

What is Python? What's so special about it?



University at Buffalo

The State University of New York

- very-high-level programming language with many great attributes
 - scope of a **full, general-purpose programming language** (incl. scripting)
 - minimalist syntax, high **clarity/readability/compactness**
 - many complicated aspects hidden/**automated**
 - **easy debugging** during execution rather than compilation
 - supports different coding paradigms/styles (beginners to experts)
 - **extensible** design, modular setup
 - free (GPL); interactive; cross-platform portable with same interface;...
 - exceedingly **powerful, versatile, easy to learn, fun to use**



What is Python? What's so special about it?



University at Buffalo

The State University of New York

DEEP LEARNING

PYTHON



```
import deeplearning
deeplearning.train()
```

2 HOURS LATER



WOOHOO, I'M DONE.

C++



```
#include <iostream>
template <class T>
class Tensor {
public:
    :
    :
    :
    :
    :
```

5000 LINES & 5 DAYS LATER



WHAT DOES IT MEAN BY:

"error: incomplete type 'Tensor' used."

What is Python? What's so special about it?



University at Buffalo

The State University of New York

- easy access *via* Anaconda distribution

The screenshot shows the Anaconda Distribution download page. At the top, it says "Download Anaconda Distribution" with "Version 5.1 | Release Date: February 15, 2018". Below this, it says "Download For:" followed by icons for Windows, macOS, and Linux. The page is divided into three columns: "High-Performance Distribution" (Easily install 1,000+ [data science packages](#)), "Package Management" (Manage packages, dependencies and environments with [conda](#)), and "Portal to Data Science" (Uncover insights in your data and create interactive visualizations). Below these columns are buttons for "Windows", "macOS", and "Linux". The "Windows" button is selected, leading to the "Anaconda 5.1 For Windows Installer" section. This section has two options: "Python 3.6 version *" and "Python 2.7 version *". Each option has a green "Download" button. Below the "Python 3.6 version *" button, it says "64-Bit Graphical Installer (537 MB) (?)". Below the "Python 2.7 version *" button, it says "64-Bit Graphical Installer (523 MB) (?)".

Download Anaconda Distribution
Version 5.1 | Release Date: February 15, 2018

Download For:

High-Performance Distribution
Easily install 1,000+ [data science packages](#)

Package Management
Manage packages, dependencies and environments with [conda](#)

Portal to Data Science
Uncover insights in your data and create interactive visualizations

Windows macOS Linux

Anaconda 5.1 For Windows Installer

Python 3.6 version *
[Download](#)
64-Bit Graphical Installer (537 MB) (?)

Python 2.7 version *
[Download](#)
64-Bit Graphical Installer (523 MB) (?)

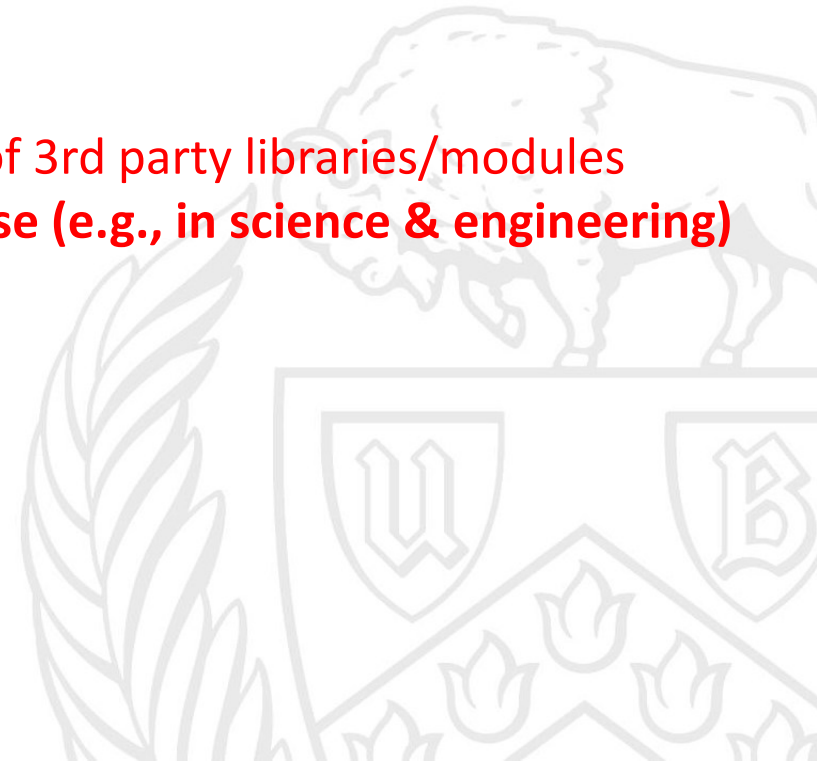
What is Python? What's so special about it?



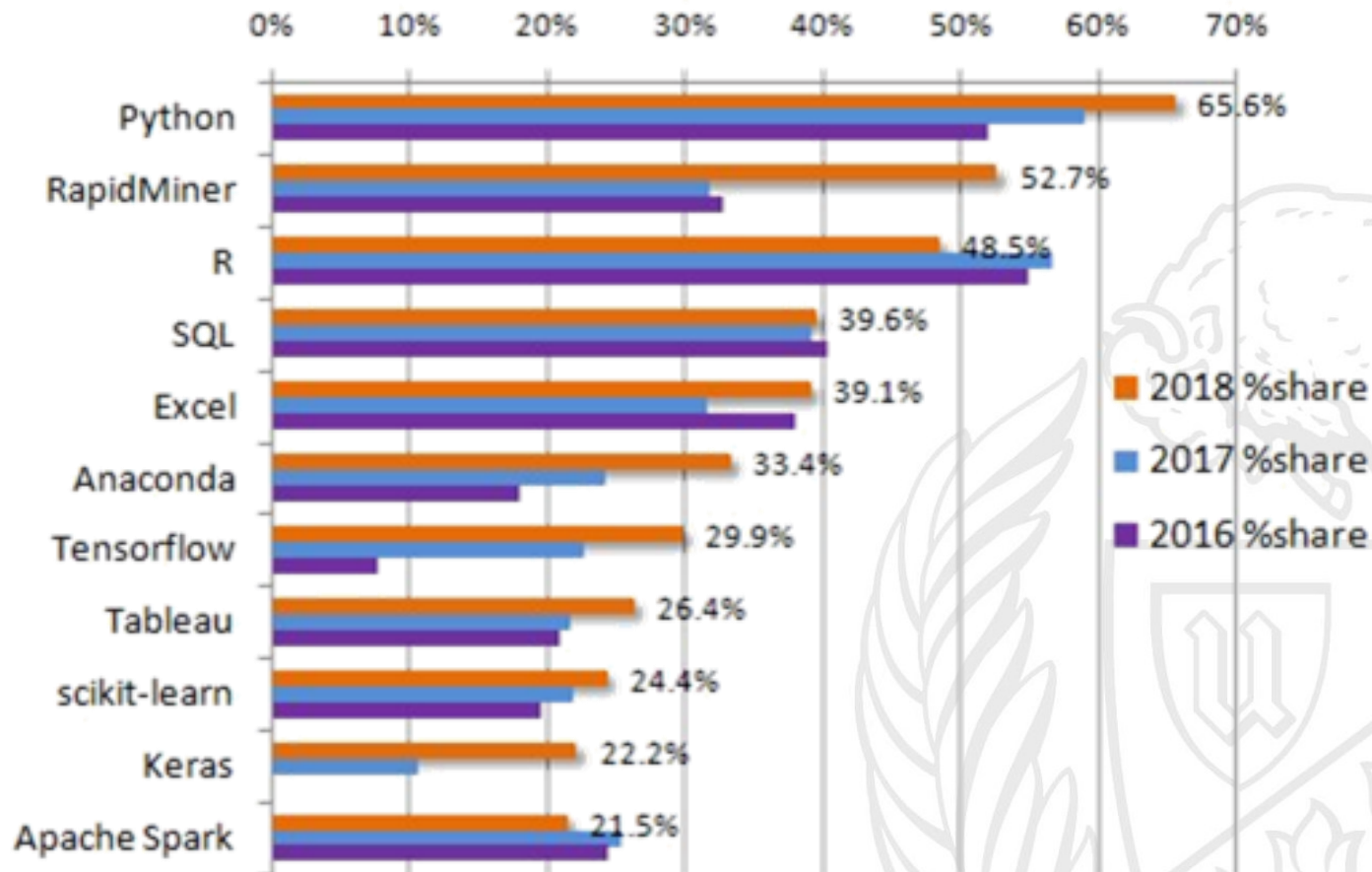
University at Buffalo

The State University of New York

- very-high-level programming language with many great attributes
 - scope of a **full, general-purpose programming language** (incl. scripting)
 - minimalist syntax, high **clarity/readability/compactness**
 - many complicated aspects hidden/**automated**
 - **easy debugging** during execution rather than compilation
 - supports different coding paradigms/styles (beginners to experts)
 - **extensible design, modular setup**
 - seamless embedding/interfacing of 3rd party libraries/modules
 - **primed for special-purpose use (e.g., in science & engineering)**



KDnuggets Analytics, Data Science, Machine Learning Software Poll, 2016-2018





SciPy (pronounced "Sigh Pie") is a Python-based ecosystem of open-source software for mathematics, science, and engineering. In particular, these are some of the core packages:



NumPy
Base N-dimensional array package



SciPy library
Fundamental library for scientific computing



Matplotlib
Comprehensive 2D Plotting



IPython
Enhanced Interactive Console



Sympy
Symbolic mathematics



pandas
Data structures & analysis

[More information...](#)

News

SciPy 0.14.0 released (2014-05-03) See [Obtaining NumPy & SciPy libraries](#).

NumPy 1.8.1 released (2014-03-26) See [Obtaining NumPy & SciPy libraries](#).

EuroSciPy 2014 [EuroSciPy](#) is the European gathering for scientists using Python. The 2014 edition will take place in Cambridge, UK, Aug. 27-31.

SciPy 2014 [SciPy](#) is an annual conference for scientists using Python. The 2014 edition will take place in Austin, Texas, July 6-12.

[About SciPy](#)
[Install](#)
[Getting Started](#)
[Documentation](#)
[Bug Reports](#)
[Topical Software](#)
[Cookbook](#) 
[SciPy Central](#) 
[Wiki](#) 
[SciPy Conferences](#) 
[Blogs](#) 
[NumFOCUS](#) 

CORE PACKAGES:

[Numpy](#) 
[SciPy library](#) 
[Matplotlib](#) 
[IPython](#) 
[SymPy](#) 
[Pandas](#) 

Search

Go



SciPy.org Docs

index modules next

SciPy

Release: 0.13.0

Date: October 21, 2013

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

- [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 IO \(`scipy.io`\)](#)
 - [Weave \(`scipy.weave`\)](#)
- [Contributing to SciPy](#)
- [API - importing from Scipy](#)
- [Release Notes](#)

Reference

- [Clustering package \(`scipy.cluster`\)](#)
- [Constants \(`scipy.constants`\)](#)
- [Discrete Fourier transforms \(`scipy.fftpack`\)](#)
- [Integration and ODEs \(`scipy.integrate`\)](#)

Table Of Contents

- [SciPy](#)
 - [Reference](#)

Next topic

[SciPy Tutorial](#)



NumPy

[Scipy.org](https://scipy.org)

NumPy

NumPy is the fundamental package for scientific computing with Python. It contains among other things:

- a powerful N-dimensional array object
- sophisticated (broadcasting) functions
- tools for integrating C/C++ and Fortran code
- useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

NumPy is licensed under the [BSD license](#), enabling reuse with few restrictions.

Getting Started

- [Getting Numpy](#)
- [Installing the SciPy Stack](#)
- [NumPy and SciPy documentation page](#)
- [NumPy Tutorial](#)
- [NumPy for MATLAB® Users](#)
- [NumPy functions by category](#)
- [NumPy Mailing List](#)

More Information

- [NumPy Sourceforge Home Page](#)
- [SciPy Home Page](#)
- [Interfacing with compiled code](#)
- [Older python array packages](#)

[About Numpy](#)

[License](#)

[Old array packages](#)

Ohloh

253K Lines



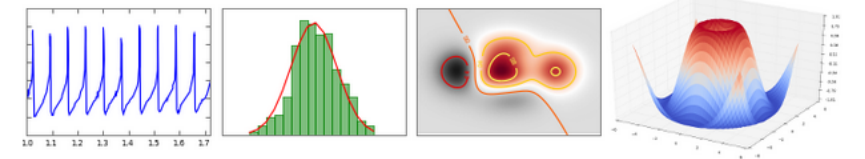
Fork me on GitHub

home | examples | gallery | pyplot | docs »

modules | index

Introduction

matplotlib is a python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. matplotlib can be used in python scripts, the python and [ipython](#) shell (ala MATLAB[®] or Mathematica[®]), web application servers, and six graphical user interface toolkits.



matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, errorcharts, scatterplots, etc, with just a few lines of code. For a sampling, see the [screenshots](#), [thumbnail](#) gallery, and [examples](#) directory

For example, using "`ipython --pylab`" to provide an interactive environment, to generate 10,000 gaussian random numbers and plot a histogram with 100 bins, you simply need to type

```
x = randn(10000)
hist(x, 100)
```

For the power user, you have full control of line styles, font properties, axes properties, etc, via an object oriented interface or via a set of functions familiar to MATLAB users.

Download

Visit the [matplotlib downloads page](#).

Documentation

This is the documentation for matplotlib version 1.3.1.

Other versions are available:

- [1.3.1](#) Latest stable version.
- [1.3.0](#) Previous stable version.
- [1.2.1](#) Previous stable version.

John Hunter (1968–2012)



On August 28 2012, John D. Hunter, the creator of matplotlib, died from complications arising from cancer treatment, after a brief but intense battle with this terrible illness. John is survived by his wife Miriam, his three daughters Rahel, Ava and Clara, his sisters Layne and Mary, and his mother Sarah.

If you have benefited from John's many contributions, please say thanks in the way that would matter most to him. Please consider making a donation to the [John Hunter Memorial Fund](#).

Trying to learn how to do a particular kind of plot? Check out the [gallery](#), [examples](#), or the [list of plotting commands](#).

seaborn

0.9.0

Gallery

Tutorial

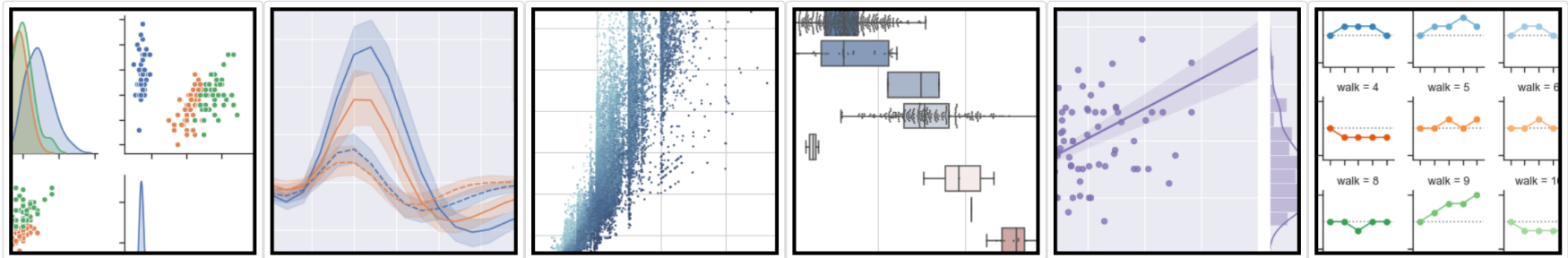
API

Site ▼

Page ▼

Search

seaborn: statistical data visualization



Seaborn is a Python data visualization library based on [matplotlib](#). It provides a high-level interface for drawing attractive and informative statistical graphics.

For a brief introduction to the ideas behind the library, you can read the [introductory notes](#). Visit the [installation page](#) to see how you can download the package. You can browse the [example gallery](#) to see what you can do with seaborn, and then check out the [tutorial](#) and [API reference](#) to find out how.

To see the code or report a bug, please visit the [github repository](#). General support issues are most at home on [stackoverflow](#), where there is a seaborn tag.

Contents

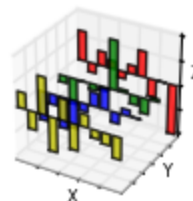
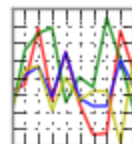
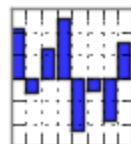
- [Introduction](#)
- [Release notes](#)
- [Installing](#)
- [Example gallery](#)
- [Tutorial](#)
- [API reference](#)

Features

- Relational: [API](#) | [Tutorial](#)
- Categorical: [API](#) | [Tutorial](#)
- Distributions: [API](#) | [Tutorial](#)
- Regressions: [API](#) | [Tutorial](#)
- Multiples: [API](#) | [Tutorial](#)
- Style: [API](#) | [Tutorial](#)
- Color: [API](#) | [Tutorial](#)

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



[overview](#) // [get pandas](#) // [documentation](#) // [community](#) // [talks](#)

Python Data Analysis Library

pandas is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the [Python](#) programming language.

0.13.1 released (February 3, 2014)

This is a scheduled bugfix release for 0.13.0, with multiple issues and regressions addressed, and several back-compatible enhancements introduced. See the [Release Notes](#) to read all about it.

For binaries and source archives of v0.13.1 see the [Download page](#).

Quick vignette

VERSIONS

Official Release

0.13.1 – February 2014

[downloads](#) // [docs](#) // [pdf](#)

Development

0.14 - May 2014

[github](#) // [docs](#)

Previous Releases

0.13.0 - [download](#) // [docs](#) // [pdf](#)

0.12.0 - [download](#) // [docs](#) // [pdf](#)

0.11.0 - [download](#) // [docs](#) // [pdf](#)

0.10.1 - [download](#) // [docs](#) // [pdf](#)

0.10.0 - [download](#) // [docs](#) // [pdf](#)

0.9.1 - [download](#) // [docs](#) // [pdf](#)

0.9.0 - [download](#) // [docs](#) // [pdf](#)



[Home](#) [Installation](#) [Documentation](#) [Examples](#)

Google™ Custom Search

Search x



scikit-learn

Machine Learning in Python

- Simple and efficient tools for data mining and data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

Classification

Identifying to which set of categories a new observation belong to.

Applications: Spam detection, Image recognition.

Algorithms: *SVM, nearest neighbors, random forest, ...* — Examples

Regression

Predicting a continuous value for a new example.

Applications: Drug response, Stock prices.

Algorithms: *SVR, ridge regression, Lasso, ...* — Examples

Clustering

Automatic grouping of similar objects into sets.

Applications: Customer segmentation, Grouping experiment outcomes

Algorithms: *k-Means, spectral clustering, mean-shift, ...* — Examples

Dimensionality reduction

Reducing the number of random variables to consider.

Applications: Visualization, Increased efficiency

Algorithms: *PCA, Isomap, non-negative matrix factorization.* — Examples

Model selection

Comparing, validating and choosing parameters and models.

Goal: Improved accuracy via parameter tuning

Modules: *grid search, cross validation, metrics.* — Examples

Preprocessing

Feature extraction and normalization.

Application: Transforming input data such as text for use with machine learning algorithms.

Modules: *preprocessing, feature extraction.* — Examples

- ▶ Building Graphs
- ▶ Asserts and boolean checks.
- ▶ Constants, Sequences, and Random Values
- ▶ Variables
- ▶ Tensor Transformations
- ▶ Math
- ▶ Strings
- ▶ Histograms
- ▶ Control Flow
- ▶ Higher Order Functions
- ▶ TensorArray Operations
- ▶ Tensor Handle Operations
- ▶ Images
- ▶ Sparse Tensors
- ▶ Inputs and Readers
- ▶ Data IO (Python functions)
- ▶ Neural Network
- ▶ Neural Network RNN Cells
- ▶ Running Graphs
- ▶ Training
- ▶ Wraps python functions
- ▶ Summary Operations
- ▶ Testing

Tensorflow Python API

Building Graphs > Core graph data structures

Members

<code>class tf.Graph</code>	A TensorFlow computation, represented as a dataflow graph.
<code>tf.Graph.__init__()</code>	Creates a new, empty Graph.
<code>tf.Graph.as_default()</code>	Returns a context manager that makes this Graph the default graph.
<code>tf.Graph.as_graph_def(from_version=None, add_shapes=False)</code>	Returns a serialized GraphDef representation of this graph.
<code>tf.Graph.finalize()</code>	Finalizes this graph, making it read-only.
<code>tf.Graph.finalized</code>	True if this graph has been finalized.
<code>tf.Graph.control_dependencies(control_inputs)</code>	Returns a context manager that specifies control dependencies.
<code>tf.Graph.device(device_name_or_function)</code>	Returns a context manager that specifies the default device to use.
<code>tf.Graph.name_scope(name)</code>	Returns a context manager that creates hierarchical names for operations.
<code>tf.Graph.add_to_collection(name, value)</code>	Stores value in the collection with the given name.

Contents

- Building Graphs > Core graph data structures
- Building Graphs > Tensor types
- Building Graphs > Utility functions
- Building Graphs > Graph collections
- Building Graphs > Defining new operations
- Building Graphs > For libraries building on TensorFlow
- Building Graphs > Other Functions and Classes
- Asserts and boolean checks. > Asserts and Boolean Checks
- Constants, Sequences, and Random Values > Constant Value Tensors
- Constants, Sequences, and Random Values > Sequences
- Constants, Sequences, and Random Values > Random Tensors
- Variables > Variables

PYTORCH

[Get Started](#)

[About](#)

[Blog](#)

[Support](#)

[Tutorials](#)

[Discuss](#)

[Docs](#)

Tensors and Dynamic neural networks in Python
with strong GPU acceleration.

PyTorch is a deep learning framework that puts Python first.

We are in an early-release Beta. Expect some adventures.

[Learn More](#)

Get Started.

OS

Linux

OSX

Package Manager

conda

pip

Source

 Keras Documentation

Search docs

Home

Keras: The Python Deep Learning library

You have just found Keras.

Guiding principles

Getting started: 30 seconds to Keras

Installation

Using a different backend than TensorFlow

Support

Why this name, Keras?

Why use Keras

Getting started

Guide to the Sequential model

Guide to the Functional API

FAQ

Models

About Keras models

Sequential

Model (functional API)

Layers

About Keras layers

Docs » Home

 [Edit on GitHub](#)

Keras: The Python Deep Learning library



You have just found Keras.

Keras is a high-level neural networks API, written in Python and capable of running on top of [TensorFlow](#), [CNTK](#), or [Theano](#). It was developed with a focus on enabling fast experimentation. *Being able to go from idea to result with the least possible delay is key to doing good research.*

Use Keras if you need a deep learning library that:

- Allows for easy and fast prototyping (through user friendliness, modularity, and extensibility).
- Supports both convolutional networks and recurrent networks, as well as combinations of the two.
- Runs seamlessly on CPU and GPU.

Read the documentation at [Keras.io](#).

Keras is compatible with: **Python 2.7-3.6**.

[Home](#)[Download](#)[Documentation](#)[Weblog](#)[Community](#)[Code](#)

The Web framework for perfectionists with deadlines.

Django makes it easier to build better Web apps more quickly and with less code.

Meet Django

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design.

Developed by a fast-moving online-news operation, Django was designed to handle two challenges: the intensive deadlines of a newsroom and the stringent requirements of the experienced Web developers who wrote it. It lets you build high-performing, elegant Web applications quickly.

Django focuses on automating as much as possible and adhering to the [DRY principle](#).

Dive in by [reading the overview](#) —

When you're ready to code, read the [installation guide](#) and [tutorial](#).

The Django framework

Object-relational mapper

Define your [data models](#) entirely in Python. You get a rich, [dynamic database-access API](#) for free — but you can still write SQL if needed.

Automatic admin interface

Save yourself the tedious work of creating interfaces for people to add and update content. [Django does that automatically](#), and it's production-ready.

Elegant URL design

Design pretty, [cruff-free URLs](#) with no framework-specific limitations. Be as flexible as you like.

Template system

Use Django's powerful, extensible and designer-friendly [template language](#) to separate design, content and Python code.

Download

Latest release: **1.6.4**

Open source, [BSD license](#)

Documentation

[Installation guide](#)[Tutorial](#)[Full index...](#)

Sites that use Django

Disqus

Beautiful, real-time, engaging comments threads for your website.

Instagram

A fast, beautiful and fun photo-sharing app for iOS and Android.

Mozilla

Makers of Firefox, and defenders of the openness, innovation, and opportunity on the web.

OpenStack

Open source software for building private and public clouds.

Pinterest

Pinterest is a virtual pinboard to share beautiful things you find on the web.

PolitiFact.com

The Pulitzer Prize-winning fact-checking site.

Rdio

Your music library in the cloud.

Weblog

Bugfix releases issued

by [James Bennett](#) on Apr. 28, 2014

Today the Django team is issuing releases to fix a minor non-security-related bug from last week's security fixes.

[Read more](#)

Security releases issued

by [James Bennett](#) on Apr. 21, 2014

Today the Django team is issuing multiple releases -- Django 1.4.11, Django 1.5.6, Django 1.6.3 and Django 1.7 beta 2 -- which fix three security issues.

[Read more](#)

Getting ready for Django 1.7

by [Russell Keith-Magee](#) on Apr. 10, 2014

Django 1.7 is almost here! How can you help? Plus, a special celebration surprise...

[Read more](#)

Django Update - 2014-03-23 to 2014-04-05

by [Curtis Maloney](#) on Apr. 9, 2014

Django Update

[Read more](#)



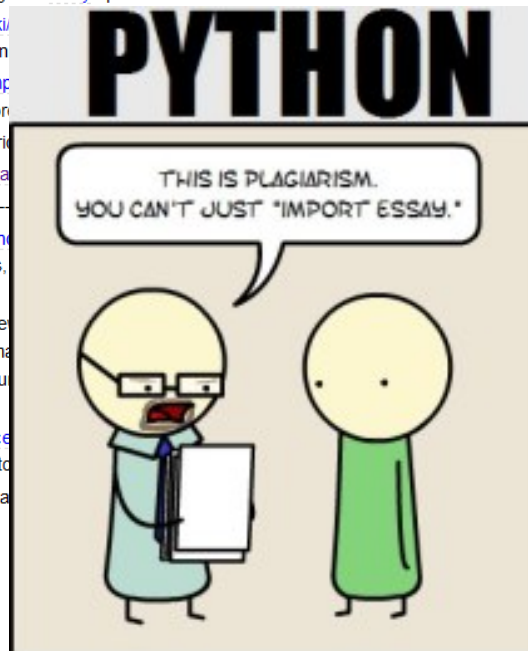
Search titles text

» NumericAndScientific

» NumericAndScientific

Numeric and Scientific

- » **NumPy** - <http://www.numpy.org/> -- Numerical Python adds a fast, compact, multidimensional array facility to Python. **NumPy** is the successor to both Numeric and Numarray.
 - » **Deprecated: Numeric** -- Numerical Python adds a fast, compact, multidimensional array language facility to Python. (Note: superseded by **NumPy**)
 - » **Deprecated: NumArray** - <http://stsdas.stsci.edu/numarray/index.html> -- Numarray is a reimplementation of Numeric which adds the ability to efficiently manipulate large numeric arrays in ways similar to Matlab and IDL. (Note: superseded by **NumPy**)
- » **SciPy** - <http://www.scipy.org/> SciPy is an open source library of scientific tools for Python. SciPy supplements the popular **NumPy** module, gathering a variety of high level science and engineering modules together as a single package. SciPy includes modules for linear algebra, optimization, integration, special functions, signal and image processing, statistics, genetic algorithms, ODE solvers, and others.
- » **Numba** - <http://numba.pydata.org/> Numba is an open source, **NumPy**-aware Python compiler specifically suited to scientific codes.
- » **ad** is an open-source Python package for transparently performing first- and second-order automatic differentiation calculations with any of the base numeric types (int, float, complex, etc.). Utility functions designed for working with SciPy optimization routines.
- » **APM Python** - <http://apmonitor.com/wiki/> available as a web-service through Python
- » **SymPy** - <http://code.google.com/p/sympy/> as simple as possible in order to be compr
- » **ALGLIB** - <http://www.alglib.net/> - numeric
- » **Python Data Analysis Library** - <http://pa>
- » **PyGSL** - <http://pygsl.sourceforge.net/> -
- » **FuncDesigner** - <http://openopt.org/Func> integration, interpolation, interval analysis, FuncDesigner by OpenOpt.
- » **OpenOpt** - <http://openopt.org> - a frame optimize FuncDesigner models with automa numerical optimization "MUST HAVE" featur purposes)
- » **SpaceFuncs** - <http://openopt.org/Space> systems of geometrical equations with auto
- » **!Nlopt** - <http://ab-initio.mit.edu/nlopt> - a



nd optimization environment with an interface to Python. The software is problems.

1. Its aim is to become a full featured CAS in Python, while keeping the code

ices.

easy-to-use data structures and data analysis tools for the Python .

y (gsl).

and get their derivatives via automatic differentiation. Also you can perform on-linear/ODE equations and numerical optimization problems coded in

ations. Connects to dozens of solvers (some are C- or Fortran-written). Can factor analysis tool for scientific experiments planning and some more addon (commercial yet, free for small-scaled academic and research

ssibilities of parametrized calculations, numerical optimization and solving

optimization algorithms written in C, with a Python interface (as well as

What is Python? What's so special about it?



University at Buffalo

The State University of New York

Sponsored By



Numba

Numba is a just-in-time specializing compiler which compiles annotated Python and NumPy code to LLVM (through decorators). Its goal is to seamlessly integrate with the Python scientific software stack and produce optimized native code, as well as integrate with native foreign languages.

Example

```
from numba import jit
from numpy import arange

# jit decorator tells Numba to compile this function.
# The argument types will be inferred by Numba when function is called.
@jit
def sum2d(arr):
    M, N = arr.shape
    result = 0.0
    for i in range(M):
        for j in range(N):
            result += arr[i,j]
    return result

a = arange(9).reshape(3,3)
print(sum2d(a))
```

More examples: [examples](#).

Documentation

- <http://numba.pydata.org/doc.html>

Source and Downloads

- Github: <https://github.com/numba/numba>

Releases

Stable: 0.13 – March 2014

[Downloads](#)

[Documentation](#)

[Release Notes](#)



Links

- [Issue tracker](#)
- Mailing lists:
[User](#)

Development

- Download [ZIP File](#)
- Download [TAR Ball](#)
- View on [GitHub](#)

What is Python? What's so special about it?



University at Buffalo

The State University of New York

```
In [1]: import numpy as np
X = np.random.random((1000, 3))
```

```
In [3]: def pairwise_python(X):
    M = X.shape[0]
    N = X.shape[1]
    D = np.empty((M, M), dtype=np.float)
    for i in range(M):
        for j in range(M):
            d = 0.0
            for k in range(N):
                tmp = X[i, k] - X[j, k]
                d += tmp * tmp
            D[i, j] = np.sqrt(d)
    return D
%timeit pairwise_python(X)
```

1 loops, best of 3: 13.4 s per loop

```
In [4]: from numba import double
from numba.decorators import jit, autojit

pairwise_numba = autojit(pairwise_python)

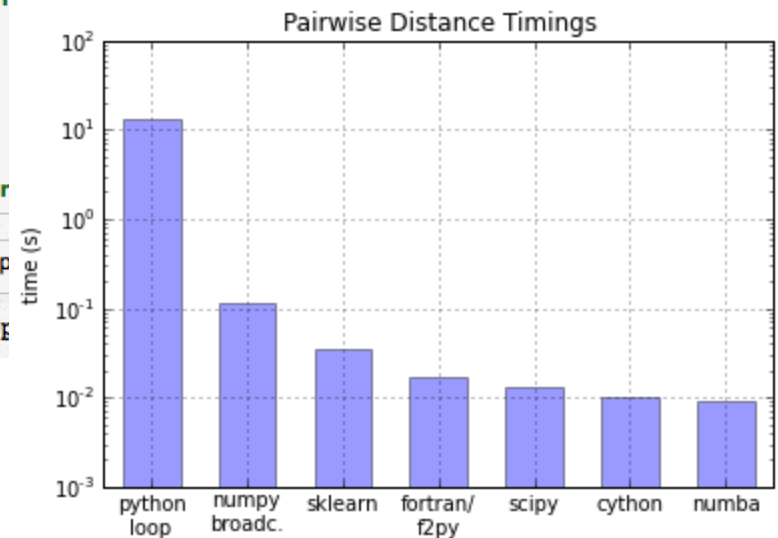
%timeit pairwise_numba(X)
```

1 loops, best of 3: 9.12 ms per loop

```
In [6]: %%cython
import numpy as np
cimport cython
from libc.math cimport sqrt

@cython.boundscheck(False)
@cython.wraparound(False)
def pairwise_cython(double[:, ::1] X):
    cdef int M = X.shape[0]
    cdef int N = X.shape[1]
    cdef double tmp, d
    cdef double[:, ::1] D = np.empty((M, M), dtype=np.float64)
    for i in range(M):
        for j in range(M):
            d = 0.0
            for k in range(N):
                tmp = X[i, k] - X[j, k]
                d += tmp * tmp
            D[i, j] = sqrt(d)
    return D
```

```
In [7]: %timeit pairwise_cython(X)
100 loops
```



➤ Python can be made fast!



 View Your Cart

[HOME](#) [PRODUCTS](#) [CONSULTING](#) [TRAINING](#) [COMPANY](#) [CONTACT US](#)

Anaconda

Completely free enterprise-ready Python distribution for large-scale data processing, predictive analytics, and scientific computing

- 330+ of the [most popular Python packages](#) for science, math, engineering, data analysis
- Completely **free** - including for commercial use and [even redistribution](#)
- Cross platform on Linux, Windows, Mac
- Installs into a single directory and doesn't affect other Python installations on your system. Doesn't require root or local administrator privileges
- Stay up-to-date by easily updating packages from our [free, online repository](#)
- Easily switch between Python 2.6, 2.7, 3.3, 3.4, and experiment with multiple versions of libraries, using our [conda](#) package manager and its great support for [virtual environments](#)
- Comes with tools to connect and [integrate with Excel](#)

Why Are We Just Giving This Away?

- We want to ensure that Python, [NumPy](#), [SciPy](#), [Pandas](#), [IPython](#), [Matplotlib](#), [Numba](#), [Blaze](#), [Bokeh](#), and other great Python data analysis tools can be used everywhere.
- We want to make it easier for Python evangelists and teachers to promote the use of Python.
- We want to give back to the Python community that we love being a part of.

But all of this takes hard work and resources!

Help us out -- Check out our [products](#), sign up for our [virtual and on-site courses](#), and [contact us](#) about doing a data science or SciPy/NumPy [consulting project](#)!

Using Anaconda in a professional environment?

Check out [Anaconda Server](#) to take control of the deployment and management of Python, R, and internal packages behind your firewall and proxy. Integration tools and install support included.

[Download Anaconda](#)

Download Anaconda

Full Version is Completely Free



Anaconda

Anaconda Add-Ons

Accelerate	\$499.00	Free Trial
IOPro	\$199.00	Free Trial
MKL Optimizations	\$99.00	Free Trial

All Products are Free for Academic Use

Anaconda Server

Manage the deployment of Python, R, and

[Learn](#)

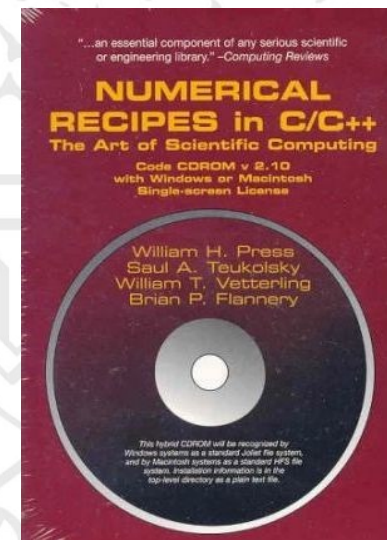
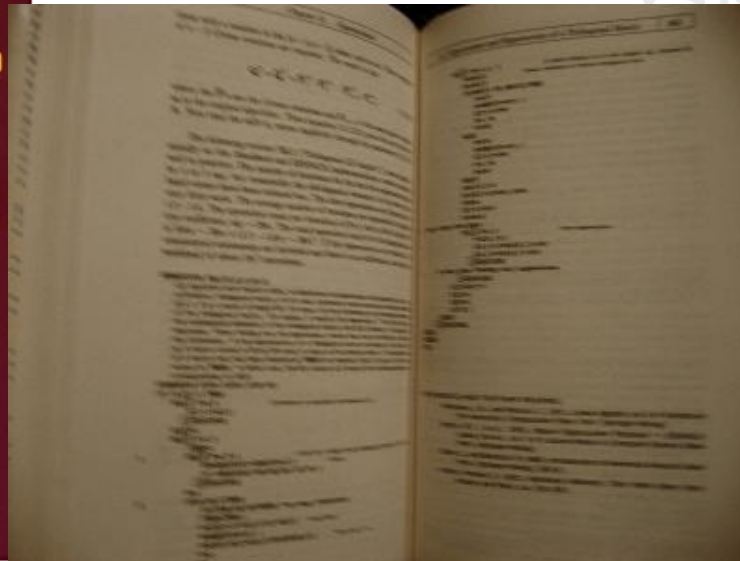
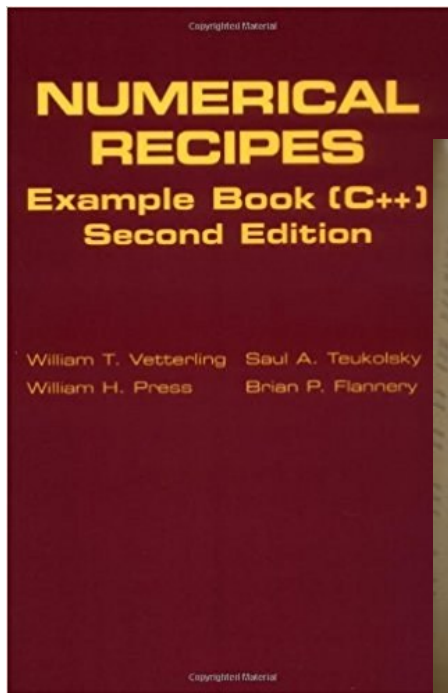
How to learn Python?



University at Buffalo

The State University of New York

- old days of coding: Fortran, C, C++
 - learn the language (for a very long time), then use the language
- new days of coding: Python
 - learn the basics (for a couple of days), then learn by doing
 - cookbooks, templates, example/code snippet collections (NR on steroids)



How to learn Python?



University at Buffalo

The State University of New York

- old days of coding: Fortran, C, C++
 - learn the language (for a very long time), then use the language
- new days of coding: Python
 - learn the basics (for a couple of days), then learn by doing
 - cookbooks, templates, example/code snippet collections

The screenshot shows the SciPy Cookbook website. The sidebar on the left lists various categories: Graphics, Input & Output, Interfacing With Other Languages, Interpolation, Linear Algebra, Matplotlib, Mayavi, Numpy, Optimization and fitting, Ordinary differential equations, Other examples, Performance, Root finding, Scientific GUIs, Scientific Scripts, and Outdated. The main content area is titled 'SciPy Cookbook' and includes a search bar. Below the title, there is a description of the cookbook as a collection of user-contributed recipes. The 'Graphics' section lists 'Line Integral Convolution' and 'Vtk volume rendering'. The 'Input & Output' section lists 'Data Acquisition with NIDAQmx', 'Data acquisition with PyUL', 'Fortran I/O Formats', 'Input and output', 'LAS reader', 'Reading SPE file from CCD camera', 'Reading mat files', and 'hdf5 in Matlab'. The 'Interfacing With Other Languages' section lists 'C Extensions for Using NumPy Arrays', 'C extensions', 'Compiling Extension Modules on Windows using mingw', 'Ctypes', 'F2py', 'Inline Weave With Basic Array Conversion (no Blitz)', 'Pyrex And NumPy', 'SWIG Numpy examples', 'SWIG and Numpy', 'SWIG memory deallocation', and 'f2py and numpy'. The 'Interpolation' section lists 'Interpolation' and 'Using radial basis functions for smoothing/interpolation'. The 'Linear Algebra' section lists 'Rank and nullspace of a matrix'.



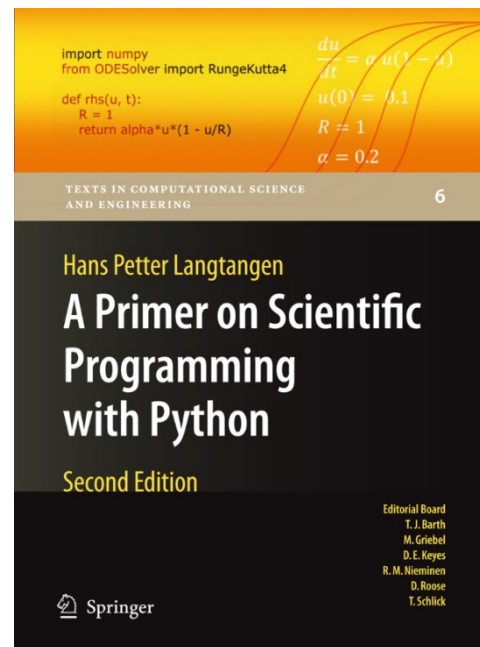
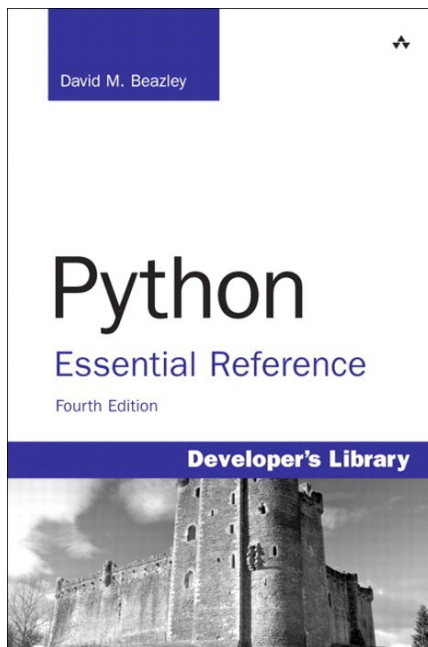
How to learn Python? Recommended reading



University at Buffalo

The State University of New York

- Python online documentation and tutorials
- a few good books



How to learn Python? Recommended reading



University at Buffalo

The State University of New York

- Python online documentation and tutorials
- a few good books
- stackoverflow.com

 **stackoverflow** [Questions](#) [Jobs](#) [Documentation](#) BETA [Tags](#) [Users](#) [?](#) [≡](#) [Log In](#) [Sign Up](#)

[Tagged Questions](#) [info](#) [newest](#) [27 featured](#) [frequent](#) [votes](#) [active](#) [unanswered](#)

Python is a dynamic and strongly typed programming language designed to emphasize usability. Two similar but mostly incompatible versions of Python are in widespread use (2 and 3). If you have a version-specific Python question consider using the [python-2.7] or [python-3.x] tags in addition to the [...]

[learn more...](#) [top users](#) [synonyms \(3\)](#) [python jobs](#)

-1
votes

0
answers

6 views

[Saving files to a subdirectory](#)

I've been working on a scraper to get large amounts of HTML and images from a website. I've got the scraper working, but the directory fills massively, making it hard to navigate. How would I go about ...

[python](#)

asked 1 min ago

 [Jammy Dodger](#)
56 • 8

0
votes

0
answers

2 views

[OpenCV/Python: cv2.minAreaRect won't return a rotated rectangle](#)

I want to deskew an image using. To do that I wrote (admittedly with lots of help) a program that: transforms image to be a easier to compute (thresh, dilation, etc.) draws contours around all ...

[python](#) [image](#) [python-3.x](#) [opencv](#) [image-processing](#)

asked 1 min ago

 [MrVocabulary](#)
67 • 9

700,745
questions tagged

[python](#) [about »](#)

BLOG

 [How Stack Overflow Redesigned the Top Navigation](#)

Related Tags

[django](#) × 68669

[python-2.7](#) × 41336

[python-3.x](#) × 35127

[pandas](#) × 32488

[numpy](#) × 29608

[list](#) × 23185

[matplotlib](#) × 18528

[dictionary](#) × 16352

GRACIAS
ARIGATO
SHUKURIA
JUSPAXAR
DANKSCHEEN
TASHAKKUR ATU
YAQHANYELAY
SUKSAMA
EKGHMET
TINGKI
BIYAN
SHUKRIA
THANK
YOU
BOLZIN
MERCİ