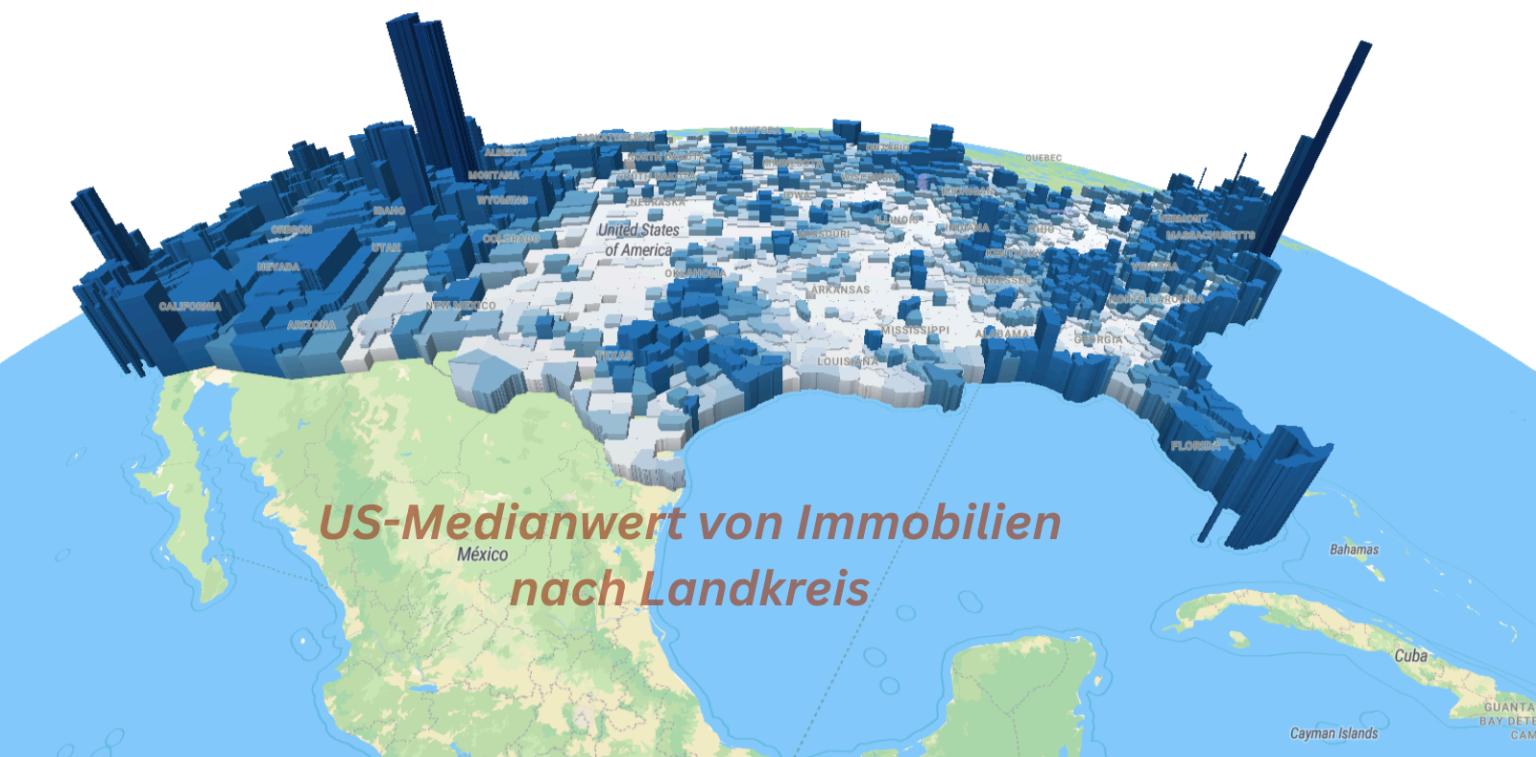


Einführung in die GIS- Programmierung

Ein praktischer Python-Leitfaden für
Open-Source-Geoinformatik-Tools

Qiusheng Wu



Einführung in die GIS-Programmierung

Ein praktischer Python-Leitfaden für Open-
Source-Geoinformatik-Tools

Qiusheng Wu
2025

Contents

Vorwort	1
Einführung	3
Für wen dieses Buch ist	3
Was dieses Buch abdeckt	4
Das Beste aus diesem Buch herausholen	5
In diesem Buch verwendete Konventionen	6
Herunterladen der Codebeispiele	6
Video-Tutorials	7
Kontakt aufnehmen	7
Danksagungen	7
Über den Autor	8
Lizenzierung und Urheberrecht	9
I: Softwareeinrichtung	11
1. <i>Überblick über Software-Tools</i>	13
1.1. Einführung	13
1.2. Lernziele	13
1.3. Wesentliche Software-Tools	13
1.4. Tool-Integration und Workflow	15
1.5. Code-Beispiele ausführen	16
1.6. Wichtige Erkenntnisse	16
2. <i>Einführung in das Python-Paketmanagement</i>	17
2.1. Einführung	17
2.2. Lernziele	17
2.3. Installation von Conda (Miniconda)	18
2.4. Conda-Konzepte verstehen	20
2.5. Ihre erste Geodaten-Umgebung erstellen	20
2.6. Conda-Fehlerbehebung	21
2.7. Wesentliche Conda-Befehle	22
2.8. Einführung in uv: Die schnelle Alternative	26
2.9. Best Practices für Package Management	27
2.10. Wichtige Erkenntnisse	28
2.11. Übungen	29
3. <i>Visual Studio Code einrichten</i>	31
3.1. Einführung	31
3.2. Lernziele	31
3.3. Visual Studio Code installieren	32
3.4. Wichtige Erweiterungen für die Python-Programmierung	32
3.5. VS Code für die Python-Entwicklung konfigurieren	34
3.6. Wichtige Tastenkombinationen	36
3.7. Referenzen und weiterführendes Lernen	37
3.8. Wichtige Erkenntnisse	38
3.9. Übungen	38
4. <i>Versionskontrolle mit Git</i>	40
4.1. Einführung	40
4.2. Lernziele	41
4.3. GitHub-Konto einrichten	41

4.4. Git installieren	41
4.5. Git konfigurieren	42
4.6. Git-Konzepte verstehen	42
4.7. Wesentliche Git-Befehle	43
4.8. GitHub verwenden	46
4.9. Integration mit VS Code	47
4.10. Best Practices für Geospatial-Projekte	47
4.11. Wichtige Erkenntnisse	48
4.12. Übungen	49
5. Google Colab verwenden	51
5.1. Einführung	51
5.2. Lernziele	51
5.3. Erste Schritte mit Google Colab	51
5.4. Ihre Geoinformations-Umgebung einrichten	53
5.5. Wesentliche Colab-Funktionen	54
5.6. Code-Beispiele in Colab ausführen	56
5.7. Wichtige Erkenntnisse	56
5.8. Übungen	56
6. Arbeiten mit JupyterLab	58
6.1. Einführung	58
6.2. Lernziele	58
6.3. Installation und Einrichtung von JupyterLab	58
6.4. Erste Schritte mit JupyterLab	59
6.5. Wichtige Tastenkürzel	62
6.6. Code-Beispiele auf MyBinder ausführen	64
6.7. Wichtige Erkenntnisse	64
6.8. Übungen	65
7. Docker verwenden	67
7.1. Einleitung	67
7.2. Lernziele	67
7.3. Docker Desktop installieren	67
7.4. Grundlegende Konzepte	69
7.5. Code-Beispiele in Docker ausführen	70
7.6. Häufige Docker-Befehle	71
7.7. Wichtige Erkenntnisse	72
7.8. Übungen	72
II: Grundlagen der Python-Programmierung	75
8. Variablen und Datentypen	77
8.1. Einführung	77
8.2. Lernziele	77
8.3. Variablen in Python	77
8.4. Benennung von Variablen	78
8.5. Datentypen	79
8.6. Escape-Zeichen	80
8.7. Kommentare in Python	81
8.8. Arbeiten mit Variablen und Datentypen	81
8.9. Grundlegende String-Operationen	81
8.10. Wichtige Erkenntnisse	82

8.11. Übungen	83
9. Python-Datenstrukturen	84
9.1. Einführung	84
9.2. Lernziele	84
9.3. Tupel	84
9.4. Listen	85
9.5. Mengen (Sets)	88
9.6. Wörterbücher (Dictionaries)	90
9.7. Leitfaden zur Auswahl von Datenstrukturen	93
9.8. Wichtige Erkenntnisse	94
9.9. Übungen	94
10. String-Operationen	97
10.1. Einführung	97
10.2. Lernziele	97
10.3. Erstellen und Manipulieren von Strings	97
10.4. String-Methoden für Geodaten	99
10.5. String-Formatierung	102
10.6. Entscheidungshilfe für String-Operationen	105
10.7. Wichtige Erkenntnisse	106
10.8. Übungen	106
11. Schleifen und bedingte Anweisungen	108
11.1. Einführung	108
11.2. Lernziele	108
11.3. For-Schleifen	108
11.4. While-Schleifen	110
11.5. Steuerungsanweisungen: Entscheidungen in Ihrem Code treffen	111
11.6. Kombination von Schleifen und Steuerungsanweisungen	113
11.7. Entscheidungshilfe für Schleifen und Steuerungsanweisungen	115
11.8. Wichtige Erkenntnisse	115
11.9. Übungen	116
12. Funktionen und Klassen	118
12.1. Einführung	118
12.2. Lernziele	118
12.3. Funktionen: Erstellen wiederverwendbarer Code-Blöcke	118
12.4. Klassen: Organisation von Daten und Verhalten zusammen	124
12.5. Kombination von Funktionen und Klassen	126
12.6. Richtlinien für Funktions- und Klassendesign	126
12.7. Wichtige Erkenntnisse	127
12.8. Übungen	127
13. Arbeiten mit Dateien	129
13.1. Einführung	129
13.2. Lernziele	129
13.3. Erstellen einer Beispieldatei	129
13.4. Lesen und Schreiben von Dateien	130
13.5. Ausnahmebehandlung	132
13.6. Kombination von Dateiverarbeitung und Ausnahmebehandlung	133
13.7. Arbeiten mit verschiedenen Dateiformaten	135
13.8. Wichtige Erkenntnisse	137

13.9.	Übungen	137
14.	Datenanalyse mit NumPy und Pandas	140
14.1.	Einführung	140
14.2.	Lernziele	140
14.3.	Einführung in NumPy	141
14.4.	Einführung in Pandas	151
14.5.	NumPy und Pandas kombinieren	159
14.6.	Wichtige Erkenntnisse	160
14.7.	Weiterführende Literatur	160
14.8.	Übungen	160
III:	Geoinformatik-Programmierung mit Python	163
15.	Einführung in Geospatial Python	165
15.1.	Einführung	165
15.2.	Das Geospatial Python-Ökosystem	165
15.3.	Verständnis der Bibliotheksbeziehungen	166
15.4.	Einrichtung Ihrer Umgebung	166
15.5.	Verifikation und erste Schritte	168
15.6.	Lernpfad und Kapitelübersicht	168
15.7.	Wichtige Konzepte zum Merken	169
15.8.	Hilfe erhalten und Ressourcen	170
15.9.	Nächste Schritte	170
15.10.	Übungen	170
16.	Vektordatenanalyse mit GeoPandas	172
16.1.	Einführung	172
16.2.	Lernziele	172
16.3.	Grundkonzepte	173
16.4.	Installation von GeoPandas	173
16.5.	Erstellen von GeoDataFrames	173
16.6.	Lesen und Schreiben von Geodaten	174
16.7.	Projektionen und Koordinatenreferenzsysteme (CRS)	176
16.8.	Räumliche Messungen und Analysen	177
16.9.	Visualisierung von Geodaten	180
16.10.	Erweiterte geometrische Operationen	184
16.11.	Räumliche Beziehungen und Abfragen	188
16.12.	Best Practices und Leistungsüberlegungen	189
16.13.	Wichtige Erkenntnisse	189
16.14.	Übungen	190
17.	Arbeiten mit Rasterdaten mit Rasterio	191
17.1.	Einführung	191
17.2.	Lernziele	191
17.3.	Rasterio installieren	192
17.4.	Rasterdaten lesen	192
17.5.	Rasterdaten visualisieren	195
17.6.	Zugriff auf und Manipulation von Rasterbändern	203
17.7.	Rasterdaten schreiben	204
17.8.	Rasterdaten ausschneiden	205
17.9.	Wichtige Erkenntnisse	208
17.10.	Übungen	208

18. Mehrdimensionale Datenanalyse mit Xarray	211
18.1. Einführung	211
18.2. Lernziele	211
18.3. Xarrays Datenmodell verstehen	212
18.4. Einrichten Ihrer Umgebung	213
18.5. Laden und Erkunden echter Klimadaten	213
18.6. Arbeiten mit DataArrays	215
18.7. Intuitive Datenauswahl und Indizierung	217
18.8. Durchführung von Operationen auf mehrdimensionalen Daten	218
18.9. Datenvisualisierung mit Xarray	220
18.10. Arbeiten mit Datasets: Mehrere Variablen	223
18.11. Die Kraft label-basierter Operationen	224
18.12. Erweiterte Indizierungstechniken	225
18.13. Hochrangige Rechenoperationen	226
18.14. Datenein- und -ausgabe	230
18.15. Wichtige Erkenntnisse	231
18.16. Weiterführende Literatur	231
18.17. Übungen	231
19. Rasteranalyse mit Rioxarray	234
19.1. Einführung	234
19.2. Lernziele	234
19.3. Einrichten Ihrer Rioxarray-Umgebung	235
19.4. Laden und Erkunden georeferenzierter Rasterdaten	235
19.5. Grundlegende Geodatenoperationen	238
19.6. Arbeiten mit räumlichen Dimensionen und Auflösung	240
19.7. Visualisierung von Geodaten-Rasterdaten	242
19.8. Datenspeicherung und Dateiverwaltung	245
19.9. Koordinatensystem-Vergleiche	247
19.10. Einführung in Bandmathematik	249
19.11. Wichtige Erkenntnisse	253
19.12. Übungen	253
20. Interaktive Visualisierung mit Leafmap	256
20.1. Einführung	256
20.2. Lernziele	257
20.3. Installation und Einrichtung von Leafmap	257
20.4. Erstellen interaktiver Karten	258
20.5. Ändern von Basemaps	261
20.6. Visualisierung von Vektordaten	266
20.7. Erstellen von Choropleth-Karten	271
20.8. Visualisierung von GeoParquet-Daten	273
20.9. Visualisierung von PMTiles	275
20.10. Visualisierung von Rasterdaten	279
20.11. Zugriff auf und Visualisierung von Maxar Open Data	286
20.12. Wichtige Erkenntnisse	292
20.13. Übungen	293
21. Geoverarbeitung mit WhiteboxTools	296
21.1. Einführung	296
21.2. Lernziele	296

21.3.	Warum Whitebox?	297
21.4.	Nützliche Ressourcen für Whitebox	298
21.5.	Installation von Whitebox	299
21.6.	Einzugsgebietsanalyse	299
21.7.	LiDAR-Datenanalyse	313
21.8.	Wichtige Erkenntnisse	321
21.9.	Übungen	322
22.	<i>3D-Kartierung mit MapLibre</i>	325
22.1.	Einführung	325
22.2.	Lernziele	325
22.3.	Nützliche Ressourcen	325
22.4.	Installation und Einrichtung	325
22.5.	Interaktive Karten erstellen	326
22.6.	Hinzufügen von Kartensteuerelementen	328
22.7.	Hinzufügen von Ebenen	330
22.8.	Verwendung von MapTiler	332
22.9.	3D-Kartierung	334
22.10.	Visualisierung von Vektordaten	341
22.11.	Visualisierung von Rasterdaten	351
22.12.	Hinzufügen benutzerdefinierter Komponenten	353
22.13.	Visualisierung von PMTiles	360
22.14.	Hinzufügen von DeckGL-Ebenen	365
22.15.	Export nach HTML	368
22.16.	Wichtige Erkenntnisse	369
22.17.	Übungen	369
23.	<i>Cloud Computing mit Earth Engine und Geemap</i>	371
23.1.	Einführung	371
23.2.	Lernziele	371
23.3.	Einführung in Google Earth Engine	371
23.4.	Einführung in interaktive Karten und Tools	374
23.5.	Der Earth Engine Datenkatalog	380
23.6.	Earth Engine Datentypen	382
23.7.	Earth Engine Rasterdaten	382
23.8.	Earth Engine Vektordaten	384
23.9.	Weitere Tools zur Visualisierung von Earth Engine-Daten	387
23.10.	Vektordatenverarbeitung	397
23.11.	Rasterdatenverarbeitung	399
23.12.	Exportieren von Earth Engine-Daten	406
23.13.	Erstellen von Zeitrafferanimationen	408
23.14.	Darstellung von Earth Engine-Daten	414
23.15.	Wichtige Erkenntnisse	444
23.16.	Übungen	445
24.	<i>Hyperspektrale Datenvizualisierung mit HyperCoast</i>	447
24.1.	Einführung	447
24.2.	Lernziele	447
24.3.	Umgebungseinrichtung	448
24.4.	Finden hyperspektraler Daten	448
24.5.	Herunterladen hyperspektraler Daten	451

24.6.	Lesen hyperspektraler Daten	451
24.7.	Visualisierung hyperspektraler Daten	452
24.8.	Erstellen von Bildwürfeln	453
24.9.	Interaktives Schneiden	454
24.10.	Interaktive Schwellenwertbildung	456
24.11.	Wichtige Erkenntnisse	457
24.12.	Übungen	458
25.	Hochleistungs-Geospatial-Analytik mit DuckDB	459
25.1.	Einführung	459
25.2.	Lernziele	459
25.3.	Installation und Einrichtung	460
25.4.	SQL-Grundlagen für räumliche Analysen	462
25.5.	Python API-Integration	467
25.6.	Datenimport	469
25.7.	Datenexport	473
25.8.	Arbeiten mit Geometrien	475
25.9.	Räumliche Beziehungen	478
25.10.	Räumliche Joins	480
25.11.	Großangelegte Datenanalyse	484
25.12.	Wichtige Erkenntnisse	491
25.13.	Übungen	492
26.	Geospatale Datenverarbeitung mit GDAL und OGR	496
26.1.	Einführung	496
26.2.	Lernziele	497
26.3.	Installation und Einrichtung	497
26.4.	Beispieldatensätze	497
26.5.	Verstehen Ihrer Daten	498
26.6.	Koordinatentransformation	498
26.7.	Formatkonvertierung	499
26.8.	Beschneiden und Maskieren	500
26.9.	Rasteranalyse und Berechnungen	501
26.10.	Konvertierung zwischen Raster und Vektor	502
26.11.	Geometrieverarbeitung	503
26.12.	Verwaltung von Feldern und Ebenen	504
26.13.	Kachelung und Datenverwaltung	505
26.14.	Fortgeschrittene Rasterverarbeitung	507
26.15.	Geländeanalyse	508
26.16.	Wichtige Erkenntnisse	514
26.17.	Referenzen und weiterführende Literatur	515
26.18.	Übungen	515
27.	Interaktive Dashboards mit Voilà und Solara erstellen	518
27.1.	Einführung	518
27.2.	Lernziele	519
27.3.	Installation von Voilà und Solara	519
27.4.	Einführung in Hugging Face Spaces	520
27.5.	Erstellen einer grundlegenden Voilà-Anwendung	520
27.6.	Erstellen einer erweiterten Webanwendung mit Solara	526
27.7.	Wichtige Erkenntnisse	532

27.8. Übungen	533
28. Verteiltes Computing mit Apache Sedona	534
28.1. Einführung	534
28.2. Lernziele	535
28.3. Installation und Einrichtung von Apache Sedona	535
28.4. Herunterladen von Beispieldaten	537
28.5. Kernkonzepte und Datenstrukturen	538
28.6. Räumliche Operationen und Funktionen	540
28.7. Räumliche Joins und Indizierung	544
28.8. Erweiterte räumliche Analyse	547
28.9. Lesen von Vektordaten	549
28.10. Visualisierung von Vektordaten	552
28.11. Schreiben von Vektordaten	555
28.12. Lesen von Rasterdaten	555
28.13. Visualisierung von Rasterdaten	557
28.14. Raster-Kartenalgebra	558
28.15. Raster-Zonal-Statistiken	559
28.16. Schreiben von Rasterdaten	560
28.17. Integration mit GeoPandas	561
28.18. Reale Anwendungsfälle	564
28.19. Wichtige Erkenntnisse	566
28.20. Referenzen und weiterführende Literatur	567
28.21. Übungen	567

Vorwort

Einführung

Geographische Informationssysteme (GIS) und geospatiale Analysen sind zu grundlegenden Werkzeugen in zahlreichen Disziplinen geworden, von der Umweltwissenschaft und Stadtplanung bis hin zur Geschäftsanalytik und öffentlichen Gesundheit. Da das Volumen und die Komplexität geospatialer Daten exponentiell weiter wachsen, ist die Fähigkeit, diese Daten programmatisch zu verarbeiten, zu analysieren und zu visualisieren, zu einer wesentlichen Fertigkeit für Forscher, Analysten und Fachkräfte geworden, die mit räumlichen Informationen arbeiten.

Python hat sich als führende Programmiersprache für geospatiale Analysen etabliert und bietet ein reichhaltiges Ökosystem von Bibliotheken und Werkzeugen, die komplexe räumliche Operationen sowohl für Anfänger als auch für Experten zugänglich machen. Dennoch kann der Weg vom Python-Anfänger zum selbstbewussten geospatialen Programmierer entmutigend erscheinen, mit zahlreichen Bibliotheken, die es zu erlernen gibt, und Konzepten, die es zu beherrschen gilt.

Dieses Buch überbrückt diese Lücke, indem es einen strukturierten, praxisorientierten Ansatz zum Erlernen der geospatialen Programmierung mit Python bietet. Anstatt Sie von Anfang an mit fortgeschrittenen Techniken zu überfordern, konzentrieren wir uns darauf, ein solides Fundament wesentlicher Fertigkeiten aufzubauen, das Ihnen während Ihrer gesamten geospatialen Programmierreise dienen wird. Jedes Kapitel baut auf den vorherigen auf und stellt sicher, dass Sie sowohl theoretisches Verständnis als auch praktische Expertise entwickeln.

Der in diesem Buch verwendete Ansatz ist praxisorientiert und beispielgesteuert. Sie werden mit echten geospatialen Datensätzen arbeiten, praktische Probleme lösen und Projekte erstellen, die die Kraft von Python für geospatiale Analyse und Visualisierung demonstrieren. Am Ende dieses Buches werden Sie das Vertrauen und die Fertigkeiten haben, Ihre eigenen geospatialen Programmierherausforderungen anzugehen.

Für wen dieses Buch ist

Dieses Buch ist für eine vielfältige Zielgruppe von Lernenden konzipiert, die die Kraft von Python für geospatiale Analyse und Visualisierung nutzen möchten:

Studierende und Forscher in Geographie, Umweltwissenschaften, Stadtplanung, Datenwissenschaft und verwandten Bereichen, die räumliche Daten als Teil ihrer Studien oder Forschung analysieren müssen. Es werden keine Programmiererfahrungen vorausgesetzt, obwohl grundlegende Vertrautheit mit Computern und Datenanalysekonzepten hilfreich ist.

GIS-Fachkräfte, die derzeit Desktop-GIS-Software verwenden und ihr Toolkit um Programmierungsfähigkeiten erweitern möchten. Wenn Sie mit ArcGIS, QGIS oder ähnlichen Werkzeugen gearbeitet haben und Arbeitsabläufe automatisieren oder Analysen durchführen möchten, die in traditioneller GIS-Software herausfordernd sind, wird Ihnen dieses Buch bei diesem Übergang helfen.

Datenwissenschaftler und Analysten, die mit standortbasierten Daten arbeiten und räumliche Analysefähigkeiten zu ihrem Skillset hinzufügen möchten. Wenn Sie sich mit Python-Grundlagen auskennen, aber neu in geospatialen Konzepten sind, führt Sie dieses Buch in das räumliche Denken und die Werkzeuge ein, die Sie benötigen.

Softwareentwickler, die sich für die Entwicklung von Anwendungen interessieren, die mit geospatialen Daten arbeiten. Egal, ob Sie Web-Mapping-Anwendungen, mobile Apps mit Standortfunktionen oder Datenverarbeitungs-Pipelines entwickeln, dieses Buch bietet das Fundament, das Sie benötigen.

Selbstlerner und Karrierewechsler, die sich für das wachsende Feld der geospatialen Datenwissenschaft interessieren. Das Buch setzt keine Vorkenntnisse in Python-Programmierung oder GIS-Konzepten voraus und macht es für motivierte Anfänger zugänglich.

Fachkräfte in Regierung und Industrie, die räumliche Analysen in ihre Arbeit einbeziehen müssen, wie Stadtplaner, Umweltberater, Marktforscher, Logistikkordinatoren oder Beamte im öffentlichen Gesundheitswesen.

Die wichtigste Voraussetzung ist Neugier und Lernbereitschaft. Obwohl Programmiererfahrung hilfreich ist, ist sie nicht notwendig. Wir beginnen mit den Grundlagen und bauen systematisch auf.

Was dieses Buch abdeckt

Dieses Buch ist in drei progressive Abschnitte gegliedert, die Sie von der Software-Installation über Python-Grundlagen bis hin zur fortgeschrittenen geospatialen Programmierung führen:

Software-Installation bereitet Ihre Entwicklungsumgebung mit allem vor, was Sie für die geospatialen Programmierung benötigen. Sie lernen, wie Sie wesentliche Werkzeuge installieren und konfigurieren, einschließlich Miniconda für das Paketmanagement, VS Code für die Entwicklung, Git für die Versionskontrolle und cloudbasierte Alternativen wie Google Colab und JupyterLab. Dieser Abschnitt stellt sicher, dass Sie ein solides Fundament haben, bevor Sie in die Programmierung eintauchen.

Python-Programmiergrundlagen baut Ihre Kernprogrammierfähigkeiten durch sieben umfassende Kapitel auf. Beginnend mit Python-Grundlagen meistern Sie Variablen und Datentypen, Datenstrukturen (Listen, Wörterbücher, Mengen), String-Operationen, Kontrollfluss mit Schleifen und Bedingungen, Funktionen und Klassen, Dateienhandhabung und Datenanalyse mit NumPy und Pandas. Diese Fertigkeiten bilden das Fundament für alle geospatialen Programmieraufgaben.

Geospatiale Programmierung mit Python umfasst vierzehn spezialisierte Kapitel, die Sie zu einem selbstbewussten geospatialen Programmierer transformieren:

- **Einführung in Geospatial Python** - Kernkonzepte und das Python-geospatiale Ökosystem
- **Vektordatenanalyse mit GeoPandas** - Arbeiten mit Punkten, Linien und Polygonen
- **Rasterdaten mit Rasterio** - Verarbeitung von Satellitenbildern und Rasterdatensätzen
- **Mehrdimensionale Datenanalyse mit Xarray** - Umgang mit komplexen wissenschaftlichen Datensätzen
- **Rasteranalyse mit Rioxarray** - Fortgeschrittene Rasterverarbeitung und -analyse
- **Interaktive Visualisierung mit Leafmap** - Erstellen dynamischer, interaktiver Karten
- **Geoprozessierung mit WhiteboxTools** - Fortgeschrittene räumliche Analyseoperationen
- **3D-Kartierung mit MapLibre** - Erstellung dreidimensionaler Visualisierungen
- **Cloud Computing mit Earth Engine und Geemap** - Nutzung von Google Earth Engine für großskalige Analysen
- **Hyperspektrale Datenvisualisierung mit HyperCoast** - Arbeiten mit hyperspektralen Daten
- **Hochleistungs-Geospatiale Analytik mit DuckDB** - Hochleistungs-räumliche Datenverarbeitung
- **Geospatiale Datenverarbeitung mit GDAL und OGR** - Arbeiten mit verschiedenen geospatialen Datenformaten
- **Erstellung interaktiver Dashboards mit Solara** - Erstellen interaktiver Dashboards für geospatiale Anwendungen
- **Verteiltes Rechnen mit Apache Sedona** - Verarbeitung großer geospatialer Datensätze in einer verteilten Umgebung

Jedes Kapitel folgt einer konsistenten Struktur:

- Klare Konzepterklärungen mit realweltlichem Kontext
- Schritt-für-Schritt-Codebeispiele mit detaillierten Anmerkungen
- Praktische Übungen mit authentischen geospatialen Datensätzen
- Häufige Fallstricke und Fehlerbehebungsanleitung
- Verweise auf zusätzliche Ressourcen und weiterführende Literatur

Die Progression ist sorgfältig so gestaltet, dass jedes Kapitel auf vorherigen Konzepten aufbaut und gleichzeitig neue Fähigkeiten einführt, wodurch sichergestellt wird, dass Sie sowohl Breite als auch Tiefe in der geospatialen Programmierung entwickeln.

Das Beste aus diesem Buch herausholen

Um Ihre Lernerfahrung mit diesem Buch zu maximieren, beachten Sie die folgenden Empfehlungen:

Richten Sie eine ordnungsgemäße Entwicklungsumgebung ein: Installieren Sie Python und die erforderlichen Bibliotheken wie im ersten Abschnitt des Buches beschrieben. Eine gut konfigurierte Umgebung wird Ihnen Zeit und Frustration während Ihrer Lernreise ersparen. Ziehen Sie die Verwendung von conda oder uv zur Verwaltung Ihrer Python-Pakete in Betracht, da dies die Installation geospatialer Bibliotheken vereinfacht.

Folgen Sie den Codebeispielen: Dieses Buch ist interaktiv konzipiert. Lesen Sie nicht nur den Code - tippen Sie ihn ab, führen Sie ihn aus und experimentieren Sie mit Änderungen. Verständnis kommt durch Übung, und jedes Beispiel baut Fertigkeiten auf, die Sie später benötigen werden.

Arbeiten Sie die Übungen durch: Jedes Kapitel enthält Übungen, die darauf ausgelegt sind, die erlernten Konzepte zu verstärken. Diese sind keine optionalen Extras - sie sind ein integraler Bestandteil des Lernprozesses. Beginnen Sie mit den angeleiteten Übungen und fordern Sie sich dann mit eigenen Projekten heraus.

Verwenden Sie echte Daten: Während das Buch Datensätze für Beispiele und Übungen bereitstellt, versuchen Sie, die Techniken auf Daten aus Ihrem eigenen Bereich oder Ihren Interessen anzuwenden. Dies wird Ihnen helfen zu verstehen, wie die Konzepte auf realweltliche Szenarien angewendet werden, und Vertrauen in Ihre Fähigkeiten aufzubauen.

Erstellen Sie Projekte: Während Sie durch das Buch voranschreiten, ziehen Sie es in Betracht, an einem persönlichen Projekt zu arbeiten, das Sie interessiert. Dies könnte die Analyse von Daten aus Ihrer Forschung, die Erstellung von Karten für Ihre Gemeinde oder die Lösung eines Problems sein, dem Sie bei Ihrer Arbeit begegnet sind.

Haben Sie Geduld mit sich selbst: Programmieren kann frustrierend sein, besonders wenn Sie lernen. Erwarten Sie, auf Fehler zu stoßen, Zeit mit dem Debuggen zu verbringen und sich gelegentlich festgefahren zu fühlen. Dies ist normal und Teil des Lernprozesses. Machen Sie Pausen, wenn nötig, und denken Sie daran, dass sich Expertise allmählich durch konsequente Übung entwickelt. Wenn Sie feststecken, zögern Sie nicht, um Hilfe im GitHub-Repository des Buches zu bitten.

Üben Sie weiter: Die Fertigkeiten in diesem Buch erfordern regelmäßige Übung zur Aufrechterhaltung und Entwicklung. Planen Sie regelmäßig Zeit für die Arbeit an geospatialen Programmierprojekten ein, auch wenn es kleine sind.

In diesem Buch verwendete Konventionen

Dieses Buch verwendet mehrere Konventionen, um Ihnen bei der Navigation durch den Inhalt und dem Verständnis der Codebeispiele zu helfen:

Code-Formatierung: Aller Python-Code erscheint in Monospace-Schriftart innerhalb von Codeblöcken. Wenn Code im regulären Text erscheint, ist er so formatiert. Datei- und Verzeichnisnamen sind ebenfalls in Monospace-Schriftart formatiert.

Codebeispiele: Die meisten Codebeispiele sind vollständig und ausführbar. Sie enthalten Kommentare, die die wichtigsten Konzepte und demonstrierten Techniken erklären. Zeilennummern können zur Referenz im begleitenden Text enthalten sein.

```
# Dies ist ein Beispiel für einen Codeblock
import leafmap
m = leafmap.Map()
m.add_basemap("OpenTopoMap") # eine Basiskarte zur Karte hinzufügen
m
```

Befehlszeilenanweisungen: Befehle, die in der Befehlszeile oder im Terminal eingegeben werden sollen, sind mit einem \$ -Prompt dargestellt:

```
$ pip install leafmap
$ python script.py
```

Herunterladen der Codebeispiele

Alle Codebeispiele, Datensätze und ergänzenden Materialien für dieses Buch sind kostenlos auf GitHub verfügbar:

<https://github.com/giswqs/intro-gispro>

Um die Materialien herunterzuladen, können Sie eine der folgenden Methoden verwenden:

- **Repository klonen** (wenn Sie Git installiert haben):

```
$ git clone https://github.com/giswqs/intro-gispro.git
```

- **Als ZIP herunterladen** (wenn Sie Git nicht verwenden möchten):

- Besuchen Sie die GitHub-Repository-Seite
 - Klicken Sie auf den grünen **Code**-Button
 - Wählen Sie **Download ZIP**
 - Extrahieren Sie die Dateien an Ihren bevorzugten Ort
- **Einzelne Dateien durchsuchen** online über die GitHub-Oberfläche, wenn Sie nur spezifische Beispiele benötigen

Das Repository wird regelmäßig mit Korrekturen, Verbesserungen und zusätzlichen Beispielen aktualisiert. Schauen Sie regelmäßig nach Updates oder **verfolgen** Sie das Repository auf GitHub, um über Änderungen benachrichtigt zu werden.

Wenn Sie Fehler im Code finden oder Verbesserungsvorschläge haben, öffnen Sie bitte ein Issue oder reichen Sie einen Pull Request auf GitHub ein. Community-Beiträge helfen dabei, diese Ressource für alle besser zu machen.

Video-Tutorials

Ergänzend zum schriftlichen Inhalt wird dieses Buch durch eine umfassende Serie von Video-Tutorials unterstützt, die wichtige Konzepte durchgehen und zusätzliche Beispiele bieten:

<https://tinyurl.com/intro-gispro-videos>

Die Videos sind darauf ausgelegt, das schriftliche Material zu ergänzen, nicht zu ersetzen. Sie sind besonders hilfreich für:

- Visuelle Lerner, die davon profitieren, Code geschrieben und ausgeführt zu sehen
- Das Verständnis komplexer Konzepte durch mehrere Erklärungen
- Das Erlernen des Entwicklungsworflows und bewährter Praktiken
- Das Sehen, wie man Probleme angeht und Probleme debuggt

Die Playlist ist so organisiert, dass sie der Struktur des Buches folgt. Sie können sie der Reihe nach ansehen, während Sie durch das Buch voranschreiten, oder zu spezifischen Themen springen, wie benötigt.

Die Videos wurden im Herbst 2024 erstellt, als ich den Kurs **Introduction to GIS Programming**¹ an der University of Tennessee unterrichtete. Obwohl der Kurs abgeschlossen ist, bleiben die Videos relevant und können als Referenz für das Buch verwendet werden. Zusätzliche Videos werden in Zukunft hinzugefügt.

Kontakt aufnehmen

Ich begrüße Feedback, Fragen und Vorschläge von Lesern. Ihr Input hilft dabei, das Buch zu verbessern und es für die geospatiale Programmiergemeinschaft nützlicher zu machen.

Für buchbezogene Fragen und Diskussionen:

- GitHub Issues: <https://github.com/giswqs/intro-gispro/issues>
- GitHub Discussions: <https://github.com/giswqs/intro-gispro/discussions>

Arten von Feedback, die besonders hilfreich sind:

- Fehler oder unklare Erklärungen im Text oder Code
- Vorschläge für zusätzliche Beispiele oder Anwendungsfälle
- Ideen für neue Themen oder Kapitel
- Berichte über Kompatibilitätsprobleme mit verschiedenen Betriebssystemen oder Bibliotheksversionen
- Erfolgsgeschichten, wie Sie die Techniken aus dem Buch angewendet haben

Danksagungen

Dieses Buch wäre ohne die Beiträge und Unterstützung vieler Einzelpersonen und der breiteren Open-Source-Geospatial-Community nicht möglich gewesen.

Die Open-Source-Community: Dieses Buch baut auf der unglaublichen Arbeit unzähliger Open-Source-Entwickler auf, die das Python-geospatiale Ökosystem geschaffen und gepflegt haben. Besonderen Dank an die Entwickler und Betreuer von NumPy, Pandas, GeoPandas, Rasterio, Xarray, Rioxarray,

¹<https://geog-312.gishub.org>

Folium, ipyleaflet, MapLibre, GDAL und den vielen anderen Bibliotheken, die geospatiale Programmierung zugänglich machen.

Studierende und Kollegen: Die Fragen, Herausforderungen und Einsichten der Studierenden in meinen geospatialen Programmierkursen an der University of Tennessee haben den Ansatz und Inhalt dieses Buches geprägt. Ihr Feedback darüber, was funktioniert und was nicht, war von unschätzbarem Wert bei der Erstellung von Materialien, die Lernenden wirklich dienen.

Forschungspartner: Kollegen und Mitarbeiter in der geospatialen Forschungsgemeinschaft haben realweltliche Anwendungsfälle, Datensätze und Problemszenarien bereitgestellt, die die praktischen Beispiele im gesamten Buch informieren.

Familie und Freunde: Das Schreiben eines technischen Buches erfordert erhebliche Zeit und Konzentration. Ich bin dankbar für die Geduld und Unterstützung von Familie und Freunden, die die vielen Abende und Wochenenden verstanden haben, die diesem Projekt gewidmet waren.

Die breitere GIS-Community: Das geospatale Feld basiert auf einem Fundament des Wissens- und Werkzeugaustauschs. Dieses Buch ist Teil dieser Tradition, und ich fühle mich geehrt, zu den verfügbaren Ressourcen für das Erlernen geospatialer Programmierung beizutragen.

Dieses Buch wurde mit [MyST Markdown](#)² geschrieben und mit [Typst](#)³ und der [min-book](#)⁴ Vorlage kompiliert. Danke an die Entwickler und Betreuer der Typst- und MyST Markdown-Projekte. Besonderen Dank an [Maycon F. Melo](#)⁵ für die min-book-Vorlage und ihre Hilfe bei der Anpassung der Vorlage für dieses Buch.

Alle Fehler oder Auslassungen in diesem Buch bleiben meine Verantwortung. Ich bin verpflichtet, Probleme anzugehen und den Inhalt basierend auf Leserfeedback zu verbessern.

Über den Autor

Dr. Qiusheng Wu ist außerordentlicher Professor und Direktor der Graduiertenstudien im Department of Geography & Sustainability an der University of Tennessee, Knoxville. Er dient auch als Amazon Scholar. Dr. Wus Forschung konzentriert sich auf geospatiale Datenwissenschaft und Open-Source-Softwareentwicklung, mit Schwerpunkt auf der Nutzung großer geospatialer Daten und Cloud Computing zur Untersuchung von Umweltveränderungen, insbesondere der Dynamik von Oberflächenwasser- und Feuchtgebietsüberflutungen. Er ist der Schöpfer mehrerer weit verbreiteter Open-Source-Python-Pakete, einschließlich [geemap](#)⁶, [leafmap](#)⁷, [segment-geospatial](#)⁸ und [geoai](#)⁹, die fortgeschrittene geospatiale Analyse und interaktive Visualisierung unterstützen. Seine Open-Source-Arbeit ist bei [Open Geospatial Solutions](#)¹⁰ auf GitHub verfügbar.

²<https://mystmd.org>

³<https://github.com/typst/typst>

⁴<https://github.com/mayconfmelo/min-book>

⁵<https://github.com/mayconfmelo>

⁶<https://geemap.org>

⁷<https://leafmap.org>

⁸<https://samgeo.github.io>

⁹<https://opengeoai.org>

¹⁰<https://github.com/opengeos>

Lizenzierung und Urheberrecht

Dieses Buch umfasst die Prinzipien der offenen Wissenschaft und offenen Bildung. Zur Unterstützung von Transparenz, Lernen und Wiederverwendung werden die **Codebeispiele** in diesem Buch unter einer [Creative Commons Attribution 4.0 International \(CC BY 4.0\)](#) Lizenz veröffentlicht. Dies bedeutet, dass Sie den Code frei kopieren, modifizieren und verteilen dürfen, auch für kommerzielle Zwecke, solange eine angemessene Anerkennung gegeben wird.

Bitte führen Sie die Codeverwendung auf, indem Sie das Buch zitieren oder auf das GitHub-Repository verlinken:

Wu, Q. (2025). *Introduction to GIS Programming: A Practical Python Guide to Open Source Geospatial Tools*. <https://gispro.gishub.org>

Während der Code frei verfügbar ist, sind der **Text, Abbildungen und Bilder** in diesem Buch **urheberrechtlich geschützt** durch den Autor und dürfen ohne ausdrückliche Genehmigung nicht reproduziert, weiterverteilt oder modifiziert werden. Dies umfasst alle geschriebenen Inhalte, benutzerdefinierten Diagramme und eingebetteten Visualisierungen, sofern nicht anders angegeben.

Wenn Sie nicht-Code-Material aus dem Buch wiederverwenden oder anpassen möchten - zum Beispiel für Lehre, Präsentationen oder Publikationen - kontaktieren Sie bitte den Autor, um eine Genehmigung zu beantragen.

Dieser duale Lizenzierungsansatz hilft dabei, den offenen Zugang zu Lernmaterialien mit dem Schutz originaler kreativer Arbeit in Einklang zu bringen. Vielen Dank, dass Sie diese Bedingungen respektieren und die Open-Source-Geospatial-Community unterstützen.

Einführung in die GIS-Programmierung

Ein praktischer Python-Leitfaden für Open-Source-Geoinformatik-Tools

Da die Nachfrage nach räumlicher Datenanalyse in verschiedenen Disziplinen – von Umweltwissenschaften und Stadtplanung bis hin zu öffentlicher Gesundheit und Wirtschaft – stetig wächst, steigt auch der Bedarf an zugänglichen und skalierbaren Werkzeugen, die mit der Komplexität geospatialer Daten umgehen können. Einführung in die GIS-Programmierung mit Python ist Ihr praktischer Leitfaden zur Beherrschung der räumlichen Analyse.

Ob Studentin, GIS-Fachkraft, Datenanalystin oder Entwickler*in – dieses praxisorientierte Buch bietet einen strukturierten und einsteigerfreundlichen Einstieg in die geoinformatische Programmierung mit Python. Sie beginnen mit den Grundlagen von Python und arbeiten sich schrittweise zu fortgeschrittenen räumlicher Analyse und interaktiver Visualisierung vor – unter Nutzung leistungsstarker Open-Source-Bibliotheken wie GeoPandas, Rasterio, Leafmap, Geemap und GDAL.

Was dieses Buch besonders auszeichnet, ist sein beispielorientierter Ansatz: Sie arbeiten mit echten Datensätzen, lösen reale Probleme und erstellen aussagekräftige Ergebnisse – von statischen Karten bis hin zu dynamischen Webanwendungen. Jedes Kapitel ist sorgfältig aufgebaut, um Ihre Fähigkeiten Schritt für Schritt zu entwickeln – mit einer Kombination aus Theorie, kommentiertem Code, praxisnahen Übungen und hilfreichen Tipps zur Fehlerbehebung.

Was Sie lernen werden:

- Wie Sie Ihre Entwicklungsumgebung mit conda, VS Code, Git, Docker und Cloud-Computing-Tools einrichten
- Zentrale Python-Programmierkenntnisse – von Variablen und Datenstrukturen bis hin zu Datei-Ein-/Ausgabe und Pandas
- Verarbeitung von Vektor- und Rasterdaten, interaktive Kartierung, 3D-Visualisierung und Geoprozessierung
- Cloudbasierte Geodatenverarbeitung mit Google Earth Engine sowie fortgeschrittene Themen wie hyperspektrale Datenanalyse und räumliches SQL



Scannen Sie den QR-Code, um
die Buch-Website zu besuchen
und die Codebeispiele von
<https://gispro.gishub.org>
herunterzuladen