The Journal of Open Source Software

Solar Data Tools: a Python library for automated analysis of unlabeled PV data

Sara A. Miskovich [1] and **Bennet Meyers** [1¶]

1 SLAC National Accelerator Laboratory, Menlo Park, CA, 94025, USA ¶ Corresponding author

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Open Journals

Reviewers:

- @openjournals

Submitted: 01 January 1970
Published: unpublished

Summary

Solar Data Tools is an open-source Python library for analyzing photovoltaic (PV) power (and irradiance) time-series data. It enables automated analysis of *unlabeled* PV data, i.e. with no model, no meteorological data, and no performance index required, by taking a statistical signal processing approach in the algorithms used in the library's main data processing pipeline.

Solar Data Tools provides methods for data I/O, cleaning, filtering, plotting, and data quality and loss analysis. These methods are largely automated and require little to no input from the user regardless of system type. This library is for anyone dealing with photovoltaic data, especially data with no meteorological information (unlabeled). This includes photovoltaic professionals (in private solar industry or utility companies for example), researchers and students in the solar power domain, community solar owners, and anyone with a rooftop system. The scientific goal of the library is to empower PV system fleet owners or operators to analyze system performance, even when they only have access to the most basic data stream—power output of the system.

Statement of need

With the growing number of real-world installations of photovoltaic (PV) systems worldwide, it is crucial to have tools that can process and analyze the data from system of all sizes and configurations. The data is typically in the form of time-series measurements of real power production reported as average power over some interval of time (between the scale of one minute to one hour) over a number of years, possibly with missing entries.

Historically, PV data analysis tools have focused on data that are combined with local meteorological measurements and system configuration information (such as those from large power plants). The data cleaning tasks have largely been manual, the analyses have relied on metrics such as the performance index (townsend), which rely on having access to accurate site models and meteorological data. In the case of smaller, distributed rooftop PV systems, meteorological information is typically lacking, and it's often difficult to model the system accurately. For such systems, we desire to get insights from just the PV power data in isolation (what we call *unlabeled* data), for which it is difficult or impossible to form a performance index. Given that distributed rooftop PV systems make up over 40% of the installed capacity in 2020 (Davis et al., 2021), there is a clear need for automated and model-free data processing and analysis tools that can enable the remote monitoring of system health and optimization of operations and maintenance activities of these systems.

Solar Data Tools (SDT) (B. Meyers et al., 2020; Bennet Meyers & others, 2024) is an open-source Python library for automatic data processing and analysis of unlabeled PV data signals. SDT automates the cleaning, filtering, and analyzing PV power data, including loss factor estimation analysis, eliminating the need for user configuration or "babysitting" regardless of data quality or system configuration, from large, utility-scale trackers to small, multi-pitch rooftops. SDT provides practical tools for both small and fleet-scale PV performance analyses, without the need to calculate performance indices for each system.

There are two other libraries that are similar in that they offer data analysis tools for solar applications: (Perry et al., 2022) and (**?**). In contrast to SDT, they are both model driven, and require the user to define their own analysis. PVAnalytics focuses on preprocessing and QA, while RdTools focuses on loss factor analysis. SDT provides both data quality and loss factor analysis, runs *automatically* with little to no setup, and is **model-free** and does not require any weather or other information. SDT is most suited for when users want a pre-defined pipeline to get information on complex systems that can't be modeled easily and that have no meteorological data–which is frequently the case for small, distributed systems. (cite tutorial here for more info?)

Figures

Add example plots (heatmaps, loss analysis, what else?)

Figures can be included like this: Caption for example figure. and referenced from text using section .

Figure sizes can be customized by adding an optional second parameter: Caption for example figure.

Acknowledgements

This work is supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under the Solar Energy Technologies Office Award Number 38529.

References

Davis, M., Smith, C., White, B., Goldstein, R., Sun, X., Cox, M., Curtin, G., Manghani, R., Rumery, S., Silver, C., & Baca, J. (2021). *U.S. Solar market insight executive summary, 2020 year in review*. Wood Mackenzie; SEIA.

Meyers, B., Apostolaki-Iosifidou, E., & Schelhas, L. (2020). *Solar data tools: Automatic solar data processing pipeline* (pp. 0655–0656). https://doi.org/10.1109/PVSC45281.2020.9300847

Meyers, Bennet, & others. (2024). *Slacgismo/solar-data-tools: v1.2.2* (Version v1.2.2). Zenodo. https://doi.org/10.5281/zenodo.10888385

Perry, K., Vining, W., Anderson, K., Muller, M., & Hansen, C. (2022). *PVAnalytics: A python package for automated processing of solar time series data*. https://www.osti.gov/biblio/1887283