Package 'SunsVoc'

May 7, 2026

Type Package

Title Constructing Suns-Voc from Outdoor Time-Series I-V Curves

Version 0.1.2

Maintainer Tyler J. Burleyson <tjb152@case.edu>

Description Suns-Voc (or Isc-Voc) curves can provide the current-voltage (I-V) characteristics of the diode of photovoltaic cells without the effect of series resistance.
Here, Suns-Voc curves can be constructed with outdoor time-series I-V curves [1,2,3] of full-size photovoltaic (PV) modules instead of having to be measured in the lab.
Time series of four different power loss modes can be calculated based on obtained Isc-Voc curves.
This material is based upon work supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under Solar Energy Technologies Office (SETO) Agreement Number DE-EE0008172.
Jennifer L. Braid is supported by the U.S. Department of Energy (DOE) Office of Energy Efficiency and Renewable Energy administered by the Oak Ridge Institute for Science and Education (ORISE) for the DOE.
ORISE is managed by Oak Ridge Associated Universities (ORAU) under DOE contract number DE-SC0014664.
[1] Wang, M. et al, 2018.
<doi:10.1109/PVSC.2018.8547772>.
[2] Walters et al, 2018
<doi:10.1109/PVSC.2018.8548187>.
[3] Guo, S. et al, 2016.
<doi:10.1117/12.2236939>.

License BSD_3_clause + file LICENSE

Encoding UTF-8

LazyData true

LazyDataCompression xz

RoxygenNote 7.1.1

Depends R (>= 3.5.0)

Imports ddiv, magrittr, stringr, dplyr, purrr, data.table, rlang

Suggests testthat (>= 2.1.0), knitr, rmarkdown, ggplot2

VignetteBuilder knitr

NeedsCompilation no

Author Menghong Wang [aut] (ORCID: <https://orcid.org/0000-0001-7349-699X>),
Tyler J. Burleyson [aut, cre] (ORCID:
<https://orcid.org/0000-0002-6356-5354>),
Jiqi Liu [aut] (ORCID: <https://orcid.org/0000-0003-2016-4160>),
Alan J. Curran [aut] (ORCID: <https://orcid.org/0000-0002-4505-8359>),
Abdulkerim Gok [aut, ctb] (ORCID:
<https://orcid.org/0000-0003-3433-7106>),
Eric J. Schneller [aut] (ORCID:
<https://orcid.org/0000-0002-2104-0066>),
Kristopher O. Davis [aut] (ORCID:
<https://orcid.org/0000-0002-5772-6254>),
Jennifer L. Braid [aut] (ORCID:
<https://orcid.org/0000-0002-0677-7756>),
Roger H. French [aut, cph] (ORCID:
<https://orcid.org/0000-0002-6162-0532>)

Repository CRAN

Date/Publication 2021-04-30 09:30:06 UTC

Contents

`char_to_df` *Convert From Hbase Char String to Dataframe*

Description

This function parses the I-V curve string and creates an additional dataframe with current and voltage columns from it.

Usage

```
char_to_df(str)
```

Arguments

str	The character string to be converted to a Dataframe.

Value

Dataframe containing IV curve data.

Examples

```
char_to_df(df_wbw$ivdf[1])
```

`df2chr` *Convert From Dataframe to Hbase Char String*

Description

The companion function to this one, `char_to_df`, parses this string and creates an additional dataframe from it. This function works the other way, converting that dataframe back to a character string.

Usage

```
df2chr(df)
```

Arguments

df	The dataframe, typically named "ivdf", to be converted.

Value

Returns a character string representing an IV curve.

Examples

```
df2chr(char_to_df(df_wbw$ivdf[1]))
```

df_wbw *1 Year of raw outdoor IV curve data.*

Description

This dataframe contains 1 year of anonymous raw outdoor time-series IV curve data.

Usage

```
df_wbw
```

Format

Dataframe with 4140 objects in 10 variables:

tmst A local PosixCT Timestamp

ivdf IV Dataframe

modt Module Temperature

poa Plane of array (POA) irradiance

isc Extracted Short Circuit Current by ddiv

voc Extracted Open Current Voltage by ddiv

rs Extracted Rs (series resistance) by ddiv

pmp Extracted Maximum Power by ddiv

imp Extracted Current at Maximum Power Point

vmp Extracted Voltage at Maximum Power Point

isc.1sun *Obtain 1-sun Isc Value*

Description

This function is used internally by IVxbyx and others for the calculation of 1-sun Isc values based on Isc and Irradiance measurements.

Usage

```
isc.1sun(isc, Irrad)
```

Arguments

isc	Isc values
Irrad	Irradiance values

Value

Returns a calculated Isc value at 1-sun Irradiance.

Examples

```
isc_1sun <- isc.1sun(df_wbw$isc, df_wbw$poa)
```

IVXbyX *Create x-by-x Psuedo-IV Curves*

Description

This function moves through IV curve data x-by-x, generating psuedo IV curves and binding them together into the output.

Usage

```
IVXbyX(df, corr_temp = "median", N_c)
```

Arguments

df	Dataframe containing IV data. Typically, a raw dataframe after filtering by current accuracy based on plot of irradiance vs. temperature. Within this package environment, use read_df_raw to filter the dataset and generate period index.
corr_temp	The temperature for all I-V features to standardize to. Pass the string "median" in order for the function to automatically calculate the median module temperature at 1 sun irradiance and use it, or input a number directly, the temperature unit should be Celsius.
N_c	Number of cells in series; the total number of cells in the system.

Value

Psuedo-IV Curve data with features extracted and evaluation parameters of fitting grouped by time periods of set length.

Examples

```
df <- read_df_raw(df_wbw,0.02,7)
df_full <- IVXbyX(df, corr_temp = "median", 60)
```

median_temp *Calculate Median Temperature*

Description

This function calculates the median module temperature throughout the data to be used in corrections.

Usage

```
median_temp(df)
```

Arguments

df	Dataframe containing timeseries irradiance (column name must be poa) and module temperature (column name must be modt) in unit of Celsius.

Value

Returns an integer value of median reported module temperature of the data.

Examples

```
T_corr <- median_temp(df_wbw)
```

power_loss_phys *Physical Model Power Loss Modes of Single period*

Description

This function decouples power loss to different loss modes, specifically current mismatch, recombination, uniform current loss, and Rs loss, from the changes in IV features for a given period, used internally for power_loss_phys_bat function.

Usage

```
power_loss_phys(init_piv, init_isc1sun, init_prs, degr_row)
```

Arguments

init_piv	Dataframe containing current andvolatge psuedo-IV curves of initial period, calculated from the the result of p_iv.week function applied to timeseries dataframe of initial period.
init_isc1sun	1 sun isc generated from the p_iv.week output from the initial period dataframe.
init_prs	extracted rs from the psuedo-IV curve of the initial period.
degr_row	one row of output from IVXbyX function.

Value

dataframe containing information about power loss due to various power loss modes for one given period

Examples

```
df <- read_df_raw(df_wbw,0.02,7)
df_init <- select_init_df(df, days = 21)
init <- p_iv.week(df_init, temp = 30, N_c = 60)
init_piv <- data.frame(V = init$voc_corr, I = init$isc_1sun - init$isc)
init_piv <- dplyr::arrange(init_piv, .data$V)
init_pivf <- ddiv::IVfeature(I = init_piv$I, V = init_piv$V, crtvalb = 0.06)
init_prs <- init_pivf$Rs
init_isc1sun <- init$isc_1sun
df_full <- IVXbyX(df, corr_temp = 30, 60)
power_loss_phys(init_piv,init_isc1sun, init_prs,df_full[1,])
```

---

power_loss_phys_bat *Physical Model Power Loss Modes*

Description

This function decouples power loss to different loss modes, specifically current mismatch, recombination, uniform current loss, and Rs loss, from the changes in IV features.

Usage

```
power_loss_phys_bat(df_iv, init_df, corr_T = 40, N_c, ddiv_param = 0.06)
```

Arguments

df_iv	Dataframe containing psuedo-IV curves. Generate with IVXbyX.
init_df	Dataframe containing first several weeks of real-world IV data. Generate with select_init_df
corr_T	The temperature from which to create the correction factor. Pass only numeric values.
N_c	Number of cells in series; The total number of cells in the system.
ddiv_param	Parameter passed to ddiv::IVfeature for 'crtvalb'

Value

dataframe containing information about power loss due to various power loss modes

Examples

```
df <- read_df_raw(df_wbw,0.02,7)
df_init <- select_init_df(df, days = 21)
df_full <- IVXbyX(df, corr_temp = 30, 60)
res <- power_loss_phys_bat(df_full, df_init, corr_T = 30, N_c = 60)
```

`p_iv.week` *Obtain Psuedo IV Curve and other predicted IV features at reference conditions for a given week, used internally in IVxbyx function.*

Description

Obtain Psuedo IV Curve and other predicted IV features at reference conditions for a given week, used internally in IVxbyx function.

Usage

```
p_iv.week(df, temp, N_c, isc_1sun = NULL)
```

Arguments

`df`	A dataframe containing timeseries I-V features of one period.
`temp`	The reference module temperature to correct the Psuedo IV curve to be, unit of the temperature should be Celsius.
`N_c`	Number of cells in series. Equal to the total number of cells in the system.
`isc_1sun`	(optional) Input an Isc 1-Sun value manually. Leave NULL to have one generated from the dataframe.

Value

Psuedo-IV Curve data with features extracted and evaluation parameters of fitting grouped for a single, given period.

Examples

```
df <- read_df_raw(df_wbw,0.02,7)
df_slice <- dplyr::filter(df, df$n_period == 1)
# Check that this has enough data! needs more than 10 rows to be meaningful
nrow(df_slice)
# needs median temperature
temp <- median_temp(df_wbw)
res <- p_iv.week(df_slice, temp = temp, N_c = 60)
```

read_df_raw *Read in Raw Data from Dataframe*

Description

Given an imported datafram of Hbase-Formatted IV curve data, this function reads in the data, filters missing temperature data, and checks for a nonzero maximum power point, in case of power loss. It resets the timestamps based on the minimum timestamp, and filters Isc values for the tracer's accuracy. Finally, a n_period counter is added to the dataframe.

Usage

```
read_df_raw(df, tracer_accuracy, t_period)
```

Arguments

df	dataframe; the IV curve data to be filtered
tracer_accuracy	The accuracy of the IV tracer used. See the device's manual to find the exact value at which Isc readings are no longer accurate.
t_period	Data period for the Psuedo-IV curves. Addded as a column to the dataframe based on the timestamp. Use units of days, i.e. daily periods should have t_period 1, weekly periods should have t_period 7, etc.

Value

df_raw, a dataframe containing the raw IV curve data

Examples

```
df_test <- read_df_raw(df_wbw, 1, 7) # Weekly periods
df_test <- read_df_raw(df_wbw, 1, 1) # Daily periods
```

select_init_df *Generate Initial Dataframe for power_loss_bat*

Description

The power_loss_bat function makes use of a dataframe containing the first several days of psuedo-IV curves. This function creates that dataframe for the user from the raw dataframe containing IV Curve measurement data.

Usage

```
select_init_df(df_raw, days = 21)
```

Arguments

df_raw	The raw dataframe containing request input columns the same as the example df_wbw dataset and the column of day, which starts from 1 and column of n_period, which is the index of the period, decided by how many days to be grouped as one period, the column of day and n_period can be generated from the function read_df_raw
days	The number of initial days to subsample. Default: 21

Value

Subset of input dataframe within the first several days decided by the input parameter "days".

Examples

```
df <- read_df_raw(df_wbw, 1, 7)
df_init <- select_init_df(df, days = 21)
```

---

voc.corr *Correct Voc to certain reference conditions, used internally in piv_iv_week function.*

Description

This method uses a physical model to correct Voc to certain reference conditions. Since indoor Suns-Voc is conducted at a steady 25C, a correction on the outdoor readings is necessary for meaningful comparison.

Usage

```
voc.corr(df)
```

Arguments

df	A dataframe time series data with columns of voc, lnSun and modt, the dataframe is converted from the request input dataframe like df_wbw by the piv_iv_week function before using voc_corr function.

Value

returns a list object of Voc model.

Examples

```
df <- read_df_raw(df_wbw,0.02,7)
# subset data to first period
df_slice <- dplyr::filter(df, df$n_period == 1)
N_c <- 60 # true of the example data. N_c is the number of cells in series

df_slice <- dplyr::mutate(df_slice, T_K = .data$modt + 273.15,
lnSun = 1.38e-23/1.6e-19 * N_c * .data$T_K * log(.data$isc),
isc2 = .data$isc^2,
T_lnIsc2 = .data$T_K * .data$lnSun * log(.data$isc),
rs = as.numeric(.data$rs),
expVoc = exp(-.data$voc/.data$T_K) / .data$isc,
I0 = (N_c * 1.38e-23 * .data$T_K) / (1.6e-19 * .data$isc))
voc_mod <- voc.corr(df_slice)
```

Index