
pvct
Release 0.0.0

Matthew Rowles

Dec 03, 2021

CONTENTS

| | | |
|----------|---|-----------|
| 1 | Overview | 1 |
| 1.1 | Installation | 1 |
| 1.2 | Documentation | 1 |
| 1.3 | Development | 2 |
| 2 | Installation | 3 |
| 3 | Usage | 5 |
| 4 | Reference | 7 |
| 4.1 | pvct | 7 |
| 5 | Contributing | 9 |
| 5.1 | Bug reports | 9 |
| 5.2 | Documentation improvements | 9 |
| 5.3 | Feature requests and feedback | 9 |
| 5.4 | Development | 10 |
| 6 | Authors | 11 |
| 7 | Changelog | 13 |
| 7.1 | 0.0.0 (2021-11-06) | 13 |
| 8 | Indices and tables | 15 |

OVERVIEW

| | |
|---------|--|
| docs | |
| tests | |
| package | |

A program to construct a pseudo-variable count time diffraction pattern from many fixed-time diffraction patterns.

- Free software: Apache Software License 2.0

1.1 Installation

```
pip install pvct
```

You can also install the in-development version with:

```
pip install https://github.com/rowlesmr/pvct/archive/master.zip
```

1.2 Documentation

<https://pvct.readthedocs.io/>

1.3 Development

Ask me, and I can help.

INSTALLATION

At the command line:

```
pip install pvct
```


USAGE

To use pvct in a project:

```
import pvct
```

or, more likely, to run as a command-line program with arguments:

```
pvct lam mono B abs formula min max [-f... | -D]
```

This program is used to create a pseudo-variable-count-time (pVCT) diffraction pattern from many fixed-count-time (FCT) diffraction patterns.

To do so, information must be given to the program to calculate the way in which to sum the given diffraction data. Compulsory information is:

- **lam**: (float) wavelength in Å or energy in keV.
- **mono**: (float) monochromator angle in deg 2Th (0 for none, 90 for synchrotron).
- **B**: (float) average temperature factor (0.5 is a good start).
- **abs**: (float) capillary absorption in μ^*R or incident angle for fixed-incident-beam. (0 for no correction).
- **formula**: (string) Chemical formula → "La 1 B 6" or "Fe 1".
- **min**: (float) angle of first peak in deg 2Th.
- **max**: (float) maximum angle in deg 2Th.

You must also give one of the following options:

- **-filenames**: (string) list of file names - XY or XYE format. Wildcards accepted eg *.xy, la?.xy
- **-D**: (int) number of diffraction patterns to simulate.

One suggested workflow is to run pvct in simulation mode using the known experimental parameters with $D \approx 20$. An FCT diffraction pattern is then collected to assess both the factor, D , by which the intensities at that position must be increased to match the intensities of the low angle peaks, and the required collection time to give adequate intensities for the low angle peaks. D FCT diffraction patterns are collected, and pvct is then used to create a pVCT diffraction pattern.

Two examples of valid invocations of pvct are

```
pvct 0.7093 0 0.5 7 "La 1 B 6" 13 -1 -f *.xy  
pvct 16.5 90 0.3 0.9 "Mg 1.8 Fe 0.2 Si 1 0 4" 3 165 -D 20
```

The first will create a pVCT diffraction pattern from all XY files in the directory, using a wavelength of 0.7093 Å, no monochromator correction, an overall thermal parameter of 0.5 Å², an incident angle of 7°, and a specimen composition

of LaB_6 . The pattern summation calculations are taken with respect to the values at $13^\circ 2\theta$ and are carried out to a maximum angle as defined in the input data. The output files are written to the same directory as the source data.

The second will create a pVCT summary simulation using an energy of 16.5 keV, a monochromator angle of 90° , an overall thermal parameter of 0.3^2 , a capillary absorption of 0.9, and a specimen composition of $\text{Mg}_{1.8}\text{Fe}_{0.2}\text{SiO}_4$ over the range $3 - 165^\circ 2\theta$, assuming 20 diffraction patterns.

If the program is run with no command-line arguments, a minimalist GUI is launched to enable the user to enter the same information.

REFERENCE

4.1 pvct

CONTRIBUTING

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

5.1 Bug reports

When [reporting a bug](#) please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

5.2 Documentation improvements

pvct could always use more documentation, whether as part of the official pvct docs, in docstrings, or even on the web in blog posts, articles, and such.

5.3 Feature requests and feedback

The best way to send feedback is to file an issue at <https://github.com/rowlesmr/pvct/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that code contributions are welcome :)

5.4 Development

To set up *pvct* for local development:

1. Fork *pvct* (look for the “Fork” button).
2. Clone your fork locally:

```
git clone git@github.com:YOURGITHUBNAME/pvct.git
```

3. Create a branch for local development:

```
git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

4. Commit your changes and push your branch to GitHub:

```
git add .  
git commit -m "Your detailed description of your changes."  
git push origin name-of-your-bugfix-or-feature
```

5. Submit a pull request through the GitHub website.

5.4.1 Pull Request Guidelines

If you need some code review or feedback while you’re developing the code just make the pull request.

For merging, you should:

1. Update documentation when there’s new API, functionality etc.
2. Add a note to `CHANGELOG.rst` about the changes.
3. Add yourself to `AUTHORS.rst`.

CHAPTER
SIX

AUTHORS

- Matthew Rowles - .

CHANGELOG

7.1 0.0.0 (2021-11-06)

- First release on PyPI.

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`