

# Package ‘SunsVoc’

June 29, 2020

**Type** Package

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

**Version** 0.1.0

**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 Suns-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

**RoxygenNote** 7.1.0

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

**Repository** CRAN

**Date/Publication** 2020-06-29 10:40:03 UTC

## R topics documented:

char_to_df . . . . .	2
df2chr . . . . .	3
df_wbw . . . . .	4
filter_df_raw . . . . .	5
isc.lsun . . . . .	6
IVcorr . . . . .	6
IVcorr_full . . . . .	7
IVXbyX . . . . .	7
median_temp . . . . .	8
power_loss_phys_bat . . . . .	8
p_iv.week . . . . .	9
read_df_raw . . . . .	10
read_df_raw_from_csv . . . . .	11
select_init_df . . . . .	12
voc.corr . . . . .	12

**Index** **14**

---

char_to_df	<i>Convert From Hbase Char String to Dataframe</i>
------------	--

---

## Description

Hbase saves some data automatically as a long string in a column of a dataframe. This function parses this string and creates an additional dataframe 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**

```
# generate Psuedo-IV Curves
df_full <- IVXbyX(df_wbw, corr_temp = "median", 4)

# subset a single row
degr_row <- df_full[1,]

degr_piv <- char_to_df(degr_row$piv)
```

---

df2chr

*Convert From Dataframe to Hbase Char String*

---

**Description**

Hbase saves some data automatically as a long string in a column of a dataframe. 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**

```
# generate subset of first period
df_slice <- dplyr::filter(df_wbw, df_wbw$n_period ==1)

temp <- median_temp(df_wbw)
res <- p_iv.week(df_slice, temp = temp, N_c = 4)

isc_1sun <- res[[3]]
df_piv <- data.frame(voc_corr = res[[1]], isc_corr = isc_1sun - res[[2]])

chr_piv <- df2chr(df_piv)
```

---

df\_wbw

*1 Year of raw outdoor IV curve data.*

---

### Description

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

### Usage

df\_wbw

### Format

Dataframe with 28649 objects in 20 variables:

**tmst** A local PosixCT Timestamp

**vocc** Open Current Voltage

**ghir** Global Horizontal Irradiance

**ivdf** IV Dataframe

**imxp** Current Max Power

**pmpp** Maximum Power Point

**ishc** Short Circuit Current

**modt** Module Temperature

**vmpp** Voltage Max Power Point

**ffff** Fill Factor

**tmst\_1**

**poa\_ratio** Ratio between POA and GHI

**poay** POA irradiance

**eisc** Extracted Isc (short circuit current)

**ersh** Extracted Rsh (shunt resistance)

**evoc** Extracted Voc (open voltage voltage)

**exrs** Extracted Rs (series resistance)

**epmp** Extracted Pmp (maximum power point)

**eimp** Extracted Imp (max power current)

**evmp** Extracted Vmp (max power voltage)

**exff** Extracted Fill Factor

**day** the day, with 1 being the first day

**n\_period** The period this belongs to. Created by grouping specified in method

---

filter_df_raw	<i>Perform Temperature Filters on Raw Data</i>
---------------	--

---

### Description

Filters a raw dataframe of IV curve data according to specified irradiance levels and temperature range. The function removes data below the low irradiance and above the high irradiance thresholds, calculates mean temperature of the remaining data, and then removes data outside of the temp range around mean temperature (Celsius).

### Usage

```
filter_df_raw(  
  df_raw,  
  low_irrad_thresh = 950,  
  high_irrad_thresh = 1050,  
  temp_range = 3  
)
```

### Arguments

df_raw	A dataframe containing IV curve data.
low_irrad_thresh	A lower bound for irradiance. Default = 995
high_irrad_thresh	An upper bound for irradiance. Default = 1005
temp_range	An acceptable range of temperature from the mean. The filter will allow this value above and below the mean temperature, in Celcius.

### Value

df\_reduced, a necessary dataframe for IV Curve Translation (see [IVcorr\\_full](#)) This dataframe has been filtered as described above.

### Examples

```
df_reduced <- filter_df_raw(df_wbw)
```

---

isc.1sun	<i>Obtain 1-sun Isc Value</i>
----------	-------------------------------

---

**Description**

This function is used internally by IVwbyw 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	Ir radiance values

**Value**

Returns a calculated Isc value at 1-sun of Irradiance.

**Examples**

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

---

IVcorr	<i>Correct individual IV curve by 1-sun Voc and Isc</i>
--------	---

---

**Description**

This method produces a dataframe with corrected values that can be passed to power\_loss\_bat in order to calculate power loss from IV curve data. This function is a component of [IVcorr\\_full](#), which runs this function on an entire dataframe in a loop

**Usage**

```
IVcorr(iv_row)
```

**Arguments**

iv_row	A single row of psuedo-IV curve data.
--------	---------------------------------------

**Value**

Returns a single row of corrected psuedo-IV curve data.

---

IVcorr_full	<i>Correct psuedo-IV curves by 1-sun Voc and Isc</i>
-------------	--

---

**Description**

This method produces a dataframe with corrected IV curve values that can be passed to `power_loss_bat` in order to calculate power loss from IV curve data.

**Usage**

```
IVcorr_full(df_red, df_piv)
```

**Arguments**

<code>df_red</code>	A filtered dataframe containing IV curve data. Obtain with <a href="#">filter_df_raw</a>
<code>df_piv</code>	A dataframe of psuedo-IV curve data. Obtain with <a href="#">read_df_raw</a> or <a href="#">read_df_raw_from_csv</a> .

**Value**

Returns A dataframe with corrected IV curves.

---

IVXbyX	<i>Create x-by-x Psuedo-IV Curves</i>
--------	---------------------------------------

---

**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**

<code>df</code>	Dataframe containing IV data. Typically, a raw dataframe prior to filtering for irradiance/temperature. Within this package environment, use <a href="#">read_df_raw</a> to generate this from a .csv
<code>corr_temp</code>	The temperature from which to create the correction factor. Pass the string "median" in order for the function to automatically calculate the median module temperature at 1 sun irradiance and use it.
<code>N_c</code>	Number of Cells in series; the total number of cells in the system.

**Value**

Psuedo-IV Curve data (dataframe) grouped from time periods of set length.

**Examples**

```
df_full <- IVXbyX(df_wbw, corr_temp = "median", 4)
```

---

median_temp	<i>Calculate Median Temperature</i>
-------------	-------------------------------------

---

**Description**

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

**Usage**

```
median_temp(df)
```

**Arguments**

df                      Dataframe containing IV Curve data (make sure it includes temperature).

**Value**

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

**Examples**

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

---

power_loss_phys_bat	<i>Physical Model Power Loss Modes</i>
---------------------	--

---

**Description**

This function decouples power loss to different loss modes, specifically I 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 = "median", N_c)
```



**Arguments**

df_iv	Dataframe containing psuedo-IV curves. Generate with <a href="#">IVXbyX</a> .
init_df	Dataframe containing first several weeks of real-world IV data. Generate with <a href="#">select_init_df</a>
corr_T	The temperature from which to create the correction factor. Pass the string "mean" in order for the function to automatically calculate the mean temperature and use it.
N_c	Number of cells in series; The total number of cells in the system.

**Value**

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

**Examples**

```
df_full <- IVXbyX(df_wbw, corr_temp = "median", 4)
df_init <- select_init_df(df_wbw, days = 21)
T_corr <- median_temp(df_wbw)
res <- power_loss_phys_bat(df_full, df_init, corr_T = T_corr, N_c = 4);
```

---

p\_iv.week

*Obtain Psuedo IV Curve for given week*


---

**Description**

Obtain Psuedo IV Curve for given week

**Usage**

```
p_iv.week(df, temp, alpha = 0.5, N_c, isc_1sun = NULL)
```

**Arguments**

df	A dataframe containing IV curve data.
temp	The temperature at which the data was obtained
alpha	Parameter for "emp" and "val" methods.
N_c	Number of cells in series. Equal to the total number of cells in the system.
isc_1sun	(optional) Input a Isc 1-Sun value manually. Leave NULL to have one generated from the dataframe.

**Value**

Pseudo-IV Curve data (dataframe) for a single, given week.

**Examples**

```
df_slice <- dplyr::filter(df_wbw, df_wbw$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 = 4)
```

---

read\_df\_raw

*Read in Raw Data from Dataframe*


---

**Description**

Given an imported dataframe 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 Pseudo-IV curves. Added 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
```

---

read\_df\_raw\_from\_csv *Read in Raw Data*

---

### Description

Given the file location of a .csv 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 I<sub>sc</sub> values for the tracer's accuracy. Finally, a n\_period counter is added to the dataframe.

### Usage

```
read_df_raw_from_csv(file_loc, tracer_accuracy, t_period)
```

### Arguments

file_loc	String; The location of the file, including the file name.
tracer_accuracy	The accuracy of the IV tracer used. See the device's manual to find the exact value at which I <sub>sc</sub> readings are no longer accurate.
t_period	Data period for the Psuedo-IV curves. Added 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_subset <- head(df_wbw, 3)

tf <- tempfile()
write.csv(df_subset, tf)

# uses 0 as tracer accuracy to avoid filtering
df_test <- read_df_raw_from_csv(tf, 0, 7)
```

---

select_init_df	<i>Generate Initial Dataframe for power_loss_bat</i>
----------------	--

---

### 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 IV Curve measurement data.
days	The number of initial days to subsample. Default: 21

### Value

Subset of input dataframe.

### Examples

```
df_init <- select_init_df(df_wbw, days = 21)
```

---

voc.corr	<i>Correct Voc for Module Temperature</i>
----------	---

---

### Description

This method uses a physical model to correct Voc for the module temperature. Since indoor Sun-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 of IV curve data, containing an InSun column generated by the <a href="#">p_iv.week</a> function.
----	---

**Value**

returns a list object of modeled Voc.

**Examples**

```
# subset data to first period
df_slice <- dplyr::filter(df_wbw, df_wbw$n_period == 1)
N_c <- 4 # 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$ishc),
                        T_lnIsc2 = .data$lnSun * log(.data$ishc),
                        exrs = as.numeric(.data$exrs),
                        expVoc = exp(-.data$vocc/.data$T_K) / .data$ishc,
                        I0 = (N_c * 1.38e-23 * .data$T_K) / (1.6e-19 * .data$ishc))

voc_mod <- voc.corr(df_slice)
```

# Index

## \*Topic **datasets**

df\_wbw, 4

char\_to\_df, 2, 3

df2chr, 3

df\_wbw, 4

filter\_df\_raw, 5, 7

isc.1sun, 6

IVcorr, 6

IVcorr\_full, 5, 6, 7

IVXbyX, 7, 9

median\_temp, 8

p\_iv.week, 9, 12

power\_loss\_phys\_bat, 8

read\_df\_raw, 7, 10

read\_df\_raw\_from\_csv, 7, 11

select\_init\_df, 9, 12

voc.corr, 12