

Python 利用環境構築ガイド(2023年10月10日版)

このガイドは、Miniforge と呼ばれるパッケージソフトウェアを利用して、メッシュ農業気象データ処理のための Python 利用環境を PC にセットアップする手順を説明するものです。初めて Python を使用する人と、「以前人に勧められて Python をインストールしたけど結局使っていない」という人を対象としています。きちんと保守された Python 環境をすでにお持ちの方は、4. で示すモジュールを必要に応じて追加するだけで差し支えありません。

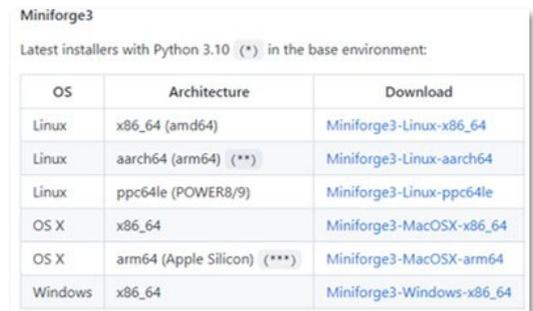
1. Miniforge について

Miniforge は、プログラミング言語 Python の本体と、これに関連する各種のソフトウェアの管理機能（インストールやアンインストール、アップデートなど）を提供するパッケージソフトウェアです。このような製品としては、Anaconda が大変有名ですが、利用者によっては有償であること、インストールされるソフトウェアが膨大であることなどから、このガイドでは、だれでも無償で利用でき、コンパクトな Miniforge を採用しています。なお、Anaconda の縮小版として、Miniconda というパッケージもありますが、Miniforge はこれよりもさらにコンパクトです。

2. Miniforge の入手

Miniforge のインストーラは、下記の Web ページから入手します。Windows 版、Mac 版、Linux 版があるので、インストール対象に合ったものを選んでください（図1）。

<https://github.com/conda-forge/miniforge>



OS	Architecture	Download
Linux	x86_64 (amd64)	Miniforge3-Linux-x86_64
Linux	aarch64 (arm64) (**)	Miniforge3-Linux-aarch64
Linux	ppc64le (POWER8/9)	Miniforge3-Linux-ppc64le
OS X	x86_64	Miniforge3-MacOSX-x86_64
OS X	arm64 (Apple Silicon) (***)	Miniforge3-MacOSX-arm64
Windows	x86_64	Miniforge3-Windows-x86_64

図 1. Miniforge のダウンロードページ。インストールする PC の環境に合ったものを選択する。

3. Miniforge のインストール

「以前人に勧められて Python をインストールしたけど結局使っていない」という人は、まず、古い Python 製品をアンインストールしてください。アンインストールの手順は OS やそれぞれの製品で異なりますが、一般的には、一般のプログラムと同様、コントロールパネルの「プログラムと機能」からアンインストールをします (Windows の場合)。製品によっては、インストールの際、環境変数に手を加えるものもあるので、



図 2. 古い Anaconda のアンインストール。「プログラムと機能」から Python 3.6... を選択する。

そのような場合は、環境変数も綺麗に戻しておいてください。なお、Anaconda3 5.3.1以前のバージョンをインストールしていた場合は、「プログラムと機能」のリストに「Anaconda」とは表示されず、「Python 2.7.x(Anaconda3.x.x)」などのように表示されるので注意してください(図2)。

次に、これからインストールするユーザアカウントについて確認してください。インストールは管理者では行わず、一般ユーザで行います。また、ユーザアカウント名に漢字やひらがな等日本語が使われているとMiniforgeが正常に動作しなくなることがあるので、アルファベットだけのアカウント名に変更するか、Miniforgeを利用する専用のアカウントを新規作成してください。

上記2点が確認出来たら、ダウンロードしたインストーラのアイコンをダブルクリックしてインストールを開始します。基本的には、デフォルトのまま「Next >」ボタンをクリックして進めば大丈夫です。利用者だけがMiniforgeを利用できるようにインストールする方法(インストールウィザードでは「Just Me」)と、PCの全利用者が利用できるようにインストールする方法(同「All users」)とを選択することが求められますが、その時は、前者(デフォルト)を選択してください。インストール先は

「C:\Users\username\miniforge3」としてください(usernameは利用者アカウント名)。また、Windowsにおける設定について聞かれますが、「Create start menu shortcuts(supported packages only).」を選択してください。図3は、Windows PCに、Miniforgeをインストールしているときのウィザード画面です。図3を参考にインストールを実行してください。

Mac版(Intel/M1)の場合は、インストーラではなくインストールプログラムの形式(シェルスクリプト)で提供されています。お持ちの端末にあったMiniforgeの導入方法をwebなどでお調べください。また、モジュール追加は、Windows版と同様のコマンド操作を試してください。



図 3. Miniforge のインストールウィザード。

4. ライブラリのインストール

Python はプログラム言語の一つで、文法が平易なため誰でも使い易く、様々な分野で利用されています。

ライブラリは、一言でいうと、Python に機能を追加するためのソフトウェアです。利用者が必要なライブラリを選んでインストールすることで、その人にとって最適な Python にすることができます。メッシュ農業気象データを処理するために必要な外部モジュールを以下に示します。

Numpy 多次元配列を効率よく取り扱えるようになります

Scipy 科学技術計算の関数ができるようになります

matplotlib 分布図やグラフを描けるようになります

netcdf4 NetCDF と呼ばれる形式のファイルを取り扱えるようになります

pyproj 地図投影に必要な座標変換等ができるようになります

xarray 座標が定義されている多次元データを効率よく取り扱えるようになります

上記のライブラリは、インストーラではなく、Miniforge をインストールすることで利用可能となる、コマンドラインプログラム「conda」を使ってインストールします。

Windows では、スタートメニューから、**Miniforge3** > **Miniforge Prompt** を選択してください (図4)。暫くして、黒いウィンドウが現れるので、以下の文字列を書き込み [Enter] キーを押してください。これで上記6つのライブラリが一度にインストールされます。



図4. Windows のスタートメニューで「Miniforge3」「Miniforge Prompt」を選択し、Miniforge Prompt を起動する。

```
conda install -y -c conda-forge numpy scipy matplotlib netcdf4 pyproj  
xarray
```

これらのライブラリは、それぞれがさらに別なライブラリを必要としているので、このコマンドを実行するそれらもインストールされます。その数は数十個に上ります。インストールされたモジュールの一覧を確認するには、以下を入力します。

```
conda list
```

現在の環境と conda や Python のバージョン、参照するチャンネル等を確認するには、以下のように入力します。

```
conda info
```

5. Jupyter Lab のインストール

ライブラリのインストールと同様に、Miniforge Prompt で、以下に示すコマンドを実行してください。

```
conda install -y -c conda-forge jupyterlab [エンターキー]
```

6. 利用ツール、テストプログラム、チュートリアル の配置と設定

6-1. 作業用フォルダのダウンロードと配置

メッシュ農業気象データを配信サーバーから取得するための利用ツール、インストールが正しく行えたかを確認するためのテストプログラム、メッシュ農業気象データの基本的な利用方法を学習するためのチュートリアルが、メッシュ農業気象データホームページ (<https://amu.rd.naro.go.jp/>) に用意されているので、これをダウンロードします。ホームページの「→登録利用者専用ページ (ログインしないと進めません)」のリンクをクリックし、「Python Tips」「Tip 0.初めて Python を利用される方へ」の項目3、「この圧縮ファイル」をダウンロードし、展開してください。

展開したフォルダは、C:/users/*username*/ (エクスプローラーでは、PC > ローカルディスク(C:) > ユーザー > *username* と表示されます、ここで、*username* は皆さんのユーザー名です) の下に置いてください。

デスクトップがローカルディスクに作られている場合 (デスクトップに何かのファイルを置いたとき、それが C:/Users/*username*/Desktop に格納される設定の場合) 展開したフォルダは単純にデスクトップへ配置することをお勧めします。デスクトップが OneDrive に作られている場合 (デスクトップに何かのファイルを置いて、それは C:/Users/*username*/Desktop には置かれず、クラウド上の、OneDrive に格納される設定の場合) C:/Users/*username*/ の直下にダウンロードしたファイルを展開することをお勧めします。

6-2. データ配信サーバーへのアクセス認証情報の設定

メッシュ農業気象データのデータ配信サーバーへのアクセスを可能とするため、利用ツール「AMD_Tools3.py」に認証情報 (ID とパスワード) を設定します。

「メモ帳」等で「AMD_Tools3.py」を開き、49行目と50行目にご自身の ID とパスワードをそれぞれ書き込み、保存・終了します (図5)。半角英数で書き込んでください。パスワードについては、書き込む場所が2個用意されていますが、どちらか一方に書き込めば大丈夫です。

```
43 | 20131118 3次メッシュコードをキーとする表として出力する関数(PurCSV
44 | 20130314 地理情報を読み取る関数の追加↓
45 | 20121205 安定版初版↓
46 | Copyright (C) OHNO, Hiroyuki↓
47 | ""↓
48 | #_引用符の中を、通知された認証情報で書き換えてください。 _____↓
49 | USER=""利用者ID"↓
50 | PASSWORDS=["パスワード","パスワード"] #二つのうち、どちらかが正しい↓
51 | ↓
52 | #_プロキシサーバー経由で接続する方は下も設定してください。 _____↓
53 | # (使用しない場合はこのままにしてください) ↓
54 | PROXY_IP = "" #プロキシサーバーのIPアドレス(文字列で)↓
55 | PROXY_PORT = "" #使用するポート番号(文字列で)↓
56 | ↓
57 | # _____↓
58 | #////////// 以下には変更を加えないでください ////////////↓
```

図 5. 利用ツール「AMD_Tools3」への認証情報の記入
ファイルを「メモ帳」等で開き、文字列「利用者 ID」と
文字列「パスワード」を通知された認証情報に変更する。

7. Jupyter Lab の利用

7-1. Jupyter Lab の開始と終了

Jupyter Lab の開始は、OS のスタートメニューから「Miniforge Prompt」を選択し、コマンドプロンプトを開いて、以下に示すコマンドを実行してください。

```
jupyter lab [エンターキー]
```

すると、いつもお使いの Web ブラウザが起動し、図6のような画面が開きます。Jupyter Lab が開始しても、Miniforge Prompt の黒いウィンドウは閉じません。無理に終了すると、Jupyter Lab も終了してしまうので注意してください。

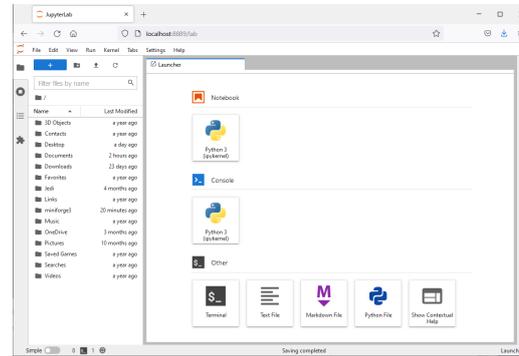


図 6. Jupyter Lab の画面
Jupyter Lab はブラウザ上で動作する

Jupyter Lab を終了するときは、メニューから File>Save Notebook としてデータを保存した後、メニューから File>Shut down を選択して終了してください。ブラウザが終了します。時に、ブラウザが閉じず、終了した旨を示すダイアログボックスが表示されるだけのこともあります。その際は強制的にブラウザを終了して構いません。その後に Miniforge Prompt の黒いウィンドウを終了します。

Jupyter Lab の起動と終了が普通のアプリとこのように異なるのは、Jupyter Lab が一般のアプリケーションソフトとは少し違う仕組みで動いているからです。お使いの PC の中にホームページサーバーが一時的に構築され、そこにブラウザがアクセスしているのです。このため、単にブラウザを閉じただけでは、ホームページサーバーが内部的に動き続けることとなります。

7-2. Jupyter Lab の操作

Jupyter Lab は、文字修飾等が可能なテキスト、実行可能なプログラムとその実行結果が共存するドキュメントを作成することができるソフトウェアで、Cell(セル)と呼ばれる矩形を単位とし、これを連ねることでドキュメントを形作ります。Cell は、文章を保持する Markdown Cell と、Python のスクリプトを保持する Code Cell の2種類からなります。Markdown Cell では Markdown 記法が利用でき、限定的ながら文字修飾などが行えます。Code Cell には Python プログラムを書き込むことができ、それは実行することが可能です。

Markdown Cell は、シングルクリックで選択でき、さらに、ダブルクリックすることで編集モードに移行し、Markdown 記法を用いた文章の編集が可能となります。この時、背景は灰色となります。編集後は、ツールバーの ▶ ボタンをクリックするか、[Ctrl] キーを押しながら [Enter] キーを押して確定します。

Code Cell は、配色がちょうど逆で、通常状態では灰色、クリックすると白色となり編集が可能となります。そして、ツールバーの ▶ ボタンをクリックするか、[Ctrl] キーを押しながら [Enter]

キーを押すことで実行されます。

実行後も、各変数の値はそのまま保持されているので、その下のセルを実行すれば、プログラムを継続することができます。この仕様は、設定→データ取得→描画のような、一本道のプログラムにおいては大変便利です。例えば、グラフの体裁を変えたいときは、そのセルだけを何度も手直し/実行すればよいからです。ただし、修正により読み込んだデータ自体に変更が加えられるような処理がなされる場合はデータ読込も再実行する必要があります。一からやり直したいときは、カーネルの再起動(Kernel > Restart the kernel)を実行します。

Jupyter Lab を一旦終了後、再び Notebook を開くと、前回の計算結果やグラフが表示されますが、変数の値については再現されていません。従って、プログラミングを再開するには、再開場所よりも上にある Code Cell すべてを最初から順に実行しなおす必要があります。

8. テストプログラムの実行

Jupyter Lab の画面左側のフォルダリスト(もし、表示されていない場合はウィンドウ左上方にある黒色のフォルダのアイコンをクリック)の上方、検索ボックスの下にあるフォルダのアイコンをまずクリックしてからその下のリスト順次クリックし、作業用フォルダ「PythonWorks」を見つけて、その中の「test.ipynb」をダブルクリックして開いてください。図7のような表示となります。

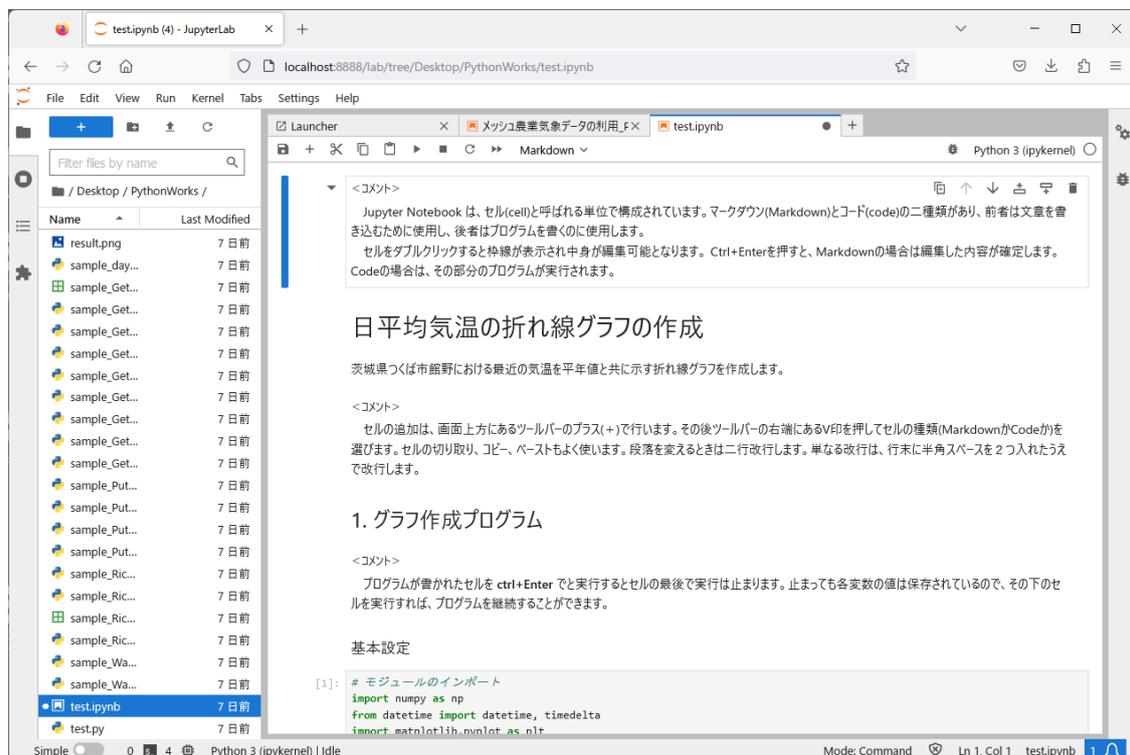


図7. テストプログラム「test.ipynb」を開いた画面
Jupyter Lab では、マークダウン文書中にプログラムを記述でき実行できる。

Notebook 内の記述を読んで内容を理解しつつ Code Cell を順次実行してください。最終的に図

8のようなグラフが表示されれば、利用環境は正しく構築されています。

確認出来たら、必要に応じ結果を保存 (Ctrl+S) し、Jupyter Lab を終了 (File > Shut Down、黒窓に exit) してください。これで環境構築作業は終了です。お疲れさまでした。

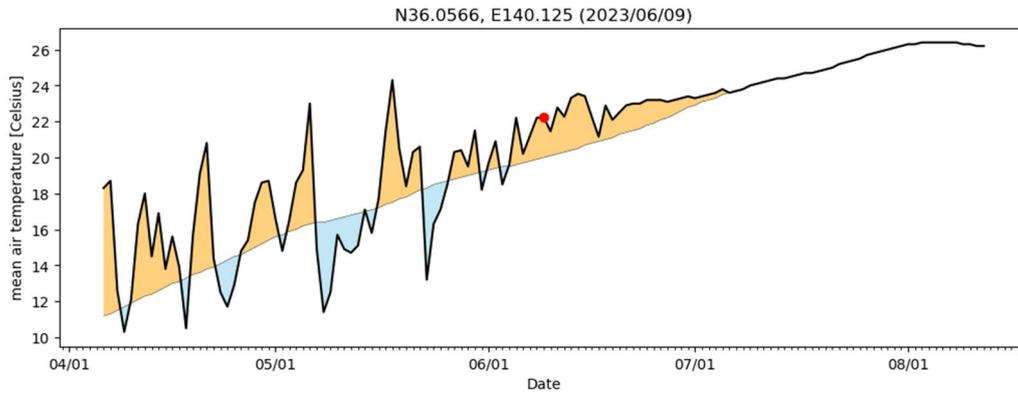


図 8. テストプログラム「test.ipynb」で作成されるグラフの例
特定地点における平均気温の推移が、プログラム実行日の前後 60 日間について示される。