



**Sandia  
National  
Laboratories**



2022



**PV Performance**  
MODELING COLLABORATIVE

pvlib v0.9.3

pvlib v0.9.4

pvlib v0.9.5

2023



**PV Performance**  
MODELING COLLABORATIVE

# User's group: pvlib python and pvanalytics

PV Performance Modeling and Monitoring Workshop  
Salt Lake City, May 10, 2023

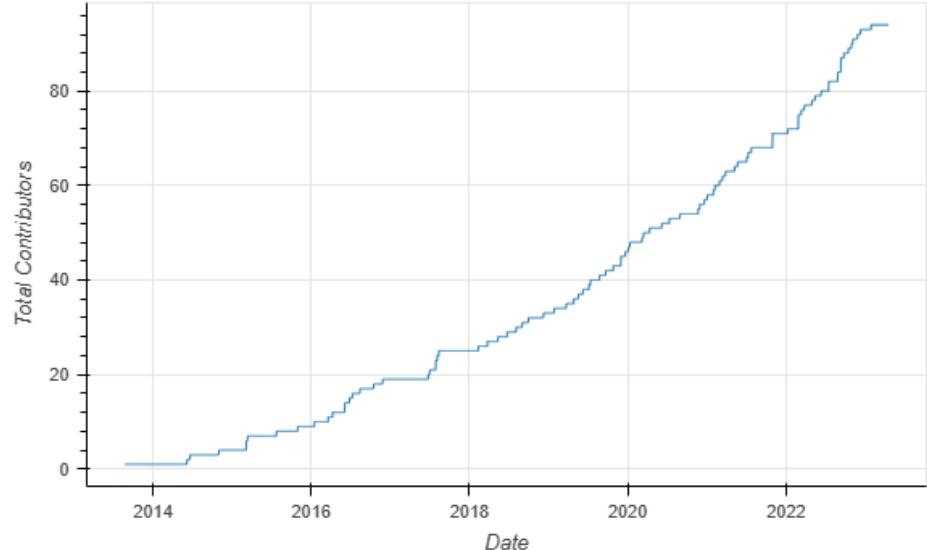
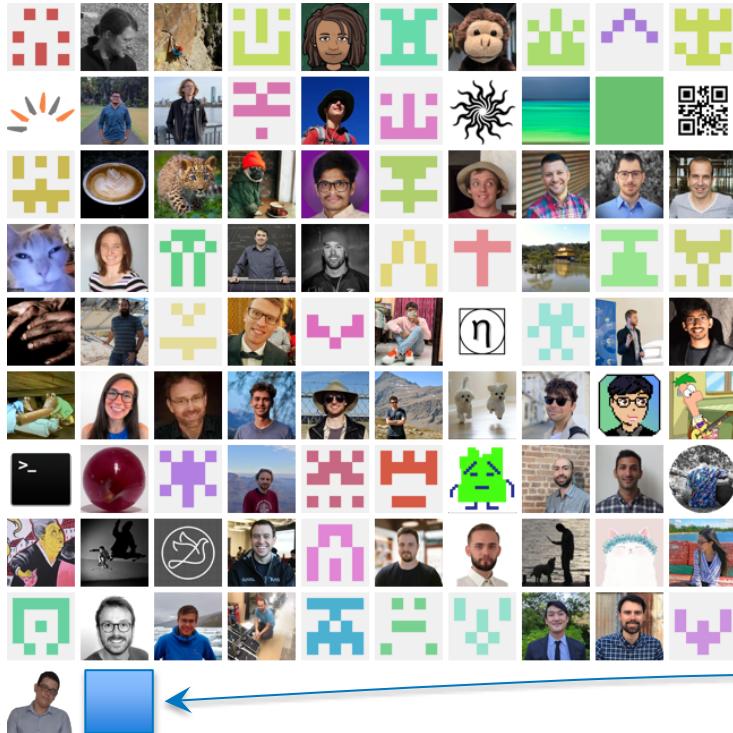
## **pvlib-python**

- Cliff Hansen (Sandia)
- Kevin Anderson (Sandia)
- Will Holmgren (DNV)
- Mark Mikofski (DNV)
- Adam R. Jensen (DTU)

## **pvanalytics**

- Cliff Hansen (Sandia)
- Will Vining (Sandia)
- Kevin Anderson (Sandia)
- Kirsten Perry (NREL)

# pvlb python: GitHub Contributors



\*Not all contributions are code!

This software is made possible by contributions from people like you. You can help!

<https://pvlb-python.readthedocs.io/en/stable/contributing.html>

## Does 1.0.0 come right after 0.9.\*?

- No. Expect 0.10.0 to come next. There is no ETA for pvlib 1.0.0 yet ☺

## What does 1.0 mean?

- A declaration that pvlib is no longer “beta” (whatever that means)
- Mostly, no more changes that break people’s code (until 2.0, anyway)

## What needs to happen before 1.0?

- Change all the things we want to change:
  - Package-wide consistency in naming (mostly there already, but still room for improvement)
- Fill in some modeling gaps: transformer losses, direct shading, etc.
- Rewrite/reorg the docs to follow an intentional strategy instead of the current ad-hoc “pile of info”
- **What else? We’d love to get your feedback!**

Additions/improvements where publications and/or data are available

- ✓ **Spectral irradiance mismatch calculations**
- ✓ **Functions and models for module efficiency**
- Functions for horizon shading (e.g., input digital elevation data, output angle from ground to horizon)
- Function(s) for LCOE
- Cell, module and string electrical mismatch calculations (stay tuned)
- **Degradation of DC components**

Additions where new publications and/or data may be needed

- Functions to translate parameters among models: temperature, incidence angle modifier (IAM), IV curve models
- **Models for transformer losses, inverter operations off unity power factor, inverter temperature derating**
- **“Smarter clipping” (find both voltage and power from IV curve rather than simply capping power)**

## Library of data for model benchmarking

- Non-expert users can be challenged when faced with e.g. seven clear-sky models
- Provide some guide to assist in selecting appropriate models
- Illustrate model accuracy and limitations

## Overhaul the documentation

- The “middle” layer between the home page for pvlib-python and each function’s document page

## More examples

- ✓ **More frequent releases (quarterly)**
- Quarterly community meetings (contact [cwhanse@sandia.gov](mailto:cwhanse@sandia.gov))

[www.github.com/pvlib/pvanalytics.git](https://www.github.com/pvlib/pvanalytics.git)

## Welcoming contributors, developers, maintainers

Project will mature through contact with users

Guiding principles:

- Built from the bottom up
  - Workflow independent library of documented functions
  - Common signature: input time series, output Boolean series
    - E.g., clear sky labeling, clipped inverters, GHI passes quality checks
- Combine functions into workflows
  - Prioritize flexibility

New Features?

Classes to help automate workflows

- Gallery of examples should help to clarify workflow definitions

# What are your ideas?



# Thank You

---

[www.github.com/pvlib/pvlib-python](https://github.com/pvlib/pvlib-python)  
<https://pvlib-python.readthedocs.io>

[www.github.com/pvlib/pvanalytics](https://github.com/pvlib/pvanalytics)  
<https://pvanalytics.readthedocs.io>