



PYTHON FOR GEOSPATIAL ANALYSIS

COURSE GUIDE (V2): JANUARY – JULY 2021

© 2021 PYTHON CHARMERS®

Python for Geospatial Analysis

A specialist course

Audience: This is a course for GIS analysts, scientists, engineers, surveyors, and other analysts working with geospatial data sets.

Context: Spatial data is ubiquitous and location analytics are more important than ever. A well drawn map is not only beautiful to look at, but can change how you see the world. In the last 10 years Python has become the go-to language for scientific computing and spatial science.

Overview: You will learn to work with and analyze general, scientific, and geospatial datasets in many useful formats and learn to solve a wide range of analytical problems in Python for geospatial applications. You will also learn about the elegance and power of the Python language and the breadth of its amazing ecosystem of powerful packages for geospatial analysis.

Format: Live instructor-led training (online). Each topic is a mixture of expert instruction, worked examples, and hands-on exercises with help from the instructor(s).

Expert instructors: See bios below.

Duration: 5 days

Modules:

Days 1–5: “Python for Geospatial Analysis”

Days 1–2: “Introduction to Python”

Days 3–5: for graduates of Python Charmers courses or people with significant Python experience (3–6 months)

Price:

Regular course (5 days): AUD \$3,500 (excl GST)

Modular / partial courses: AUD \$800 (excl GST) per day

Dates (January – July 2021):

1–5 February

15–19 March

19–23 April

24–28 May

28 June–2 July

PYTHON
C H A R M E R S

Skills & Activities

Skills

Days 1–2: You will gain a solid understanding of the Python language and experience using Python for a wide range of scripting and data-manipulation tasks with data in useful formats (CSV, Excel, SQL); and creating beautiful statistical graphics and simple dashboards.

Day 3: You will learn about time-series analysis; manipulating vector/matrix data; performing Monte Carlo simulations; constructing statistical models; linear regression; clustering; outlier / anomaly detection.

Days 4: You will learn all about working with geospatial data using Python, including accessing spatial data formats (shapefiles, KML, raster and vector formats), analyzing raster and vector geospatial datasets; and visualizing the results on a map.

Day 5: You will gain an overview of available scientific routines, including for gridded data interpolation, curve fitting, optimization, signal and image processing. You will also learn how to speed up slow code and parallelize it across multiple cores or a cluster for large datasets.

Activities

Exercises: There will be practical exercises throughout the training course. These will be challenging and fun, and the solutions will be discussed after each exercise and provided as source code. During the exercises, the trainer will offer help and suggestions.

Worked examples: To prepare you for the exercises, the trainer will present worked examples and demos and help you to follow along on your own computer.



Topic outline

Day 1: Python basics

Day 1 covers how to use Python for basic scripting and automation tasks, including tips and tricks for making this easy:

- Why Python? What's possible?
- The *Jupyter* notebook for rapid prototyping
- Modules and packages
- Python concepts: an introduction through examples
- Essential data types: strings, tuples, lists, dicts
- Worked example: retrieving real-time data from a REST web API
- Raising and handling exceptions

Topic outline

Day 2: Handling, analyzing, and presenting data in Python

Python offers amazingly productive tools like Pandas for working with different kinds of data. Day 2 gives a thorough introduction to analyzing and visualizing data easily:

- Reading and writing essential data formats:
CSV, Excel, SQL, time-series (others on request)
- Indexing and selecting data in *Pandas*
- Data fusion: joining & merging datasets
- Summarization with “group by” operations; pivot tables
- Visualization and statistical graphics with *Seaborn*
- Automated reporting; interactive dashboards *ipywidgets* and *voilà*

Topic outline

Day 3: Time-series, simulation, inference and modelling

Day 3 shows you how to manipulate time-series and matrix/vector data. It then describes simulation methods and walks you through using powerful methods of inference and modelling, clustering and outlier detection:

- Time-series analysis: parsing dates; resampling; interpolation; joining
- Introduction to *NumPy* for manipulating vector and matrix data: data types, powerful indexing, reshaping, *ufuncs*
- Monte Carlo simulation and applications
- Statistical modelling and density estimation with *scipy* and *scikit-learn*
- Linear regression with *statsmodels*
- Clustering with *scikit-learn*, with geospatial applications
- Outlier and anomaly detection with *pyod*

Topic outline

Day 4: Spatial analysis in Python

Day 4 will provide a comprehensive tutorial in working with geospatial data using Python. It will cover spatial data access, spatial analysis, and visualizing the results on a map.

- Reading & writing vector data with *Geopandas* and *GDAL*
- Reading and writing rasters with *Rasterio*
- Working with *NetCDF* data with *xarray*
- Projections with *Geopandas*, *pyproj* and *shapely*
- Creating beautiful maps and overlaying statistical data
- Introduction to vector and raster image analysis with *PySAL* and *SciPy*

Topic outline

Day 5: Scientific computing with Python

Day 5 teaches you specialized tools in Python for scientific and engineering computing. It gives you a comprehensive introduction to *SciPy* and the broader package ecosystem. It then teaches you how to profile and speed up slow numerical code and how to parallelize code for large datasets across several cores/processors or distribute them across a cluster.

Morning: scientific computing

- Tour of *SciPy* and related packages, with fancy demos:
 - dense & sparse linear algebra
 - interpolation (e.g. for gridded data)
 - optimization and curve fitting
 - integration / ODEs
 - signal & image processing
 - handling scientific units and uncertainties

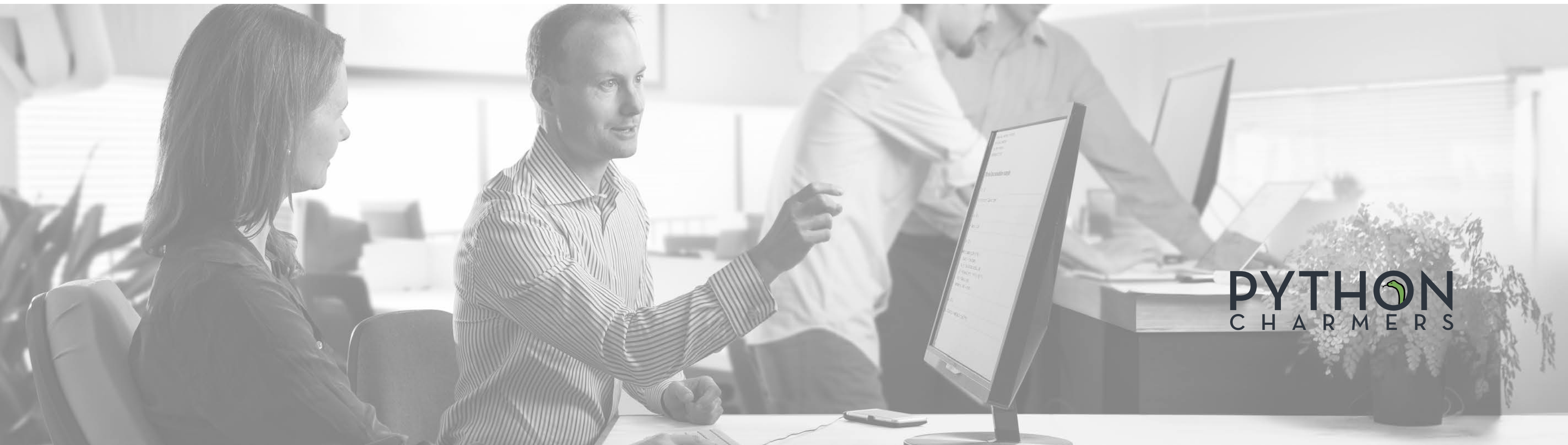
Afternoon: scaling up

- Speeding up code by 4x to 10,000x:
 - profiling, vectorization, JIT compilation with *numba*
 - parallel and distributed computing with *dask*



Personal help

We are happy to offer on-the-spot problem-solving after each day of the training for you to ask one-on-one questions — whether about the course content and exercises or about specific problems you face in your work and how to solve them. If you would like us to prepare for this in advance, you are welcome to send us background info before the course.



PYTHON
CHARMERS

Other information

Format: Courses are conducted online via video meeting using Python Charmers' cloud notebook server for sharing code with the trainer(s).

Computer:

- **Hardware:** we recommend ≥ 8 GB of RAM and a webcam.
Preferably also multiple screens and a quiet room (or headset mic).
- **Software:** a modern browser: Chrome, Firefox, or Safari (not IE or Edge);
and Zoom.
- **Coding:** we have a cloud-based coding server that supports running code
and sharing code with the trainer(s).

Timing: Most courses will run from 9:00 to roughly 17:00 (AEDT) each day, with breaks of 50 minutes for lunch and 20 minutes each for morning and afternoon tea.

Certificate of completion: We will provide you a certificate if you complete the course and successfully answer the majority of the exercise questions.

Materials: You will have access to all the course materials via the cloud server.

We will also send you a bound copy of the course notes, cheat sheets, and a USB stick containing the materials, exercise solutions, and further resources.

```
data, cmap='winter')
```

```
python3.7/site-packages/matplotlib  
cnis1- elementwise comparison
```

```
but the future will perform
```

```
str('
```

PYTHON
CHARMERS

Instructor bio



Henry Walshaw

Henry has almost 15 years of experience in Python application development and has trained hundreds of people in how to use Python from organisations including AGL, the Bureau of Meteorology, ESRI, the NSW Department of Finance, National Australia Bank, and Telstra.

Henry's core technical expertise relates to the development and analysis of large scale spatial datasets (primarily using Python), and communicating this understanding to both subject matter experts and the general public.

Before joining Python Charmers, Henry worked in both government and industry — at Geoscience Australia, the Victorian Department of Sustainability and Environment, and the Environmental Protection Agency (EPA); as a consultant with Sinclair Knight Merz (SKM), a manager at we-do-IT, and as CTO of a startup. He holds a Bachelors in Computational Science.



PYTHON
C H A R M E R S

Instructor bio



Dr Edward Schofield

Ed has consulted to or trained over 2500 people from dozens of organisations in Python, including Atlassian, the Bureau of Meteorology, Cisco, CSIRO, Dolby, Geoscience Australia, Harvard University, Singtel Optus, Oracle, Shell, Telstra, Toyota, Verizon, and Westpac. He is well-known in the Python community as a former release manager of *SciPy* and the author of the widely used *future* package. He regularly presents at conferences in data analytics and Python in Australia and internationally.

Ed holds a PhD in machine learning from Imperial College London. He also holds BA and MA (Hons) degrees in mathematics and computer science from Trinity College, University of Cambridge. He has 20+ years of experience in programming, teaching, and public speaking.



Instructor bio



Dr Robert Layton

Robert is the author of the book “Data Mining in Python”, published by Packt. He provides analysis, consultancy, research and development work to businesses, primarily using Python. Robert has worked with government, financial and security sectors, in both a consultancy and academic role. He is also a Research Fellow at the Internet Commerce Security Laboratory, investigating cybercrime analytics and data-mining algorithms for attribution and profiling.

Robert is a contributor to the Python-based *scikit-learn* open source project for machine learning and writes regularly on data mining for a number of outlets. He is also the author of the website “LearningTensorflow.com”. He has presented regularly at a number of international conferences in Python, data analysis, and its applications.



PYTHON
CHARMERS

Instructor bio



Dr Juan Nunez-Iglesias

Juan Nunez-Iglesias is co-author of the book *Elegant SciPy*, published by O'Reilly Media. Juan is a core developer of the *scikit-image* Python library, and has contributed to many others in the scientific Python ecosystem, including *SciPy*, *NetworkX*, and *Matplotlib*. He has taught and presented at the SciPy conference in Austin, EuroSciPy, PyCon Australia, the Advanced Scientific Programming in Python summer school, and Software Carpentry workshops.

Juan is a research fellow at Monash University, with interests in neuroscience and biological image analysis. He also has a particular interest in renewable energy and the environment.

Juan has Bachelor's degree in Biomedical Science from the University of Melbourne and both an MSc in Statistics and PhD in Computational Biology and Bioinformatics from the University of Southern California.



PYTHON
CHARMERS



About Python Charmers®

Python Charmers is the leading provider of Python training in the Asia-Pacific region, based in Australia and Singapore. Since 2010, Python Charmers has given over 450 training courses and bootcamps to over 4,500 delighted people from organizations such as AGL, Atlassian, Barclays, CSIRO, Cisco, Deloitte, Dolby, IMC, pwc, Singtel Optus, Shell, Sportsbet, Telstra, Toyota, Transurban, Verizon, VicRoads, Westpac, and Woolworths. Python Charmers specializes in teaching programming and data science to scientists, engineers, data analysts, quants, and computer scientists in the Python language.

Python Charmers' trainers boast years of Python experience and deep roots in the open source community, as both speakers at events and contributors to well-known open source projects, including *NumPy*, *SciPy*, *Scikit-Learn*, *Pandas*, and *Python-Future*.

Testimonials: Testimonials from past participants of similar bootcamps and training courses are available at

<https://pythoncharmners.com/testimonials/>

Questions: We are happy to customise this program further on request. Please let us know if you would like to discuss this or have any other questions.

Contact:

Phone: +61 1300 963 160

Email: info@pythoncharmners.com

Web: pythoncharmners.com

The logo for Python Charmers, featuring the word "PYTHON" in a large, bold, sans-serif font, with a small green Python snake icon integrated into the letter "O". Below "PYTHON" is the word "CHARMERS" in a smaller, all-caps, sans-serif font.

PYTHON
CHARMERS



PYTHON
CHARMERS®