**PYTHON LIBRARIES**

1. **PANDAS:** Pandas is a python library which serves the purpose of data manipulation and analysis. It offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license. Pandas allows converting data structures to DataFrame objects, handling missing data, and adding/deleting columns from DataFrame, imputing missing files, and plotting data with histogram or plot box. It’s a must-have for data wrangling, manipulation, and visualization.
2. **NUMPY:** It is a python library which adds support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. It is a perfect tool for scientific computing and performing basic and advanced array operations. The library offers many handy features performing operations on n-arrays and matrices in Python. It helps to process arrays that store values of the same data type and makes performing math operations on arrays (and their vectorization) easier. In fact, the vectorization of mathematical operations on the NumPy array type increases performance and accelerates the execution time.
3. **OPENCV:** OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc.
4. **SCIPY:** This useful library includes modules for linear algebra, integration, optimization, and statistics. Its main functionality was built upon NumPy, so its arrays make use of this library. SciPy works great for all kinds of scientific programming projects (science, mathematics, and engineering). It offers efficient numerical routines such as numerical optimization, integration, and others in submodules. The extensive documentation makes working with this library really easy.
5. **SCIKIT-LEARN:** This is an industry-standard for data science projects based in Python. Scikits is a group of packages in the SciPy Stack that were created for specific functionalities – for example, image processing. Scikit-learn uses the math operations of SciPy to expose a concise interface to the most common machine learning algorithms. Data scientists use it for handling standard machine learning and data mining tasks such as clustering, regression, model selection, dimensionality reduction, and classification. Another advantage? It comes with quality documentation and offers high performance.
6. **MATPLOTLIB:** This is a standard data science library that helps to generate data visualizations such as two-dimensional diagrams and graphs (histograms, scatterplots, non-Cartesian coordinates graphs). Matplotlib is one of those plotting libraries that are really useful in data science projects, it  provides an object-oriented API for embedding plots into applications. Because of this library Python can compete with scientific tools like MatLab or Mathematica. Popular plotting libraries work seamlessly with Matplotlib.

**STEPS TO INSTALL A PYTHON LIBRARY IN A GIVEN ENVIRONMENT**

1. Type Command Prompt in the Windows search box.
2. Right click on the Windows Command Prompt. Then, select **Run as administrator**(by running the Command Prompt as an administrator, you’ll avoid any permission issues).
3. In the Command Prompt, type “cd\” as this command will ensure that your starting point has only the drive name.
4. Press **Enter**. Now you’ll see the drive name of C:\>.
5. Locate your Python **Scripts** path. The Scripts folder can be found within the Python application folder, where you originally installed Python. In the Command Prompt, type **cd** followed by your Python **Scripts** path.
6. Press **Enter**.
7. Now, type the pip install command to install your Python package. The pip install command has the following structure:
   1. pip install package name
8. Finally, press **Enter**, and you’ll notice that the package will be installed.