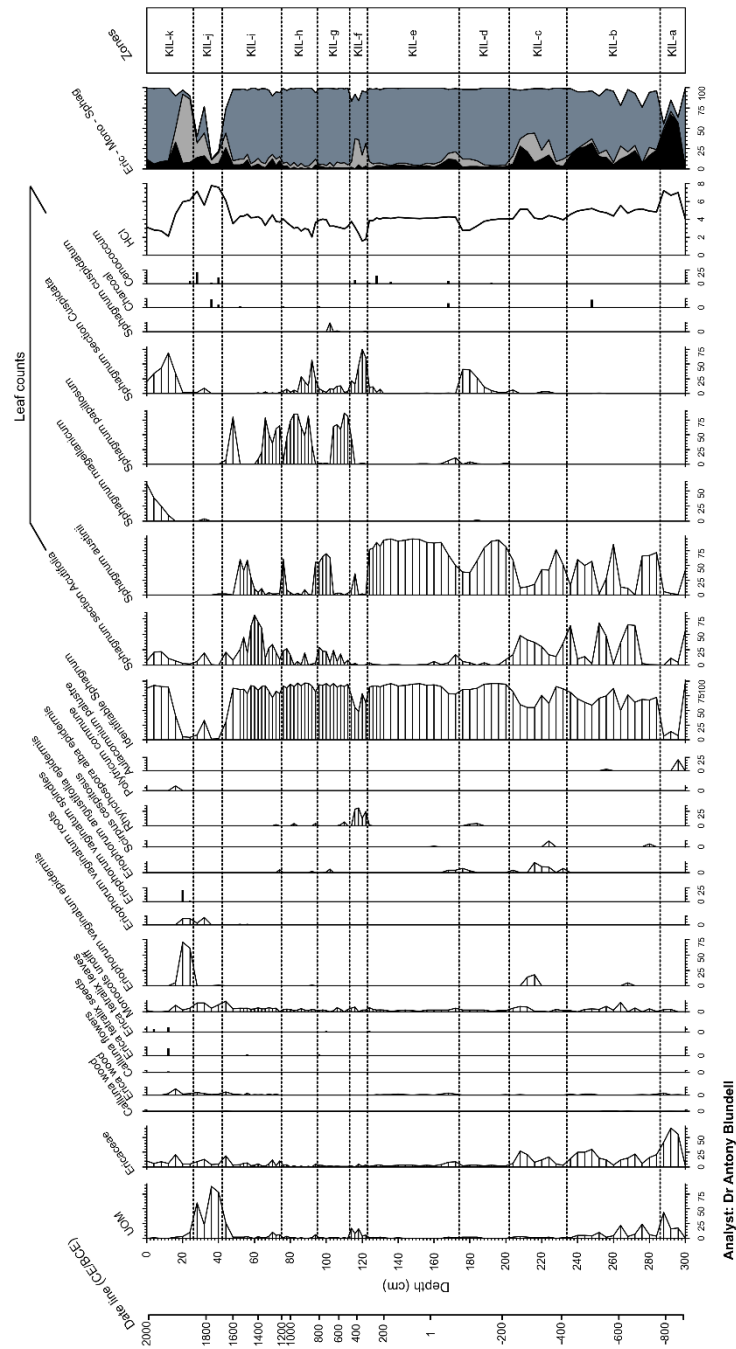


# Python Palaeo Plotting Program Manual

## Version 0.2



Written by Dr Antony Blundell

School of Geography, University of Leeds 2025

# 1 Introduction

P4 is a Python based program (Blundell, 2023) used to display palaeo style data such as pollen, testate amoebae, or macrofossils. The program enables the rapid production of high-quality figures. The program produces plots such as those in Figures 1.1 and 1.2. Additional annotations are possible within P4, but complex annotations are best added to the output via a separate drawing package outside of P4 such as Inkscape or CorelDraw. The program has not been designed to do complex statistics or produce percentage data it simply plots the data the user provides.

All features evident in the example plots (Figures 1.1 and 1.2) are editable in P4. One of the example figures (Figure 1.1) is based on Ardkill Bog (ARD) testate amoebae data from Blundell et al. (2008) and the other (Figure 1.2) is based on macrofossil data from Keighley Moor from Blundell and Holden (2015). This manual primarily uses the testate amoebae data from ARD (Blundell et al., 2008) as an example of how to use the program.

## 1.1 Files supplied with the download from Zenodo

P4 program is available from the repository Zenodo ([ANTB25/P4-PYTHON-PALAEO-PLOTTING-PROGRAM: P4-PYTHON-PALAEO-PLOTTING-PROGRAM](#)) and has been described and discussed in Blundell (2023). Be sure to download the latest version of the program. Versions are listed at the base of the web page. As well as the program (Python code) the download contains other files including a licence. To enable the user to familiarise themselves with how to use the program **Input** and **Parameter** files are provided for both ARD and KM example sites. Once the program has been setup users can run it immediately with these files to create and observe outputs. Template files are also supplied and can be filled with entries from scratch, or the ARD or KM files can be edited to suit the user's requirements for their own sites. A requirements file (requirements.txt) is also supplied and can aid in one of the ways to setup the program detailed in Section 1.3. All files supplied are listed below.

### PROGRAM FILE

---

**Program file** - P4\_v02.py

### ARD FILES

---

**Parameter file** – ARD\_Testate\_Parameter.csv

**Parameter file** – ARD\_Testate\_Parameter.xlsx (for demo only)

**Input file** – ARD\_Teststate\_Input.csv

**Input file** – ARD\_Teststate\_Input.xlsx (for demo only)

**Extra Input file** – ARD\_Teststate\_Extra Input.csv

**Extra Input file** – ARD\_Teststate\_Extra\_Input.xlsx (for demo only)

**Resultant figure (png) at 150 dpi** – ARD\_Teststate\_Plot\_150\_DPI.png

**Resultant figure (png) at 600 dpi** – ARD\_Teststate\_Plot\_600\_DPI.png

## KM FILES

---

**Parameter file** - KM\_Macro\_Parameter.csv

**Parameter file** - KM\_Macro\_Parameter.xlsx (for demo only)

**Input file** – KM\_Macro\_Input.csv

**Input file** – KM\_Macro\_Input.xlsx (for demo only)

**Resultant figure (png) at 150 dpi** – KM\_Macro\_Plot\_150DPI.png

**Resultant figure (png) at 600 dpi** - KM\_Macro\_Plot\_600\_DPI.png

**Parameter file** - KM\_Macro\_Parameter\_LOI.csv

**Input file** - KM\_Macro\_Input\_LOI.csv

**Extra Input file** - KM\_Macro\_Extra\_Input\_LOI.csv

**Resultant figure (png) at 150 dpi** - KM\_Macro\_Plot\_150\_DPI\_LOI.png

**Resultant figure (png) at 600 dpi** - KM\_Macro\_Plot\_600\_DPI\_LOI.png

## TEMPLATE FILES

---

**Parameter file** – Template\_Parameter.csv

**Parameter file** – Template\_Parameter.xlsx (for demo only)

**Input file** – Template\_Input.csv

**Input file** – Template\_Input.xlsx (for demo only)

**Extra Input file** – Template\_Extra Input.csv

**Extra Input file** – Template\_Extra Input.xlsx (for demo only)

## REQUIREMENTS FILE

---

requirements.txt

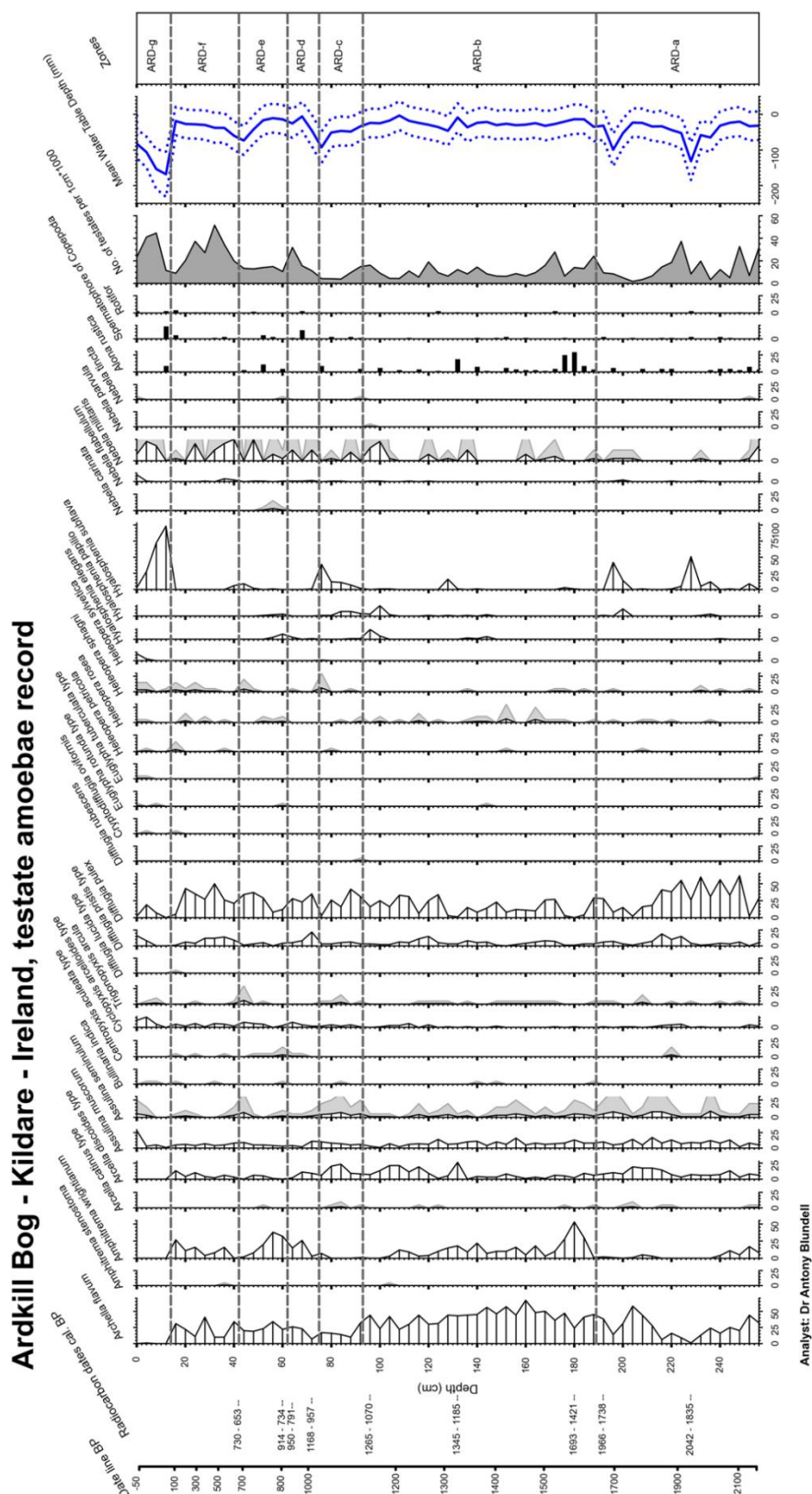


Figure 1.1 Example of P4 output for testate amoebae data from Ardkill Moss (ARD), Kildare, Ireland (modified from Blundell et al 2008).

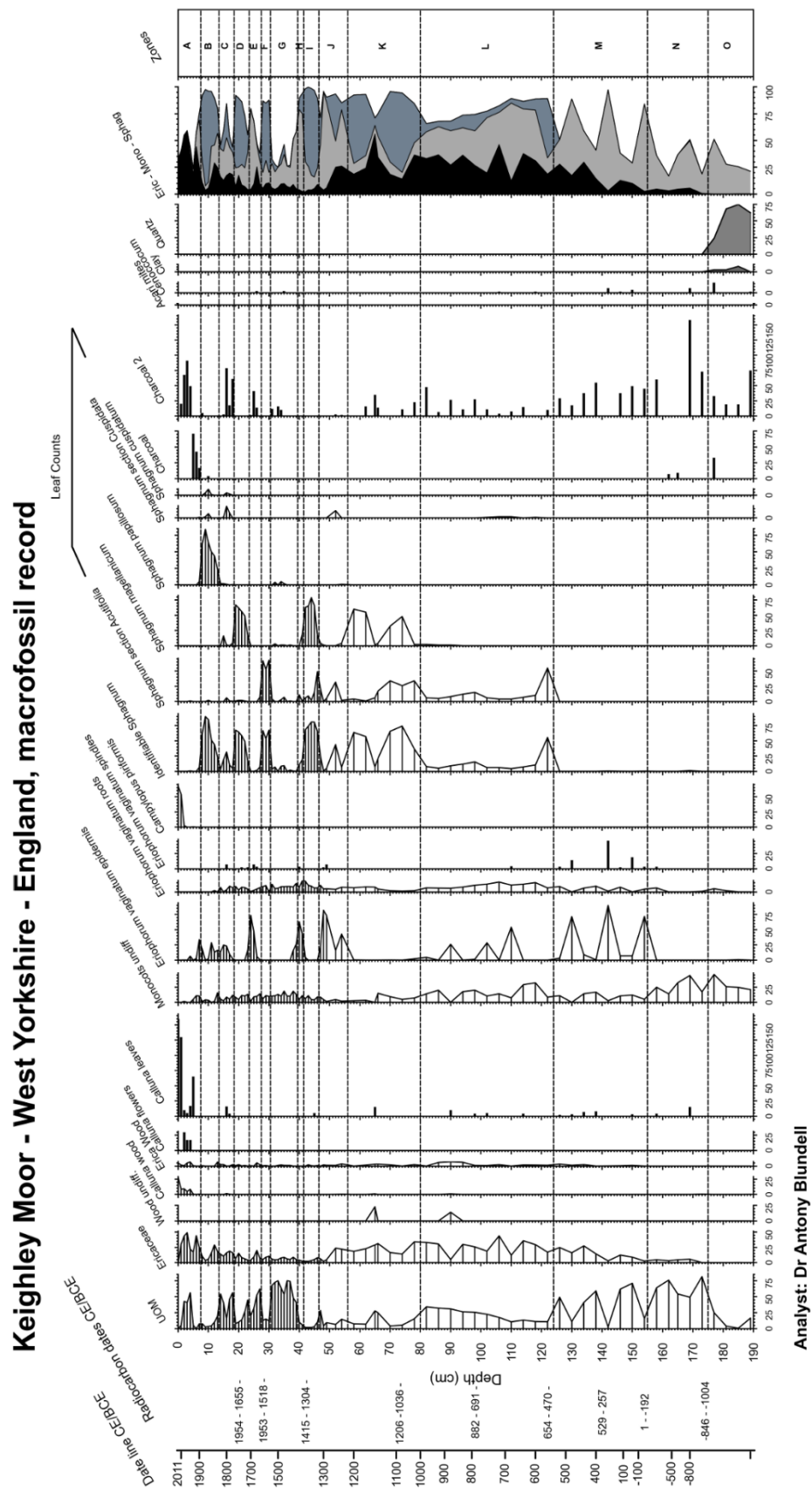


Figure 1.2. Example of P4 output for macrofossil data from Keighley Moor (KM) in West Yorkshire, England (modified from Blundell and Holden 2015).

## 1.2 The basic building blocks of the program

P4 is coded in Python (originally version 3.7.6, now updated for version 3.12.1) and is run from the command terminal. The code can be viewed in any text viewer or an Integrated Development Environment (IDE). The code has been tested successfully on computers running Windows 7, 10 and 11 using version coded in 3.7.6 and in 11 using the version coded in 3.12.1 here.

The code has also been tested using a computer running Mac and Linux systems. The code has been primarily tested with the author's data and some from other palaeoecologists and students. Despite extensive testing a bug may be encountered. If so these should be reported to the author.

The program is set up to receive two elements of information from the 'command line statement' that is used when running the program in the command or terminal window. If an experienced user wants to run P4 from within an IDE, the code will need to be altered slightly. This only concerns those with a degree of Python programming experience and/or a wish to tailor or improve the code. These users are likely to know how to run the code independent of the command line after examining the code. This involves simply removing the 'argparse section' (see comments in the code).

In this manual it is assumed that the reader has no knowledge of Python or the windows/Linux/Mac command line environment. The reader does not require any prior coding knowledge to use the program. Thoroughly reading this manual is all that is required.

## 1.3 Setting up the program for use

Running the code from a command terminal is how the program has been set up and is a simple operation once configured. **This requires minimal time and is a one-off operation.**

Firstly, the correct Python version must be installed. In this case that is **Python 3.12.1** from [Python Release Python 3.12.1 | Python.org](#) (Figure 1.3). When installing ensure the 'add Python to PATH box' is checked (Figure 1.4). The PATH can be added manually later if need be and guides as to how to do that are available on-line.

Python programs usually rely on packages to enable them to run and these need to be installed. There are numerous ways to store these packages and keep them separate

from other projects the user may have if they use Python. This is often achieved using an ‘environment’ and can be accomplished with or without **Anaconda**.

Files						
Version	Operating System	Description	MDS Sum	File Size	GPG	Sigstore
<a href="#">Gzipped source tarball</a>	Source release		51c5c22dbc698483734dff5c8028606	25.8 MB	<a href="#">SIG</a>	<a href="#">.sigstore</a>
<a href="#">XZ compressed source tarball</a>	Source release		50f827c800483776c8ef86e6a53831fa	19.6 MB	<a href="#">SIG</a>	<a href="#">.sigstore</a>
<a href="#">macOS 64-bit universal2 installer</a>	macOS	for macOS 10.9 and later	eae2d617cbd978a4a6c167924b287572	43.3 MB	<a href="#">SIG</a>	<a href="#">.sigstore</a>
<a href="#">Windows installer (64-bit)</a>	Windows	Recommended	3e3b6550e58772d324f7519bfa8066dc	25.4 MB	<a href="#">SIG</a>	<a href="#">.sigstore</a>
<a href="#">Windows installer (32-bit)</a>	Windows		37a89a0913888e6331ec279d68fea8e	24.1 MB	<a href="#">SIG</a>	<a href="#">.sigstore</a>
<a href="#">Windows installer (ARM64)</a>	Windows	Experimental	25fb741f175dc98d5630520f2df931ec	24.6 MB	<a href="#">SIG</a>	<a href="#">.sigstore</a>
<a href="#">Windows embeddable package (64-bit)</a>	Windows		019788d34af2c60a7be45bf8273e361f	10.5 MB	<a href="#">SIG</a>	<a href="#">.sigstore</a>
<a href="#">Windows embeddable package (32-bit)</a>	Windows		acc28815c74facc402469e917c8f8433	9.4 MB	<a href="#">SIG</a>	<a href="#">.sigstore</a>
<a href="#">Windows embeddable package (ARM64)</a>	Windows		18058aa3c8ccbf3e8fee53386d38c711	9.8 MB	<a href="#">SIG</a>	<a href="#">.sigstore</a>

Figure 1.3. The bottom of the python web page ([Python Release Python 3.12.1 | Python.org](#)) where Python is downloaded. Here most windows users would likely choose the Windows installer 64 bit option.

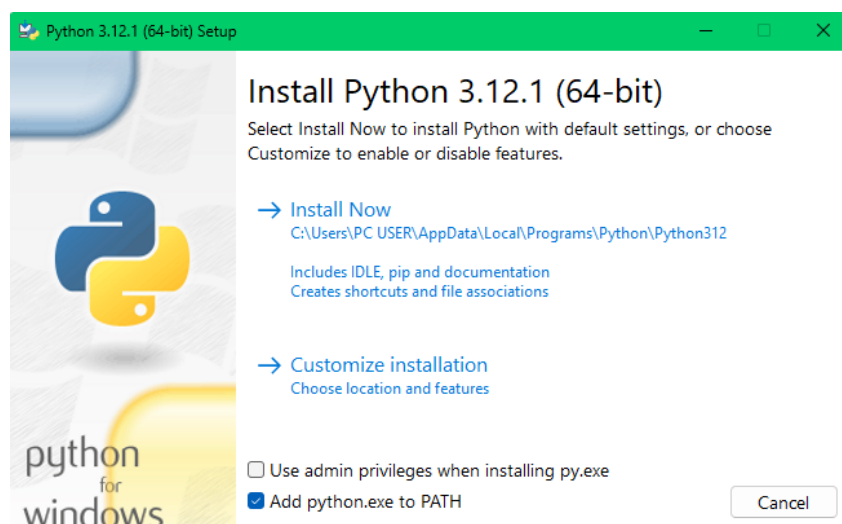


Figure 1.4. Adding Python to PATH when downloading Python. This figure is just for purposes of illustrating the need to check the Python to PATH box. The program is compatible with Python 3.12.1 so it is recommended to use 3.12.1.

**Anaconda** is a scientific application created for Python users that contains lots of packages built in and allows the creation of separate ‘conda environments’. Anaconda can be downloaded from [Download Anaconda Distribution | Anaconda](#). This is a large application in terms of storage size so the user may choose to use **Miniconda** (see link on same page as Anaconda) instead which is a substantially slimmed down version.

A specific environment can also be created outside of Anaconda/Miniconda and is the way the author deploys P4. Use of Anaconda/Miniconda is advantageous for those who



already use it, and it can help avoid institutional conflicts when trying to run the code on a work computer. Contact your IT department to resolve these issues if they occur.

Directions for program setup are weighted towards windows users as that is the system the author uses. However, instructions are available here for Mac and Linux. It is probable there are other ways deployment can be achieved with these systems so users can of course use other methods they are aware of.

### 1.3.1 Setting up an environment

The following instructions describe steps for setting up an **environment** on an external USB drive (F:) but it could be placed in any location. **Setup does require that you are connected to the internet** as files need to be downloaded.

#### *Without using Ananconda/Miniconda*

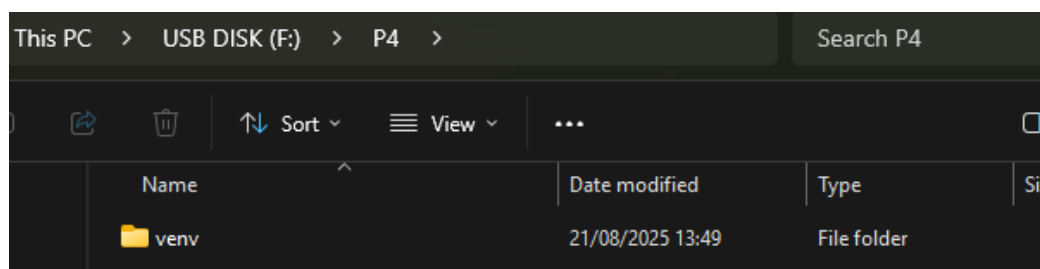
- 1) After installing Python on the computer open a windows terminal by typing **cmd** into the search window (or open the terminal app in Linux/Mac OS).
- 2) Install **virtualenv** which will aid the creation of a virtual environment. To do so type the following in the cmd window (terminal).  
**pip3 install virtualenv**
- 3) Once step 2 is completed navigate to the drive where the virtual environment is to be located. In this example a USB stick (F:) is used. To do this the user would type the drive name followed by a colon (F:) and press return.
- 4) If the user wants the virtual environment in a specific folder on the drive (P4 folder is used as an example here and throughout), they should navigate to that folder once in the drive using 'cd' such as cd F:\P4. The user can create the folder via windows explorer or by using the **mkdir** command followed by the folder name.
- 5) Once navigated to the desired folder enter the following statement to create the virtual environment. Ensure there is a single space where the arrows are located

**virtualenv** ↓ ↓ ↓ **-p 3.12.1 venv**

```
F:\P4>virtualenv -p 3.12.1 venv
created virtual environment CPython3.12.1.final.0-64 in 16991ms
creator CPython3Windows(dest=F:\P4\venv, clear=False, no_vcs_ignore=False, global=False)
seeder FromAppData(download=False, pip=bundle, via=copy, app_data_dir=C:\Users\minda\AppData\Local\pypa\virtualenv)
added seed packages: pip==25.1.1
activators BashActivator,BatchActivator,FishActivator,NushellActivator,PowerShellActivator,PythonActivator
```



- 6) This will only work if the version of Python stated in the command has been previously downloaded on to the user's machine. There is no problem having multiple versions of Python on a single computer.
- 7) Creating the virtual environment will take a few minutes. The user can substitute '**venv**' in the above statement with any name they wish for the environment.
- 8) To check the environment has been created successfully the user can use windows explorer or Linux/Mac OS equivalent to look in the working directory (in this case P4), the new environment should be visible. If the user enters the folder, they should observe content including directories called 'Lib' and 'Scripts'.



- 9) With the virtual environment successfully created the required packages can now be installed. The best way to achieve this is using the **requirements.txt** file provided in the download.
- 10) First activate the newly created virtual environment in the **cmd** window (or Linux/Mac OS terminal) using one of the following commands.

**venv\Scripts\activate.bat** (for Windows)

**source venv/bin/activate** (for MAC and Linux)

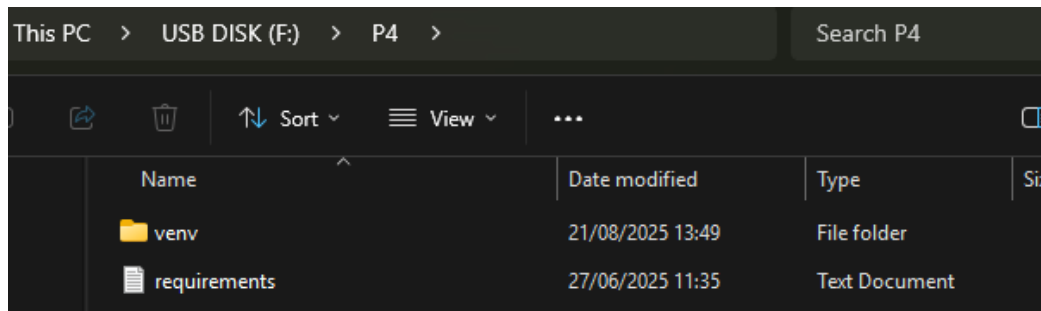
**echo \$VIRTUAL\_ENV** (to show which venv is activated (Mac / Linux))

```
F:\P4>venv\Scripts\activate.bat
(venv) F:\P4>
```

- 11) After the virtual environment is activated, the user can also check that the correct version of Python has been installed by typing the following command in the cmd (terminal) window.

**python --version**

- 12) Place a copy of the **requirements.txt** file (in download from Zenodo) in the working directory (here P4).



- 13) With the environment activated, in the cmd window (or Linux/Mac OS terminal), now use the following command to download all the packages the program uses.

**pip3 install -r requirements.txt**

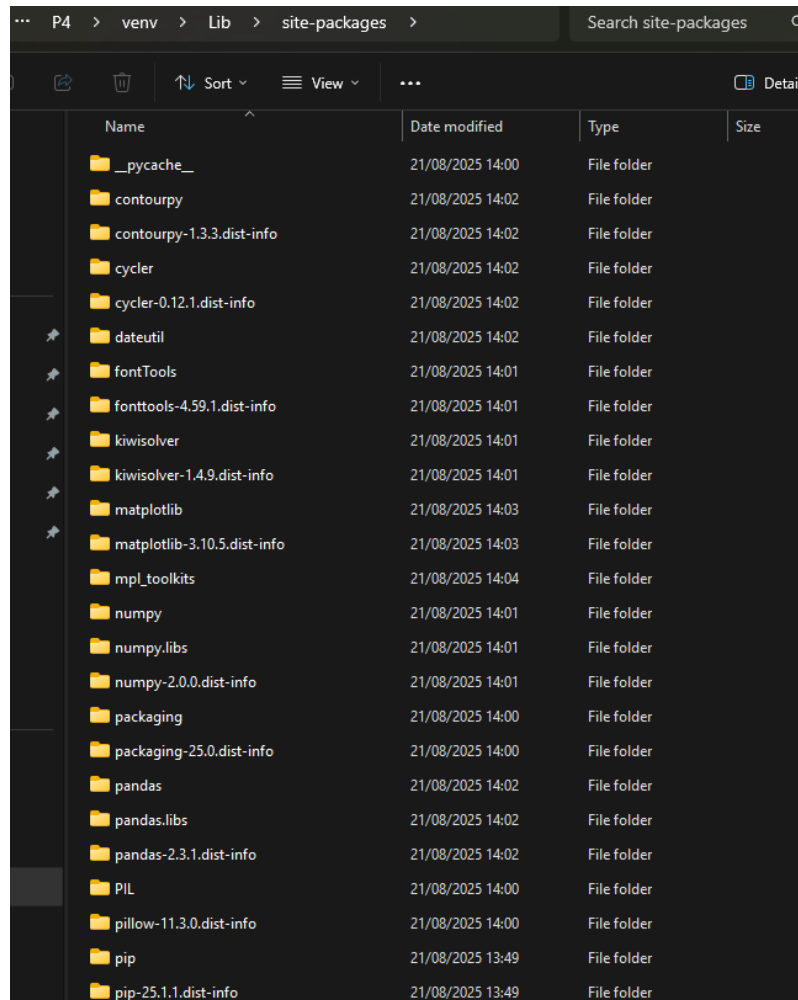
```
(venv) F:\P4>pip3 install -r requirements.txt|
```

- 14) If this is unsuccessful the following may be used  
**pip3 install --only-binary=:all: -r requirements.txt**

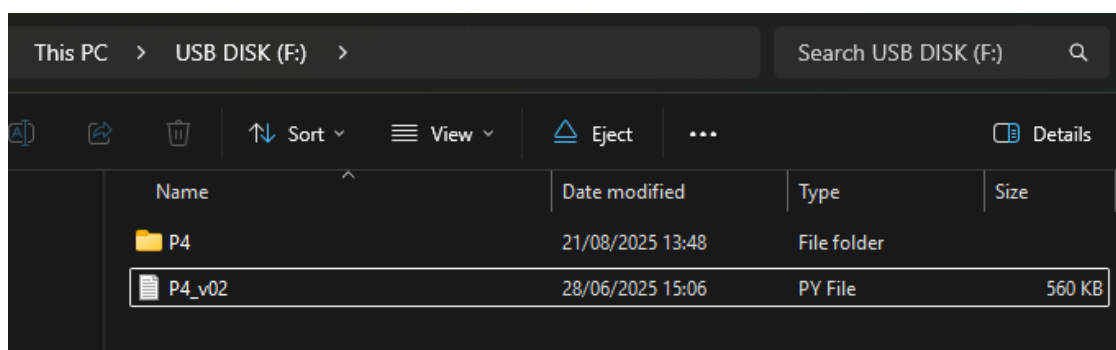
- 15) All packages should be downloaded after a few minutes. Packages and their required version numbers are as follows.

```
pandas==2.3.1  
numpy==2.0  
matplotlib==3.10.5
```

- 16) Packages can be added separately using the pip3 command, but this should be **unnecessary** if the **requirements.txt** file is used. Check dependencies are present via windows explorer or Linux/Mac OS equivalent by navigating to the site-packages folder within the Lib folder in the venv folder.



17) Now the environment has been created with packages installed, the P4 program (P4\_v02.py) file should be added in the folder where the virtual environment is located.

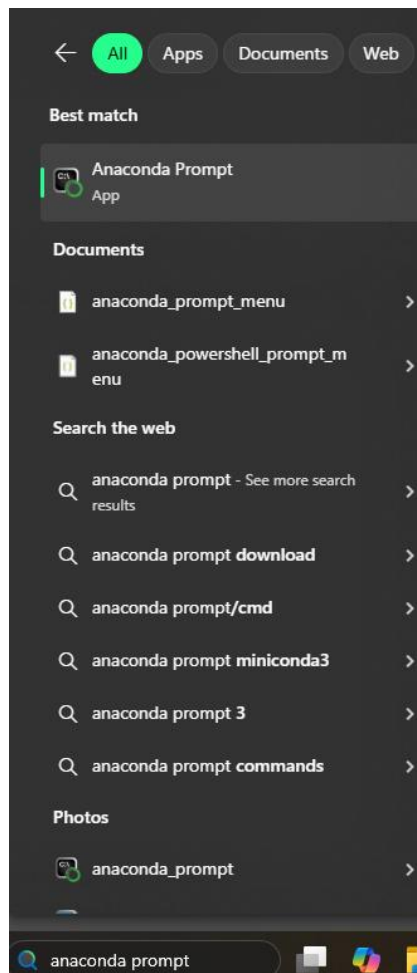


18) **Input** and **Parameter files** (Section 2) for each site can also be added in this folder. Although it is not necessary it can be useful to keep all files together.

## Using Anaconda/Miniconda

---

- 1) Install Anaconda or Miniconda [Download Anaconda Distribution | Anaconda](#).
- 2) Open the 'anaconda prompt' by typing 'anaconda prompt' into your search window.



- 3) Create the new conda environment and install the required packages with the following statement. Here the environment is named P4, but any name can be used. Ensure there are no spaces in the name.

```
conda create -n P4 python=3.12.1 matplotlib==3.10 pandas==2.3.1  
numpy==2.0
```

- 4) Upon returning to a new session the conda environment can be activated by the following statement for the P4 example here.

```
conda activate P4
```

### 1.3.2 Linux users

- 1) Linux users can use Anaconda as detailed above or can use pip to install required packages. Firstly, the user could make a new virtual environment using.

**python3-m venv P4**

- 2) Activate the environment with the following.

**../P4/bin/activate**

- 3) Install required packages with the following.

pip install -U pip

pip install numpy==2.0 pandas==2.3.1 matplotlib==3.10.5

### 1.1.3 Mac users

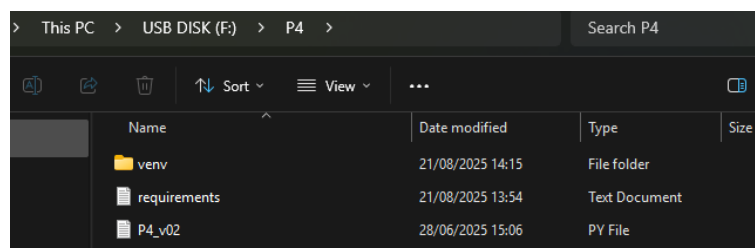
- 1) Install python3. For example, using a package manager like homebrew (<https://brew.sh>) or Macports (<https://www.macports.org/install.php>)
- 2) Install pip3 using method as in 1).
- 3) Using pip3, install numpy, matplotlib, pandas like the windows version above but using pip3.
- 4) The input for file location in the Parameter file see subsection 2.2 should be set to /Users/username/path to files.
- 5) Use the command in the terminal like that listed above for windows with a few exceptions to activate the program.

## 1.4 Running the program

Once steps detailed above have been taken, depending on your computer operating system, the user will be able to run the program.

### 1.4.1 Running the program using a virtual environment

- 1) If using a virtual environment as opposed to a conda environment the program file P4\_v02.py should be moved to the folder where the virtual environment is located.



- 2) The user can add in the Input and Parameter files (Section 2) for the site to be plotted in the folder where the P4\_v02.py file has been located. This is not necessary but can be useful to keep all files together and organised.
- 3) The user should navigate to the location where the 'P4' environment is via the cmd line.

- 4) Activate the environment.

**venv\Scripts\activate.bat** (for Windows)

**source venv/bin/activate** (for MAC and Linux)

**echo \$VIRTUAL\_ENV** (to show which venv is activated (Mac / Linux))

- 5) The program is then run via a single statement.

**python P4 v02.py --input F:\P4 ARD\_Testate\_Parameter.csv**

Here it can be seen that the command line statement is made up of the following.

- a) The word **python** followed by a **single space**.
- b) The name of the program **P4 v02.py** and a **further single space**.
- c) **Two hyphens** and the word **input** followed by a **single space**.

- d) The route for the Parameter file location (**F:\P4**) followed by a **single space**.
- e) The file name for the Parameter file (**ARD\_Testate\_Parameter.csv**) with the .csv extension included.

**\*\*Do not execute this command until Input file and Parameter file have been configured, correctly located and saved.\*\***

#### **1.4.2 Running the program using a conda environment**

- 1) When using a conda environment the P4\_v02.py file can be located in any named folder.
- 2) Navigate to the folder where the P4\_v02.py file is in the Anaconda command prompt window.
- 3) Activate the conda environment (P4 in this example).

**conda activate P4**

- 4) The program is then run using the same single statement as used for the virtual environment.

**python P4 v02.py --input F:\P4 ARD\_Testate\_Parameter.csv**

- 5) Linux users should activate the conda environment using the following statement.

**./P4/bin/activate**

- 6) Linux users should use the following statement to run the program.

**python P4 v02.py--input . ARD Testate Parameter.csv**

#### **1.4.3 Mac users**

After carrying out procedures in 1.1.4 Mac users should employ the following statement to run the program.

**python3 P4 v02.py--input . ARD Testate Parameter.csv**



## 1.5 User defined files

Before the program is run the user needs to configure the two or potentially three csv files the program needs to supply it with the palaeo data and user options to create the plot. These files are a combination of one or two **Input files** and a **Parameter file**. Information on how to configure these files is detailed in Section 2.

1) **Input file** (name must be supplied in the Parameter file). This provides the 'palaeo data' and the coding required for aesthetic attributes particular to each individual taxon to be displayed.

2) **Extra Input file** (name if used must be supplied in the Parameter file). This file is used if more than one element is required on the same individual plot. For example, if reconstructed testate water table depths were to be plotted, instead of a single entry of Mean Water Table depths alone the user may wish to plot error lines with for example a minimum and maximum error. Therefore, this plot would have three lines, if a line plot was chosen (Figure 1.1). It can also be used to plot auxiliary data that has a different sampling resolution (Section 2.3).

3) **Parameter file** The name of this file is used as part of the command line statement to make the program run (Section 1.4) and provides details used to customise the more general aesthetics of the plot.

## 1.6 Upon running P4

Running the program once the statement is entered takes only 5 seconds or less leading to resultant plots being saved in a format (png, svg or pdf) specified by the user within the Parameter file. The output is plotted in portrait and should be rotated for inspection. On windows systems simply double click on the file in explorer and the image can be rotated in 'Photos' or the default viewing program.

Entries in the input and parameter files are altered, saved and the program rerun, until the user is satisfied with the output. This of course is an iterative process but with practice the number of these steps will reduce dramatically. Although the number of possible attributes that can be adjusted in the Input and Parameter files may appear daunting, they are easily altered without the need of numerous windows, clicks or hunting for buttons etc.

Many of the inputs required in the Input files to control an individual plots' appearance can simply be 'copy and pasted' across different taxa. Completion of Input and Parameter files can be remarkably rapid especially if those provided for the example

figures at KM or ARD are used as a starting point and then modified. Once the user successfully completes a plot it is likely that many of the parameters and attributes will be similar for subsequent plots for other sites so the Parameter/Input files from the first plot can be edited.

The program has many inbuilt checks to detect user error and will provide 'helpful' messages to help resolve the issue in most cases.

## 2 Input and Parameter files

### 2.1 The Input file\

A blank template is provided (**Template Input.csv**) for the Input file and the Extra Input file if required. This can be filled with the user's data and specifications, or alternatively the user can modify one of the supplied Input files from the KM or ARD examples. These are called **ARD Testate Input.csv** and **KM Macro Input.csv** and demonstrate how the input csv is configured for these two sites using testate and macrofossil data respectively. Editing these files is best carried out in **Excel** or an Open Office equivalent such as **Calc**. These files can be edited in a text editor, but it is much less convenient. **Always** save edited files as the same format supplied, **a comma delimited csv** file, with the csv extension.

ARD testate Input file (ARD Testate Input.csv) is used here to show how to configure the Input file. A screen shot of the layout is shown in Figure 2.1. The figure has reduced number of taxa from the actual csv file and figure displayed in (Figure 1.1) to facilitate a clear display in this manual. The file not only contains the palaeo data (typically abundance of taxa) but also sample depths, an area for the codes the program employs for defining each graph style and colour and so forth, and some limited information related to configuring zones (if required). Close attention is required as to how the file is laid out and how users should modify it for their own data input. It is advised new users start with the ARD or KM example and modify it.

The first column is **Depth**. This heading 'Depth' **needs to be retained and not altered**. The name can be changed for display purposes for the plot in the Parameter file if required. The final column is **always 'Zones'** and must be retained whether zones are required or not. Columns in between are for the taxon abundances or whatever it is the user is plotting.

Names entered are the names that will be displayed in the figure for different taxa and the order they are in is the order in which they are plotted. The names supplied here are the names that will appear on the plot so editing them is carried out here in the Input file. If extra space is required for more taxa simply insert new column or columns between the Depth and Zones columns and add a new name in the corresponding cell in row 1. The program does have a **limit of 60 plots per figure**. This is rarely an issue as when 60 is approached the figure is getting hard to read and should be split into two figures for clarity.

The program does not carry out statistical analyses or create percentages, however up to two 'stack' plots can be produced and if specified percentage of the groups

nominated will be calculated (Section 2.1.1). Numbers in the Input file are the numbers that will be displayed. Testate amoebae data for example is often given as percentages so these would be calculated and then added to the Input file as percentages. The example file for ARD has a mix (Figure 2.1) of percentage and absolute count data.

In Figure 1.1 and 2.1 there are also 'No of testates per cm<sup>3</sup>' and 'Mean Water Table Depth' plots displayed, the latter from a transfer function. The former is an example of when the large numbers involved would lead to an individual plot much larger than those of the taxa data, if scaled proportionately. In this case data can be expressed as number of testates \*1000 to bring about a similar scale to the taxa. A percentage range is normally 0 - 100 but if count data is likely to far exceed this, for example over 300 it is recommended to create a number  $x*10$  or  $x*100$ . This is so the individual graphs can scale proportionately to each other without one graph occupying the entire plotting space.

However, in Figure 1.1 both 'No of testates per cm<sup>3</sup>' and 'Mean Water Table Depth' have been displayed using the **Non Standard Scaling (NSC)** capability of the program. This is designed for unusual non taxa like data that is to be plotted on unusual scales compared to those of the taxa plots. There are options to display **up to five** of these types of plots and may include measures such as pH, or conductivity, or anything operating on an unusual scale. How to nominate these and set accompanying scales is explained in Section 2.2.1.

### 2.1.1 Configuring the Input file

The Input file is set out in a specific way and rules must be adhered to for the program to run successfully. The first row contains the '**Depth**' label and then the names of taxa the user wants to plot. The final column has the label '**Zones**' (Figure 2.1). Often palaeo plots are split into zones of differing assemblages. This column is there to allow a 'Zone' column to be plotted at the end of the entire figure. The zone depths and names are provided in the Parameter file (Section 2.2). The final column must always be named 'Zones' so adding extra taxa columns is carried out by inserting columns between the Depth and Zones columns in the template files or by modifying space between these columns in the ARD or KM example files. The '**Zones**' heading text can be altered in the Parameter file to whatever name the user requires. Whether zones are displayed as part of the figure can also be specified in the Parameter file.

The **2nd to 39th row** of the Input file contains number codes to inform the program of the various taxa specific aesthetic attributes the user requires such as graph type and various colours. The number codes and their possible entries are listed below. Not all

entries are required depending on the graph type selected for the individual plot. It is advised to make all entries 0 (zero) to start with and then alter them from there. If many taxa require the same aesthetics, entries can be copied across (often most colours will be black for example or code 1). Graph types (line or scatter etc) that require an entry for the attributes listed below are listed next to the attribute title. For example, an entry for 'Depth Bar Colour' is required for Graph types 1, 2, 5.

**Please see areas of the Input file that should not be altered in red (Figure 2.1). Areas in green are there to be edited. If no entry is required leave as 0. The xlsx versions of the various Input files supplied show colour coding of grey, areas to not be altered and green where entries can be altered.**

**\*\*Important note about Zones column in Input file\*\***

The user will notice a data entry for Zones of 100 in the example Input csv files. A single entry of 100 with the rest being zeros. This gives the zone part of the plot a 'width' value as the program code uses the maximum and range of values for each taxon to define and scale the plot width. So, although Zones has no value data as such, to gain a column display as in the example figure 1.1 it requires a value to provide a column. So, Zones column values beyond row 39 should contain all zeros apart from a single value to govern its width. This width is based on the data units of the main palaeo proxy data. In the Ardkill Input file example 100 is used as this gives a column that would be the maximum of the percentage data supplied. However, a single entry of 200 would make it twice as wide and 50, half as wide.

Entries for **rows 1 to 39 for Zones will all be zero** except potentially those on **rows 27-30** (X Tick Major Colour to Y Tick Minor Colour). If X ticks depth axis only\*\* entry in the Parameter file (Section 2.2) is 'off' and therefore ticks are requested on all plot X axes, then these colour entries can be added for the Zones column.

**These are the only entries that can be used for the Zones column in the plot aesthetic rows (1 to 39) in the Input and Extra Input files.**

## **Attributes**

These apply to the graph types stated. If no graph type is stated they apply to all graph types.

### **Graph Type - Required for all taxa/data to be plotted**

---

- 1 = Bar plot
- 2 = Line plot with depth bars and fill colour
- 3 = Line plot
- 4 = Shadow plot (line plot with area below filled in)
- 5 = Line and marker plot
- 6 = Scatter plot (markers only)
- 7 = Option for creating stack plots (see stack plot 1 and 2 below)

### **Depth Bar Colour – Graph type 1 and 2**

---

Colour of the bars in graph types 1 and 2. Colour codes are listed once here but the same colour codes are used throughout. Please note American spelling is used here for many instructions within the program such as '**gray**' for **grey**. Colour codes are as follows.

- 1 = black
- 2 = gray
- 3 = dim gray (darker than dark gray)
- 4 = dark grey
- 5 = slate gray
- 6 = light gray
- 7 = red
- 8 = dark red
- 9 = orange red
- 10 = coral
- 11 = green
- 12 = dark green
- 13 = olive
- 14 = light green
- 15 = blue
- 16 = dark blue
- 17 = light blue
- 18 = cyan
- 19 = yellow
- 20 = brown
- 21 = magenta

22 = orange

23 = white

---

**Depth Bar width – Graph type 1**

Bar width values of 0.5 to 2 are generally appropriate.

---

**Depth Bar width - Graph type 2**

Bar width values of 0.5 to 2 are generally appropriate.

---

**Line Style - Graph type 2, 3, 4 and 5**

Defines line style for plots with lines (not bars). Line style codes are listed here once and are the same for all line styles throughout the program.

1 = solid

2 = dotted

3 = dashed

4 = dashdot

---

**Line Colour – Graph type 2, 3, 4 and 5**

Define line colour. See colour codes above.

---

**Fill Colour – Graph type 2 and 4**

Define fill colour. See colour codes above.

---

**Fill Colour Transparency – Graph type 2 and 4**

Define fill transparency from 0 to 1. 0 is totally transparent and 1 no transparency.

---

**Line width – Graph type 2, 3, 4 and 5**

Define line width. Between 0.5 and 2 is typical.

---

**Marker Type – Graph type 5 and 6**

Define marker type. Codes are as follows.

1 = circle

2 = cross

3 = upward facing triangle

4 = downward facing triangle

5 = diamond

6 = asterisk



---

**Marker Size- Graph type 5 and 6**

Marker size. Typically, between 1- 5 is large enough.

---

**Marker Face Colour- Graph type 5 and 6**

Central colour of marker. See codes above.

---

**Marker Edge Colour- Graph type 5 and 6**

Outside colour of marker. See colour codes above.

---

**Marker Edge Width- Graph type 5 and 6**

Width of edge line round marker. Typically, 0.5 to 1 is sufficient.

---

**Taxon Title Text Colour**

Colour for individual taxon title text. See colour codes.

---

**Taxon Title Text Bold**

Taxon title text bold or not bold, 1 or 0.

---

**Plot Vertical Spine Width**

Width of main spine on individual plot. Typically, 0.5 to 2 is sufficient.

---

**Plot Vertical Spine Style**

Style of line used for spine for individual plot. See line style codes above.

---

**Plot Vertical Spine Colour**

Colour of line used for spine for individual plot. See colour codes above

---

**Plot Left Spine Width, Plot Left Spine Style, Plot Left Spine Colour, Plot Right Spine Width, Plot Right Spine Style, Plot Right Spine Colour**

These are all as for vertical spine above.

---

**X Tick Major Colour**

Colour of major ticks on what is typically the depth axis. See colour codes above. Unlike all other attributes, this, X Tick Minor Colour, Y Tick Major Colour and Y Tick Minor Colour are available for the zones plot column if it has been specified in the Parameter file. In Figure 2.1 these entries have been altered to values of 1, indicating black, for the input example for ARD Testate Input.csv.

---

**X Tick Minor Colour**

Colour of minor ticks on what is typically the depth axis. See colour codes above.

---

**Y Tick Major Colour**

Colour of major ticks on what is typically the taxon abundance axis. See colour codes above.

---

**Y Tick Minor Colour**

Colour of minor ticks on what is typically the taxon abundance axis. See colour codes above.

---

**Exaggeration multiplier**

If exaggeration is required on the taxon plot, add the multiplier required. To exaggerate the plot by 5 times, add an entry of 5. Leave as 0 if no exaggeration is required. See exaggeration applied on the ARD plot in Figure 1.1.

---

**Exaggeration graph type**

Exaggeration is **only available on graph types 2, 3 and 4** and the **graph type used for the exaggeration** can be either **graph type 3** (line) or **graph type 4** (line with fill below or just fill). If not required leave as 0.

---

**Exaggeration fill colour**

Colour used for the fill of the exaggeration if graph type 4 has been specified as the exaggeration graph type. See colour codes above or leave as 0.

---

**Exaggeration fill transparency**

Define fill transparency from 0 to 1. 0 is totally transparent and 1 is no transparency. If not required leave as 0.

---

**Exaggeration line colour**

Define line colour for exaggeration. See colour codes above. If not required leave as 0.

---

**Exaggeration line width**

Define exaggeration line width. Between 0.5 and 2 is typical. If not required leave as 0.

---

**Exaggeration line style**

Define exaggeration line style. See line style codes above. Leave as 0 if not required.

### Stack plot 1

---

Here groups are nominated to be included in a stack plot. The user should code the taxa to be grouped together using numbers. For an example see entries for KM in the KM Macro Input.csv file. Taxa within the Ericaceae, Monocots and Sphagnum groups have been entered as 1, 2 and 3. These are displayed as a stack plot (Figure 1.2). These groups can also be made into percentages based on the total of all the groups provided if required. This needs to be specified in the Parameter file (Section 2.2).

In the Input file a new column must be inserted for the stack plot and called whatever name the user wishes and attributes in rows 2 to 27 filled in correctly. The spaces where data would normally be filled in should be as for the Zones column noted above, with all zeros except for a single entry to govern the width of the plot. See the KM Parameter file provided for an example. **The number of groups is presently limited to five.** The numbers 1 to 5 for groupings need to be employed. So, if there are 3 groupings use numbers 1 to 3 to group them as in the example KM file.

### Stack plot 2

---

Here there is the option to have a second stack summary stack plot. See instructions for stack plot 1 above.

Once data and coding has been added the Input file should be named and saved (as a comma delimited csv file) in a folder. P4 does try to give the user warnings if the data added exhibits errors but not every eventuality will be accounted for in the error checking procedures. If a problem occurs without a useful error message re-trace the preparation steps. If there is an error that has not been picked up by the program always keep an original template (the one provided with the program) so inputs can be restarted from a robust base. Data entry for the Input and Parameter files is relatively simple so few issues should be encountered once the user has read this manual.

### *The Extra Input file*

---

Graph types 3, 5 and 6 (line, line and marker, and scatter) can have more than one plot as demonstrated in Figure 1.1 for the Mean Water Table Depth (mm) plot. To do this a further file is required that is essentially a repeat of the Input file. For the ARD data ARD Testate Extra Input.csv is used (Figure 2.2). For extra plots to be drawn the same taxon name (contained in main Input file) must be contained within the Extra file's taxa name. For example, in the ARD Testate Extra Input.csv file, the additional two lines to be

plotted on the Mean Water Table Depth (mm) plot are called Mean Water Table Depth (mm) 1 and Mean Water Table Depth (mm) 2. The ARD Testate Extra Input.csv file contains the same first two columns as the main Input file followed by the additional data for the extra plots.

The attributes can be altered as for the main Input file. This additional csv Input file can be called anything (here it is called ARD Testate Extra Input.csv) but the name needs to be nominated in the Parameter file. If this functionality is not required, the entry in the Parameter file 'Extra Input file name' (Section 2.2) should be 'none'. Otherwise add the name such as ARD Testate Extra Input.csv. This additional file should be saved in the same location as the main Input file that is nominated in the Parameter file (Section 2.2). The Extra Input file can have an additional use to plot data that is not sampled at the same resolution as the main proxy being displayed (detailed in section 2.3).



	A	B	C	D	E
1		Depth	Mean Water Table Depth (mm) 1	Mean Water Table Depth (mm) 2	Zones
2	Graph Type	0	3	3	0
3	Depth Bar Colour - Graphs 1, 2	0	0	0	0
4	Depth Bar width - Graphs 1	0	0	0	0
5	Depth Bar width - Graphs 2	0	0	0	0
6	Line Style - Graphs 2, 3, 4, 5	0	2	2	0
7	Line Colour - Graphs 2, 3, 4, 5	0	15	15	0
8	Fill Colour- Graphs 2,4	0	0	0	0
9	Fill Colour Transparency - Graphs 2,4	0	0	0	0
10	Line width - Graphs 2, 3, 4, 5	0	1	1	0
11	Marker Type - Graphs 5, 6	0	0	0	0
12	Marker Size - Graphs 5, 6	0	0	0	0
13	Marker Face Colour - Graphs 5, 6	0	0	0	0
14	Marker Edge Colour - Graphs 5, 6	0	0	0	0
15	Marker Edge Width - Graphs 5, 6	0	0	0	0
16	Taxon Title Text Colour	0	1	1	0
17	Taxon Title Text Bold	0	0	0	0
18	Plot Vertical Spine Width	0	0.5	0.5	0
19	Plot Vertical Spine Style	0	1	1	0
20	Plot Vertical Spine Colour	0	1	1	0
21	Plot Left Spine Width	0	0.5	0.5	0
22	Plot Left Spine Style	0	1	1	0
23	Plot Left Spine Colour	0	1	1	0
24	Plot Right Spine Width	0	0.5	0.5	0
25	Plot Right Spine Style	0	1	1	0
26	Plot Right Spine Colour	0	1	1	0
27	X Tick Major Colour	0	1	1	1
28	X Tick Minor Colour	0	1	1	1
29	Y Tick Major Colour	0	1	1	1
30	Y Tick Minor Colour	0	1	1	1
31	Exaggeration multiplier	0	0	0	0
32	Exaggeration graph type - Graph 3 or 4	0	0	0	0
33	Exaggeration fill colour - Ex style Graph 4	0	0	0	0
34	Exaggeration fill transparency - Ex style Graphs 4	0	0	0	0
35	Exaggeration line colour - Ex style Graphs 3,4	0	0	0	0
36	Exaggeration line width - Ex style Graphs 3,4	0	0	0	0
37	Exaggeration line style - Ex style Graphs 3,4	0	0	0	0
38	Stack plot 1	0	0	0	0
39	Stack plot 2	0	0	0	0
40	DATA	0	-123.2	-41.76	100
41	DATA	4	-149.4	-62.84	0
42	DATA	8	-209.48	-97.7	0
43	DATA	12	-231.78	-104.7	0
44	DATA	16	-58.79	21.53	0
45	DATA	20	-67.4	13.52	0
46	DATA	24	-68.05	11.97	0
47	DATA	28	-70.99	11.45	0
48	DATA	32	-79.77	4.33	0
49	DATA	36	-79.3	3.36	0

Figure 2.2. View of EXTRA.csv file allowing extra lines to be plotted on Mean water table depth (mm) plot

## 2.2 The Parameter file

This file contains **general non-taxon specific user defined parameters** that can be altered to change the appearance of the plot. Part of the Parameter file for the ARD example data used here (ARD Testate Parameter.csv) is shown in Figure 2.3. Column A (first column) is employed by the program to locate the different parameters and Column C (third column) provides description and instruction to the user for each parameter. **Only Column B should be edited by the user.** Do not change column A or C or the program may not function correctly.

The xlsx version of Parameter files supplied show areas of grey that are not to be altered and areas of green where entries can be changed. These files are for illustration only as all Input and Parameter files must be saved as csv files.

Columns can be more easily read when opened in Excel or Open Office Calc if the column widths are expanded, and column text is centred for Column B. The file can of course be edited in Notepad, WordPad, or any text editor but it is easier to follow in either Excel or the Open Office equivalent.

Parameters are arranged in **15 logical groups** separated by empty rows. Parameters followed by double **\*\* are essential** and must have entries or the program will ask for them and close. Notes in column C are relatively self-explanatory but further explanation is provided below. The Parameter file can be saved with any name if it is a comma delimited csv file with the .csv extension and can be placed in any location. The Parameter file name and the route to it are required in the command statement in the windows command window to run the program (Section1.4)

### 2.2.1 Parameter groups

#### Group 1 Files/Directories

##### Directory\*\*

Provide the location route of the Input file containing the main palaeo data. See the example in the example Parameter files for ARD/KM here. Make sure the levels of the route are separated by a backslash.

F:\P4

(possible example based on setup example)

**\*\*This is where the main and extra (if required) Input files with palaeo data are to be located and where the completed plot will be sent.\*\***



Only edit this area of the parameter file. Explanations as to the meaning of each parameter are given in column C. Do not alter anything in column A or C only Column B is to be altered with user preferences. There are many more parameters, this is just the upper screen shot of the ARD Parameter file.

A		B	C
PARAMETERS	ENTRY	NOTES	
GROUP 1			
FILES/DIRECTORIES			
Directory**	C:\Users		
Input file name**	ARD_Teststate_Input.csv	Provide location of directory where input data is and where output will go separated by \	
Extra input file name**	ARD_Teststate_Extra_Input.csv	Provide the 'extra' input file name (needs to be a comma delimited csv file or.xlsx. include name and the .csv or .xlsx extension here)	
Output file name**	ARD_Teststate_Plot_600_DPI	Provide a filename for the output plot. Do not add extension like the name.	
Save as**	png	Provide instruction to save as either .svg, .pdf or .png using the terms .svg, .pdf or .png. Specify a single format or two or three separated by comma	
Png dpi	600	Provide dots per inch for png output if png has been selected. A number can be left here whether png has been selected or not	
GROUP 2			
OVERALL SIZE			
Overall figure size X**	12	Specify overall size of entire plot X dimension in centimetres (X is the depth axis dimension as plot is plotted in portrait. see manual for illustration)	
Overall figure size Y**	25	Specify overall size of entire plot Y dimension in centimetres (Y is the plot size across with all graphs, see manual for illustration). Size dictated by overall figure size X	
GROUP 3			
OVERALL FONT			
Font style**	Arial	Provide font style of entire plot from either Arial, Calibri, DejaVu sans or Times New Roman.	
GROUP 4			
OVERALL TITLE			
Overall title text	Ardkull Bog - Kildare - Ireland, teststate amoebae record	Provide title text if an overall title is required. If this overruns in to plots can use overall dimensions to adjust or reduce font size	
Overall title text on/off**	on	Specify if title text is to be visible or not. Enter word "on" or "off".	
Overall title X position	0	X position of start of title. In page units not data units. 0 is far left of visible page. 1 far right. Remember plots are in portrait so X is the depth axis	
Overall title Y position	0.1	Y position of start of title. In page units not data units. 0 is far left of visible page. 1 far right. Remember plots are in portrait so Y is the axes of the plot	
Overall title font size	12	Specify font size for title	
Overall title rotation	90	Specify rotation of title in degrees ( 90 degrees if as in example plots in manual)	
Overall title gap	0.2	Adjust gap between title and plots (most likely range is between 0 - 0.5)	
Overall title text colour	1	Specify title text colour (1 = black, 2 = gray, 3 = dimgray, 4 = darkgray, 5 = slategray, 6 = light gray, 7 = red, 8 = darkred, 9 = orange, 10 = coral, 11 = blue)	
Overall title text bold on/off	on	Specify if title text is to be bold or not. Enter word "on" or "off".	

Figure 2.3. Screen shot of the Parameter file showing some of the parameter names used by the program in column A, the parts the user should change in column B and explanations/directions for the individual parameters in column C. Parameters are grouped into 15 logical groupings. Only alter entries in column B.

**Input file name\*\***

Add the Input file name with .csv extension.

**Extra Input file name\*\***

Add the additional file name with .csv extension used for supplying data for multiple lines on plots. If no additional extra Input file is required 'none' should be entered.

**Output file name\*\***

Add file name for the output image. Do not supply extension as that is dictated by the 'Save as' option detailed below. The output is saved in the location specified in the Directory location, see above.

**Save as\*\***

Select the required file type for output figure. Can be **png**, **pdf** or **svg**. Once the user is satisfied with the plot all available formats can be requested if required at once by separating each entry with a comma (no spaces) as such **png,svg,pdf** and all will be saved with the same output name. It is not recommended to do this for early iterations as the user builds the plot as the extra saving procedures may make the run time longer in duration.

**Png dpi**

State the dpi resolution if png is selected above. For example, 150 for low, 300 for medium, and 600 or >600 for high resolution. This number can remain whether using png or not.

**Group 2 Overall size**

---

**Overall figure size X\*\***

Overall X size of figure in centimetres. The plot is created in portrait, so X here refers to the depth axis side of the figure dimension (figure 8).

**Overall figure size Y\*\***

Overall Y size of figure in centimetres. The plot is formed in portrait, so the Y here refers to the length of the figure from the initial depth axis to the zone part of the figure if specified (Figure 2.4).

## Group 3 Overall font

### Font style\*\*

Choose the font style for the entire figure. Possible options are **arial**, **calibri**, **dejavu sans** and **times new roman**. Use lower case as here and do not leave spaces after the entry.

## Group 4 Overall title

### Overall title text

Supply the text required for the overall title

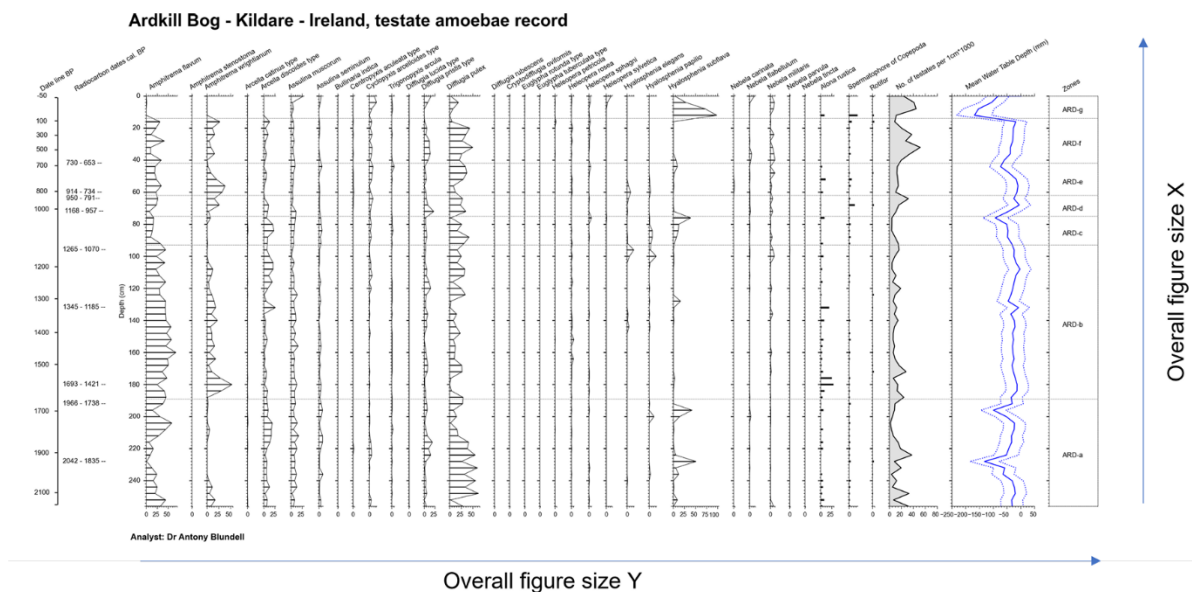


Figure 2.4. Arrows showing overall size in X and Y.

### Overall title text on/off\*\*

Whether a title is displayed or not. An entry of '**on**' or '**off**' is required.

### Overall title X position

Position of start of title in '**page units**' not 'data units'. 0 is far left on visible page, 1 far right. See example Parameter files. Remember figures are plotted in portrait.

### Overall title Y position

Position of start of title in page not data units. 0 is lowest position on visible page, 1 highest. Remember figures are plotted in portrait.

### Overall title font size

Font size for overall title text.

**Overall title rotation**

Rotation in degrees for overall title text. Usually, would be 90 degrees.

**Overall title gap**

Adjusts gap between main title (if specified) and the taxa plots. Adjust between 0 and 0.5.

**Overall title text colour**

Supply the text colour required for the overall title. See colour codes.

**Overall title text bold on/off**

Title text bold or not bold. An entry of '**on**' or '**off**' is required.

**Group 5 Footer title**

---

**Footer text**

Supply the text required for the footer title.

**Footer text on/off\*\***

Whether a footer title is displayed or not an entry of '**off**' will stop the footer title being plotted, '**on**' will permit display.

**Footer X position**

Position of start of footer in '**page units**' not data units. 0 is far left on visible page, 1 far right. Remember figures are plotted in portrait.

**Footer Y position**

Position of the start of the footer. Number is in **page** not data units. 0 is lowest position on visible page and 1 is the highest on the visible page. Remember figures are plotted in portrait.

**Footer font size**

Font size for footer title text.

**Footer rotation**

Rotation in degrees for footer title text. Usually 90 degrees.

**Footer text colour**

Supply the text colour required for the footer title. See colour codes.

**Footer text bold on/off**

Footer text bold or not bold. An entry of '**on**' or '**off**' is required.

## Group 6 X title

---

### **X title text\*\***

Enter text for the X title. This, as the plot is produced in portrait, is the 'depth' axis in the example plot.

### **X title font size\*\***

Font size for X title text.

### **X title rotation\*\***

Rotation in degrees for X title text.

### **X title text colour\*\***

Colour of X title text.

### **X title text bold on/off\*\***

X title text bold or not bold. An entry of '**on**' or '**off**' is required.

## Group 7 X limits, ticks and labels

---

### **X limit top\*\***

Upper limit of X axis (depth) often 0 cm indicating the surface.

### **X limit base\*\***

Lower limit of X axis (depth), deepest depth.

### **X major interval\*\***

Major X tick interval unit (the depth axis, plots are plotted in portrait).

### **X minor interval**

Minor X tick interval unit (the depth axis, plots are plotted in portrait).

### **X minor ticks on/off\*\***

If 'on' minor ticks for X (depth) axis are visible. An entry of '**on**' or '**off**' is required.

### **X ticks depth axis only\*\***

If this option is 'off' depth major and/or minor marks (depending on whether minor have been selected above) are displayed on all axes not just the first plot. An entry of '**on**' or '**off**' is required.

### **X major tick length\*\***

Length of major tick marks.

**X minor tick length**

Length of minor tick marks.

**X major tick width**

Width of major tick marks.

**X minor tick width**

Width of minor tick marks.

**X label font size**

Font size of X axis tick labels.

**X label rotation\*\***

Rotation of X label (normally 90 degrees as plot made in portrait). The X labels will typically be numbers relating to for example cm depth.

**X label colour\*\***

Colour of X labels. See colour codes.

**Group 8 Y Title**

---

**Y title font size\*\***

Font size for Y title text.

**Y title rotation\*\***

Rotation of Y titles in degrees. The Y titles are the names of each taxon.

**Group 9 Y ticks and labels**

---

**Y major tick interval\*\***

Y axis major tick interval (interval applied to all taxon graphs, unless is Non Standard Scaling).

**Y minor tick interval**

Y axis minor tick interval (interval applied to all taxon graphs, unless is Non Standard Scaling).

**Y minor ticks on/off\*\***

If 'on' minor ticks on Y axis are visible. An entry of 'on' or 'off' is required.

**Y ticks both ends of plot on/off\*\***

If this is 'on' the program plots taxon abundance axis ticks at both ends of the plot. An entry of '**on**' or '**off**' is required.

**Y major tick length\*\***

Length of Y axis major tick marks.

**Y minor tick length**

Length of Y axis minor tick marks.

**Y major tick width\*\***

Width of Y axis major tick marks.

**Y minor tick width**

Width of Y axis minor tick marks.

**Y label font size\*\***

Font size of Y axis tick labels.

**Y label rotation\*\***

Rotation of Y labels (normally 90 degrees).

**Y label gap\*\***

Gap between Y axis titles and plots. Between 0.1 and 0.5 is usually sufficient.

**Space between plots\*\***

Space between each plot. Between 0.05 and 0.4 is normally sufficient.

---

**Group 10 Zone depths, labels, title, appearance**

Although used primarily for zones this section can be used to draw lines across the figure at the same depths for any purpose such as presence of tephra layers for example.

**Zones on/off\*\***

Zones included in plot or not. An entry of '**on**' or '**off**' is required.

**Zone depths**

Enter zone depth boundaries separated by commas only. See example in examples Parameter files.



**Zone labels**

Zone labels separated by commas only. See example in example Parameter files. If the number of zone depths is not one less than the number of names a warning will be issued, and the program stopped.

**Zone title**

Enter the zone title required.

**Zone title font size**

Zone title font size.

**Zone title rotation**

Zone title rotation. 120 degrees in one of the example Parameter files.

**Zone title colour**

Colour of zone title text. See colour codes.

**Zone title bold on/off**

Zone title in bold or not. An entry of **on** or **off** is required.

**Zone labels on/off**

Zone labels visible or not. An entry of 'on' or 'off' is required.

**Zone label font size**

Zone label font size.

**Zone label rotation**

Zone label rotation in degrees. 90 degrees in example Parameter files.

**Zone label position**

Distance from edge of zone boundary of plot in 'data units'.

**Zone label position correction**

If zones are narrow they may need minor correction to their default location to keep the label within zone lines. Apply correction here as a unit of depth. A negative number will move all labels upwards a positive one downwards.

**Zone label colour**

Colour of zone labels. See colour codes.

**Zone label bold on/off**

Zone labels in bold. An entry of '**on**' or '**off**' is required.

**Zone line style**

Zone line styles. Add line style code for each zone line separated by comma only. See line style coding above and example Parameter file.

**Zone line width**

Width of zone lines. Add line width for each zone line separated by comma only. See example Parameter file.

**Zone line col list**

Colour of zone lines. Add line colour for each zone line separated by comma only. See example Parameter file and colour codes.

**Zone boundary line left on/off**

Zone boundary line left. An entry of '**on**' or '**off**' is required.

**Zone boundary line right on/off**

Zone boundary line right. An entry of '**on**' or '**off**' is required.

**Zone boundary line top on/off**

Zone boundary line top. An entry of '**on**' or '**off**' is required.

**Zone boundary line bottom on/off**

Zone boundary line bottom. An entry of '**on**' or '**off**' is required.

**Zone boundary line style**

Zone boundary box line style. See line style coding.

**Zone boundary line colour**

Colour of boundary box zone lines. See colour codes.

**Zone boundary line width**

Width of zone boundary box zone lines.

**Zone X major tick colour**

Colour of depth major tick marks for zones if X ticks depth axis only parameter is 'off'.

**Zone X minor tick colour**

Colour of depth minor tick marks for zones if X ticks depth axis only parameter is 'off' and minor ticks are specified.

**Group 11 Radiocarbon title, depths, labels and appearance**

---

**RC ages on/off\*\***

If listed makes RC dates visible or not. An entry of 'on' or 'off' is required.

**RC age title on/off**

RC title visible or not. An entry of 'on' or 'off' is required.

**RC age title**

Provide title text for RC dates.

**RC age title depth position**

Provide depth at which title is to be displayed. Use minus number if is required above the 0 cm position. Units are data units.

**RC age title offset position**

Specify offset (Figure 2.5) from first plot in data units (RC dates are plotted to side of first taxon plotted next to depth axis). Therefore, likely to be minus number as to side of first taxon plot. See examples in example Parameter files.

**RC age title font size**

Font size of RC title text.

**RC age title colour**

Colour of RC title text. See colour codes.

**RC age title bold on/off**

RC title text bold or not bold. An entry of 'on' or 'off' is required.

**RC age title rotation**

RC title text rotation in degrees.

**RC age labels**

Provide RC age labels separating each with comma only.

### RC age label font size

Font size of RC label text.

### RC age labels colour

Colour of RC title text. See colour codes.

### RC age labels bold on/off

RC label text bold or not bold. An entry of 'on' or 'off' is required.

### RC age labels rotation

RC label text rotation in degrees.

### RC age depth label positions

RC depth locations separated by commas only.

### RC age label position offset

Offset from first plot in data units (RC dates are plotted to side of first taxon plotted next to depth axis). Therefore, likely to be minus number as to side of first taxon plot. See examples in example Parameter files.

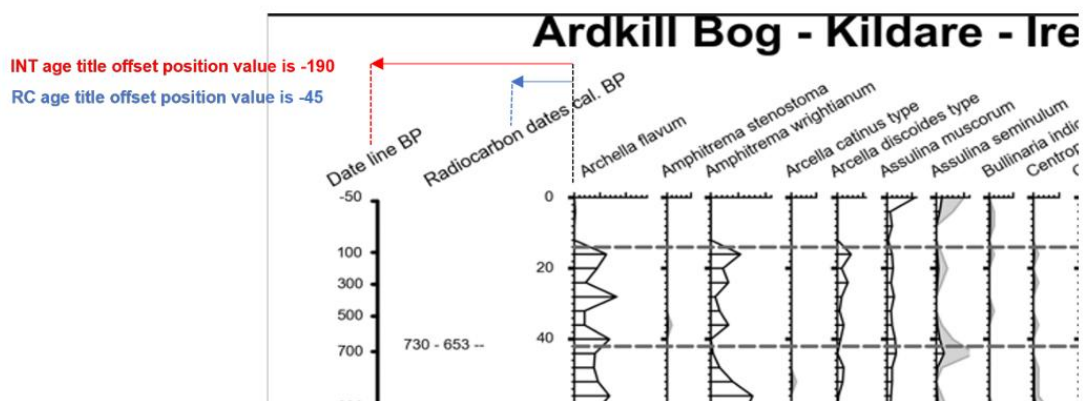


Figure 2.5. Arrows show minus offset from first plot for both RC (blue) and INT (red) age titles. Units are data units of the first plot. Offsets are required for the titles, labels and for the INT date line if required.

## Group 12 Interpolated dates title, depths, labels, appearance

### INT ages on/off\*\*

If listed makes INT dates and line visible or not. An entry of 'on' or 'off' is required.

**INT age title on/off**

RC title visible or not. An entry of '**on**' or '**off**' is required.

**INT age title**

Provide title text for INT dates.

**INT age title depth position**

Provide depth at which title is to be displayed. Use minus number if is required above the 0 cm position. Units are data units.

**INT age title offset position**

Specify offset from first plot in data units (INT ages are plotted to side of first taxon plotted next to depth axis). Therefore, likely to be minus number as to side of first taxon plot (Figure 2.5). See in example Parameter files.

**INT age title font size**

Font size of INT title text.

**INT age title colour**

Colour of INT title text. See colour codes.

**INT age title bold on/off**

INT title text bold or not bold. An entry of '**on**' or '**off**' is required.

**INT age title rotation**

INT title text rotation in degrees.

**INT age labels**

Provide INT age labels separating each with commas only.

**INT age label font size**

Font size of INT label text.

**INT age labels colour**

Colour of INT title text. See colour codes.

**INT age labels bold on/off**

INT label text bold or not bold. An entry of '**on**' or '**off**' is required.

**INT age labels rotation**

INT label text rotation in degrees.

**INT age depth label positions**

INT depth locations separated by commas only.

**INT age label position offset**

Offset from first plot in data units (INT dates are plotted to side of first taxon plotted next to depth axis). Will be minus number. See examples in example Parameter files.

**INT lines offset**

Offset of date line from first plot in data units (INT date line is plotted to side of first taxon plotted next to depth axis). Will be minus number (figure 9). See example in example Parameter files.

**INT upper most depth age**

Upmost depth of the first INT date to be used at the surface.

**INT lines width**

Date line line width.

**INT lines colour**

Date line colour. See colour codes.

**INT lines depth length**

Length of tick lines. This uses data units and is based on the data units in the first plot (abundance for *Archella flavum* or UOM in either example provided).

**INT lines width correction**

If tick lines do not meet main line exactly can use correction here to adjust. If no correction needed add 0. This uses data units so if the plot is based around depth as in the examples provided these will be the units.

**INT lines length correction**

If the main vertical line does not meet main line exactly can use correction here to adjust. If no correction needed add 0.

## Group 13 Grouping annotations

---

There are 10 options for including grouping annotations in the Parameter file. Groups 2- 10 annotation attributes are a repeat of Grouping 1 detailed here.

### **Grouping annotation 1 on/off\*\***

Display grouping annotation or not. An entry of '**on**' or '**off**' is required.

### **Grouping annotation 1 start**

Supply the taxon (y axis plot title) from where the annotation should start from, *Sphagnum* section *Acutifolia* in Figure 2.6 (see examples in example Parameter files).

### **Grouping annotation 1 title**

Supply text for the title of the group annotation.

### **Grouping annotation 1 title font size**

Font size for the text title.

### **Grouping annotation 1 title bold on/off**

Text bold or not bold. An entry of '**on**' or '**off**' is required.

### **Grouping annotation 1 title colour**

Text colour. See colour codes.

### **Grouping annotation 1 line colour**

Grouping annotation line colour. See colour codes.

### **Grouping annotation 1 line width**

Grouping annotation line width.

### **Grouping annotation 1 line start x**

Offset from nominated start taxon on x axis (depth axis, Figure 2.6, example value -35). Data units from start taxon.

### **Grouping annotation 1 line start y**

Offset from nominated start taxon on y axis (abundance axis, see Figure 2.6, example value 260). In data units.

### Grouping annotation 1 line end x

Offset for end of group annotation from nominated start taxon on x axis (depth axis, see Figure 2.6, example value -35, same as Grouping annotation 1 line start x). In data units.

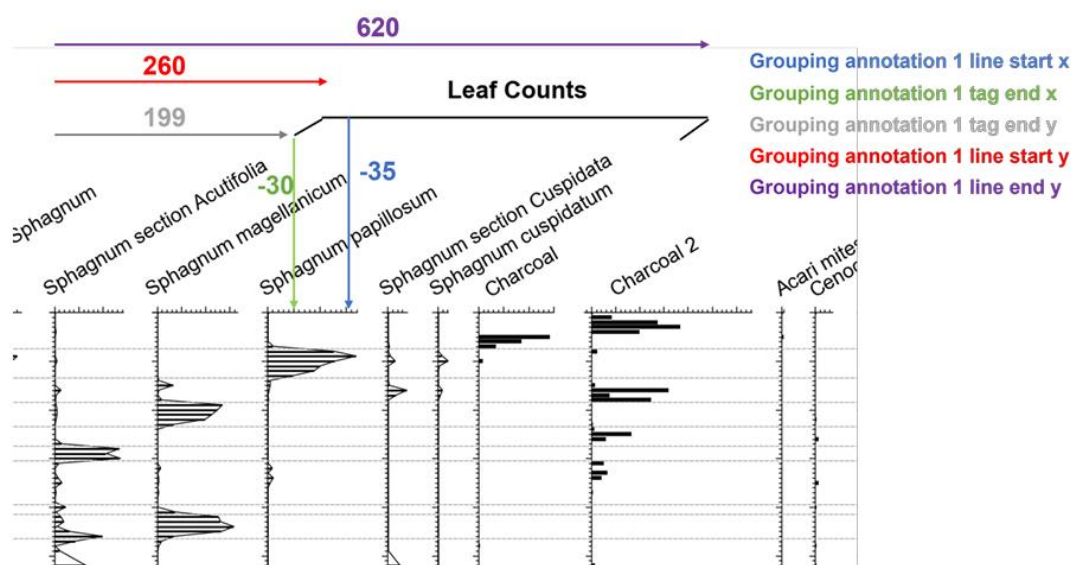


Figure 2.6. Arrows show dimensions required to draw the group annotation seen in figure 2. The numbers and arrows are colour coded to match the Parameter contained in the Parameter file for KM macrofossils and listed on the figure. All dimensions are based on the starting position from *Sphagnum* section *Acutifolia* as nominated in the Parameter file. Dimensions are in data units.

### Grouping annotation 1 line end y

Offset for end of group annotation from nominated start taxon on y axis (abundance axis, see Figure 2.6, example value 620). In data units.

### Grouping annotation 1 tag end x

Offset of end of tag end if required on x axis (see Figure 2.6, example value -30). In data units.

### Grouping annotation 1 tag end y

Offset of end of tag end if required on y axis (see Figure 2.6, example value 199). In data units.

### Grouping annotation 1 tag correction

Grouping annotation vertical correction in data units. If this grouping is being used this must be filled in. The tag part of the annotation occasionally does not meet the horizontal line. If this needs to be



corrected it can be done so here using data units. If no correction needed add 0.

#### **Grouping annotation 1 line correction**

Grouping annotation horizontal correction in data units. If this grouping is being used this must be filled in. The horizontal part of the annotation occasionally does not meet the tag lines. If this needs to be corrected it can be done so here using data units. If no correction needed add 0.

#### **Group 14 Non Standard Scaling (NSC) 1- 5, ticks, min and max limits**

---

If percentage taxa data is provided, it is likely to be within the 0 to 100 range. However, if also plotting pH for example the range may be from 3.5 to 5. Non Std Scaling (NSC) elements are for this purpose.

Here nominate the exact name of the measure used from the Input file and it will be selected to be treated differently to the other taxa data. This is necessary as the plots need to be scaled against each other which causes no problem if they are operating in similar ranges. There are 5 possible places for accommodating data with unusual scaling. These are NSC 1 - 5.

Options listed below are available for all 5 but will be listed here just for NSC 1. Please see the example in the ARD Testate Parameter.csv file. NSC plots used in this example are the '**Mean Water Table Depth (mm)**' and '**No. of testates per 1cm\*1000**'. If, as in this example, NSC entries 3- 5 are not being used then an entry of '**none**' is required. If using only two NSC options use NSC entries for NSC 1 and NSC 2. Do not use for example NSC 3, 4 or 5. Clearly if using 4 NSC options use NSC 1- 4.

#### **NSC 1\*\***

State data name **from Input file** that needs unusual scale. If not in use enter '**none**'.

#### **NSC 1 Y major tick interval**

Major tick interval for Y axis for NSC 1.

#### **NSC 1 Y minor tick interval**

Minor tick interval for Y axis for NSC 1.

#### **NSC 1 Y min**

Minimum value on Y axis.

**NSC 1 Y max**

Maximum value on Y axis.

**NSC 1 spine on/off**

Is spine required on Y axis. An entry of '**on**' or '**off**' is required.

**NSC 1 size**

Width of plot compared to maximum width of other NSC data. Expressed as decimal. 1 is 100 percent and 1.5 is 150 percent.

**NSC spine start 1**

Spine to start at 0 or minimum value. An entry of **0** or '**mini**' is required.

**Group 15 Stack Plots**

---

If stack plots are required information regarding codes for aesthetics are entered here. At present there is the option for two stack plots although this could be expanded by the author if demand exists. These stack plots are each able to display up to five groups. The example in Figure 1.2 displays three groups Ericaceae, Monocots and Sphagnum, shortened to Eric- Mono- Sphag in the title. These groups were specified in the Input file KM Macro Input.csv and the aesthetic attributes listed below are in the KM Macro Parameter.csv file for this example. These can be simply grouped and plotted, or they can be grouped and a new percentage of each individual group based on the sum of all groups calculated and plotted instead. These plots by default are plotted at a set size. This can be altered with the stack plot size attribute.

Attributes for Stack plot 1 listed below are simply repeated for Stack Plot 2 if two are required.

**Stack plot 1 on/off\*\***

Specify if stack plot required or not. Entry of '**on**' or '**off**'.

**Stack plot 1 title**

Add the title to be displayed above the stack plot.

**Stack plot calculation 1**

Is a recalculation of percentage based on the sum of the groups supplied required? A 'yes' or 'no' entry is required.

**Stack plot size 1**

Supply factor to size plot. Initially use 1. If plot is too small an entry of 1.2 will make it 20% larger or if too large an entry of 0.9 will make it 10% smaller.

**Stack plot col list 1**

Supply colours for your groups based on colour codes detailed in Section 2.1. Separate each code with a comma (See example in KM Parameter file) and no spaces.

**Stack plot line width 1**

Provide line width for lines separating groups in stack plot. If no lines required leave blank.

**Stack plot line colour 1**

Supply a single colour code for the lines between the groups in the stack plot. See colour codes. If no lines required leave blank.

## **2.3 Plotting data with a different sampling resolution**

The program assumes that for the main proxy be it pollen, testates, macrofossils or anything else that all taxa will have been measured at the same depths which would be commonplace. However, secondary data that a user may wish to be displayed such as charcoal, pH or loss on ignition (LOI) for example may not have been carried out at the same depth resolution. The program assumes everything has been measured at the same depths, but the Extra Input file here can be used to surmount this issue.

This has been carried out on the second version of the KM macrofossil plot (Figure 2.7). Separate Parameter, Input and Extra Input files have been supplied ending in LOI to demonstrate how (Figure 2.7) was achieved. LOI at KM has mostly been measured every 2 cm whereas the macrofossil data is recorded in places at 1 and in others 4 cm. Displaying the LOI uses both the **Extra Input file** and the **Non Standard Scaling** function. As this uses the Non Standard Scaling function these plots will be limited both to graph types 3, 5 or 6 and 5 instances of Non Standard Scaling. Listed below are the steps required to add LOI in this case to the KM plot. At present this is a work around and, in the future, if the need is demonstrated a more formal procedure will be developed with greater user options.

- 1) Create a dummy column in the main Input file in the desired position.

- 2) Name it, in this case LOI. Fill in the aesthetic options desired for graph type 3, 5 or 6. **This is the dummy column where data will be meaningless so make the plot colour aesthetics white.**
- 3) Add a single value entry in the initial depth for this dummy column that is extremely low. In KM Macro Input LOI.csv a value of 0.001 was added at the 0 cm depth position with all remaining depth slots being filled with zeros. This simply creates an empty graphing space where the user can then add the data at frequencies at odds with the main proxy.
- 4) Use the Extra Input file. See KM Extra Input LOI for layout. Insert a column with same name as that in the dummy column in the main Input file and fill in aesthetic values as for the same graph type as quoted in the dummy column but with values that the user would like the data displayed with.
- 5) The actual data here in Extra Input file in the depth slots should be the real data that the user wishes to display.
- 6) Save both Input files and make sure both are named in the Parameter file in the appropriate places in Group 1.
- 7) The LOI data in this example needs to be quoted as one of the NSC options in the Parameter file. Always use the 5 possible slots in order. Do not use slot 2 if only using one set of data with a NSC requirement. Use the slots sequentially.
- 8) Fill in options for the NSC in Group 14 of the Parameter file. See KM Macro Parameter LOI.csv file for the layout employed (Figure 2.7).
- 9) Once all files are saved the program can be run and iterations of option changes continued until the user is satisfied with the plot.
- 10) The example in Figure 2.7 uses graph type 5 with a brown line and black crosses at sampling points.

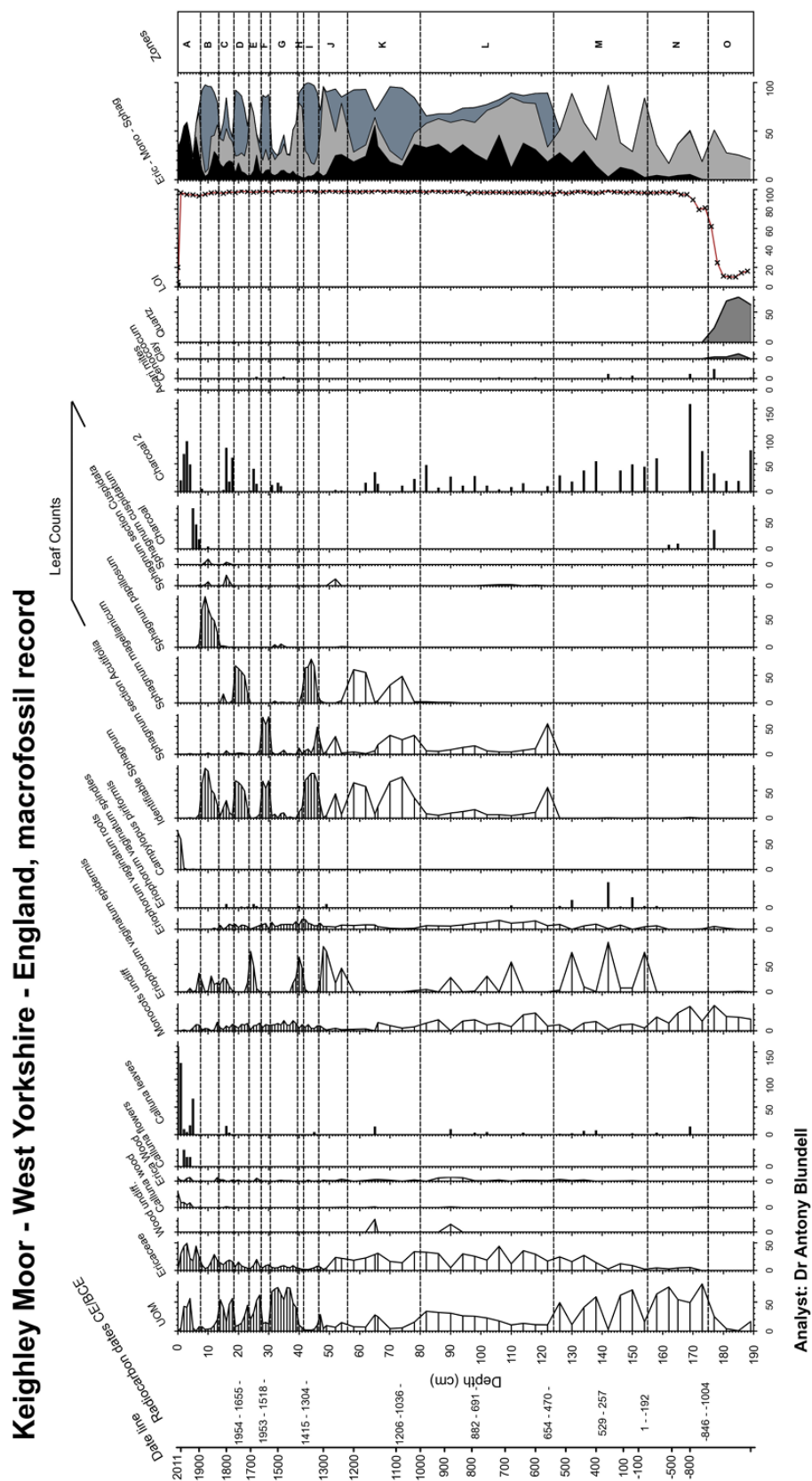


Figure 2.7. Keighley Moor macrofossil dataset with LOI plotted. This plot and associated example files show how to plot data, in the case LOI, outside the main proxy that is of different sampling resolution.

### 3 General procedure checklist

- 1) Set up P4 to run via a virtual environment or from a conda environment (**one off operation**).
- 2) Place program file (P4\_v02.py) in correct/desired location.
- 3) Fill out template Parameter file or modify one of the example Parameter files with preferences as directed in the manual and save as csv.
- 4) Fill out template Input file or modify one of the example Input file/s with desired preferences (as directed by manual) and palaeo data.
- 5) If required fill out template Extra Input file or modify example from ARD with desired preferences (as directed by manual) and palaeo data.
- 6) Place Input file/files and Parameter file in desired locations (easier but not essential if all in same location).
- 7) Activate environment and navigate to where program file is located. Execute command line statement to run program.
- 8) Fix any issues highlighted by the inbuilt user error checks that will provide messages pointing towards the problem.
- 9) Examine plot and change Input/Parameter files to make desired changes and re-save.
- 10) Execute command line statement to run program and repeat editing of the Input/Parameter files until desired plot is achieved.
- 11) Remember to cite the program, using citation from Zenodo or the publication (Blundell 2023).

## 4 References

- Blundell, A. 2023. Outline of the new palaeo plotting program P4. Mires and Peat 29, 1-11. (Online: <http://www.mires-and-peat.net/pages/volumes/map29/map2914.php>); doi: [10.19189/MaP.2023.OMB.Sc.2000309](https://doi.org/10.19189/MaP.2023.OMB.Sc.2000309)
- Blundell, A., Charman, D.J., Barber, K.B. 2008. Multi-proxy late Holocene records from Ireland: towards a regional palaeoclimate curve. *Journal of Quaternary Science* 23, 59-71.
- Blundell, A., Holden, J. 2015. Using palaeoecology to support blanket peatland management. *Ecological Indicators* 49, 110-120.