The Journal of Open Source Software

POLEVAL: A Python package for HAXPES analysis

Robin Yoël Engel [1,2] **and Patrick Lömker** [1¶]

1 Chemical Physics, Department of Physics, Stockholm University, Roslagstullsbacken 21, 114 21 Stockholm **2** Wallenberg Initiative Materials Science for Sustainability, Stockholm University, Roslagstullsbacken 21, 114 21 Stockholm ¶ Corresponding author

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Lucy Whalley

Reviewers:

- @tyronerees
- @nluetts

Submitted: 21 March 2025
Published: unpublished

Summary

POLEVAL provides a software toolbox for collaborative, persistent and reproducible analysis of X-ray Photoelectron Spectroscopy (XPS), and particularly Hard X-ray Photo-Electron Spectroscopy (HAXPES) experiments. It allows to treat, analyse and visualise the results of an extended experimental campaign in a single Python notebook in a consistent manner. Managing experimental data in adequate objects enables experimentalists to process and analyse measurements in very few lines of code, so as to provide decision aids through online data analysis during e.g. beamtime experiments. The persistent and self-documentary style of the notebook-based analysis allows for easy communication of intermediate results and enables progressive refinements into publishable figures or exporting the results to other programs.

The toolbox facilitates various routines for data treatment (normalization, cropping, etc.) and aggregation of spectra into groups to analyse trends. It also enables quantitative analysis with three major functions:

First, normalization to the photoionization cross-section and probability of emission into the analyser cone allows for quantitative comparisons between intensities from different core levels. The integrated *haxquantpy* package allows easy retrieval of literature values (Trzhaskovskaya & Yarzhemsky, 2018) for this purpose.

Second, an extensive fitting functionality is implemented to treat groups of spectra together, rather than spectrum-by-spectrum. This grouping allows reinforcing the fit algorithm with prior knowledge, such as the equivalence of peak widths or positions between spectra, which enables for more consistent, and importantly, more confident fit results for sets of potentially noisy spectra.

Third, a simple formalism to estimate the thickness of adsorbate layers based on the ratio between the substrate's and adsorbate's XPS signal is implemented.

Statement of Need

XPS spectra are often analyzed using home developed tools to perform the analysis. Here, results are reported by specifying spectral shapes and background treatment, but often without a public discussion or publication of the actual procedures, or with the use of commercial software (Casa XPS (N. Fairley, 2018; Neal Fairley et al., 2021), UniFit (Hesse et al., 1999), etc.).

We find that common open-source source algorithms, including their publication, discussion and common development can be extremely beneficial for both the efficiency and scientific rigour of experimental work.

While some open-source software for XPS is available (i.e. (Nakajima, 2024) or (Stansbury & Lanzara, 2020)), published software providing a persistent, notebook-style analysis with

capabilities for treating, plotting and fitting inter-dependent sets of XPS spectra quickly and reproducibly is so far lacking.

Our work is inspired by the special environment of synchrotron-based research, which necessitates a frequent information hand-over and near-online data analysis. This motivates the choice of the Jupyter ecosystem, where a documented form of the proceedings (markdown descriptions, headers and source code together with spectra and analysis results) is inherently generated and can be worked on continually by several shifts of scientists performing an experiment over several days. Complex datasets often probe a certain dimension and vary one parameter over sets of spectra. Consequently POLEVAL enables quick plotting with many functions being able to get a single line of code to combine a complex data set, while also enabling a smooth transition into non-standard in-depth analysis within the same framework.

The next step after loading and manipulating data is analysing the spectral contributions. The ability to enforce consistent (or even equal) peak shape parameters over entire measurement series is essential, especially in chemically sensitive applications (Major et al., 2020), yet no software published for XPS so far provides this functionality, to the best of the authors knowledge.

The big strength of XPS is that it is quantitative, a spectrum can be related to the absolute abundance of chemical species in the sample surface-near region. Experiments in the HAXPES regime using grazing incidence X-rays necessitate a modelling of the X-ray penetration length in combination with the electron emission pathway. This feature is implemented in the tool such that, given the incidence angle and an average adsorbate composition, sample surface coverages can be quantified in terms of mono-layers.

Implementation

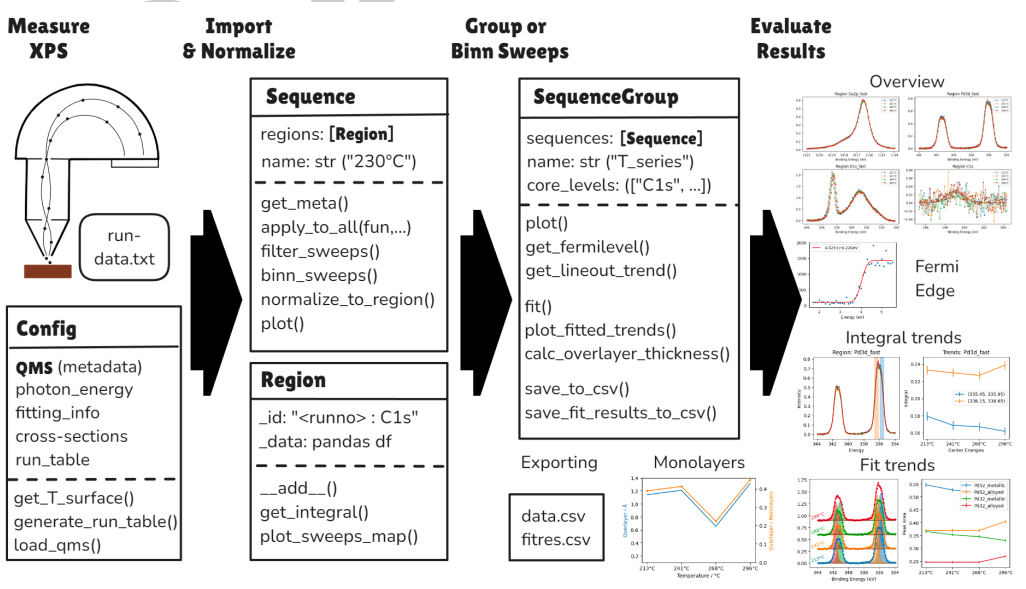

Figure 1: Data pipeline using the poleval package, showing the key classes and and their most important properties and methods as well as typical result plots.

The principal functions of POLEVAL are illustrated in **Figure 1**. Spectra and acquisition settings are read from text files generated by the SES software which controls the Scienta XPS analyser. While XPS data from CSV-, SPECS, and XY-files can be read and other formats could easily be added, the import from Scienta's SES text files is tested most.

Spectra are organised in `Region` objects that represent electronic core levels. Each `Region` object represents a single core level spectrum and retains information of the individual sweeps as well as the normalizations applied in the import step.

An experiment typically compares sets of regions, which are recorded together in an interleaved manner, `Region` objects are thus organised into `Sequence` objects, which correspond to a loaded file. Key scientific insights often emerge from analysing trends - be it dependent on time, temperature, gas pressure or other experimental variables. Such comparisons are enabled by the `SequenceGroup` class.

A `SequenceGroup` can be generated either from a list of separately imported and treated `Sequence` objects, or if applicable, by breaking up and binning the sweeps contained in the `Regions` of a single `Sequence`. Each of these classes contains rich functionality to facilitate the manipulation and intermediate visualization of data, while also maintaining reproducibility of each data treatment step taken. A `Config` object is used to store global settings, such as the photon energy, but also fit models for each core level and metadata in form of a `QMS` object, which can be used for binning or generating a run table as an experiment overview, as well as associating the recorded spectra with simultaneously recorded Quadrupole Mass Spectroscometer (QMS) and sample temperature data.

The peak fitting algorithm is based on the `lmfit` package, but operates on all `Regions` of the same core level within one `SequenceGroup`, and allows for dependency constraints between fit parameters. Common applications are to enforce a common (yet variable) peak width across multiple peaks and spectra, or defining a fixed ratio or distance between peak parameters in one spectrum. Defining such constraints by imposing prior knowledge helps to restrict the information content that must be extracted from potentially noisy data, and thus reduces the uncertainty of the fit results.

Similarly, the estimation of the adsorbate overlayer thickness requires a set of parameters, such as the lattice and atomic constants of each material involved, to be defined explicitly before an estimate can be computed. The mathematic model is laid out in the README.

Acknowledgements

This work was partially supported by the Wallenberg Initiative Materials Science for Sustainability (WISE) funded by the Knut and Alice Wallenberg Foundation.

We acknowledge Mikhail Shipilin for his conceptualization and basic implementation of specqp (Shipilin, 2025) that we used as the starting point to this project.

References

Fairley, N. (2018). *CasaXPS: Processing software for XPS, AES, SIMS and more*. Casa Software Ltd. http://www.casaxps.com/

Fairley, Neal, Fernandez, V., Richard-Plouet, M., Guillot-Deudon, C., Walton, J., Smith, E., Flahaut, D., Greiner, M., Biesinger, M., Tougaard, S., Morgan, D., & Baltrusaitis, J. (2021). Systematic and collaborative approach to problem solving using x-ray photoelectron spectroscopy. *Applied Surface Science Advances*, *5*, 100112. https://doi.org/https://doi.org/10.1016/j.apsadv.2021.100112

Hesse, R., Chassé, T., & Szargan, R. (1999). Peak shape analysis of core level photoelectron spectra using UNIFIT for WINDOWS. *Fresenius' Journal of Analytical Chemistry*, *365*(1), 48–54. https://doi.org/10.1007/s002160051443

Major, G. H., Fairley, N., Sherwood, P. M. A., Linford, M. R., Terry, J., Fernandez, V., & Artyushkova, K. (2020). Practical guide for curve fitting in x-ray photoelectron spectroscopy.

Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films, *38*(6), 061203. https://doi.org/10.1116/6.0000377

Nakajima, H. (2024). *LG4X* (Version v0.084). Zenodo. https://doi.org/10.5281/ZENODO.10477914

Shipilin, M. (2025, March 20). *SPECQP stands for SPECtroscopy Quick Peak*. https://github.com/Shipilin/specqp

Stansbury, C., & Lanzara, A. (2020). PyARPES: An analysis framework for multimodal angle-resolved photoemission spectroscopies. *SoftwareX*, *11*, 100472. https://doi.org/10.1016/j.softx.2020.100472

Trzhaskovskaya, M. B., & Yarzhemsky, V. G. (2018). Dirac–Fock photoionization parameters for HAXPES applications. *Atomic Data and Nuclear Data Tables*, *119*, 99–174. https://doi.org/10.1016/j.adt.2017.04.003