XRDpy User Manual

Andrew Garcia

Updated: July 2020

Summary

What is XRDpy?

XRDpy (*est. 2018*) is an XRD pattern plotting program which calculates crystallite size in an easy way. This program executes through the Terminal ('command line') and the execution is easy and straightforward. The program is

currently divided into 2 main scripts, XRD.py and XRDsingle.py.

The first script, XRD.py, calculates the Scherrer widths for multiple XRD patterns and plots them in a single figure, where the order of the plots can be easily customized by the user through the Terminal.

The second script, XRDsingle.py, does the same but for a single XRD pattern, giving a single plot in a figure.

These scripts are currently separate of one another, but a single script integration has been entertained as it may facilitate ease of use. Nonetheless, would such be the case the functionality of the program is expected to remain the same or improve.

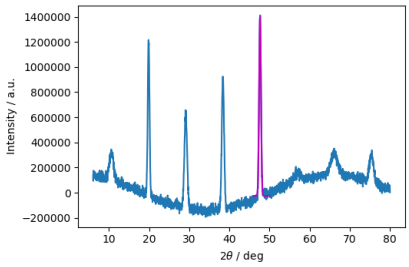


Figure 1: This is an output of the program

Compatibility

XRDpy runs on Python, which technically makes it multiplatform. However, I have only tested it in **Windows** and **Linux (Ubuntu)** operating systems, and it runs wonderfully on both. One should not find many complications running it with **Apple** systems iOS either, though the path format (directories) to access your excel and csv files may need to be changed accordingly.

How do I start using it?

Taking a pragmatic approach you may find the use of XRDpy to be very straightforward. You're encouraged to watch the instructional video of XRDpy which can be found by going to the main XRDpy Github repository (**github.com/andrewrgarcia/XRDpy**), scrolling down to the README and clicking on the black YouTube button.

I am drafting up a short manual here and thankfully Linux and the TeX language is making it way easier for me to write something relatively quick. Sorry

about the lorem ipsum dolor here and there...

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisissem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi necante...

Clone the repository

Needed: **Git** (git-scm.com)
Optional: **Github account** (github.com)
You may need to learn the basics of Git, which aren't complicated to do. Creating a Github account will allow you to fork my repository and share any changes you may make of your forked version(s).

In the main XRDpy Github repository, there should be a button called "Code" or clone. Click on it and copy the HTTP address

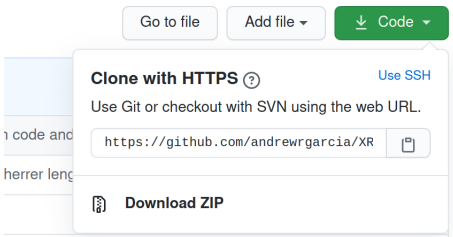


Figure 2: it looks like this

Open your Terminal and select the folder or path where you want to place the XRDpy program (i.e. use cd command).

In the Terminal, type: **git clone [pasted HTTP address]**

Run a simple command in the Terminal

Open your Terminal (or command line, or shell, whatever you call it)

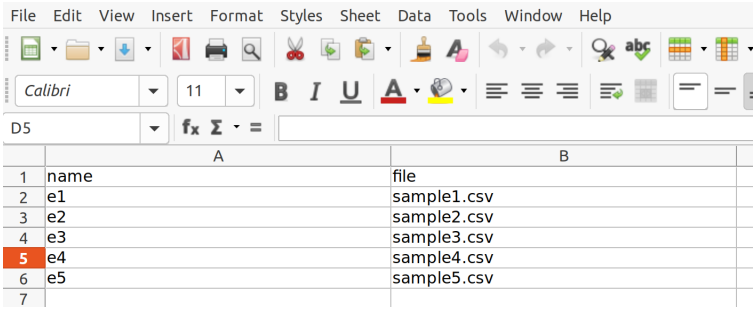
Change to the folder containing your cloned version of XRDpy (Use the cd command to get there)

While in that folder, type: **python XRD.py -h**. This should bring up the lists of all the arguments available to customize and make your plot(s).

```
:~/scripts/XRDpy$ python XRD.py -h
```

Make your database.xlsx file

Your database file should contain the names of all your files with their .csv extension (oh yes, they should be converted or be in csv format) in the right column, as well as a short nickname in the left column that will be used to call them by the XRDpy program. Please use the **database_template.xlsx** file provided in your cloned XRDpy folder. It makes things easier.



	A	B
1	name	file
2	e1	sample1.csv
3	e2	sample2.csv
4	e3	sample3.csv
5	e4	sample4.csv
6	e5	sample5.csv
7		

Set your common XRD file directories

Open XRD.py and XRD.py in a Python interpreter or text editor and change the paths where you are housing your database (see above) and your XRD files. Though you can call to switch the path in the Terminal by python XRD.py -p **['your-pathhere/.../...']** it is more efficient to have your permanent paths as defaults.
The argument IDs should be -p and -p2 for XRD.py and -p for XRD.single.py

For an interesting demonstration, leave the **database_template.xlsx** file unchanged and set the -p2 path to the folder named **XRD-patterns-fake** inside your XRDpy folder

```
ap.add_argument("-p", "--path_database_file", \
                default = '/home/andrew/XRD/database.xlsx',\
                type = str, help="Path and filename of Excel database which is used\
                to call all your files. Please update with your path and databse file name.")
ap.add_argument("-p2", "--path_files_folder", \
                default = '/home/andrew/scripts/XRDpy/XRD-patterns-fake/',\
                type = str, help="path to FOLDER containing XRD files listed in Excel database'")
```

Run an example command for XRD.py

Leaving the **database_template.xlsx** file unchanged and having set the -p2 path to the **XRD-patterns-fake** folder, type the following commands in your terminal (with the Terminal path set to the XRDpy folder):

python XRD.py (this will display your database; an alternative to **python XRD.py -d True**)

python XRD.py -o e1 e2 e3 e4 e5 -x 3 2 -s e3

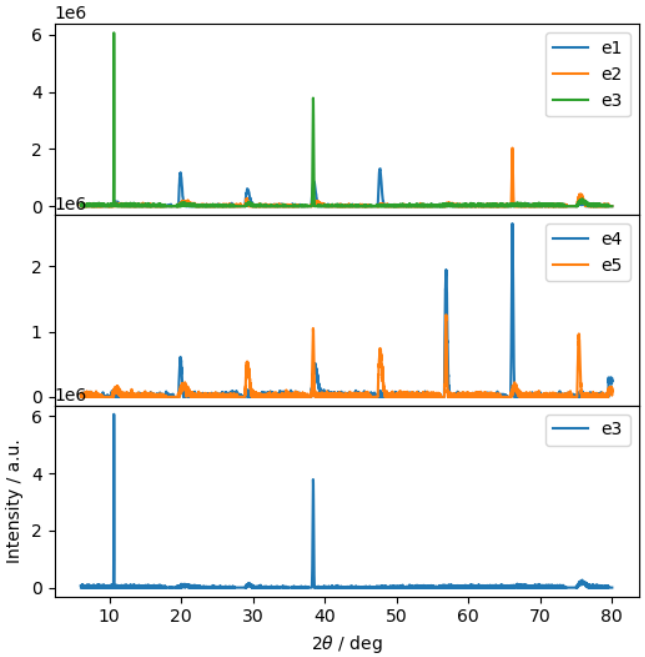


Figure 3: Output of above command. -o stands for overlaid with -x being the split of the declared overlaid plots and -s being the single plot

XRD.py arguments

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisissem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi necante...

-h, –help

Typing **python XRD.py -h** in your command prompt Terminal will give you the complete list of arguments you can run to process your XRD pattern [all the ones you see here below] .

-p, –path_database_file

python XRD.py -p r'C:\Users\...\database.xlsx' (Windows)
python XRD.py -p '/home/user-name/.../database.xlsx' (Ubuntu)

This specifies the address and file name of your excel database of XRD patterns, where the second column displays the file names of your XRD patterns (which should be changed to .csv extensions) and the first column can be a short name you use to easily call the .csv file (see database_template.xlsx) For easy execution, please update the default address to your address with your excel file on the Python script and save it (line 23 XRD.py)

-p2, –path_files_folder

python XRD.py -p2 r'C:\Users\...\XRD-files/' (Windows)
python XRD.py -p2 '/home/user-name/.../XRD-files/' (Ubuntu)

This specifies the path of your folder housing all your XRD patterns which should be mentioned in your Excel database. For easy execution, please update the default address to your address with your excel file on the Python script and save it (line 27 XRD.py)

-d –see_database

python XRD.py -d True

Boolean which, when set to True, displays every common name of the XRD patterns written in your Excel database. It is defaulted to False; if you ever want to look at the contents of your database before running the plotting script, type the above underlined command.

-ka, –K_alpha_wavelength

-b, –background_sub

-o, –overlaid

-x, –overlaid_split

-s, –single

-u, –units

-r, –Scherrer_range

-K, –shape_factor_K

XRDsingle.py arguments

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisissem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi

necante...