

# Python Programming Basic

## Python Third-party Libraries



# Foreword

In this course, we are going to learn the third-party libraries of Python programming, including NumPy, pandas, matplotlib, SciPy, and scikit-learn.





# Objectives

Upon finishing the course, you will be able to :

- ◆ Know how to install python third-party libraries
- ◆ Know the concepts and basic usages of NumPy, pandas, matplotlib, SciPy, and scikit-learn

# Contents

## 1. Third-party Libraries Installing

2. NumPy

3. Pandas

4. Matplotlib

5. SciPy

6. Scikit-Learn



# Third-party Libraries Installing

Input the following pip commands in your command prompt in Windows, or terminal in Linux and MacOS.

- pip install numpy
- pip install pandas
- pip install matplotlib
- pip install scipy
- pip install scikit-learn

```
(TEMP) C:\Users\[REDACTED]>pip install numpy
Collecting numpy
  Downloading numpy-1.19.4-cp38-cp38-win_amd64.whl (13.0 MB)
    | [REDACTED] 13.0 MB 3.3 MB/s
Installing collected packages: numpy
Successfully installed numpy-1.19.4

(TEMP) C:\Users\[REDACTED]>_
```



# Contents

1. Third-party Libraries Installing

2. NumPy

3. Pandas

4. Matplotlib

5. SciPy

6. Scikit-Learn



# NumPy

<https://numpy.org/>

NumPy (Numerical Python) is an open source Python library that's used in almost every field of science and engineering.



The fundamental package for scientific computing with Python

GET STARTED



# NumPy

The core of the NumPy package, is the ndarray object, a homogeneous n-dimensional array object, with methods to efficiently operate on it.

NumPy is partially written in Python, but most of the parts that require fast computation are written in C or C++.

The NumPy API is used extensively in Pandas, SciPy, Matplotlib, scikit-learn, scikit-image and most other data science and scientific Python packages.





# Why NumPy?

NumPy can be used to perform a wide variety of mathematical operations on arrays. It adds powerful data structures to Python that guarantee efficient calculations with arrays and matrices.

Python lists are slow to process. NumPy ndarray is up to 50x faster than traditional Python lists.

NumPy supplies an enormous library of high-level mathematical functions that operate on ndarrays.

# Contents

1. Third-party Libraries Installing

2. NumPy

**3. Pandas**

4. Matplotlib

5. SciPy

6. Scikit-Learn



# Pandas

<https://pandas.pydata.org/>

## pandas

**pandas** is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language.

Install pandas now!



# Pandas

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.

Provides high-level building block for doing practical data analysis in Python.

Powerful and flexible open source data analysis tool.



# Features of pandas

- Accessible to everyone
- Free for users to use and modify
- Flexible
- Powerful
- Easy to use
- Fast





# Data structures of pandas

There are three main data structures in pandas:

- Series
- DataFrame
- Panel

The most widely used pandas data structure is DataFrame.





# Series

Series is a one-dimensional labeled array capable of holding any data type (integers, strings, floating point numbers, Python objects, etc.). The axis labels are collectively referred to as the index.

Series is ndarray-like.

Series is dict-like.



# DataFrame

DataFrame is a two-dimensional labeled data structure with columns of potentially different types.

Like Series, DataFrame accepts many different kinds of input:

- Dict of 1D ndarrays, lists, dicts, or Series
- 2-D numpy.ndarray
- Structured or record ndarray
- A Series
- Another DataFrame



# Panel

A panel is a 3D container of data.

The three axes of a panel are:

- items – axis 0, each item corresponds to a DataFrame contained inside.
- major\_axis – axis 1, it is the index (rows) of each of the DataFrames.
- minor\_axis – axis 2, it is the columns of each of the DataFrames.

# Contents

1. Third-party Libraries Installing

2. NumPy

3. Pandas

**4. Matplotlib**

5. SciPy

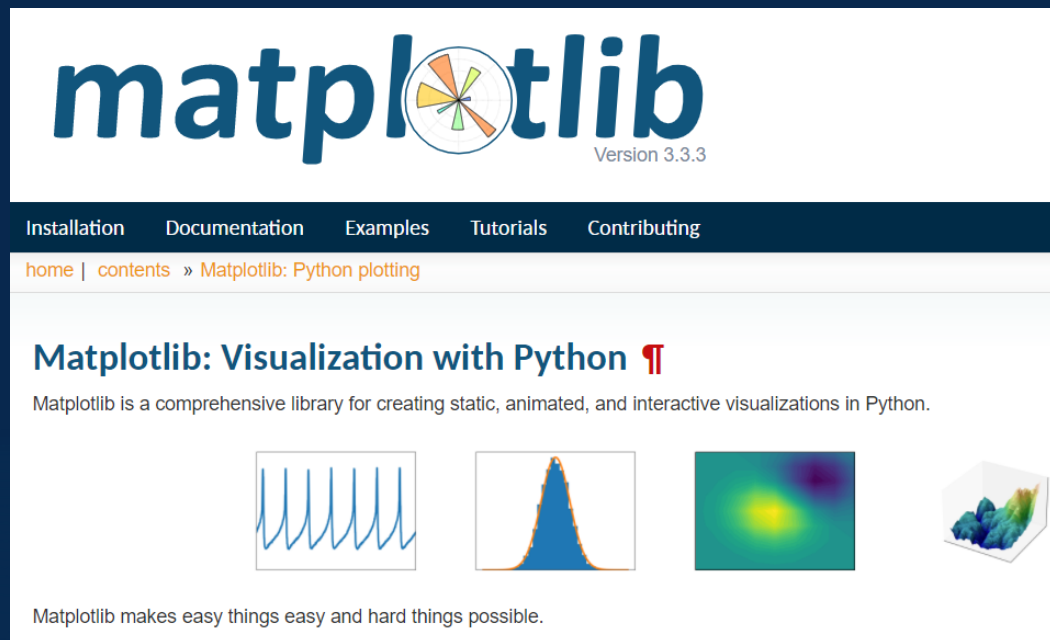
6. Scikit-Learn



# Matplotlib

<https://matplotlib.org/>

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python.





# Matplotlib

Matplotlib is a library for making plots of arrays in Python.

Although Matplotlib is written primarily in pure Python, it makes heavy use of NumPy and other extension code to provide good performance.

Matplotlib is designed with the philosophy that you should be able to create simple plots with just a few commands.



# Contents

1. Third-party Libraries Installing

2. NumPy

3. Pandas

4. Matplotlib

5. SciPy

6. Scikit-Learn

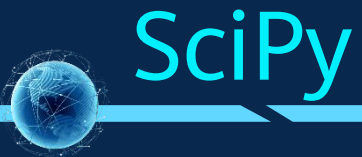


<https://www.scipy.org/>

SciPy: Scientific Python, is the Scientific computing tools for Python.

SciPy is a Python-based ecosystem of open-source software for mathematics, science, and engineering.

The screenshot shows the SciPy.org website with a blue header containing the SciPy logo and 'SciPy.org'. Below the header, there are five circular icons with labels: 'Install' (a green arrow pointing down), 'Getting started' (a yellow sun), 'Documentation' (a blue book), 'Report bugs' (a red bug), and 'Blogs' (an orange RSS feed). Below these icons, a paragraph states: '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:'. Below this paragraph, there are six items arranged in two rows. Each item consists of an icon, a package name, and a brief description. The first row includes: NumPy (a blue cube icon, 'Base N-dimensional array package'), SciPy library (a red circle with a white 'S' icon, 'Fundamental library for scientific computing'), and Matplotlib (a circular plot icon, 'Comprehensive 2-D plotting'). The second row includes: IPython (a blue 'IP[y]: IPython' icon, 'Enhanced interactive console'), SymPy (a green 'S' icon, 'Symbolic mathematics'), and pandas (a red panda face icon, 'Data structures & analysis'). At the bottom of the screenshot, there is a logo for 'NUMFOCUS' with the tagline 'OPEN CODE • BETTER SCIENCE' and a statement: 'Large parts of the SciPy ecosystem (including all six projects above) are fiscally sponsored by NumFOCUS.'



## CORE PACKAGES:

- NumPy
- SciPy library
- Matplotlib
- Pandas
- IPython
- SymPy



NumPy  
Base N-dimensional  
array package



SciPy library  
Fundamental library for  
scientific computing



Matplotlib  
Comprehensive 2-D  
plotting



IPython  
Enhanced interactive  
console



SymPy  
Symbolic mathematics



pandas  
Data structures &  
analysis



# CORE PACKAGES:

- ◆ **NumPy**, the fundamental package for numerical computation. It defines the numerical array and matrix types and basic operations on them.
- ◆ **The SciPy library**, a collection of numerical algorithms and domain-specific toolboxes, including signal processing, optimization, statistics, and much more.
- ◆ **Matplotlib**, a mature and popular plotting package that provides publication-quality 2-D plotting, as well as rudimentary 3-D plotting.
- ◆ **pandas**, providing high-performance, easy-to-use data structures.
- ◆ **IPython**, a rich interactive interface, letting you quickly process data and test ideas.
- ◆ **SymPy**, for symbolic mathematics and computer algebra.



# SciPy library

The SciPy library is one of the core packages that make up the SciPy stack. It provides many user-friendly and efficient numerical routines, such as routines for numerical integration, interpolation, optimization, linear algebra, and statistics.

SciPy is a collection of mathematical algorithms and convenience functions built on the NumPy extension of Python. It adds significant power to the interactive Python session by providing the user with high-level commands and classes for manipulating and visualizing data.



# Contents

1. Third-party Libraries Installing

2. NumPy

3. Pandas

4. Matplotlib

5. SciPy

6. Scikit-Learn





# Scikit-learn

<https://scikit-learn.org/stable/>

Scikit-learn is an open source machine learning library that supports numerous machine learning algorithms. It also provides various tools for model fitting, data preprocessing, model selection and evaluation, and many other utilities.

## scikit-learn

Machine Learning in Python

[Getting Started](#) [Release Highlights for 0.23](#) [GitHub](#)

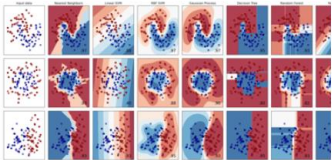
- Simple and efficient tools for predictive data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

### Classification

Identifying which category an object belongs to.

**Applications:** Spam detection, image recognition.

**Algorithms:** SVM, nearest neighbors, random forest, and more...



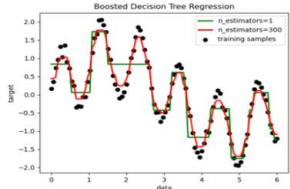
Examples

### Regression

Predicting a continuous-valued attribute associated with an object.

**Applications:** Drug response, Stock prices.

**Algorithms:** SVR, nearest neighbors, random forest, and more...




Examples

### Clustering

Automatic grouping of similar objects into sets.

**Applications:** Customer segmentation, Grouping experiment outcomes

**Algorithms:** k-Means, spectral clustering, mean-shift, and more...



Examples



# Features of scikit-learn

## Fitting and predicting: estimator basics

- Scikit-learn provides dozens of built-in machine learning algorithms and models, called estimators. Each estimator can be fitted to some data using its fit method.

## Transformers and pre-processors

- Machine learning workflows are often composed of different parts. A typical pipeline consists of a pre-processing step that transforms or imputes the data, and a final predictor that predicts target values.



# Features of scikit-learn

## Pipelines: chaining pre-processors and estimators

- Transformers and estimators (predictors) can be combined together into a Pipeline. The pipeline offers the same API as a regular estimator: it can be fitted and used for prediction with fit and predict.
- Using a pipeline will also prevent us from data leakage, i.e. disclosing some testing data in your training data.

## Model evaluation

- Fitting a model to some data does not entail that it will predict well on unseen data. This needs to be directly evaluated. Scikit-learn provides many other tools for model evaluation, in particular for cross-validation.



# Features of scikit-learn

## Automatic parameter searches

- All estimators have parameters (often called hyper-parameters) that can be tuned. The generalization accuracy of an estimator often critically depends on a few parameters.
- Quite often, it is not clear what the exact values of these parameters should be since they depend on the data at hand. Scikit-learn provides tools to automatically find the best parameter combinations (via cross-validation).



# Summary

This chapter introduces how to install python third-party libraries, the concepts and basic usages of NumPy, pandas, matplotlib, SciPy, and scikit-learn.







# More Information

## Online learning website

- <https://e.huawei.com/en/talent/#/home>

## Huawei Knowledge Base

- <https://support.huawei.com/enterprise/en/knowledge?lang=en>



# Thank you.

把数字世界带入每个人、每个家庭、  
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and  
organization for a fully connected,  
intelligent world.

Copyright©2020 Huawei Technologies Co., Ltd.  
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.