

Feasibility of the SunPy environment for analyzing solar data using Python

Vinícius Mattos dos Santos*, André Leon Sampaio Gradvohl.

Abstract

The purpose of this project is to investigate the SunPy, a Python Application Programming Interface, which was created as an alternative to acquire and analyze solar data. The project consisted in the creation of a Portuguese manual for SunPy, in a bibliographic research, and the development of scripts to find ways to work with solar data. The efforts stemmed from the project have been helping the researcher of the HighPIDS group from the School of Technology at the University of Campinas.

Key words: Python, Solar Data, Sunpy.

Introduction

Solar flares are natural phenomena caused by the chemical composition of the main star of our planetary system, combined with its physical properties as pressure and gravitational force. Some of these phenomena, like coronal mass ejections and solar flares x-ray emissions, are related to events observed on earth, such as failures in satellites and energy outages.

Some of the major space observation centers in world record these activities daily. One of the main difficulties in data analysis related to solar activity is the lack of software that enables the user to quickly and easily acquire and visualize solar data.

A software that performs satisfactorily these features is the Solarsoft. However, the Solarsoft is a proprietary software, which has a very expensive license. To offer an alternative to the Solarsoft software, a group of scientists from NASA developed an Application Programming Interface (API) written in Python named SunPy (MUMFORD et al., 2015).

The SunPy API allows the acquisition and analysis of Solar Data from the main databases of the solar weather databases. The SunPy API is multi-platform, free and has an easy to use structure. Because the SunPy is a recent API, it has not many citations on scientific papers so far. That fact made us even more excited to learn and explore the potential of this new API.

Results and Discussion

Until now, the results achieved were the development of a Portuguese manual, which explains the installation possibilities in each possible operation system, the main classes of the API, and features of the API with functional examples. In addition, we made a presentation about the API features, showing how it works.

We developed the manual to help users that never have experienced the SunPy API. We divided the manual into sections as follows: Installation, Main Functionalities, Data Types, Time Ranges, Regions of Interest and Coordinates.

Figure 1 illustrates an example of LightCurve chart feature, one of the three data types present in SunPy. This data object shows a chart, which depicts the x-ray wavelength for two x-ray sensors present on a GOES satellite. This example shows four M-class and one X-class x-ray fluxes on May 5, 2015.

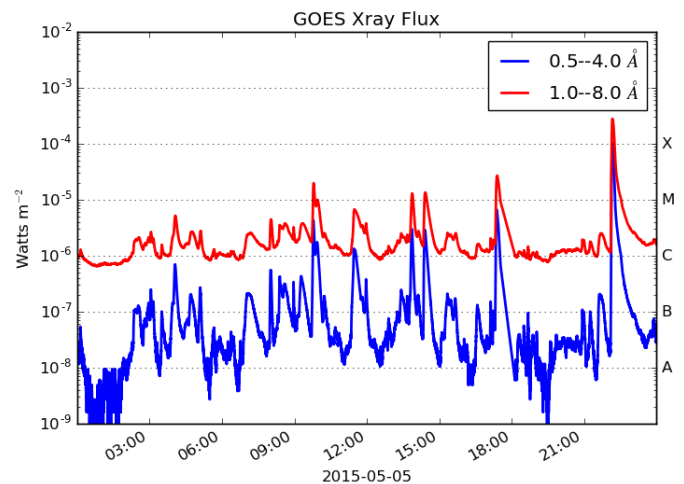


Figure 1: LightCurve for Goes x-ray flux on May 05, 2015.

Conclusions

For now, the SunPy API seems an excellent alternative to acquiring and analyses solar data. This is because it uses mature and open APIs as dependency libraries such as AstroPy, NumPy, and others. In addition, SunPy has a well-defined structure, which is easy to use and very well encapsulated.

Acknowledgement

I would like to thank my advisor, for all meetings and advices, as well as my colleagues at the National Research Center for Energy and Materials (CNPEM), where I started to use Python. In addition, I would like to thank my family for all the support.

¹ Mumford, S.; Pérez-Suárez, D.; Christe, S.; Mayer, F. e Hewett, R. J. *SunPy-Python for Solar Physics*. Computational Science and Discovery, v. 8, n. 1, art. 014009, 2015. DOI 10.1088/1749-4699/8/1/014009.