



# INTERMEDIATE GEOSPATIAL ANALYSIS IN PYTHON

COURSE GUIDE (V1): OCTOBER – DECEMBER 2020

© 2020 PYTHON CHARMERS®

# Intermediate Geospatial Analysis

A specialist course using Python

**Audience:** This is a course for GIS analysts, scientists, engineers, surveyors, and other data analysts with prior experience working with spatial data in Python.

**Prerequisites:** Completion of the Python Charmers Python for Geospatial Analysis course and six months Python programming experience.

**Overview:** This course will let you take your spatial analysis further using well-established methods to discover new information by location.

At the end of the course you will understand scientifically and statistically grounded methods of geospatial analysis that you can use to aid in your interpretation of real-world data and to solve real-world problems.

You will learn fundamentals of network analysis through automation of common geospatial tasks, the basics of dealing with network data, and more advanced spatial statistics, including measures of spatial autocorrelation and multi-dimensional interpolation and regression. You will also learn techniques for dealing with very large datasets through parallel processing and visualization.

**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).

**Exercises:** There will be practical exercises throughout the 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(s) will offer individual help and suggestions.

**Expert instructors:** See bios below.

**Duration:** 2 days

**Price:**

AUD \$1,600 (excl GST)

**Dates (October – December 2020):**

30 November – 1 December 2020

**PYTHON**  
C H A R M E R S

# Topic outline

## Day 1: Automating geospatial processes

Day 1 of the course will revise core concepts, introduce network analysis, and look at common geospatial analysis tasks to automate your workflow and analysis:

- Revision of key concepts and tools; *geopandas* and *xarray*
- Mapping the locations of addresses with geocoding with *geopy*
- Rasterization, vectorization, and skeletonization: converting old map images to vector data with *rasterio* and *scikit-image*
- Finding the most cost effective path with cost-path analysis
- Measuring the flow and capacity of a network: an introduction to network analysis with *NetworkX*

Case #1: How many people can reach the city on trains in rush-hour?

Case #2: How many houses are without water if a water-main bursts?

- Automating an analysis process in Python: updating from remote data sources, analysis, automatic map production

# Topic outline

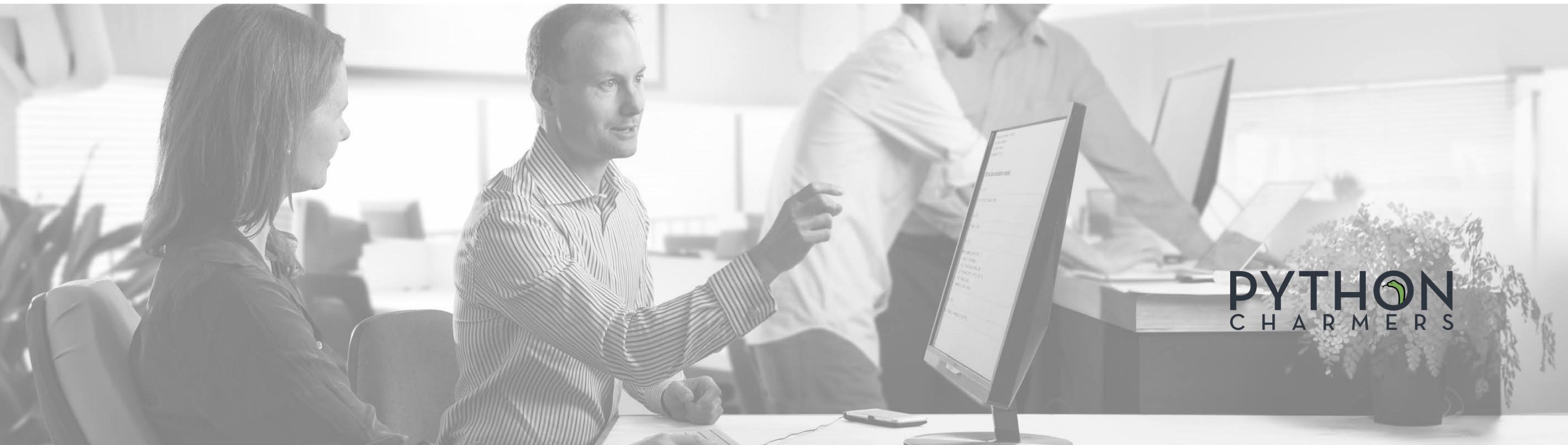
## Day 2: Extended analysis of spatial data

Day 2 looks at extending this analysis. You will learn how to perform spatial autoregression tests for spatial dependence, work with point pattern datasets for optimization and to interpolate surfaces, before finally techniques for managing and visualising large spatial datasets:

- Measuring spatial dependence with *pysal*:
  - How much do property prices depend on school locations?
  - Can you detect spatial dependence in illegal graffiti locations?
  - Measuring air pollution hotspots
- Creating better boundaries from points as concave hulls.
  - Delaunay triangulation, Voronoi tessellation, point pattern analysis with *scipy*, *shapely* and *pysal*
- Spatial data interpolation and regression with *scipy* and *scikit-learn*
  - Case study: How many pedestrians are walking through your city right now?
- Large-scale raster analysis: memory-efficient and parallel techniques: *dask*, *rasterio*, *xarray*. Example: remote-sensing indexing
- Visualization of large-scale datasets with *datashader*

## 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  $\geq 8$  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/matpl
```

```
cnis+ elementwi comparison
```

```
but the future will perfo
```

```
str('ce')
```

**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 organizations 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.





### **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](mailto:info@pythoncharmners.com)

Web: [pythoncharmners.com](http://pythoncharmners.com)

The logo features the word "PYTHON" in a bold, white, sans-serif font. The letter "O" is replaced by a stylized green Python logo. Below "PYTHON" is the word "CHARMERS" in a smaller, white, spaced-out sans-serif font, followed by a registered trademark symbol (®).

PYTHON  
CHARMERS®