Statistical analysis of experimental data
Introduction to Python

Artur Kalinowski

FACULTY OF PHYSICS
UNIVERSITY OF WARSAW

Lecture 02

October 10, 2024

Introduction

Lecture concept

The goal of the lecture is not only to present the theoretical concepts but also to demonstrate how they can be applied in practical terms to data analysis.

Different concepts as well as analysis methods will be discussed using code examples.

We will address many (relatively simple) problems and try to look at their numerical solutions.

All examples presented are based on Python and multiple additional packages, which will be shortly introduced today.

Introduction

Web resources

(New) **Kampus platform** will be used for lecture slides, Python notebooks, as well as home exercises (and final exam):

https://kampus-kursy.ckc.uw.edu.pl/course/view.php?id=802

All files will also be available from the dedicated web page:

http://www.fuw.edu.pl/~zarnecki/SAED/

accessible without USOS account...

Environment

JupyterLab

All scripts discussed during the course were developed with Jupyter Lab
Jupyter's tools are available for installation via the Python Package Index (pip)

Environment

JupyterLab

All scripts discussed during the course were developed with Jupyter Lab
Jupyter's tools are available for installation via the Python Package Index (pip)

Git and Google Colaboratory

All scripts will be available for download at Kampus and the dedicated lecture web page. They will also be uploaded to github repository: https://github.com/zarnecki/SAED

Inserted in the lecture slides are links to open them in your browser using the Google Colab.

You can also use this link CO Open in Colab to open repository and select the notebook to load.

Please remember that you need to save the file to (your) Google Disc before you make any modifications. Otherwise, all your changes will be lost!..

Environment

Containers

If you want to avoid installing and configuring all the required packages on your personal computer, you can also use virtual environments known as containers (lightweight versions of old time virtual machines). Two popular environments used to run the containers:

- Apptainer
- Docker

Both Docker and Apptainer can use the same input files.
Docker is available for any operating system, Apptainer only for Linux ones.

Notebooks presented at this lecture can be run with akalinow/root-fedora35 container prepared for the other course some time ago. This is a container based on Fedora Linux ditribution, and contains all packages we will use during the classes.

Detailed instructions on installing and starting Docker are given here

Python packages

Python tools

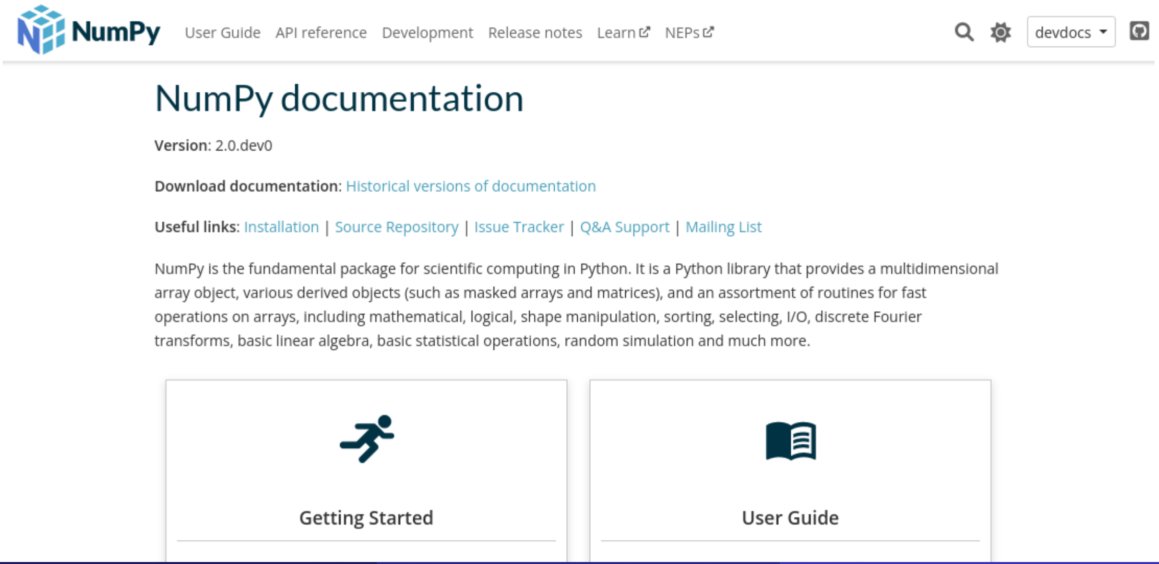
One of the main advantages of Python is the large number of diverse packages developed for various applications...

During the course we will use the following packages for computation and plotting:

- numpy - The fundamental package for scientific computing with Python
- matplotlib - Visualization with Python
- SciPy - Fundamental algorithms for scientific computing in Python
- scikit-learn - Machine Learning in Python, simple and efficient
- pandas - Handling more complex data structures

See lecture notebook for more details and examples: Open in Colab

Python packages



NumPy documentation

Version: 2.0.dev0

Download documentation: Historical versions of documentation

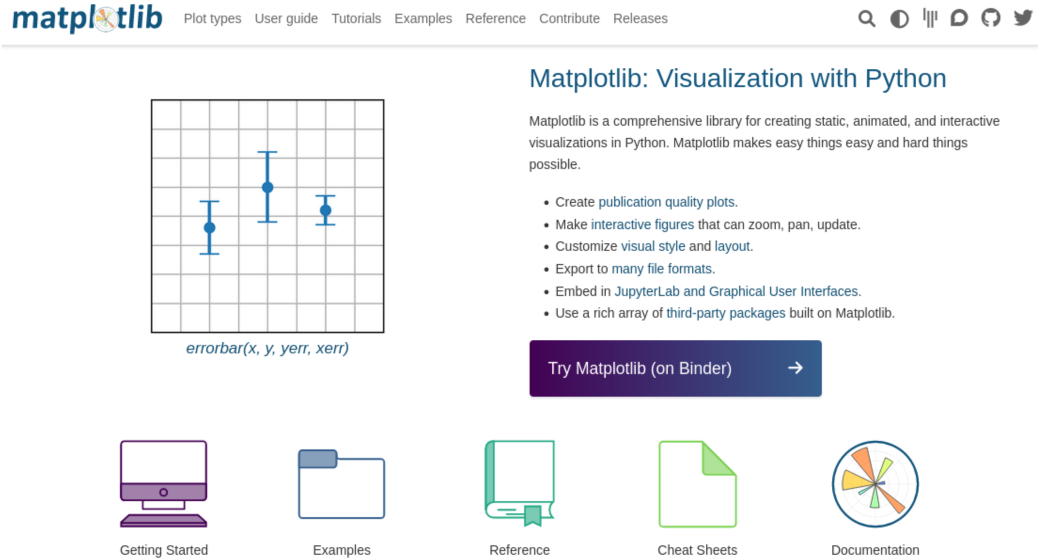
Useful links: Installation | Source Repository | Issue Tracker | Q&A Support | Mailing List

NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

Getting Started

User Guide

Python packages



errorbar(x, y, yerr, xerr)

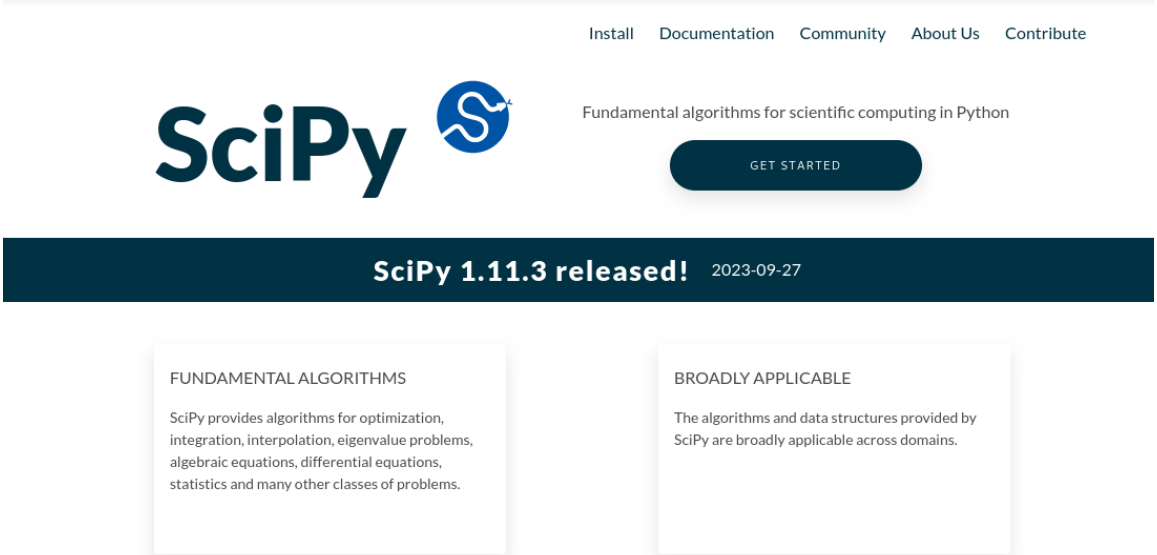
Matplotlib: Visualization with Python

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible.

- Create publication quality plots.
- Make interactive figures that can zoom, pan, update.
- Customize visual style and layout.
- Export to many file formats.
- Embed in JupyterLab and Graphical User Interfaces.
- Use a rich array of third-party packages built on Matplotlib.

Try Matplotlib (on Binder) →

Python packages



Install Documentation Community About Us Contribute

SciPy

Fundamental algorithms for scientific computing in Python

GET STARTED

SciPy 1.11.3 released! 2023-09-27

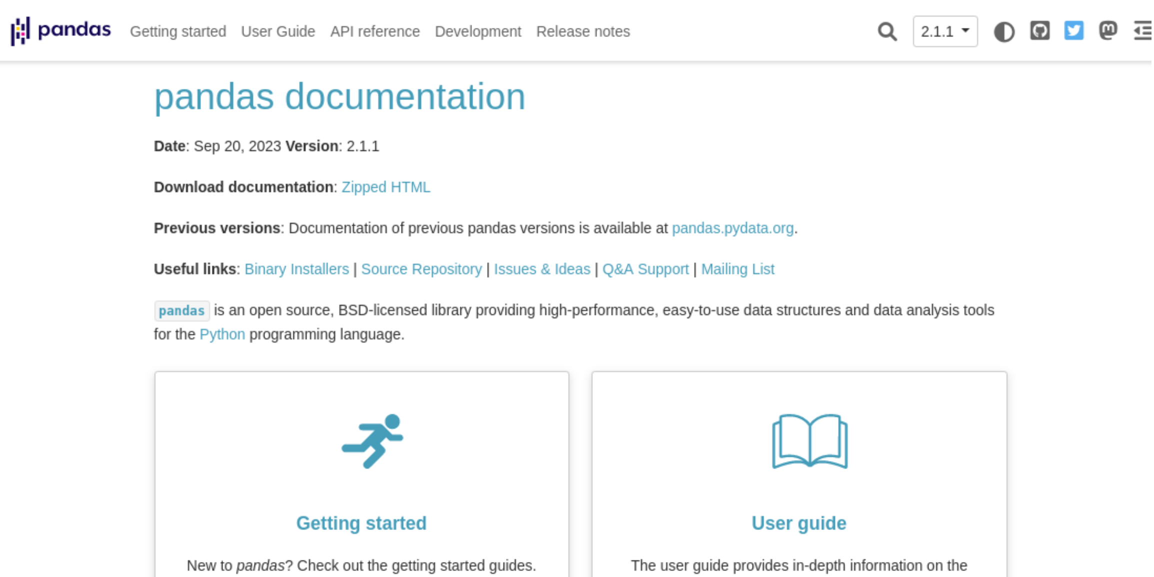
FUNDAMENTAL ALGORITHMS

SciPy provides algorithms for optimization, integration, interpolation, eigenvalue problems, algebraic equations, differential equations, statistics and many other classes of problems.

BROADLY APPLICABLE

The algorithms and data structures provided by SciPy are broadly applicable across domains.

Python packages



pandas documentation

Date: Sep 20, 2023 **Version**: 2.1.1

Download documentation: Zipped HTML

Previous versions: Documentation of previous pandas versions is available at pandas.pydata.org.

Useful links: Binary Installers | Source Repository | Issues & Ideas | Q&A Support | Mailing List

`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.

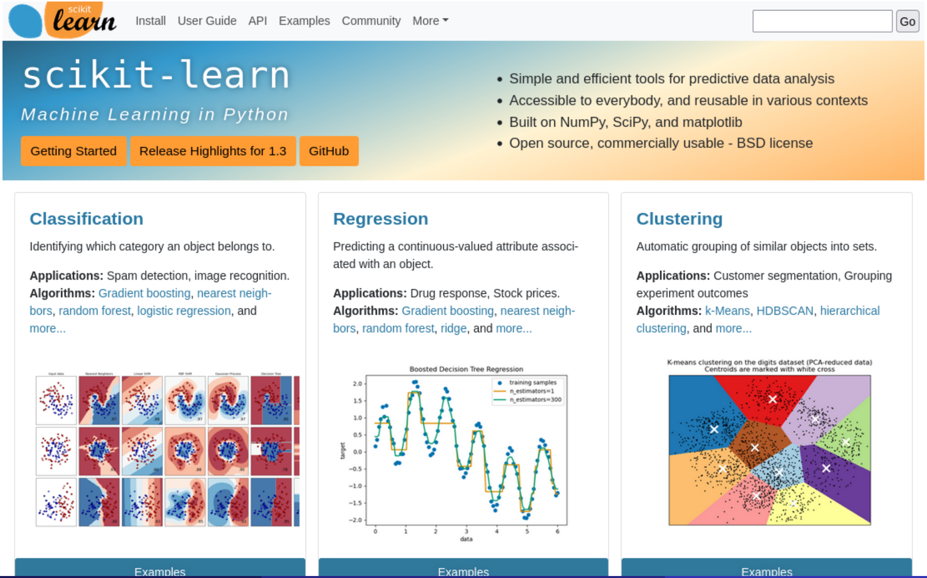
Getting started

New to *pandas*? Check out the getting started guides.

User guide

The user guide provides in-depth information on the

Python packages



scikit-learn

Machine Learning in Python

- 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: Gradient boosting, nearest neighbors, random forest, logistic regression, and more...

Regression

Predicting a continuous-valued attribute associated with an object.

Applications: Drug response, Stock prices.
Algorithms: Gradient boosting, nearest neighbors, random forest, ridge, and more...

Clustering

Automatic grouping of similar objects into sets.

Applications: Customer segmentation, Grouping experiment outcomes
Algorithms: k-Means, HDBSCAN, hierarchical clustering, and more...

K-means clustering on the digits dataset (PCA-reduced data)
Centroids are marked with white cross
Examples
Examples
Examples

Homework

Graphical presentation of the data

Look for the recent data describing age and gender structure of the population of Poland. One of the possibile sources is the Demographic Database of the Central Statistical Office (Polish: Główny Urzad Statystyczny) or Poland's Data Portal.

1. Prepare the plot showing the gender balance (men to women ratio) as a function of age.
2. Can the observed dependence be used to draw any conclusions concerning the life expectancy? What your conclusion would be? Justify your answer.

Solutions should be uploaded until October 23.

Solutions (dedicated files or printouts of the notebook with final output) should be uploaded to Kampus in readable format (PDF preferred). Upload your notebook or add the link to your notebook in Google Colab as a comment. Please share your Google Colab space with a.zarnecki@uw.edu.pl, so I can have a look at your notebook, if needed.

If you use environment different than JupyterLab, printout of the solution is mandatory!