

iminuit



```
def line(x, a, b):  
    return a + x*b  
  
def least_squares(a, b):  
    yvar = 0.01  
    comp_y = line(data_x, a, b)  
    return sum((data_y - comp_y)**2 / yvar)
```

Parameter names are detected automatically by **iminuit** by **introspection**

```
from iminuit import Minuit  
m = Minuit(least_squares, a=0, b=0,  
           error_a=1, error_b=1, errordef=1)  
m.print_param() # Nice printout in notebook
```

\pm	Name	Value	Hesse Error	Minos Error-	Minos Error+	Limit-	Limit+	Fixed?
0	a	0		1				No
1	b	0		1				No

Do the actual minimization

```
m.migrad()
```

```
FCN = 17.846433076964598  TOTAL NCALL = 36  NCALLS = 36
```

```
EDM = 3.992817215447363e-22  GOAL EDM = 1e-05  UP = 1.0
```

Valid	Valid Param	Accurate Covar	PosDef	Made PosDef
True	True	True	True	False
Hesse Fail	HasCov	Above EDM		Reach callim
False	True	False		False

\pm	Name	Value	Hesse Error	Minos Error-	Minos Error+	Limit-	Limit+	Fixed?
0	a	0.957618	0.0587754					No
1	b	2.07154	0.0990867					No

Access the fit values

```
m.values['a']
```

Hesse

`m.hesse()`

\pm	Name	Value	Hesse Error	Minos Error-	Minos Error+	Limit-	Limit+	Fixed?
0	a	0.957618	0.0587754					No
1	b	2.07154	0.0990867					No

\pm	a	b
a	1.00	-0.84
b	-0.84	1.00

Value access

`m.errors`

`m.covariance`

Minos

`m.minos()`

Minos status for a: **VALID**

Error	-0.0587753813648439	0.05877538136420765
Valid	True	True
At Limit	False	False
Max FCN	False	False
New Min	False	False

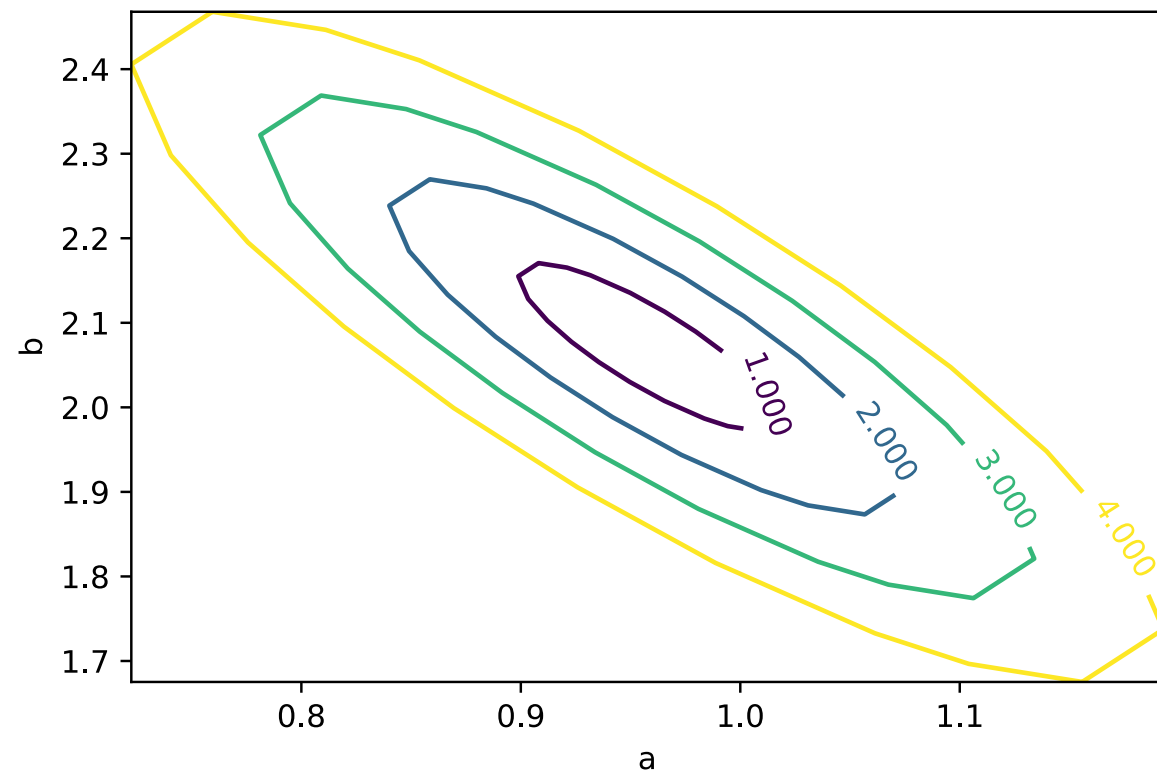
Minos status for b: **VALID**

Error	-0.09908673886192387	0.09908673886082009
Valid	True	True
At Limit	False	False
Max FCN	False	False
New Min	False	False

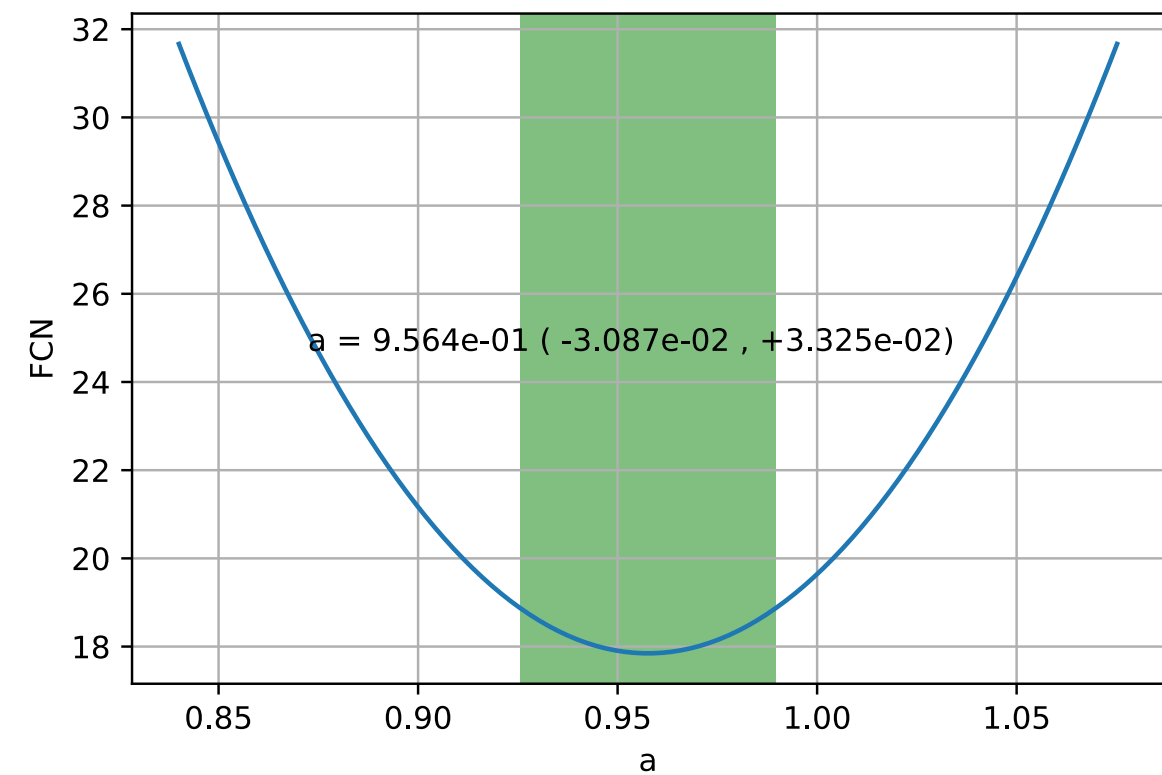
Value access

`m.merrors`

```
m.draw_mncontour('a', 'b',  
                nsigma=4)
```



```
m.draw_profile('a')
```



```
# Numpy functions are supported, too
import numpy as np

def least_squares_np(par): # par is a numpy array
    mu = np.polyval(par, data_x) # for len(par) = 2: a line
    yvar = 0.01
    return np.sum((data_y - mu) ** 2 / yvar)

m = Minuit.from_array_func(least_squares_np, (0, 0),
                          error=(1, 1), errordef=1)
m.migrad()

# Accessors for numpy arrays
m.np_values()
m.np_errors()
m.np_covariance()
```


Alternative interface to ease transition for Scipy users:

```
from iminuit import minimize
result = minimize(least_squares_np, (0, 0))
```

Same interface as `scipy.optimize.minimize!`
Result is of type `scipy.optimize.OptimizeResult`.

Standalone Minuit2

Current state of affairs

- **Previous release:** ROOT 5.20, 2008
- Poor C++11 support (warnings)
- Old build-system
- Not available via git

Many use cases for Minuit standalone

- **iminuit:** Python, pip install
- **GooFit:** GPU PDF fitter (also Python)
- **Hydra:** GPU/CPU analysis toolkit

Initial work

- Created /GooFit/Minuit2
- Some external interest
- Used internally in GooFit

Other potential users

- **TensorFlowAnalysis**
- Potential PyBind11 bindings (proof of concept in GooFit)

Use it in (your) libraries

- You can use without ROOT or even if you just have ROOT::Minuit2 missing.
- Supports all the standard ways to use a CMake libraries

Use standalone

```
add_subdirectory(minuit2)
```

Use from ROOT

```
add_subdirectory(root/math/minuit2)
```

Use prebuilt

```
find_package(Minuit2 CONFIG)
```

All methods

```
target_link_libraries(MyProgram  
    PRIVATE Minuit2::Minuit2)
```

Standard CMake builds

- Modern CMake 3.1+ build system
- Support producing source distribution without scripts
- Support add_subdirectory in and outside ROOT

```
cmake <path>
```

```
make
```

```
make install # optional
```

- <path> can be root/math/minuit2 or standalone
- make package and make package_source (read docs) work too

Package source design details

- Selected external files
 - ▶ `src/math`
 - ▶ `inc/Math`
 - ▶ `inc/Fit`
- Other files:
 - ▶ `LGPL2_1.txt`
 - ▶ `LICENSE`
 - ▶ `version_number`

Minuit2 distribution

- Sits inside ROOT 6.14+
- Submodule: (synced with master) [GitHub/GooFit/Minuit2](https://github.com/GooFit/Minuit2)
- Other downloads (source, binaries) hopefully available soon

Standalone Minuit2

CMakeLists.txt

- No conflicts: ROOT parts protected
- Options shared
- Calls `Standalone.cmake`

copy_standalone.cmake

- Can use files inplace in ROOT or standalone mode
- Copies only made if requested for creating source package

Other

- Example of CMake use in `examples/simple`
- CI tests (example too)

Tutorial documented in the book: [Modern CMake on gitlab.io](#)

README.md

- User introduction
- Instructions for building
- Instructions for CMake

DEVELOP.md

- Developer introduction
- Instructions for packaging
- Tips for maintenance

Packaged files from ROOT

- Version
- License

CMakeLists

- Lots of helpful comments

Summary

iminuit

- Powerful, pythonic interface to Minuit2
- Just a `pip install` away
- Also in Conda for macOS, Linux, and Windows

Minuit2 Standalone

- Makes it easy to include Minuit2 in a CMake project any way you want to
- Make it easy to build and use Minuit2 anywhere
- Support for macOS, Linux, and Windows
- Helpful documentation

```
conda install -c conda-forge iminuit
```

A PR is currently in progress to use standalone Minuit2 in iMinuit!

Backup

Timeline of patches, all 2018:

- Fixed Minuit2 and ROOT separation February 27
- Fixed Windows support February 28
- Standard CMake options for MPI/OpenMP (+fixes)
- **Minuit2 standalone** March 23
- Missing include in MinimumBuilder May 9
- **Standalone target name improvements** May 18
- MSVC 15 bug workaround July 19

Builds in source, no changes or copies made

```
cmake /root/math/minuit2  
make  
make install # optional
```

Library usage

```
add_subdirectory(root/math/minuit2) # Combined build  
# OR  
find_package(Minuit2 CONFIG) # Either build or install  
  
target_link_libraries(MyProgram PUBLIC Minuit2::Minuit2)
```

Builds from source package

```
cmake <root>/math/minuit2 # Optional  
make  
make install # optional
```

Library usage

```
add_subdirectory(minuit2) # Combined build  
# OR  
find_package(Minuit2 CONFIG) # Either build or install  
  
target_link_libraries(MyProgram Minuit2::Minuit2)
```

```
cmake /root/math/minuit2 -Dminuit2-standalone=ON  
make package_source  
make purge # optional
```

- Copies needed files into root/math/minuit2
- Utilizes *standard CMake technology* to make the **source** package (tar.gz and zip)
- Purge removes all copied files
- Copied files are ignored by git

```
# Linux/macOS  # Windows  
cmake ..        "C:\Program Files\CMake\bin\cmake.exe" ..  
make           "C:\Program Files\CMake\bin\cmake.exe" --build .  
make package   "C:\Program Files\CMake\bin\cmake.exe" --build .
```

- No copy needed, in or outside ROOT!
- Makes platform installers (.msi, .sh, .rpm, etc.)
- Supports Windows (tested on VC17 + OpenStack VM)