PrestoPronto: a code devoted to handling large data sets

S J A Figueroa, C Prestipino

1 Centro Nacional de Pesquisa em Energia e Materiais (CNPEM), Laboratório Nacional de Luz Sincrotron (LNLS), Rua Giuseppe Máximo Scolfaro, 10.000, Polo II de Alta Tecnologia, Caixa Postal 6192, 13083-970 Campinas, SP, Brasil.
2 Institut des Sciences Chimiques de Rennes (ISCR), UMR CNRS N°6226, 263 Avenue du Général Leclerc, 35042 RENNES, France.

E-mail: carmelo.prestipino@univ-rennes1.fr

Abstract. The software PrestoPronto consist to a full graphical user interface (GUI) program aimed to execute the analysis of large X-ray Absorption Spectroscopy data sets. Written in Python is free and open source. The code is able to read large datasets, apply calibration, alignment corrections and perform classical data analysis, from the extraction of the signal to EXAFS fit. The package includes also programs with GUIs to perform, Principal Component Analysis and Linear Combination Fits. The main benefit of this program is allow to follow quickly the evolution of time resolved experiments coming from Quick-EXAFS (QEXAFS) and dispersive EXAFS beamlines.

1. Introduction
Starting on early eighties with the first developments on fast X-ray Absorption Spectroscopy as dispersive EXAFS [1,2] and QEXAFS [3,4] beamlines, the use of time resolved experiments become widespread. From the early technical developments to the later ones [5] the amount of data that experimentalists must treat easily go from some tens of spectra to several hundred till some thousand. Here we present a new program developed in Python, a full graphic tool for the analysis of large datasets generated in time or space resolved experiments several codes have been already wrote [6–8] and analysis macro languages as Iffefit [9] and Larch were developed capable to satisfies all analysis strategy. Nevertheless the learning curve for analysis macro languages is pretty steep for users without coding experience and writing script is relatively time consuming. The main aims of PrestoPronto package is to cover the need to rapidly screen data during experiments and perform basic data analysis, dealing with an important set of spectra in a quick, intuitive and easy way, using the algorithm that are the most close to the user experience. The code has been used already in few study [10–12] it is free and open source so could be adapted to specific requirements as different data formats, peculiar setup or different type of analysis directly by users or in a collaborative way.

2. Structure of the code
The code is written in Python 2.7 using the standard Tk/Tkinter graphical toolkit for user interfaces, and mathplotlib [13] package for plotting data. In the software the sequence spectra is stored in a python list...
where each element is a class containing a spectrum with bound methods and properties. The EXAFS specific functions are performed using the Ifeffit engine [9].

In order to improve the interaction with the user and the data processing the software charge into the memory the entire sequence of spectra and define new matrix at each step. Consequently the software is limited to handling set of data that are smaller than the available memory. To our knowledge the program has been tested up to 1300x1050 data matrix, however counting the memory needed to process and store intermediate data, a modern computer with one Giga of available memory should be able to treat a set of 25000 spectra of 4000 data points or an equivalent size matrix.

The package is composed by three programs: PrestoPronto, Linearcomb_GUI and PCA_GUI.

2.1. PrestoPronto
The main goal of PrestoPronto (soon ready, from Italian) is to allow a fast analysis of a sequence spectra that come as a result of QEXAFS and Dispersive XAS experiments, principally the ones that involve in-situ and operando studies. The program is designed for rapidly calibrate, average, extract and model data in order to follow with a minimum time delay the progression of an experiment. The software shows a single notebook window (figure 1) and a terminal were information for user are printed. The notebook tabs must be used from left to right, where each tab correspond to a step of the analysis that could be plotted or saved in a multicolumn ASCII file.

The first tab “beamlines” is used to import data from different formats and corrects energy shift using a calibration sample (a reference spectra collected simultaneously with the data). The tab “Averages” could rebinning and truncate the spectra, as well could select a subset of data and cyclically average (every N spectra). The tab XANES perform a XANES normalization and plot the edge position, the edge jump and the integral of a selected region of the spectra. Tab EXAFS-FT is devoted to EXAFS signal extraction and Fourier transformation. Finally the tab Fit is devoted to EXAFS analysis using theoretical path calculated by FEFF code [14], the spectra are fitted individually (nowadays the model is s limited to only two shell). In order to simplify the choice of parameters and visualization of the full set of data during the normalization in the plot interface a slider has been added helping to stack plot or to select the relevant parameters (figure 2).
2.2. **PCA_GUI**

As additional resources to the standard data treatment the Principal Component Analysis (PCA) algorithm is added with a dedicated GUI. This module is showed by a notebook with three tabs: Setup, PCA and ITTFA. The “Setup” tab is used to input a multicolumn ASCII file and select the interested subset and the spectral range to analyse. The “PCA” tab visualizes the results of PCA analysis [15] showing several parameters to determine the number of factors: IND and MAD functions; residual variance and Reduced eigenvalue F-test (REV) (see figure 3).

Finally, the “ITTFA” tab is based on the approach used in the program Factor (developed by Marcos Fernandez Garcia) where the Iterative Target Transformation Factor Analysis is performed. This procedure allows to align theoretical components obtained from PCA analysis to physical meaningfully spectra [16,17].

2.3. **Linearcomb_GUI**

Finally to complete this package we include the Linearcomb_GUI figure 4. This part is a front end for the python library LMFIT [18] which allows to fit a set of spectra in a multicolumn file by linear combination of a series of standards. The code use the Levenberg-Marquardt algorithm to preform minimization and standard error calculate at 1σ interval of confidence.

3. **Resource**

A project page for PrestoPronto is accessible at http://soonready.github.io/PrestoPronto/. The self-installer and source are available. The screen shots in this article were made on a Windows 7 computer. The programs may be different in appearance on other systems.
Acknowledgments
The authors acknowledge the European Synchrotron Radiation Facility for beam time provision at ID24, BM23 (ex-BM29) beamlines and the support of Sakura Pascarelli and Mark A. Newton in helping to develop this program. We also thanks Matt Newville and Marcos Fernández-Garcia for gently open their codes.

References
[1] Kaminaga U, Matsushita T and Kohra K 1981 Jpn. J. Appl. Phys. 20 355
[2] Matsushita T and Phizackerley R P 1981 Jpn. J. Appl. Phys. 20 2223
[3] Frahm R 1988 Nucl. Instruments Meth. A 270 578
[4] Stötzel J, Lützenkirchen-Hecht D and Frahm R 2010 Rev. Sci. Instrum. 81 073109
[5] Rodriguez J A, Hanson J C and Chupas P J 2013 In-situ characterization of heterogeneous catalysts (New York: Wiley)
[6] Del Rio M S and Dejus R J Proc. SPIE 8141, Advances in Computational Methods for X-Ray Optics II, 814115
[7] Ressler T 1998 J. Synchrotron Radiat. 5 118
[8] San-Miguel A 1995 Phys. B: Condens. Matter 208 277
[9] Newville M 2001 J. Synchrotron Radiat. 8 322
[10] Moog I, Prestipino C, Figueroa S, Majimel J and Demourgues A 2014 J. Phys. Chem. C 118 22746
[11] Achilli E, Minguzzi A, Lugaresi O, Locatelli C, Rondinini S, Spinolo G and Ghigna P 2014 J. Spectrosc. 2014 480102
[12] Nilsson J, Carlsson P-A, Fouladvand S, Martin N M, Gustafson J, Newton M A, Lundgren E, Grönbeck H and Skoglundh M 2015 ACS Catal. 5 2481
[13] Hunter J D 2007 Comput. Sci. Eng. 9 90
[14] Rehr J J and Albers R C 1990 Phys. Rev. B 41 8139
[15] Malinowski E R 2002 Factor analysis in chemistry (New York: Wiley)
[16] Fernandez-Garcia M, Marquez Alvarez C and Haller G L 1995 J. Phys. Chem. 99 12565
[17] Rossberg A, Reich T and Bernhard G 2003 Anal. Bioanal. Chem. 376 631
[18] Newville M, Stensitzki T, Allen D B, Rawlik M, Ingargiola A and Nelson A R J 2014 LMFIT Zenodo. 10.5281/zenodo.11813