

[研究ノート]

Python を用いた地理空間データ・ インタラクティブ・ビジュアライゼーション — そのフレームワークと実装要点 —

蔣 湧

1 はじめに

本文において、地理空間データの視覚化は、地物間の隣接性や到達性などの空間的属性と、地物を持つ非空間的属性を動的に結び付き、マップとグラフを用いて表現することを指す。通常、こうした地理空間データの可視化は、地理情報システム (GIS) やデータ可視化などの技術を融合したシステムを用いて実現されるが、そのシステムの構築は容易ではない。

一方、地理空間情報が可視化されることで、物事の傾向や関連がよく見えるようになり、データに基づいた意思決定の分野において注目が集まっている。2019 年 JOHNS Hopkins University が開発した COVID-19 Dashboard (図 1) が公表された。COVID-19 Dashboard は地理空間データの Interactive Visualization として、感染地域(地物)と感染地域の統計値(地物の属性)をマップとグラフの連動で、世界規模の感染状況を時空間的に表現している。その COVID-19 Dashboard の影響を受け、近年、Microsoft Power BI、Looker Studio、metabase、ArcGIS Dashboard など含め、数多くのダッシュボードシステムがクラウドサービスの形態で提供され、情報通信技術 (ICT) の素人

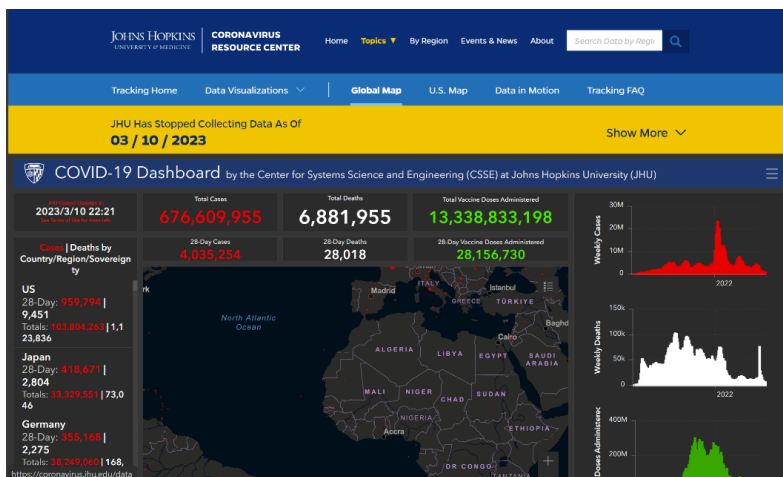


図 1 COVID-19 Dashboard (<https://coronavirus.jhu.edu/map.html>)

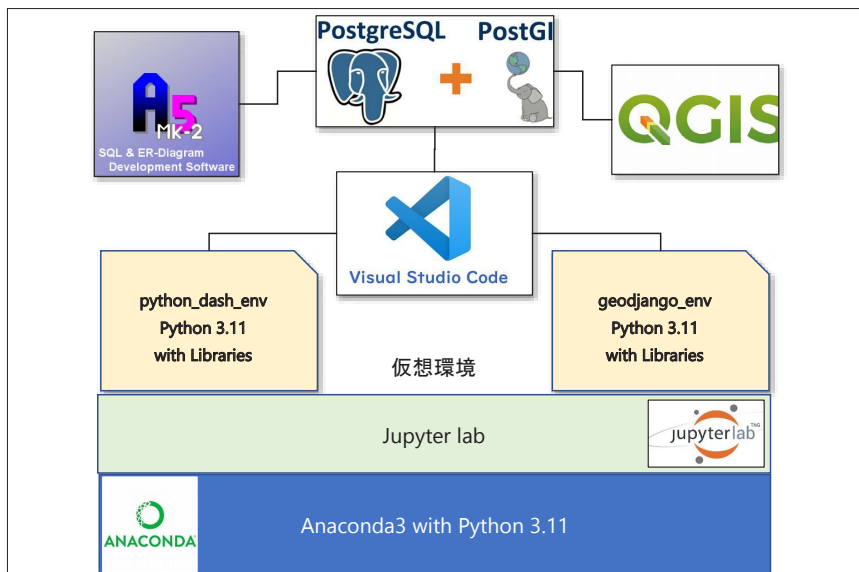


図 2 地理空間データ可視化のシステム環境

でも、自ら情報を発信することができるようになった。

しかし、住民の個人情報を扱う地方自治体や、地理空間情報・AIの研究を目指す学生と研究者は、情報管理の課題や二次開発の可能性などの理由から、こうしたダッシュボードシステムの使用が敬遠されてきた。筆者は、行政向けの「農地現況ダッシュボードシステム」の開発にあたって、Python オープンソースを用いた地理空間情報ダッシュボードシステムの構築を試みた。本稿は、研究ノートとして、Python を用いた地理空間情報可視化のフレームワークとその実装過程の要点（図 2）をまとめるものであるが、実装方法の詳細については省略する。

2 システムの概要と構築手順

2.1 システムの概要

図 2 は、Windows11 のプラットフォームにおいて、Anaconda で作成した 2 つの Python 仮想空間：ダッシュボード仮想空間と WebGIS 仮想空間に、外部のデータベース PostgreSQL、地理情報システム QGIS とプログラミング統合開発環境 (IDE) Visual Studio Code (VSCode) を組み込んだ地理空間データ可視化システムを示す。システムは 4 つの部分により構成される（表 1）。

空間データベースは PostgreSQL 16 と PostGIS 3.4 により構成する。データベースの ER 図 (Entity Relation Diagram) 作成ツールは、松原正和氏が開発した A5:SQL Mk-2 を採用する。地理空間情報システムは、2023 年 10 月の時点で最新の QGIS 3.32 を使用し、プログラミング統合開発環境はマイクロソフト社の Visual Studio Code (VSCode) の 1.83 を使う。

Python フレームワークは、Python3.11 をベースにした Anaconda を採用した。Anaconda をインストールと、Jupyter Lab の Python プログラミングエディタがすぐ使える。目標に合わせた二つの仮想環境、python_dash_env と geodiango_env を作成し、それぞれダッシュボードシステムと WebGIS の

表 1 システムの構成

構成部分	ソフトウェア・バージョン	用途
空間データベース	PostgreSQL 16	データベース本体
	PostGIS 3.4	空間拡張モジュール
	A5:SQL Mk-2	データベース ER 図の作成
地理情報システム	QGIS 3.32	地図作成と空間解析
統合開発環境	VSCoDe 1.83	Python プログラミング環境
	Jupyter Lab 3.6.3	Python プログラミングエディタ
Python フレームワーク	Anaconda 3	Python3.11 フレームワーク
	python_dash_env	ダッシュボード用の仮想空間
	geodjango_env	WebGIS 用の仮想空間

ソフトウェアは 2023 年 10 月時点でダウンロードしたもの

開発に関連する Python ライブラリ群を入れる。

2.2 システム構築の手順

図 2 に示したシステムを構築する際に、以下の順番の通り、まず、ソフトウェアをダウンロードし、インストールする。次に、ソフトウェア間の接続設定を行う。

- ① PostgreSQL16、PostGIS 3.4 のダウンロードとインストール
- ② A5:SQL Mk-2 のダウンロードとインストール
- ③ QGIS 3.32 のダウンロードとインストール
- ④ VSCoDe 1.83 のダウンロードとインストール
- ⑤ Anaconda3 と Python3.11 のダウンロードとインストール（Jupyter Lab が含まれている）
- ⑥ 仮想空間 python_dash_env と geojango_env の作成
- ⑦ 仮想環境にライブラリ群のインストール（詳細内容は表 2 ～表 5 を参照）
- ⑧ VSCoDe における仮想空間に関する環境設定
- ⑨ VSCoDe と PostgreSQL の接続設定
- ⑩ PostgreSQL と QGIS の接続設定

⑪ PostgreSQL と A5:SQL Mk-2 の接続設定

3 空間データベース環境のインストール

3.1 PostgreSQL 16 と PostGIS 3.4 のインストール

公式ホームページの URL : <https://www.postgresql.jp/> から、Windows x86-x64 の PostgreSQL 16 インストーラーをダウンロードし、インストールを行う。

PostgreSQL16 本体のインストールを完了すると、図 4 の空間拡張の画面が自動的に開かれる。ここから PostGIS 3.4 の選択にチェックを入れ、インストールを続けて行う。

インストール終了後、PostgreSQL データベース管理システム pgAdmin をクリックし、図 5 のデータベース画面が開かれる。

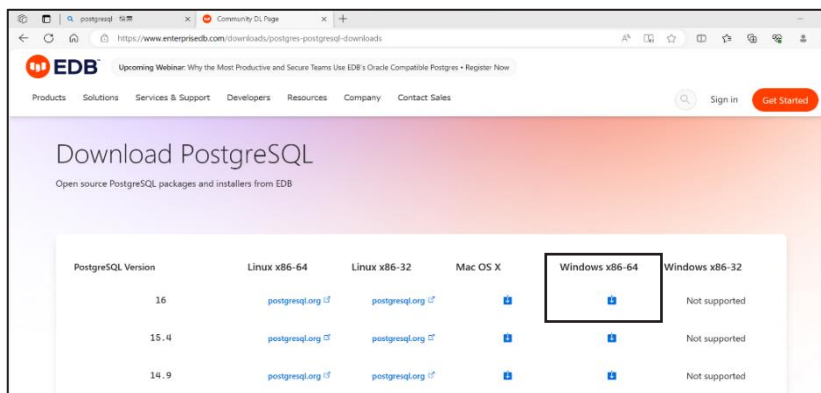


図 3 PostgreSQL16 の Windows x86-64 インストーラー

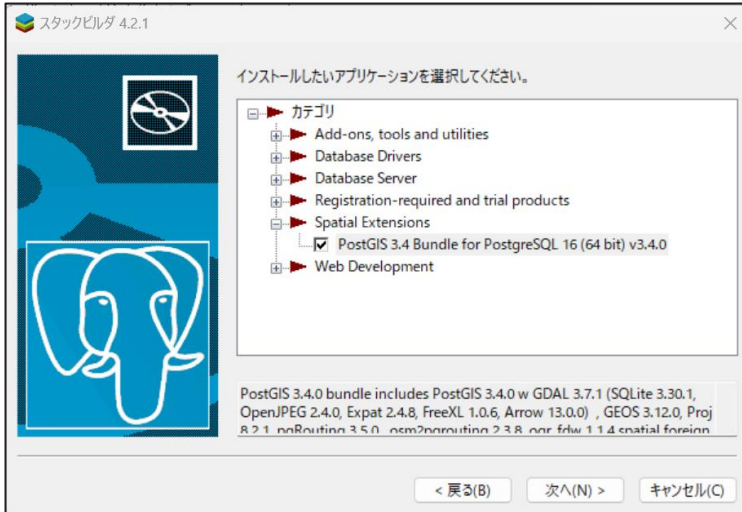


図 4 PostgreSQL の空間拡張 PostGIS のインストール

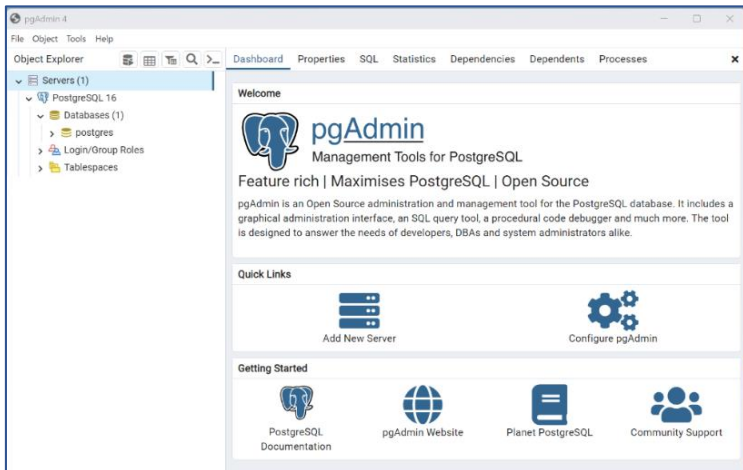


図 5 PostgreSQL の管理システム pgAdmin

3.2 ER 図作成ツールのインストール

次は、PostgreSQL データベースから自動的に ER 図を書き出すツール A5:SQL Mk-2 をインストールする。この松原正和氏が開発したツールは、サイト <https://a5m2.mmatsubara.com/> に公開している。図 6 の画面にある「Download」ボタンを押すと、インストーラダウンロードとソフトのインストール作業が一括して行われる。図 7 は完成したデータベース ER 図を示す。



図 6 A5:SQL Mk-2 のインストール

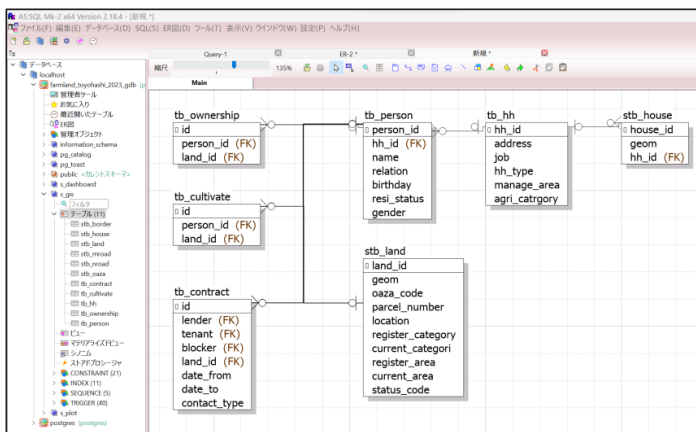


図 7 A5:SQL Mk-2 が作成したデータベース ER 図

4 地理情報システム QGIS のインストール

公式ホームページ <https://www.qgis.org/ja/site/forusers/download.html> にアクセスし、現時点（2023年10月）の最新バージョン QGIS 3.32 インストーラーをダウンロードする（図8）。

インストーラー指示の通りにインストールすると、図9に示す QGIS の画面がたどり着く。QGIS と PostgreSQL のつなぎ作業が必要であるが、詳細について後ほど紹介する。



図8 QGIS 3.32 のインストーラー

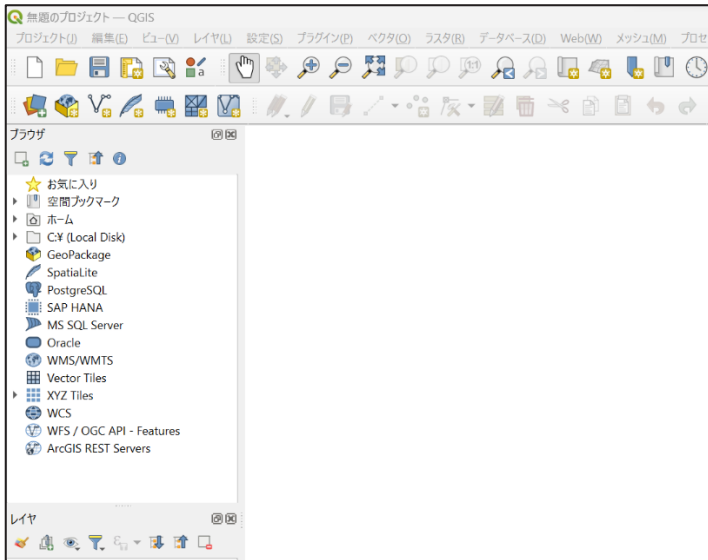


図9 QGIS 3.32 の画面

5 開発統合システムの環境 VSCoDe のインストール

公式ホームページ <https://code.visualstudio.com/Download#> から、Windows11 x64 用のインストーラーをダウンロードできる (図 10)。図 11 はインストール直後の画面を示す。现阶段において、VSCoDe の本体のみをインストールし、拡張ツールの追加や環境の設定については、後ほど触れることになる。



図 10 VSCoDe インストーラーのダウンロード

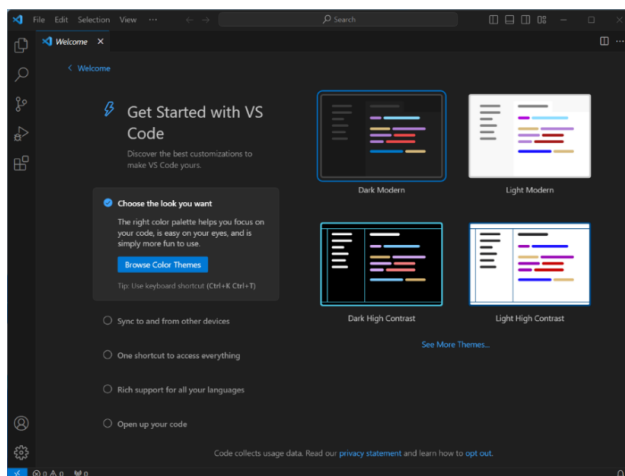


図 11 VSCoDe インストール直後の画面

6 Anaconda3 と Python3.11 のインストール

次は、Python プログラミングのフレームワーク Anaconda をインストールする。公式サイト URL : <https://www.anaconda.com/> で、トップページにアクセスする。「Free Download」リンクをクリックする (図 12)。

2023 年 10 月時点の Anaconda は Python 3.11 と対応しているので、本稿は、図 13 に示すように、Windows 64bit 用の Python 3.11 対応のインストーラーをダウンロードする。指示の通りにインストールと、図 14 に示した

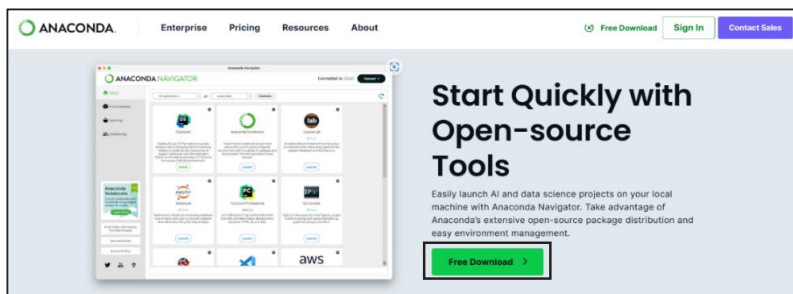


図 12 Anaconda の公式サイト

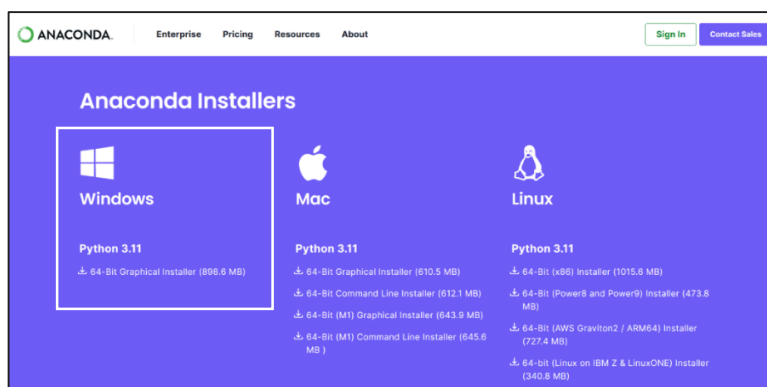


図 13 Windows 64-bit、Python 3.11 のインストーラー

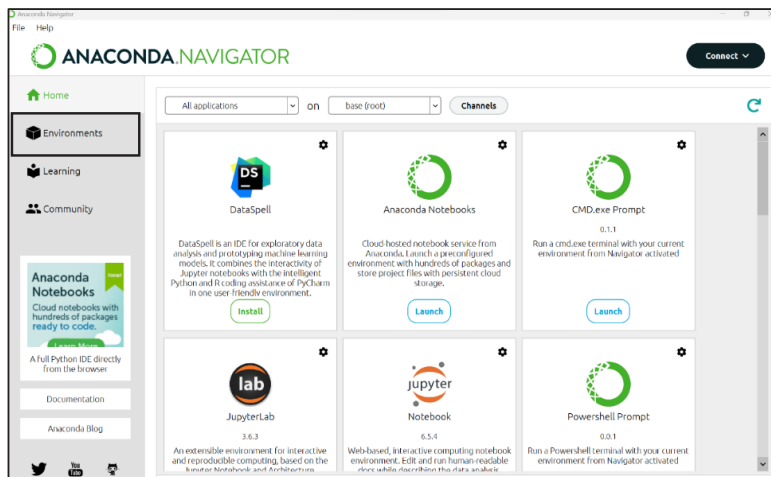


図 14 Anaconda Navigator の画面

Anaconda ナビゲーターが開かれる。

Anaconda は、データサイエンスや機械学習など分野において、およそ 8000 点以上のパッケージ（あるいはライブラリ）をオープンソースとして、バージョン管理と統合配布（distribution）を行う専門サイトである。一方、Anaconda Navigator は、Anaconda を利用する際に、PC ユーザーの手元に使用するツールである。

7 仮想環境の構築

7.1 仮想環境の新規作成

仮想環境は、特定の Python バージョンに基づき、研究に必要な Python ライブラリ群を用いたプログラミング環境を指す。Anaconda の使用は、こうした仮想環境の作成から始める。本稿は、WebGIS 構築とダッシュボードの作成、2つの目標を達成するために、それぞれ `geodjango_env` と `python_dash_env`、

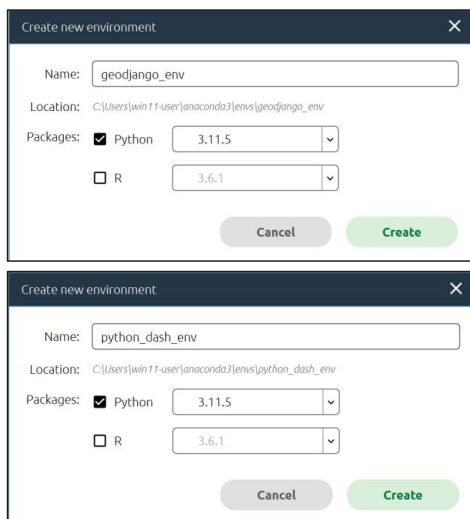


図 15 仮想空間の新規作成

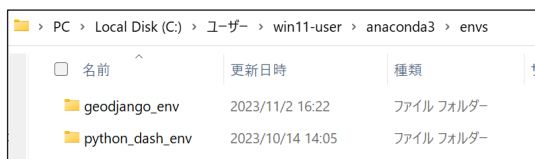


図 16 仮想空間の新規作成

二つの仮想空間を設け、それに合わせた Python プログラミングの環境構築を試みる。

図 14 の Anaconda Navigator の左側にある「Environments」をクリックし、仮想環境の作成画面が現れる。さらに、新規作成のボタンを押すと、図 15 に示す仮想空間の新規作成画面が開られる。ここに、仮想空間名と使用する Python 言語のバージョンを入れ、「Create」ボタンを押す。

仮想空間のフォルダは、パス C:\Users\win11-user\anaconda3\envs に保存される (図 16)。

7.2 仮想環境 geodjango_env への Python ライブラリ実装

表 2 は GeoDjango を用いた WebGIS の構築に必要な Python ライブラリを示す。インストール作業は、表 3 に示すインストール順番とインストールコマンドに従い、仮想空間 geodjango_env のターミナにて行う。

表 2 仮想空間 geodjango_env にインストールライブラリ群

カテゴリ	インストールライブラリ	用途
データベース	psycogp2	古い PostgreSQL 対応
	psycogp3	PostgreSQL14 以降対応
Web アプリケーション	django	WebApp フレームワーク
地理空間データ	GDAL	地理空間情報ライブラリ

表 3 インストールコマンドとライブラリのバージョン (geodjango_env)

順番	ライブラリ	インストールコマンド	Version
1	psycogp2	pip install psycogp2	2.9.9
2	psycogp3	pip install psycogp [binary]	3.1.12
3	django	conda install django	4.1
4	GDAL	conda install -c conda-forge gdal=3.6.1	3.6.1

バージョンは 2023 年 10 月時点のもの

最後に、pip list のコマンドで、インストール済みのライブラリのバージョンを確認することができる (図 17)。なお、Django に関するプロジェクト作成、アプリケーション作成、Web サーバーの起動と GDAL パス設定などの環境整備は、次の章に解説する。

```

C:\WINDOWS\system32\cmd.exe
(geodjango_env) C:\Users\win11-user>pip list
Package            Version
-----
asgiref            3.5.2
Django             4.1
GDAL               3.6.1
numpy              1.26.0
pip                23.2.1
psycogp            3.1.12
psycogp-binary    3.1.12
psycogp2          2.9.9
setuptools         68.0.0
sqlparse           0.4.4
typing_extensions  4.8.0
tzdata             2023.3
wheel              0.41.2

```

図 17 インストール済みのライブラリの確認

7.3 仮想環境 python_dash_env への Python ライブラリ実装

次の表 4 と表 5 は、それぞれ仮想空間 python_dash_env にインストールするライブラリの一覧とインストール作業のコマンドを示す。前節の作業と同様、仮想空間 python_dash_env のターミナに入り、表 5 に示す順番にライブラリをインストールするが、一部のライブラリ（括弧（）に示すライブラリ）は自動的にインストールされることになる。

表 4 仮想空間 python_dash_env にインストールライブラリ群

カテゴリ	インストールライブラリ	用途
データベース関連	psycogp2	Sqlalchemy の使用に必須
	sqlalchemy	Pandas 推奨アダプター
空間データ形式関連	geojson	地理空間データ形式
データ分析関連	pandas (numpy)	データ解析と数値計算
	geopandas	空間データの解析
グラフ・ダッシュボード関連	dash (plotly, flask)	グラフとダッシュボードの作成と配信
	matplotlib	グラフ作成

括弧（）には自動的にインストールされるライブラリ

表 5 インストールコマンドとライブラリのバージョン (python_dash_env)

順番	インストールライブラリ	インストールコマンド	Version
1	psycogp2	pip install psycogp2	2.9.9
2	sqlalchemy	pip install sqlalchemy	2.0.22
3	geojson	pip install geojson	3.0.1
4	pandas (numpy)	pip install pandas	2.1.1
5	geopandas	pip install geopandas	0.14.0
6	dash (plotly, flask)	pip install dash	2.14.0
7	matplotlib	pip install matplotlib	3.8.0

バージョンは 2023 年 10 月時点のもの

最後に、pip list でインストールされたライブラリのバージョンを確認する。

8 仮想空間と周辺装置を取り入れた VSCode の環境設定

前章まで、ダッシュボードシステムの開発に関わる主なソフトウェアと

Python ライブラリのインストールは完了した。本章は、インストール済みの各ソフトウェア間の設定について解説する。

まず、もう一度図2のシステム構成図を確認してほしい。本稿に目指すダッシュボードシステムには、主に①仮想環境 (geodjango_env と python_dash_env)、②周辺装置 (PostgreSQL データベースと QGIS)、③統合開発環境 VSCode、三つの部分により構成される。ここで、統合開発環境 VSCode は、同時に仮想空間と周辺措置とつながることで、システム構造上、さらに開発作業においても中心的な役割を果たしている。次は VSCode と仮想空間、VSCode と PostgreSQL、PostgreSQL と QGIS のつなぎ設定の要点をまとめる。設定の詳細については参照文献を参照してほしいが、本稿には省略する。

8.1 VSCode と仮想区間 geodjango_env のつながり

まず、VSCode における Python プログラミングの Workspace 環境を作成する。

【手順1】 Django プロジェクト proj_webgis の新規作成

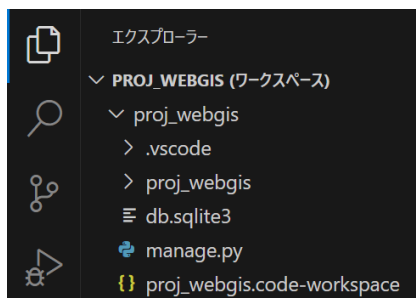
仮想空間 geodjango_env のターミナにおいて、以下のコマンドでプロジェクト proj_webgis を作成する。

```
>django-admin startproject proj_webgis
```

すると、フォルダ C:\Users\win11-user\proj_webgis が作成され、中に複数の Python ファイルが保存されている。

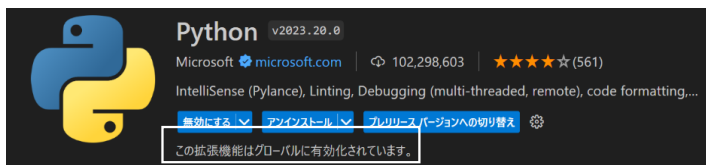
【手順2】 proj_webgis における VSCode の Workspace の作成

- VSCode の「フォルダを開く」ボタンを押し、フォルダ proj_webgis を開く。
- フォルダをワークスペースとして保存する (右図)。

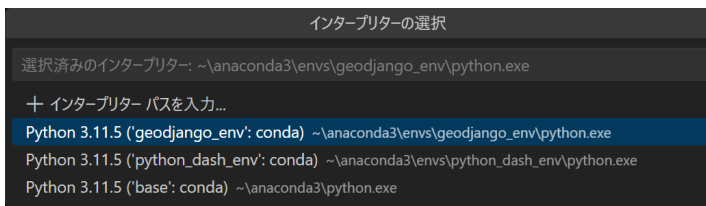


【手順 3】 Workspace に Python 環境の設定

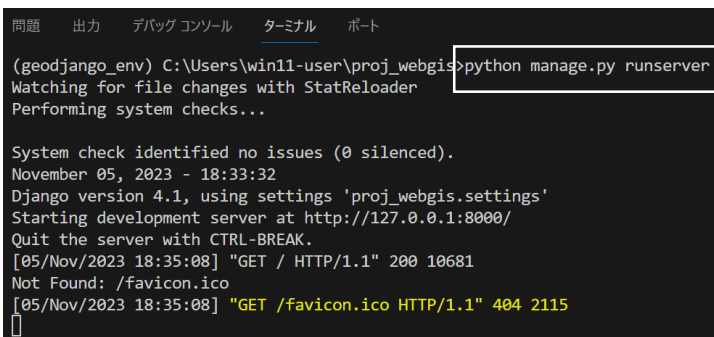
- VSCode の「拡張機能」を使って、Python 機能が有効していることを確認する。



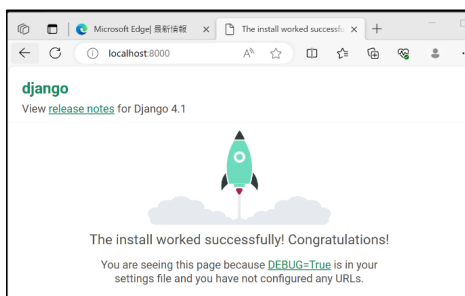
- Shift+Ctrl+P を押し、開かれた Python インタープリターの選択画面から、「Python3.11.5 ('geodjango_env':conda)」を選ぶ。こうして VSCode のワークスペースにおいて、仮想環境 geodjango_env の Python インタープリターを利用することになる。



- VSCode で仮想環境 geodjango_env のターミナルを開き、Web サーバーを起動する。



- ブラウザでサーバー起動画面を確認する



【手順4】GDAL 関連のパス設定

Django に地理空間データライブラリ GDAL (Geospatial Data Abstraction Library) を導入すると、Django のデータモデルに地理空間データの使用が可能になり、GeoDjango と称するようになる。GeoDjango を用いた WebGIS の構築において、外部の地理空間データ（例えば、シェープファイルや PostGIS のテーブルなど）と Django 内部のデータモデルの間に、データ構造のマッピングとデータの相互移行が求められる。以下は、そのための環境整備を行う。

- Django プロジェクト proj_webgis の settings.py ファイルの最後に、以下の「GDAL_LIBRARY_PATH」を設定しておく。

```
GDAL_LIBRARY_PATH =  
r'C:/Users/win11-user/anaconda3/envs/geodjango_env/Library/bin/gdal'
```

- 仮想環境 geodjango_env のターミナルにおいて、以下のコマンドで Python シェルを開く。

```
>python manage.py shell
```

すると、仮想環境 geodjango_env のターミナルが Python シェルに切り替え、ここでデータ移行の Python パッケージを実行可能になる。また、「quit()」コマンドを用いて、再び仮想環境 geodjango_env のターミナルに戻られる。

```
(geodjango_env) C:\Users\win11-user\proj_webgis>python manage.py shell
Python 3.11.5 | packaged by Anaconda, Inc. | (main, Sep 11 2023, 13:26:23) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
(InteractiveConsole)
>>> []
```

8.2 VSCode と仮想区間 pathon_dash_env のつながり

【手順1】プロジェクトフォルダ proj_python_dash_workspace の新規作成

- 単に作業専用のフォルダ proj_python_dash_workspace を作成する。

【手順2】proj_python_dash_workspace における VSCode の Workspace 作成

- 前項 8.1 の手順 2 と同様

【手順3】Workspace に Python 環境の設定

- 前項 8.1 の手順 3 と同様であるが、最後の「ブラウザでサーバー起動画面の確認」は不要。

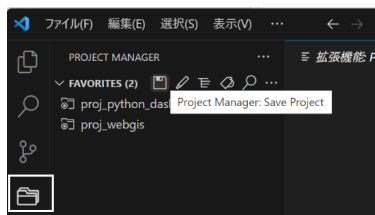
【手順4】VSCode におけるプロジェクト切り替え機能の追加

現時点において、VSCode の開発環境には二つのプロジェクトがあり、そのため 2 つのワークスペースを設けた。プログラミング作業中に、ワークスペースの切り替えが必要になり、そのために VSCode の「Project Manager」ツールを導入する。

- VSCode の「拡張機能」から「Project Manager」をインストールする。



- 1 つワークスペースを開く状態で、「Project Manager」ボタンを押し、「Save Project」をクリックするだけで、Project Manager に現在のプロジェクト



Python を用いた地理空間データ・インタラクティブ・ビジュアライゼーション

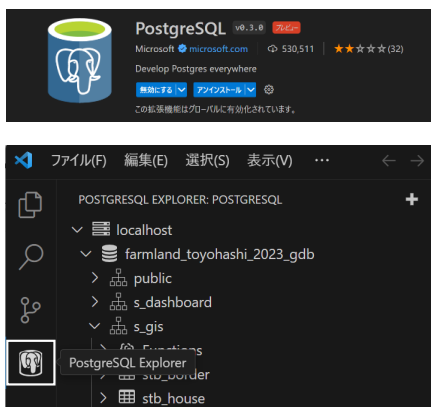
トが登録される。

- 登録されたプロジェクトをクリックすると、ワークスペースが切り替えられる。

8.3 VSCode と PostgreSQL のつながり

VSCode の開発環境に PostgreSQL との接続は可能である。

- VSCode の「拡張機能」から「PostgreSQL」ツールを追加する。
- 「PostgreSQL Explore」を押し、データベースの接続設定やデータベースクエリ作成などの作業が可能になる。



8.4 PostgreSQL と QGIS のつながり

最後に、QGIS ブラウザにある PostgreSQL への接続口を使って、データベースとの接続を行う。その結果、データベーステーブルを直接マップレイヤへ追加することができ、QGIS の「DB マネージャー」(下図) を使って、SQL 言語を用いた空間解析や GIS コンテンツの作成が可能になる。

最後に、Windows11 のプラットフォームにおいて、Python を用いた地理空間データの可視化環境システムを Windows11 のフォルダベースでまとめた(図 18)。

hh_jd	address	job	hh_type	manage_area	agri_catrgory
1	神野新田町字...	NULL	NULL	NULL	NULL
2	車呂中村町2 2...	NULL	NULL	NULL	NULL
3	車呂市場町1 2...	NULL	NULL	NULL	NULL
4	車呂公文町2 0...	NULL	NULL	NULL	NULL
5	東岩田一丁目1...	NULL	NULL	NULL	NULL
6	大府市横根町羽...	NULL	NULL	NULL	NULL
7	東盛四丁目1 7...	NULL	NULL	NULL	NULL
8	神野新田町字タ...	NULL	NULL	NULL	認定農業者
9	車呂市場町2 4...	NULL	NULL	NULL	NULL
10	神野新田町字サ...	NULL	NULL	NULL	NULL
11	車呂市場町2 9...	NULL	NULL	NULL	NULL
12	神野新田町字ロ...	NULL	NULL	NULL	NULL
13	神野新田町字ロ...	NULL	NULL	NULL	NULL

```

C:/
├─/Program Files
│   └─/PostgreSQL (データベース)
│       └─/16 (PostgreSQL 16)
│           └─/share/extension/postgis 3.4 (空間拡張)
│               └─/QGIS 3.32.3 (GIS)
│                   └─/ WindowsApps/A5SQLMk-2_2.18.4.0_x64 (ER 図関連)
├─Users
│   └─/win11¥user
│       └─-Anaconda3
│           └─-envs(仮想環境)
│               └─/python_dash_env
│                   └─/python 3.11
│                       └─/...関連ライブラリ (表2 と表3)
│               └─-geodjango_env
│                   └─python 3.11.5
│                       └─/...関連ライブラリ (表4 と表5 と GDAL)
│           └─/proj_python_dash_workspace (Workspace)
│               └─/python_dash (ダッシュボード関連コードの保存先)
│           └─/proj_webgis (Workspace)
│               └─ ( Django 関連コードの保存先)

```

図 18 Windows11 フォルダベースのシステム構造

9. まとめ

Elias Dabbas は、彼の著書 “Interactive Dashboard and Data Apps with Plotly and Dash” において、データと情報について以下のように述べている。“Information is power, someone say. Data is the new gold, others proclaim. But raw information and data are often meaningless without context. Data is an asset only when properly analyzed, interpreted, and understood”。情報こそが力だと言う人がいて、データこそが新しい黄金だと主張する人もいる。しかし、生の情報やデータはしばしば文脈がなく、意味もない。データは適切に分析され、解釈され、理解されたときこそ財産となる。データを適切に分析、解釈と理解するために、本稿の地理空間データ・インタラクティブ・ビジュアライゼーションシステムが考案された。

近年、さまざまな地理空間データダッシュボードシステムが登場しているが、その中で Python のオープンソースを活用することが注目されている。Python は豊富なライブラリ資源と知識共有のエキスパートコミュニティを備えており、これが Python オープンソースを利用する主な理由となっている。一方で、異なるソフトウェアやライブラリを組み合わせることにより、地理空間データダッシュボードシステムの構築には、知識と経験が必要であり、初学者にとっては容易ではない。

本稿は、筆者の研究実験のメモとして、①システムの構成、②オープンソースの所在、③ライブラリのインストールの要点、④システム要素間の接続設定についてまとめたものである。初学者のプロジェクトのスタートアップに役立つことを期待する。

参考文献

GeoDjango 関連

1. 蔣湧 (2024) 「地域研究のための地理空間情報統合管理システム -Python, Django, PostGIS, QGIS, VSCode を用いて-」、古今書院
2. Paul Crickard, Eric van Rees, Silas Toms (2018), “Mastring Geospatial Analysis with Python”, Puckt Publishing Lid.
3. Puckt Publishing (2014), “Leaflet.js Essentials”, Puckt Publishing Lid.
4. (株) アンク (2019) 「Django4 Web アプリ開発実装ハンドブック」、秀和システム。

Dashboard 関係

1. Teddy Petrou (2021), “Build an Interactive Data Analysis Dashboard with Python”, Printed in Great Britain by Amazon.
2. @driller、小川英幸、古木友子 (2020)、 「Python インタラクティブ・データビジュアライゼーション入門－Plotly/Dash によるデータ可視化と Web アプリ構築」、朝倉書店
3. Elias Dabbas (2021), “Interactive Dashboard and Data Apps with Plotly and Dash”, Puckt Publishing Lid.
4. Adam Schroeder, Christian Mayer, Ann Marie Ward (2022), “The Book of Dash-Build Dashboards with Python and Plotly”, No Starch Press, San Francisco.

QGIS と PostgreSQL 関連

1. 蔣湧 (2019)、「地域研究のための空間データ分析入門－QGIS と PostGIS を用いて」、古今書院。
2. 蔣湧 (2022)、「地域研究のための空間データ分析 応用編－QGIS と PostGIS を用いて」、古今書院。