Top Popular Python Libraries in Research

Samira Gholizadeh

Blast Impact and Survivability Research Unit (BISRU)
University of Cape Town, Republic of South Africa
Email: samir.gholizadeh@yahoo.com

Abstract

Python is one of the most popular programming languages in research. The structure of the language and its object-oriented approach help programmers to write logical and clear code for small and large projects. Python libraries (packages) effectively simplify many important processes such as analysing and visualizing data, retrieving unstructured data from the web, image processing, building machine learning models, and textual information [1-4]. In this article, some of the most important and popular libraries and packages in Python are described.

1- Pandas

Pandas is a fast, powerful, flexible, and easy-to-use open source data analysis and manipulation tool built on the Python programming language. Pandas is being used for data wrangling and analysis and provides simple ways for cleaning, manipulating, and transforming data. If you are dealing with a large amount of data, Pandas make it easier to work with them. Top features in Pandas can be categorized as:

1. Explore & Analyse data speedily
2. Read various file formats
3. Cleaning the data
4. Manipulating the data

Pandas works with Data Frame objects; A Pandas DataFrame is a 2 Dimensional Data Structure where the data is stored in a tabular manner in the form of rows & columns. Some

companies use Pandas as a recommendation system, such as Netflix that uses their large collection of data about their customers' preferences to provide suggestions to their users. Amazon performs extensive data analysis to create powerful recommendation systems. YouTube uses data analysis to recommend videos to their users. Pandas is also used in various domains such as Healthcare to assess the risk of chronic diseases and cancer [5, 6], Energy Sector to improve performance and reduce maintenance cost by predicting device failures [7, 8], Ecommerce organisations use Pandas for customer segmentation [9], nowadays companies analyse customer data to provide personalized Advertisement and Discovery of Services (Ads), Airline operators analyse their customer behaviour for cost cutting [4], Stock markets are using Pandas to understand market activities.

2- NumPy

NumPy is an open source library that contains multidimensional arrays. The NumPy ndarray can be used to store data in a homogeneous "n" dimensional array object [10]. NumPy is used in industry to compute arrays, for example, the data of a colored image is stored in a 3D matrix containing 1000 pixels. To manipulate those images, we need to operate on those pixels. NumPy is very useful in this scenario. NumPy is also used by advanced Python libraries like Pandas and SciPy.

NumPy is more efficient than Python's List in terms of:

- Speed
- Memory

It provides a lot of built in functions like mathematical functions, linear algebra, random sampling, etc [11] . Indexing and Slicing are used to access a subset of the data.

3- Scikit-learn

Scikit-learn is a machine learning library for the python programming language. After cleaning and manipulating your data with Panda or NumPy, Scikit-learn is used to build machine learning models, as it has thousands of tools used for modelling and predictive analysis [12]. There are several types of machine learning models that can be built using scikit-learn, namely; supervised and unsupervised learning, cross-validate the accuracy of models, and conduct feature importance. It has various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting,

k-means and DBSCAN and is designed to interact with NumPy and SciPy Python numerical and scientific libraries [13, 14].

4- Matplotlib

Matplotlib is the most popular library to explore and visualize data. You can use it to create basic charts such as line charts, scatter plots, histograms, bar charts and pie charts [15]. Matplotlib is the foundation of any other visual library. It is a plotting library for the Python programming language and NumPy numerical extension. It enables decision makers to see patterns, trends, and correlations that might go undetected in text based data [1].

- A Histogram gives the distribution of data. It is used to visualize continuous data such as sales of a company segregated by month.
- A Box Plot gives a summary of one or more numerical variables.
- A Bar Plot or Bar Chart is a plot that shows the relationship between a numerical variable and a categorical variable.
- A Scatter Plot is used to analyse the relationship between two numerical variables graphically. It is also handy in detecting the outliers in the dataset.

6- Seaborn

Seaborn is built on top of Matplotlib. Used for drawing attractive and informative statistical graphics [16]. Seaborn has a close Integration with the Pandas DataFrame and also it is a specialized support for category variables to show observations. Seaborn has tools for selecting colour palettes that reveal hidden patterns in the data. Top features of why we use Seaborn:

1. **Functionality**: Uses less syntax and has easy and interesting default themes.
2. **Flexibility:** Provides most used default themes
3. **Handling Multiple Figures:** Automates the creation of Multiple figures that may lead to Out Of Memory Issues.

5- Plotly

The Python Plotly package is an open source library built on Plotly Javascript (plotly.js). Plotly is definitely an essential tool for creating visualization because it is a powerful and easy-to-use library that is able to interact with visualizations.

There are basically two ways to create figures with plotly.py:

- Figures as dictionaries
- Figures as graph objects

The plotly.express module has functions that can create whole figures at once and is called PX. Plotly Express is an internal part of the graph library and is the recommended starting point for creating most figures. Each Plotly Express function uses graph objects internally and returns the plotly.graph_objects.Figure instance. Along with Plotly is Dash. Plotly develops Dash and also provides a platform for writing and deploying Dash applications in an enterprise environment [17]. Dash is a tool that allows you to create dynamic dashboards using Plotly visualizations.

Table 1: Matplotlib vs Seaborn vs Plotly

Features	matplotlib	Seaborn	plotly
Functionality	Mainly used for Basic Plotting using Bar Plots, Scatter Plots, etc.	Specializes in Statistical Data Visualizations	Used to create beautiful Interactive Plots
Visualization	Well-integrated with the NumPy and Pandas library of Python	More integrated for working with Pandas Data frames	High-Level API for Data Visualization
Flexibility	Highly Customizable and Powerful	Provides default themes which are most used	Provides simple syntax for complex visualizations
Animation	Not Available	Not Available	Richly interactive plots including animation in a single function call

7- TensorFlow

TensorFlow is a free, open source library for machine learning in Python. It can be used in a wide range of tasks, but has a special focus on training and inferring deep neural networks [18]. It uses multidimensional arrays, also known as tensors, which allow it to perform multiple operations on a particular input. TensorBoard is also a feature which comes with Tensorflow that helps you to visualize graphs and learn of the model [19, 20]. This helps in understanding nodes of the model and debugs it to make it better. Graph Dashboard, is a powerful tool to examine the tensorflow model as well as gives, quick view of the model's structure and design. TensorFlow APIs are hierarchically arranged, and high-level APIs are built on low-level APIs. Machine learning researchers use low-level APIs to create and

discover new machine learning algorithms. tf.keras is a TensorFlow variant of the Keras open source API. Figure 1 shows the TensorFlow toolkit:

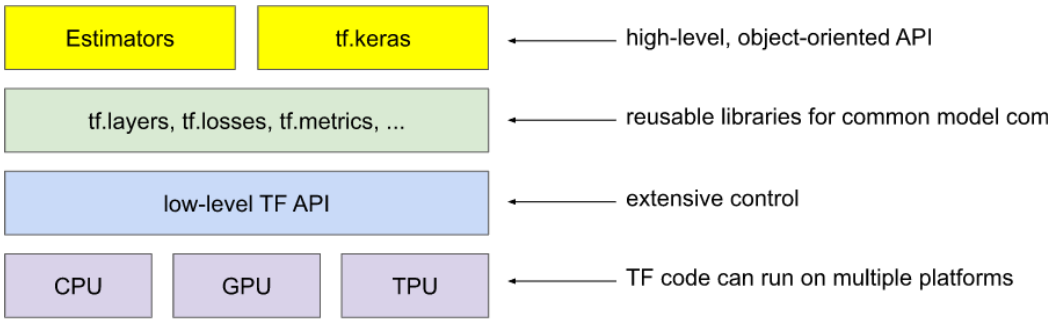

Figure 1: TensorFlow toolkit hierarchy [21]

8- Keras

Keras is a deep learning API written in Python and runs on top of the TensorFlow machine learning platform. It was developed with a focus on the possibility of rapid experiments. Keras are mainly used to create deep learning models, especially neural networks. Keras can be used to ship reliable and performant applied machine learning solutions [22], as well as in Natural Language Processing (NLP) and Computer Vision (CV) [23, 24].

9- Streamlit

Streamlit is an Open source Python library that makes it easy to build beautiful custom Web Apps for Data Science & Machine Learning. Streamlit's officially supported environment manager on Windows is Anaconda Navigator. Steamlit is an interactive graphical display of data used to understand analytical results and extract useful insights. It is the fastest way to build data apps (Interactive Dashboards). It supports many visualization libraries such as Matplotlib, Seaborn, PlotlyExpress, Bokeh, and many others. Top features of Streamlit are:

1. Improves Decision Making
2. Better Customer Experience
3. Faster Analysis
4. Increased organizational efficiency

10- Bokeh

Bokeh is a simple, interactive and powerful open source library in Python. Bokeh presents its basic grid and row / column layouts that make it quick to get started. When you need soft and responsive dashboards, you can embed bokeh designs and widgets in popular formats [25, 26]. Bokeh offers a variety of methods to embed its content in web pages: server_document for deployed Bokeh server applications, or json_items and components for standalone Bokeh output [27].

11- SciPy

SciPy in Python is an open source library used to solve math, science, engineering and technical problems. This allows users to manipulate data and visualize data using a wide range of Python commands. SciPy is based on the Python NumPy extension. SciPy extends NumPy and provides additional tools for array computing and provides specialized data structures such as scatter matrices and subsequent k-dimensional trees. It also provides algorithms for optimization, integration, interpolation, eigenvalue problems, algebraic equations, differential equations, statistics and many other classes of problems.

Some of the applications which make SciPy important are:

- Multi-dimensional image processing
- Ability to solve Fourier transforms, and differential equations
- Due to its optimized algorithms, it can do linear algebra computations very robustly and efficiently

12- OpenCv

OpenCV is an open source Python library that focuses primarily on real-time computer vision. OpenCV has a modular structure, meaning that the package consists of several shared or static libraries. The following modules are available:

Core functionality (**core**) , Image Processing (**imgproc**) [2, 3], Video Analysis (**video**) [28], Camera Calibration and 3D Reconstruction (**calib3d**) [29], 2D Features Framework (**features2d**) [30], Object Detection (**objdetect**) [31], High-level GUI (**highgui**) [32], and Video I/O (**videoio**) [33].

All the OpenCV classes and functions are placed into the cv namespace. Therefore, to access this functionality from your code, use the cv:: specifier or using namespace cv; directive.

References

1. Ronak Panchal Kamlendu Pandey, *A Study of Real World Data Visualization of COVID-19 dataset using Python* International Journal of Management and Humanities (IJMH), April 2020 **4**(8).
2. Saurabh Kulshrestha, *OpenCV For Image Processing*. 2019.
3. Naveenkumar Mahamkali and Vadivel Ayyasamy, *OpenCV for Computer Vision Applications*. 2015.
4. Fatemeh Safara, *A Computational Model to Predict Consumer Behaviour During COVID-19 Pandemic.* Computational Economics, 2020.
5. Ievgen Meniailov, Kseniia Bazilevych, Kirill Fedulov, Serhii Goranina, and Dmytro Chumachenko, *Using the K-means Method for Diagnosing Cancer Stage Using the Pandas Library*. 2019.
6. Sadrach Pierre. *Analyzing Center for Disease Control (CDC) Cancer Data using Pandas, Part 1*. 21 Oct 2019.
7. Suraj Jat, *A Project Report on Data Aggregation and analysis with Python in the field of Renewable Energy Systems MASTER OF ENGINEERING IN RENEWABLE ENERGY SYSTEMS*. 2020.
8. Katherine(Katherine Yuchen) Wang, *A machine learning framework for predictive maintenance of wind turbines*, in *Department of Electrical Engineering and Computer Science*. 2020, Massachusetts Institute of Technology. .
9. Rahul Khandelwal. *Customer Segmentation in Online Retail*. 1 Jan 2021; Available from: https://towardsdatascience.com/customer-segmentation-in-online-retail-1fc707a6f9e6.
10. Travis Oliphant, *Guide to NumPy*. 2006.
11. NumPy. Available from: https://numpy.org/.
12. Alan Fontaine, *Mastering Predictive Analytics with scikit-learn and TensorFlow*. September 2018: Packt.
13. Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, Edouard Duchesnay, and Gilles Louppe, *Scikit-learn: Machine Learning in Python.* Journal of Machine Learning Research, 2012. **12**.
14. Jiangang Hao and Tin Ho, *Machine Learning Made Easy: A Review of Scikit-learn Package in Python Programming Language.* Journal of Educational and Behavioral Statistics, 2019. **44**: p. 107699861983224.
15. Paul Barrett, J. Hunter, J. T. Miller, J. C. Hsu, and P. Greenfield, *matplotlib -- A Portable Python Plotting Package*. 2005.
16. Michael Waskom, *seaborn: statistical data visualization.* Journal of Open Source Software, 2021. **6**: p. 3021.
17. Jordi Batalla, Piotr Krawiec, D. Negru, Joachim Bruneau-Queyreix, Eugen Borcoci, Andrzej Beben, and Piotr Wiśniewski, *On providing cloud-awareness to client's DASH application by using DASH over HTTP/2.* 2015. **2015**: p. 54-64.
18. Bo Pang, Erik Nijkamp, and Ying Nian Wu, *Deep Learning With TensorFlow: A Review.* Journal of Educational and Behavioral Statistics, 2019. **45**(2): p. 227-248.
19. Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Derek Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete

Warden, and Xiaoqiang Zhang, *TensorFlow: A system for large-scale machine learning.* 2016.
20. Nikita Silaparasetty, *The Tensorflow Machine Learning Library*. 2020. p. 149-171.
21. Google's Intro to TensorFlow. Available from: https://developers.google.com/machine-learning/crash-course/first-steps-with-tensorflow/toolkit.
22. fchollet. *Introduction to Keras for Engineers*. 2020; Available from: https://keras.io/getting_started/intro_to_keras_for_engineers/.
23. fchollet. *Introduction to Keras for Researchers*. 2020; Available from: https://keras.io/getting_started/intro_to_keras_for_researchers/.
24. Palash Goyal, Sumit Pandey, and Karan Jain, *Deep Learning for Natural Language Processing*. 2018.
25. Charlie Harper, *Visualizing Data with Bokeh and Pandas.* The Programming Historian, 2018.
26. M. Tamsett and C. Group, *The NOvA software testing framework.* Journal of Physics: Conference Series, 2015. **664**: p. 062062.
27. Bokeh. Available from: https://bokeh.org/.
28. Manju Appukuttan and Valarmathie Palanisamy, *Video analytics for semantic substance extraction using OpenCV in python.* Journal of Ambient Intelligence and Humanized Computing, 2021. **12**.
29. Nikos Sarris, Michael Strintzis, B. Lei, Emile Hendriks, and Aggelos Katsaggelos, *Camera Calibration for 3D Reconstruction and View Transformation*. 2004. p. 70-129.
30. Loubrys L Rojas Reinoso, Fernando L Gutiérrez López, José C Gutiérrez, Graça Bressan, and Wilson Vicente Ruggiero, *REAL-TIME HEAD POSE ESTIMATION WITH SVM MODEL FOR FRONTAL FACE CLASSIFICATION*.
31. Chandan G, Ayush Jain, Harsh Jain, and Mohana Mohana, *Real Time Object Detection and Tracking Using Deep Learning and OpenCV*. 2018. 1305-1308.
32. Mumtazimah Mohamad, *A Review on OpenCV*. 2015.
33. Wiktor Maj, *Matrix operations on License Plate Detector and Recognizer (LPDR).* 2020: p. 1-13.