

# Top 20 Python Libraries For Data Science

1. **TensorFlow**
2. **NumPy**
3. **SciPy**
4. **Pandas**
5. **Matplotlib**
6. **Keras**
7. **SciKit-Learn**
8. **PyTorch**
9. **Scrapy**
10. **SQLAlchemy**
11. **Pyglet**
12. **Pygame**
13. **Python Twisted**
14. **Pywin32**
15. **wxPython**
16. **ipython**
17. **Flask**
18. **Sympy**
19. **Pandas**
20. **pyGTK**

# 1.TensorFlow

The first in the list of python libraries for data science is Tensor Flow. [TensorFlow](#) is a library for high-performance numerical computations with around 35,000 comments and a vibrant community of around 1,500 contributors. It's used across various scientific fields. TensorFlow is basically a framework for defining and running computations that involve tensors, which are partially defined computational objects that eventually produce a value.

## Features:

- Better computational graph visualizations
- Reduces error by 50 to 60 percent in neural machine learning
- Parallel computing to execute complex models
- Seamless library management backed by Google
- Quicker updates and frequent new releases to provide you with the latest features

TensorFlow is particularly useful for the following applications:

- Speech and image recognition
- Text-based applications
- Time-series analysis
- Video detection

# 2. SciPy

SciPy (Scientific Python) is another free and open-source Python library for data science that is extensively used for high-level computations.

SciPy has around 19,000 comments on GitHub and an active community of about 600 contributors. It's extensively used for scientific and technical computations, because it extends NumPy and provides many user-friendly and efficient routines for scientific calculations.

**Features:**

- Collection of algorithms and functions built on the NumPy extension of Python
- High-level commands for [data manipulation and visualization](#)
- Multidimensional image processing with the SciPy ndimage submodule
- Includes built-in functions for solving differential equations

**Applications:**

- Multidimensional image operations
- Solving differential equations and the Fourier transform
- Optimization algorithms
- Linear algebra

### 3. NumPy

[NumPy \(Numerical Python\)](#) is the fundamental package for numerical computation in Python; it contains a powerful N-dimensional array object. It has around 18,000 comments on GitHub and an active community of 700 contributors. It's a general-purpose array-processing package that provides high-performance multidimensional objects called arrays and tools for working with them. NumPy also addresses the slowness problem partly by providing these multidimensional arrays as well as providing functions and operators that operate efficiently on these arrays.

**Features:**

- Provides fast, precompiled functions for numerical routines
- Array-oriented computing for better efficiency
- Supports an object-oriented approach
- Compact and faster computations with vectorization

**Applications:**

- Extensively used in data analysis
- Creates powerful N-dimensional array
- Forms the base of other libraries, such as SciPy and scikit-learn

## 4. Pandas

[Pandas \(Python data analysis\)](#) is a must in the data science life cycle. It is the most popular and widely used Python library for data science, along with NumPy in matplotlib. With around 17,00 comments on GitHub and an active community of 1,200 contributors, it is heavily used for data analysis and cleaning. Pandas provides fast, flexible data structures, such as data frame CDs, which are designed to work with structured data very easily and intuitively.

Also Read: [What is Data Analysis: Methods, Process and Types Explained](#)

### Features:

- Eloquent syntax and rich functionalities that gives you the freedom to deal with missing data
- Enables you to create your own function and run it across a series of data
- High-level abstraction
- Contains high-level data structures and manipulation tools

### Applications:

- General [data wrangling](#) and [data cleaning](#)
- ETL (extract, transform, load) jobs for data transformation and data storage, as it has excellent support for loading CSV files into its data frame format  
Used in a variety of academic and commercial areas, including statistics, finance and neuroscience
- Time-series-specific functionality, such as date range generation, moving window, linear regression and date shifting.

## 5. Matplotlib

[Matplotlib](#) has powerful yet beautiful visualizations. It's a plotting library for Python with around 26,000 comments on GitHub and a very vibrant community of about 700 contributors. Because of the graphs and plots that it produces, it's extensively used for data visualization. It also provides an object-oriented API, which can be used to embed those plots into applications.

**Features:**

- Usable as a MATLAB replacement, with the advantage of being free and open source
- Supports dozens of backends and output types, which means you can use it regardless of which operating system you're using or which output format you wish to use
- Pandas itself can be used as wrappers around MATLAB API to drive MATLAB like a cleaner
- Low memory consumption and better runtime behavior

**Applications:**

- Correlation analysis of variables
- Visualize 95 percent confidence intervals of the models

Outlier detection using a scatter plot etc.

Visualize the distribution of data to gain instant insights

Also Read: [Exploring The Data Science Learning Path](#) .

## 6. Keras

Similar to TensorFlow, Keras is another popular library that is used extensively for deep learning and neural network modules. Keras supports both the TensorFlow and Theano backends, so it is a good option if you don't want to dive into the details of TensorFlow.

Also Read: [Keras vs Tensorflow vs Pytorch](#)

**Features:**

- Keras provides a vast prelabeled datasets which can be used to directly import and load.

- It contains various implemented layers and parameters that can be used for construction, configuration, training, and evaluation of neural networks

### Applications:

One of the most significant applications of Keras are the [deep learning models](#) that are available with their pretrained weights. You can use these models directly to make predictions or extract its features without creating or training your own new model.

## 7. Scikit-learn

Next in the list of the top python libraries for data science comes [Scikit-learn](#), a machine learning library that provides almost all the [machine learning algorithms](#) you might need. Scikit-learn is designed to be interpolated into NumPy and SciPy.

### Applications:

- clustering
- classification
- regression
- model selection
- dimensionality reduction

## 8. PyTorch

Next in the list of top python libraries for data science is PyTorch, which is a Pythonbased scientific computing package that uses the power of graphics processing units. PyTorch is one of the most commonly preferred deep learning research platforms built to provide maximum flexibility and speed.

### Applications:

- PyTorch is famous for providing two of the most high-level features
- tensor computations with strong GPU acceleration support
- building deep neural networks on a tape-based autograd system

## 9. Scrapy

The next known python libraries for data science is Scrapy. Scrapy is one of the most popular, fast, open-source web crawling frameworks written in Python. It is commonly used to extract the data from the web page with the help of selectors based on XPath.

### Applications:

- Scrapy helps in building crawling programs (spider bots) that can retrieve structured data from the web
- Scrapy is also used to gather data from APIs and follows a 'Don't Repeat Yourself' principle in the design of its interface, influencing users to write universal codes that can be reused for building and scaling large crawlers.

## 10. SQLAlchemy

SQLAlchemy is a library that **facilitates the communication between Python programs and databases**. Most of the times, this library is used as an Object Relational Mapper (ORM) tool that translates Python classes to tables on relational databases and automatically converts function calls to SQL statements.

### Features:

- No ORM Required.
- Varied databases support.
- Unit Of Work.
- Mature, High Performing Architecture.
- DBA Approved Non-Opinionated.
- Function-based query construction.
- Separate mapping and class design.
- Composite (multiple-column) primary keys.

## 11. Pyglet

**pyglet** is a cross-platform windowing and multimedia library for Python, intended for developing games and other visually rich applications.

### Features:

- No external dependencies or installation requirements. ...
- Take advantage of multiple windows and multi-monitor desktops. ...
- Load images, sound, music and video in almost any format. ...
- pyglet is written entirely in pure Python, and makes use of the ctypes module to interface with system libraries.

## 12. PyGame

**pyglet** is a library for the Python programming language that provides an object-oriented application programming interface for the creation of games and other multimedia applications. pyglet runs on Microsoft Windows, macOS, and Linux; it is released under BSD Licence.

It supports windowed and full-screen operations as well as multiple monitors. Images, video, and sound files in a range of formats can be done natively, with more additional capabilities supplied by the optional AVbin plugin, which uses the Libav package to provide support for audio formats including MP3, Ogg/Vorbis, and Windows Media Audio, and video formats such as DivX, MPEG-2, H.264, WMV, and XviD.

### Features:

- Pygame is a set of Python modules designed for writing video games. ...
- Multi core CPUs can be used easily. ...
- Uses optimized C and Assembly code for core functions. ...
- Comes with many operating systems. ...
- Truly portable. ...
- It's Simple and easy to use. ...
- Many games have been published. ...
- You control your main loop.

## 13. Python Twisted



Twisted is designed for complete separation between logical protocols (usually relying on stream-based connection semantics, such as HTTP or POP3) and physical transport layers supporting such stream-based semantics (such as files, sockets or SSL libraries). Connection between a logical protocol and a transport layer happens at the last possible moment — just before information is passed into the logical protocol instance. The logical protocol is informed of the transport layer instance, and can use it to send messages back and to check for the peer's identity. Note that it is still possible, in protocol code, to deeply query the transport layer on transport issues (such as checking a client-side SSL certificate). Naturally, such protocol code will fail (raise an exception) if the transport layer does not support such semantics.

### Features:

- The BuildBot continuous-integration system relies on Twisted for client/server communication.<sup>[3]</sup>
- ITA Software has developed an airline-reservation system for Air Canada that uses Twisted extensively.<sup>[4]</sup>
- SageMath, an open-source alternative to Mathematica, Maple, Magma, MATLAB, has a web-based interface, SageMath notebook,<sup>[5]</sup> that runs on a Twisted server.<sup>[6]</sup>
- Twisted was used in the Omegle one-on-one chat service<sup>[7]</sup> until it was replaced with gevent for performance reasons.<sup>[8]</sup>
- The Apple Calendar Server uses Twisted,<sup>[9]</sup> as do some internal projects of NASA.
- Conch, an implementation of the Secure Shell (SSH) protocol
- The original version of social networking and microblogging site Jaiku used Twisted.<sup>[citation needed]</sup>
- Fluidinfo, an online cloud data-store, uses Twisted extensively for internal RPC (partly in combination with Thrift and AMQP), for its internal services, and for external APIs.

## 14. Pywin32

PyWin32 is a **library of Python extensions for Windows** that enables you to use the features of the Win32 application programming interface (API) on Python. First, install PyWin32 through the command above on the command prompt. Beep () method in the win32api module makes simple sound.

### Release Process:

- Ensure CHANGES.txt has everything worth noting, commit it.
- Update setup.py with the new build number.
- Execute build.bat, wait forever, test the artifacts.
- Upload .whl artifacts to pypi - we do this before pushing the tag because they might be rejected for an invalid README.md. Done via `py -3.5 -m twine upload dist/*XXX*.whl`.
- Commit setup.py (so the new build number is in the repo), create a new git tag
- Upload the .exe installers to github.

- Update setup.py with the new build number + ".1" (eg, 123.1), to ensure future test builds aren't mistaken for the real release.
- Make sure everything is pushed to github, including the tag (ie, git push --tags)
- Send mail to python-win32.

## 15. wxPython

wxPython is a cross-platform **GUI toolkit** for the Python programming language. It allows Python programmers to create programs with a robust, highly functional graphical user interface, simply and easily. It is implemented as a set of Python extension modules that wrap the GUI components of the popular wxWidgets cross platform library, which is written in C++.

Like Python and wxWidgets, wxPython is **Open Source**, which means that it is free for anyone to use and the source code is available for anyone to look at and modify. And anyone can contribute fixes or enhancements to the project.

wxPython is a **cross-platform** toolkit. This means that the same program will run on multiple platforms without modification. Currently Supported platforms are Microsoft Windows, Mac OS X and macOS, and Linux or other unix-like systems with GTK2 or GTK3 libraries. In most cases the native widgets are used on each platform to provide a 100% native look and feel for the application.

### Features:

- wxPython is a GUI library based of the popular C++ wxWidgets library.
- It's a popular GUI library used by many for it's **cross-platform ability, clean UI and powerful and easy to use widgets**.
- Other popular GUI libraries similar to wxPython are Tkinter and PyQt.

## 16. iPython

IPython is an alternative Python interpreter. It is an interactive shell used for computing in Python. It provides many more useful features over the more popular default Python interpreter. In this article, you'll learn how to get started with IPython3 and use some of its features.

Just like there are previous versions of Python, there is also a previous version of IPython. Make sure that you install and use IPython.

### Features:

- Offers a powerful interactive Python shell.
- Acts as a main kernel for Jupyter notebook and other front end tools of Project Jupyter.
- Possesses object introspection ability. Introspection is the ability to check properties of an object during runtime.
- Syntax highlighting.
- Stores the history of interactions.
- Tab completion of keywords, variables and function names.
- Magic command system useful for controlling Python environment and performing OS tasks.
- Ability to be embedded in other Python programs.
- Provides access to Python debugger.

## 17. Flask

Flask is a web framework. Flask allows you to build a web application by providing tools, libraries, and technologies. This web application will be a web page, a wiki, or a big web-based calendar application or commercial website. Flask is classified into a micro-framework that means it has little to no dependencies on external libraries. There are some pros and cons. Pros mean there are little dependencies to upgrade and to watch security bugs and cons means by adding the plugin you will increase the dependencies. Flask has two dependencies, they are-

Werkzeug a WSGI utility library

Jinja2 is a template engine

Flask for Python based on Werkzeug, jinja2, and good intention

### Features:

- Development server and [debugger](#)
- Integrated support for [unit testing](#)
- [RESTful](#) request dispatching
- Uses Jinja templating
- Support for secure cookies (client side sessions)
- 100% [WSGI](#) 1.0 compliant
- [Unicode](#)-based
- Complete documentation
- [Google App Engine](#) compatibility

- Extensions available to extend functionality

## 18. SymPy

SymPy includes features ranging from **basic symbolic arithmetic to calculus, algebra, discrete mathematics and quantum physics**. It is capable of formatting the result of the computations as LaTeX code. SymPy is free software and is licensed under New BSD License. The lead developers are Ondřej Čertík and Aaron Meurer.

### Features:

SymPy is...

- **Free:** Licensed under BSD, SymPy is free both as in speech and as in beer.
- **Python-based:** SymPy is written entirely in Python and uses Python for its language.
- **Lightweight:** SymPy only depends on [mpmath](#), a pure Python library for arbitrary floating point arithmetic, making it easy to use.
- **A library:** Beyond use as an interactive tool, SymPy can be embedded in other applications and extended with custom functions.

## 19. Pandas

Pandas is an open-source library that is made mainly for working with relational or labeled data both easily and intuitively. It provides various data structures and operations for manipulating numerical data and time series. This library is built on top of the NumPy library. Pandas is fast and it has high performance & productivity for users.

### Features:

- Fast and efficient DataFrame object with default and customized indexing.
- Tools for loading data into in-memory data objects from different file formats.
- Data alignment and integrated handling of missing data.
- Reshaping and pivoting of data sets.
- Label-based slicing, indexing and subsetting of large data sets.
- Columns from a data structure can be deleted or inserted.
- Group by data for aggregation and transformations.
- High performance merging and joining of data.
- Time Series functionality.

## 20. PyGTK

PyGTK is a module used to create advanced graphical layouts in the python programming language so that the users can interact with the app quickly. It is the combination of python and GTK, the acronym GTK stands for GIMP ToolKit, so, the PyGTK is a collection of Python wrappers for the GTK+ graphical user interface library. It comes with a large number of graphical elements as well as other programming tools for constructing desktop apps. It's an extension of the GNOME project. It is open-source software that is released under the LGPL license and it is a cross-platform library. James Henstridge is the creator of PyGTK. PyGTK is extremely user-friendly, making it excellent for quick prototyping. The PyGTK can install on Unix and Win32, it is easy to set up using an installer version, and the Mac OS X versions are in the works.

### Features:

- Performance: PyGTK 0.X (for GTK 1.2) performs excellently. Version 2.0 (for gtk 2.X) does not have the same performance...
- PyGTK has an efficient API that is built on signal and callback propagation. Any component can send out a signal, which...
- IDE: PyGTK has a tool for creating interfaces called Glade that allows you to generate an XML reference of the interface...
- Portability: It's simple to port PyGTK apps to Windows. Normally, this is done without modifying the code.