
PV ICE

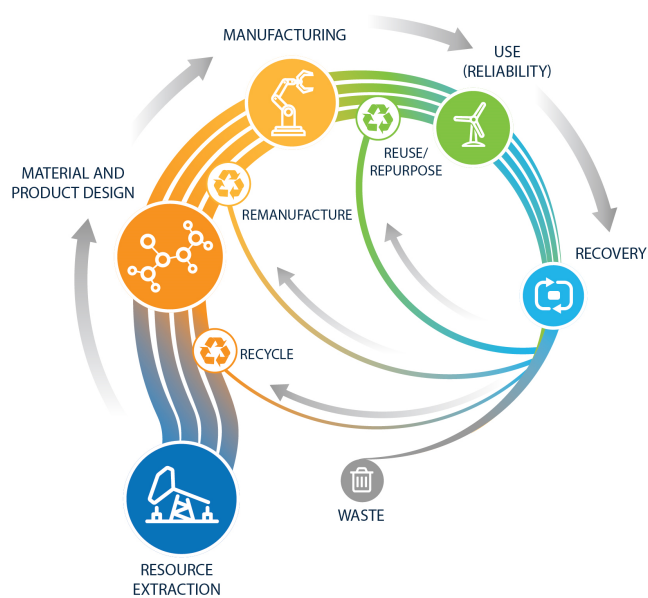
Release 0.1.0

Silvana Ayala, Heather Mirletz

Mar 24, 2021

CONTENTS:

| | | |
|----------|---|-----------|
| 1 | Citing PV_ICE | 3 |
| 2 | Contents | 5 |
| 2.1 | Package Overview | 5 |
| 2.1.1 | Framework and Definitions | 6 |
| 2.2 | Data | 7 |
| 2.2.1 | Input Data | 7 |
| 2.2.2 | Output Data | 9 |
| 2.2.3 | Data References | 10 |
| 2.3 | What’s New | 11 |
| 2.3.1 | v0.1.0 (12 / 15 / 2020) | 11 |
| 2.4 | Installation | 12 |
| 2.5 | Contributing | 12 |
| 2.6 | API | 12 |
| 2.6.1 | Flow Mass Functions | 12 |
| 2.6.2 | Reliability and Failure Functions | 13 |
| 2.6.3 | Modifying Baseline Functions | 14 |
| 2.6.4 | LCA Functions | 15 |
| 3 | Indices and tables | 17 |
| | Python Module Index | 19 |
| | Index | 21 |



The Photovoltaics in Circular Economy tool (PV ICE) is a NREL supported tool to help quantify and assign a value framework to efforts on re-design, reduction, replacement, reuse, recycling, and lifetime and reliability increases on PV, implementing Circular Economy metrics for PV materials.

The PV_ICE leverages published data from different sources on PV manufacturing and predicted technological changes.

This tool will help implement circularity metrics, quantify and assign a value framework to efforts on re-design, reduction, replacement, reuse, recycling, and lifetime and reliability increases on PV.

The intended audience ranges from PV performance and analyst researchers, Engineering Procurement Construction (EPC) companies, installers, investors, consumers and analysts of the PV industry interested in predicting and evaluating the impact of PV growth in our environment and how the improvements towards Circularity affect waste.

The source code for PV_ICE is hosted on [github](#).

Please see the [Installation](#) page for installation help.

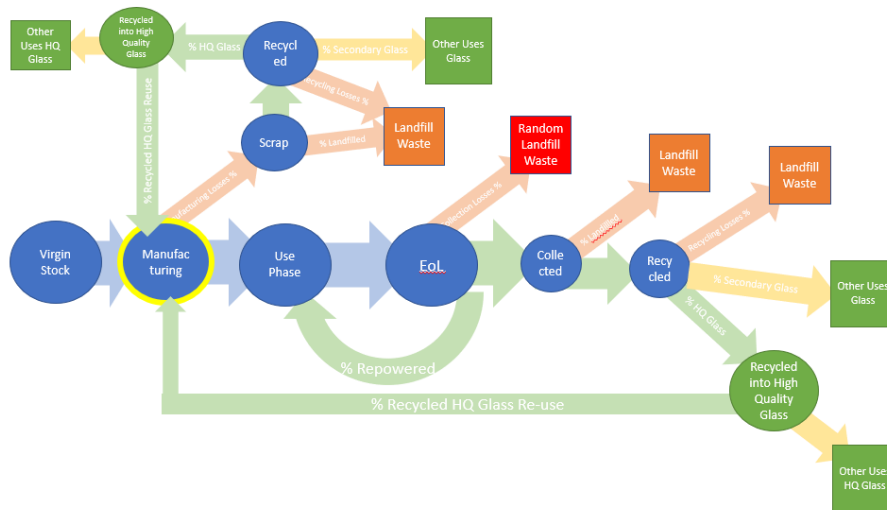
CITING PV_ICE

If you use this calculator in a published work, please cite:

Ayala Pelaez, Silvana; Mirletz, Heather; Silverman, Timothy; Carpenter, Alberta; Barnes, Teresa. “De-fluffing Circular Economy Metrics with Open-Source Calculator for PV” 2020 PV Reliability Workshop, Denver CO.

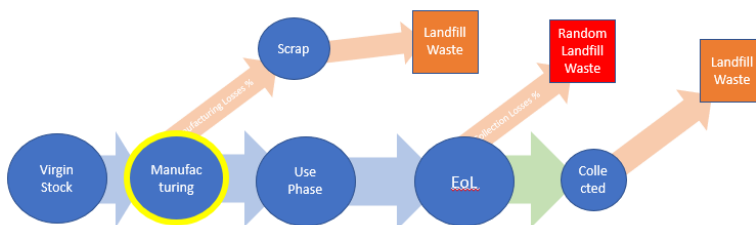
Please also cite the DOI corresponding to the specific version of PV_ICE that you used. pvlib-python DOIs are listed at [Zenodo.org](https://zenodo.org)

The PV ICE tool considers the following material flow:



Input to the calculator is an excel file with the appropriate columns to establish all the paths in the diagram. Examples for baseline input files for the US and World past and projected values are available in the PV_ICE/baselines folder.

For contrast, a completely linear economy would look like:



2.1.1 Framework and Definitions

PV Life Stages/Phases: We are setting the boundaries of the PV Life stages at mining/extraction and processing of the virgin materials, which are all represented as a single stage (“Extraction and Material Prep”). Subsequent stages are “Manufacturing”, where the processed materials are made into a module; “Lifetime/UsePhase”; where the module is installed and used to generate electricity; and “End of Life” (EOL), where the module has either failed or degraded beyond use. The “R’s” modify the linear life stages of a PV module.

EOL

Modules reach End of Life in one of two forms: Failure and Degradation. Both functions are applied to each annual cohort of modules such that each year in the model produces modules that enter the End of Life Pathway Options.

Failure

Failure is a random occurrence where the module stops working on site at any point after installation. It follows a probabilistic distribution (usually Weibull in reliability, as defined by T50 and T90 values in years). Early loss of modules are due to manufacturing defects, low quality components, or installation errors and usually represent a higher amount of failures in the first 4 years of deployment. Failure probability also increases as the modules get near their end of lifetime. Failure has the potential for “Repair”.

Degradation

The other mechanism of reaching EOL is Degradation. Modules are expected to last on the field for their working life-time/warranty, and they degrade from their nameplate capacity each year, producing less power. Once the nameplate capacity is reduced below the warranty and/or the project lifetime is complete (whichever comes first) then modules are deemed to be at EOL. Owners might choose to keep the plant going for the merchant tail benefits, or to resell or use the modules in other capacity. These options are captured through the “Reuse” circular economy pathway.

Weibull CDF / PDF - Weibull distributions are often used in product reliability to capture lifetime and failure occurrences. We are controlling our weibull curve parameters by using the T50 and T90 values, which represent the time at which 50% and 90% of the cohort has failed, respectively. For PV panels that are expected to last past their 30 year warranty, T50 and T90 must be bigger than 30, for example T50=35 years and T90=40 years. See the Weibull section for more details.

Weibull CDF / PDF

Weibull distributions are often used in product reliability to capture lifetime and failure occurrences. We are controlling our weibull curve parameters by using the T50 and T90 values, which represent the time at which 50% and 90% of the cohort has failed, respectively. For PV panels that are expected to last past their 30 year warranty, T50 and T90 must be bigger than 30, for example T50=35 years and T90=40 years. See the Weibull section for more details.

EOL Pathway Options

Landfill

Module or material is waste and gets landfilled.

Repair

A module is at EOL (through failure) and an onsite fix to the module defect or problem is possible such that the module is not demounted. If the module is not repaired, it is assumed to be at End of Life.

Refurbish

Module is at EOL (through failure or degradation), and the module is demounted and taken offsite to resolve defects or problems.

Reuse

Module is at EOL (through degradation) and is demounted and removed from the field. Offsite, the module is assessed/tested/recertified and found to be in sufficient working condition to be reinstalled at the same site or on a new site. Could be as a result of a solar PV farm “Repowering”.

Recycle

We are considering Recycle as a Circularity Pathway at two stages in the PV lifetime: during Manufacturing (“pre-consumer”), to salvage material losses due to the manufacturing inefficiencies, and EOL. When a module is at EOL and is not reused, repaired, refurbished, or sent directly to landfill it can be recycled into its constituent materials. These materials can be used to displace virgin materials for the manufacture of new modules or other products. Different quality products are considered from recycling; high quality is used for new modules (same-cycling) or for other products (“o-cycling”); low quality is considered down-cycled into products with less stringent material quality requirements.

“The R’s” - different pathways for circular economy. The idea for the calculator is to capture the material and energy benefits of each of these pathways to help inform of their impact, as they might have different requirements and efficacy. This pathways can be improved through science (for example, improving recycling efficiency of a material), as well as with policy (for example, choosing to support refurbishment research).

2.2 Data

2.2.1 Input Data

At least two files of baselines are needed for the simulation: 1) one file with module parameters throughout the years, for example *baseline_US*, and 2) one or more files with process parameters for each material being analyzed in the module, for example *baseline_glass*. See **baseline_module** and **baseline_material** for more details on input columns needed and their definition.

The baselines folder in the PV_ICE tool provides baseline scenarios for modules in the US and World level, as well as baseline_materials for glass, Silver and Aluminum. For more details on the source of these values, see the documentation on baselines.

baseline_module

year : int Year.

new_Installed_Capacity_[MW]: float New installed PV capacity in MW. Additions of PV modules in nameplate MW peak in the specified year.

module_efficiency_[%]: float Nameplate efficiency of the module. i.e.: 22.3 %.

module_reliability_t50_[years]: float (optional) The number of years after the installation year at which 50% of the cohort of modules will be removed from the field

module_reliability_t90_[years]: float (optional) The number of years after the installation year at which 90% of the cohort of modules will be removed from the field

module_degradation_[%]: float Percentage annual reduction of the module's performance, relative to its nameplate. i.e. 0.5 %.

module_manufacturing_efficiency_[%]: float Efficiency of manufacturing modules (if there are losses of modules and all associated products during production).

module_EOL_collection_losses_[%]: float Percentage of modules collected from the field at end of life for sorting and disposition. i.e. 30%

module_EOL_collected_recycled_[%] : float Percentage of collected end of life modules that are sent to recycling

module_repowering_[%]: float Percentage of modules at EOL from degradation that are repowered.

module_repairing_[%]: float Percentage of modules at EOL from failure that are repaired and go back to functioning in the field.

baseline_material

material_virgin_efficiency_[%]: float Efficiency of mining, extracting, and purifying the material up to the point of entry into the module manufacturing line.

material_masspermodule_[kg]: float Mass of specified material in kilograms per PV module

material_manufacturing_efficiency_[%]: float Efficiency of the manufacturing production line for the material - i.e. how much of the input material is incorporated into the module

material_manufacturing_scrap_recycled_[%]: float The percentage of the scrap generated at the manufacturing facility that is sent to recycling (internal or external).

material_manufacturing_scrap_recycling_efficiency_[%]: float Efficiency of the scrap recycling process

material_manufacturing_scrap_recycled_into_HighQuality_[%]: float Percentage of manufacturing scrap which is recycled into high quality material and used for non-PV module applications externally (open loop)

material_manufacturing_scrap_recycled_into_HighQuality_reused_for_module_manufacturing_[%]: float Percentage of manufacturing scrap material which is recycled and used in the manufacturing of a new module

material_EOL_collected_Recycled_[%]: float Percentage of collected end of life material which is sent to recycling

material_EOL_Recycling_efficiency_[%]: float Efficiency of the end of life recycling process, i.e. percentage of the material that is put through the process that is successfully recycled

EOL_Recycled_Material_into_HighQuality_[%]: float Percentage of collected end of life material recycled into high quality material and used for non-PV modules applications externally (open loop)

EOL_Recycled_HighQuality_Reused_for_Manufacturing_[%]: float Percentage of end of life recycled material that is recycled into high quality material and used in the manufacture of a new module (closed loop)

2.2.2 Output Data

The CEMFC calculates virgin material, landfilled and recycled materials, and the many other steps in the Mass Flow Diagram for each year dynamically. It appends these annual results to the dataframe loaded from `baseline_module` and `baseline_materials`. A description of the output columns is below.

CEMFC Outputs

`module_installedCapacity_[MW]` Summation of all cohorts of installed PV actively in the field in the specified year

`material_installedMass_[kg]`: float Summation of material associated with the total installed capacity in the field in a specified year

`material_EoL_[kg]`: ** `material_EoL_waste_[kg]`: Material in modules from all cohorts that reach that year the end-of-life stage. This value already reflects repowered, reused, or re-manufactured modules.

`material_EoL_CollectionLost`: float Summation of waste material accounting for collection efficiency on an annual basis

`material_EoL_Collected_Recycled`: float Summation of waste material sent for recycling, accounting for collection efficiency losses

`material_EoL_Collected_Landfilled`: float End of life collected material that is landfilled, as opposed to recycled, on an annual basis

`material_EoL_Recycled_Succesfully`: float End of life collected material that is successfully recycled, accounting for recycling process efficiencies, on an annual basis.

`material_EoL_Recycled_Losses_Landfilled`: float Material waste as output by the recycling process, which is landfilled, on an annual basis.

`material_EoL_Recycled_into_HQ`: float Quantity of material which is successfully recycled into high quality material, on an annual basis.

`material_EoL_Recycled_into_Secondary`: float Quantity of material recycled into low quality material, i.e. downcycled, on an annual basis.

`material_EoL_Recycled_HQ_into_Manufacturing`: float Quantity of material which is successfully recycled into high quality material and is used in closed loop for new PV modules, on an annual basis.

`material_EoL_Recycled_HQ_into_OtherUses`: float Quantity of material which is successfully recycled into high quality material and is used in open loop in other applications, on an annual basis.

`material_manufacturing_input`: float Quantity of material required to be input to the manufacturing process, accounting for inefficiencies in the production process, on an annual basis.

`material_manufacturing_scrap`: float Quantity of scrap material generated during the manufacturing process, on an annual basis.

`material_manufacturing_scrap_Recycled`: float Quantity of scrap material from the manufacturing process which is recycled, on an annual basis.

`material_manufacturing_scrap_Landfilled`: float Quantity of scrap material generated during the manufacturing process which is not recycled, on an annual basis.

`material_manufacturing_Scrap_Recycled_Succesfully`: float Quantity of scrap material generated during the manufacturing process which is successfully recycled, accounting for process efficiencies, on an annual basis.

`material_manufacturing_Scrap_Recycled_Losses_Landfilled`: float Quantity of waste material generated and landfilled from the scrap recycling process, on an annual basis.

material_Manufacturing_Recycled_into_HQ: float Quantity of manufacturing scrap material successfully recycled into high quality material, on an annual basis.

material_Manufacturing_Recycled_into_Secondary: float Quantity of manufacturing scrap material successfully recycled into low quality material, i.e. downcycled, on an annual basis.

material_Manufacturing_Recycled_HQ_into_Manufacturing: float Quantity of manufacturing scrap material successfully recycled into high quality material and input to the manufacturing process (closed loop), on an annual basis.

material_Manufacturing_Recycled_HQ_into_OtherUses: float Quantity of manufacturing scrap material successfully recycled into high quality material and used in external applications (open loop), on an annual basis.

material_virgin_stock: float Annual quantity of virgin raw material inputs to the manufacturing process to provide for the manufacturing needs. This value compensates for process and efficiency parameters such as recycled material input.

material_Total_EoL_Landfilled_Waste: float Annual quantity of material sent to the landfill from the end of life, including process inefficiencies and collection losses.

material_Total_Manufacturing_Landfilled_Waste: float Annual quantity of material sent from the manufacturer to the landfill, including process and internal recycling process inefficiencies.

material_Total_Landfilled_Waste: float EoL + Manufacturing. Annual total quantity of material from all processes, manufacturing, recycling, end of life, which are sent to the landfill.

Total_EoL_Recycled_OtherUses: float Annual total quantity of material from all processes, manufacturing, recycling, end of life, which are recycled into external applications, open loop.

2.2.3 Data References

Installed Capacity - Past

Installation data for solar pv installed in the US and globally from several IEA-PVPS T1 reports, Wood MacKenzie Power and Renewables Reports, and LBNL Utility-Scale Solar Reports. Note that installed capacity includes on and off grid, residential, commercial, and utility scale PV. Note that IEA PVPS data (US and global) pre-2009 data is assumed to be all silicon technology.

US Installations 1995 through 2008 taken from (K. Bolcar and K. Ardani, “National Survey Report of PV Power Applications in the United States 2010,” IEA-PVPS, National Survey T1-19:2010, 2010. [Online]. Available: <https://iea-pvps.org/national-survey-reports/>.)

2009 taken from (M. Bolinger, J. Seel, and D. Robson, “Utility-Scale Solar 2019,” LBNL, Dec. 2019. Accessed: Aug. 13, 2020. [Online]. Available: https://emp.lbl.gov/sites/default/files/lbnl_utility_scale_solar_2019_edition_final.pdf.)

2010 through 2019 taken from Wood Mackenzie Power & Renewables PV Forecasts Q2 of 2020 (“US PV Forecasts Q2 2020 Report,” Wood Mackenzie Power & Renewables.)

Other resources consulted include: (F. H. Morse, “IEA PVPS Task 1 1993,” IEA-PVPS, IEA PVPS T1:1993, Mar. 1995. Accessed: Aug. 13, 2020. [Online]. Available: https://iea-pvps.org/wp-content/uploads/2020/01/tr_1993.pdf.) (“IEA PVPS Task 1 1997,” IEA-PVPS, IEA PVPS T1:1997, Mar. 1997. Accessed: Aug. 13, 2020. [Online]. Available: https://iea-pvps.org/wp-content/uploads/2020/01/tr_1995_01.pdf.) (“Trends in Photovoltaic Applications 2019,” IEA-PVPS, IEA PVPS T1-36:2019, Aug. 2019. Accessed: Aug. 12, 2020. [Online]. Available: <https://iea-pvps.org/wp-content/uploads/2020/02/5319-iea-pvps-report-2019-08-lr.pdf>.) IRENA Solar Energy Data (<https://www.irena.org/solar>, and <https://irena.org/Statistics/Download-Data>)

Installed Capacity - Projections

Projection installation data for 2019 through 2050, Installations projected to increase 8.9% compound annual growth rate through 2050 (IRENA, “Future of Solar PV 2019,” IRENA, 2019. Accessed: Apr. 02, 2020. [Online]. Available: https://irena.org/-/media/Files/IRENA/Agency/Publication/2019/Nov/IRENA_Future_of_Solar_PV_2019.pdf.) OR NREL Std Scenarios 2019 mid case

Module Baselines

Degradation rate (in percentage power loss per year) (D. C. Jordan, S. R. Kurtz, K. VanSant, and J. Newmiller, “Compendium of photovoltaic degradation rates,” Progress in Photovoltaics: Research and Applications, vol. 24, no. 7, pp. 978?989, 2016, doi: 10.1002/pip.2744.)

Failure probability data, i.e. T50 and T90, in years (D. C. Jordan, B. Marion, C. Deline, T. Barnes, and M. Bolinger, “PV field reliability status - Analysis of 100 000 solar systems,” Progress in Photovoltaics: Research and Applications, vol. n/a, no. n/a, Feb. 2020, doi: 10.1002/pip.3262.)

Material Baselines

Glass thickness data

The ITRPV Results Reports for 2010 and forward provided glass thickness data, and where report data was missing, reasonable assumptions or interpolations were made. See jupyter journal “Glass per M2 Calculations” for each year’s calculations, and SupportingMaterial Glass_FrontThickness_ITRPV and Module_Conformation_ITRPV files for extracted data (“ITRPV - VDMA.” <https://itrpv.vdma.org/>).

2.3 What’s New

These are new features and improvements of note in each release.

2.3.1 v0.1.0 (12 / 15 / 2020)

First official release. This software is still under development, in particular of the baselines associated for the data of the PV modules and materials.

- Re-structured folders to include Documentation, Versioning and Setup.
- main.py populated with functions developed
- Development journals and examples moved to tutorials folder
- Added Sphinx Documentation
- Added Versioneer
- Added baseline for PV modules installations at US and World level
- Added baseline for materials Silicon & Glass
- Added simple pytest
- Pypi registration

Contributors

- Silvana Ayala (@shirubana)
- Tim Silverman (@silverman)
- Heather Mirletz (@heathermirletz)
- Dirk Jordan (@dirkjordan)
- Chris Deline (@cdeline)

2.4 Installation

PV in Circular Economy Tool can be installed using the pip and conda tools.

Install with:

```
pip install PV_ICE
```

For developer installation, download the repository, navigate to the folder location and install as:

```
pip install -e .
```

PV ICE is compatible with Python 3.5 and above.

2.5 Contributing

This is an open-source software with a license BSD-3.

To start contributing to the PV_ICE tool, please read and follow the instructions in [sign-cla.md](https://github.com/NREL/PV_ICE/blob/master/sign-cla.md)

2.6 API

Modules, methods, classes and attributes are explained here.

Main.py contains the functions to calculate the different quantities of materials in each step of the process. Reffer to the diagram on Package-Overview for the steps considered.

Support functions include Weibull functions for reliability and failure; also, functions to modify baseline values and evaluate sensitivity to the parameters.

2.6.1 Flow Mass Functions

`PV_ICE.main.Simulation` (*name=None, path=None*)

The ScenarioObj top level class is used to work on Circular Economy scenario objects, keep track of filenames, data for module and materials, operations modifying the baselines, etc.

Parameters

- **name** (*text to append to output files*)–
- **nowstr** (*current date/time string*)–

- **path**(*working directory with circular economy results*)-
- __init__** : initialize the object
- _setPath** : change the working directory

PV_ICE.main.**Scenario** (*name, file=None*)

The ScenarioObj top level class is used to work on Circular Economy scenario objects, keep track of filenames, data for module and materials, operations modifying the baselines, etc.

Parameters

- **name**(*text to append to output files*)-
 - **nowstr**(*current date/time string*)-
 - **path**(*working directory with circular economy results*)-
- __init__** : initialize the object
- _setPath** : change the working directory

PV_ICE.main.**Material** (*materialname, file*)

2.6.2 Reliability and Failure Functions

PV_ICE.main.**weibull_params** (*keypoints*)

Returns shape parameter *alpha* and scale parameter *beta* for a Weibull distribution whose CDF passes through the two time: value pairs in *keypoints*

Parameters keypoints (*list*) – Two lists of t50 and 590 values, where t50 is the year since deployment that the cohort has lost 50% of originally installed modules, and t90 is the year since deployment that the cohort has lost 90% of the originally installed modules. These values are used to calculate the shape and scale parameters for the weibull distribution.

Returns

- **alpha** (*float*) – Shape parameter *alpha* for weibull distribution.
- **beta** (*float*) – Scale parameter *beta* for weibull distribution. Often exchanged with *lifetime* like in Irena 2016, beta = 30.

PV_ICE.main.**weibull_cdf** (*alpha, beta*)

Return the CDF for a Weibull distribution having: shape parameter *alpha* scale parameter *beta*

Parameters

- **alpha** (*float*) – Shape parameter *alpha* for weibull distribution.
- **beta** (*float*) – Scale parameter *beta* for weibull distribution. Often exchanged with *lifetime* like in Irena 2016, beta = 30.

2.6.3 Modifying Baseline Functions

`PV_ICE.main.sens_StageEfficiency(df, stage, target_eff=95.0, start_year=None, goal_year=2030, plotflag=False)`

Modifies baseline scenario for evaluating sensitivity to increasing a stage in the lifetime of the module's efficiency. It either increases or decreases from the start year until the goal year the value to the target efficiency by interpolation.

Parameters

- **df** (*dataframe*) – dataframe to be modified
- **stage** (*str*) – Stage that wants to be modified. This can be any of the module or material specified efficiencies, for example: 'MFG_Material_eff', 'mat_MFG_scrap_recycled', 'mat_MFG_scrap_Recycled', 'mat_MFG_scrap_Recycled_into_HQ', 'mat_MFG_scrap_Recycled_into_HQ_Reused4MFG', 'mod_EOL_collection_losses', 'mod_EOL_collected_recycled', 'mat_EOL_Recycling_eff', 'mat_EOL_Recycled_into_HQ', 'mat_EOL_RecycledHQ_Reused4MFG', 'mod_repowering', 'mod_eff', etc.
- **start_year** (*int*) – Year to start modifying the value. This specifies the initial efficiency value that is going to be modified. If None is passed, current year is used.
- **target_eff** (*flat*) – target eff value in percentage to be reached. i.e. 95.0 %.
- **goal_year** (*int*) – year by which target efficiency will be reached. i.e. 2030. Must be higher than current year.

Returns **df** (*dataframe*) – modified dataframe

`PV_ICE.main.sens_StageImprovement(df, stage, improvement=1.3, start_year=None)`

Modifies baseline scenario for evaluating sensitivity of lifetime parameter. t50 and t90 reliability years get incresed by *improvement* parameter starting the *year_increase* year specified.

Parameters

- **df** (*dataframe*) – dataframe to be modified
- **stage** (*str*) – Stage that wants to be modified. This can be any of the module or material specified values, for example: 'MFG_Material_eff', 'mat_MFG_scrap_recycled', 'mat_MFG_scrap_Recycled', 'mat_MFG_scrap_Recycled_into_HQ', 'mat_MFG_scrap_Recycled_into_HQ_Reused4MFG', 'mod_EOL_collection_losses', 'mod_EOL_collected_recycled', 'mat_EOL_Recycling_eff', 'mat_EOL_Recycled_into_HQ', 'mat_EOL_RecycledHQ_Reused4MFG', 'mod_repowering', 'mod_eff', etc.
- **improvement** (*decimal*) – Percent increase in decimal (i.e. "1.3" for 30% increase in value) or percent decrease (i.e. "0.3") relative to values in df.
- **start_year** – the year at which the improvement occurs

Returns **df** (*dataframe*) – dataframe of expected module lifetime increased or decreased at specified year

2.6.4 LCA Functions

`PV_ICE.main.calculateLCA(PVarea, modified_impacts=None, printflag=False)`

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`

PYTHON MODULE INDEX

p

PV_ICE, [12](#)
PV_ICE.main, [12](#)

INDEX

C

`calculateLCA()` (*in module PV_ICE.main*), 15

M

`Material()` (*in module PV_ICE.main*), 13

`module`

`PV_ICE`, 12

`PV_ICE.main`, 12

P

`PV_ICE`

`module`, 12

`PV_ICE.main`

`module`, 12

S

`Scenario()` (*in module PV_ICE.main*), 13

`sens_StageEfficiency()` (*in module PV_ICE.main*), 14

`sens_StageImprovement()` (*in module PV_ICE.main*), 14

`Simulation()` (*in module PV_ICE.main*), 12

W

`weibull_cdf()` (*in module PV_ICE.main*), 13

`weibull_params()` (*in module PV_ICE.main*), 13