

# Package ‘swt’

December 4, 2025

**Type** Package

**Title** Swisstransplant R Package

**Version** 0.3

**Description** This R package provides tools for data analysis and visualization by Swisstransplant--the national organisation for organ donation and transplantation in Switzerland.

**Imports** ggplot2, grDevices, utils, hms, data.table, testit, segmented, lubridate, rms

**LazyData** true

**License** file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.3.3

**Depends** R (>= 3.5.0)

## Contents

count_perc . . . . .	2
date2num . . . . .	2
egfr_ckd_epi . . . . .	3
egfr_schwartz . . . . .	3
exam.device . . . . .	4
exam.organ . . . . .	5
exam.sumstats . . . . .	6
exam.timeseries . . . . .	7
fmt_hla . . . . .	7
get_days_in_year . . . . .	8
hla_mismatch . . . . .	8
hla_parse . . . . .	9
kidmo . . . . .	10
kidmo_hr2rank . . . . .	10
kidmo_model . . . . .	11
lifeport_d2 . . . . .	11
lifeport_process . . . . .	12
lifeport_read . . . . .	12
lifeport_sumstats . . . . .	13
mean_sd . . . . .	13
median_iqr . . . . .	14
miss_perc . . . . .	14

nearest . . . . .	15
num2date . . . . .	15
optn_kdri . . . . .	16
print.kidmo . . . . .	17
swt_colors . . . . .	17
swt_skeleton . . . . .	17
swt_style . . . . .	18
tidy_missing . . . . .	18
tidy_pvalues . . . . .	19
tidy_rmsfit . . . . .	19
uk_dcd_score . . . . .	20
uk_kdri . . . . .	20

<b>Index</b>	<b>22</b>
--------------	-----------

---

count_perc	<i>Returns count and percentage</i>
------------	-------------------------------------

---

### Description

Helper function for tidy formatting.

### Usage

```
count_perc(x, count.na = TRUE, d2 = 1)
```

### Arguments

x	logical vector
count.na	count NAs in denominator
d2	number of digits

### Value

character object

---

date2num	<i>Convert date to Excel numeric days</i>
----------	---

---

### Description

Convert POSIXct data type (date/time) to Excel days since origin.

### Usage

```
date2num(dates)
```

### Arguments

dates	character string in the form of YYYY-mm-dd
-------	--

**Value**

number of days

---

egfr_ckd_epi	<i>CKD-EPI Creatinine Equation (2021)</i>
--------------	---

---

**Description**

Calculates eGFR according to the 2021 formula.

**Usage**

```
egfr_ckd_epi(SCr, age, sex, units = "SI")
```

**Arguments**

SCr	serum creatinine in mg/dL (US) or umol/L (S)
age	age in years
sex	either "F" for female, or "M" for male
units	unit for SCr, either "SI" (umol/L; default) or "US" (mg/dL)

**Details**

See equation and references at <https://www.kidney.org/ckd-epi-creatinine-equation-2021>.

**Value**

eGFR in mL/min/1.73m<sup>2</sup>

---

egfr_schwartz	<i>Revised Schwartz Equation (2009)</i>
---------------	---

---

**Description**

Calculates eGFR for pediatric patients.

**Usage**

```
egfr_schwartz(SCr, height, units = "SI")
```

**Arguments**

SCr	serum creatinine in mg/dL (US) or umol/L (S)
height	height in cm
units	unit for SCr, either "SI" (umol/L; default) or "US" (mg/dL)

**Details**

See equation and examples at <https://www.mdcalc.com/calc/10008/revised-schwartz-equation-glomerular-filtration-rate> evidence.

**Value**

eGFR in mL/min/1.73m<sup>2</sup>

---

exam.device	<i>EXAM device data</i>
-------------	-------------------------

---

**Description**

Example data form the LifePort kidney transporter.

**Usage**

exam.device

**Format**

A data frame with self explaining variable names:

**SerialNumber** serial number of the device

**Type** type note

**SubType** subtype note

**UnitID** name given to the device

**FirmwareVersion** firmware version

**FileID** file id

**StartTime** start date and time of machine

**DataState** data state, if it is complete

**HasGaps** whether data has gaps

**Runtime** run time of the machine

**StopTime** time and date of machine stop

**Filename** file name#'

**Source**

<https://data.swisstransplant.org/>

---

exam.organ*EXAM organ data*

---

**Description**

Example data form the LifePort kidney transporter.

**Usage**

exam.organ

**Format**

A data frame with self explaining variable names:

**OrganID** identifier entered into the machine

**KidneySide** left or right kidney

**BloodType** self explaining

**CrossClampTime.Date** self explaining

**CrossClampTimezone** self explaining

**TotalIschemicTime** self explaining

**PerfusateLot** self explaining

**PerfusateExpirationDate** self explaining

**PerfusateUsed** self explaining

**Cannula** self explaining

**CannulaExpirationDate** self explaining

**CassetteLot.** self explaining

**CasetteExpirationDate** self explaining

**ID** self explaining

**DonorID** self explaining

**Source**

<https://data.swisstransplant.org/>

---

exam.sumstats

*EXAM summary statistics*


---

## Description

Example data form the LifePort kidney transporter.

## Usage

```
exam.sumstats
```

## Format

A data frame with self explaining variable names:

**perfusion.dur** perfusion duration in minutes  
**perfusion.dur.str** perfusion duration in HH:MM:SS  
**systolicPressure.md** self explaining  
**systolicPressure.mean** self explaining  
**diastolicPressure.mean** self explaining  
**flowRate.mean** self explaining  
**organResistance.mean** self explaining  
**organResistance.sd** self explaining  
**organResistance.x1** self explaining  
**organResistance.y1** self explaining  
**organResistance.x2** self explaining  
**organResistance.y2** self explaining  
**organResistance.delta** self explaining  
**organResistance.slope** self explaining  
**iceContainerTemperature.mean** self explaining  
**iceContainerTemperature.sd** self explaining  
**iceContainerTemperature.minAbove** self explaining  
**iceContainerTemperature.minAbove.str** self explaining  
**infuseTemperature.mean** self explaining  
**infuseTemperature.sd** self explaining  
**infuseTemperature.start** self explaining  
**infuseTemperature.minAbove** self explaining  
**infuseTemperature.minAbove.str** self explaining  
**D2perf** self explaining  
**Pperf** self explaining  
**D2temp** self explaining  
**Ptemp** self explaining

## Source

<https://data.swisstransplant.org/>

---

exam.timeseries	<i>EXAM time series data</i>
-----------------	------------------------------

---

**Description**

Example data form the LifePort kidney transporter.

**Usage**

```
exam.timeseries
```

**Format**

A list of data frames with time series data:

**SerialNumber** serial number of the device

**FlowRate** flow rate ...

**Source**

<https://data.swisstransplant.org/>

---

fmt_hla	<i>Format HLA</i>
---------	-------------------

---

**Description**

Helper function to format strings for broads, e.g. A(10) becomes A10 and A becomes NA.

**Usage**

```
fmt_hla(v_char)
```

**Arguments**

v\_char                  character vector

**Value**

formatted character vector

---

get_days_in_year	<i>Get the number of days in a year</i>
------------------	---

---

**Description**

Helper function useful in survival analysis to convert event times.

**Usage**

```
get_days_in_year()
```

**Value**

number of days

---

hla_mismatch	<i>Calculates HLA mismatches.</i>
--------------	-----------------------------------

---

**Description**

The function calculates HLA mismatches for SOAS data.

**Usage**

```
hla_mismatch(
  D.A1,
  D.A2,
  D.B1,
  D.B2,
  D.DR1,
  D.DR2,
  R.A1,
  R.A2,
  R.B1,
  R.B2,
  R.DR1,
  R.DR2
)
```

**Arguments**

D.A1	donor HLA Antigen on allele 1 locus A
D.A2	donor HLA Antigen on allele 2 locus A
D.B1	donor HLA Antigen on allele 1 locus B
D.B2	donor HLA Antigen on allele 2 locus B
D.DR1	donor HLA Antigen on allele 1 locus DR
D.DR2	donor HLA Antigen on allele 2 locus DR
R.A1	recipient HLA Antigen on allele 1 locus A



R.A2	recipient HLA Antigen on allele 2 locus A
R.B1	recipient HLA Antigen on allele 1 locus B
R.B2	recipient HLA Antigen on allele 2 locus B
R.DR1	recipient HLA Antigen on allele 1 locus DR
R.DR2	recipient HLA Antigen on allele 2 locus DR

### Details

The serological nomenclature in SOAS as follows: L[p, q] with L is the locus A B or DR, p and q are the two alleles of the locus L, and the convention is  $p \leq q$ . The case  $p \neq q$  is known as heterozygote, A[2, 25]. Homozygote, if  $p = q$ , such as in DR[11,11].

The HLA-matching process has to handle broad and splits. Two alleles p and r on the same locus L match if they are equal or if one of the allele is the broad of the other allele. Two different splits of same broad do not match. To calculate mismatch, we look up donor antigens and match them in the recipient. In other words, how many unknown antigens are transferred to the donor?

### Value

data frame with mismatch information

---

hla_parse	<i>Parse HLA data</i>
-----------	-----------------------

---

### Description

Parser to convert unstructured SOAS HLA information into structured data.

### Usage

```
hla_parse(D_HLA, R_HLA)
```

### Arguments

D_HLA	donor HLA antigens; character string from SOAS variable D HLA Ag.
R_HLA	recipient HLA antigens; character string from SOAS variable R HLA Ag.

### Value

data frame with structured HLA information

---

kidmo	<i>KIDMO Score</i>
-------	--------------------

---

**Description**

Calculates the KIDMO Score.

**Usage**

```
kidmo(
  D_age = 55,
  D_deathcause = "cerebral hemorrhage",
  D_diabetes = FALSE,
  D_hypertension = FALSE,
  R_age = 57,
  R_retpx = FALSE,
  R_tpxyear = 2026,
  times = c(2, 5),
  newdata = NULL
)
```

**Arguments**

D_age	donor age in years
D_deathcause	donor cause of death (cerebral hemorrhage, anoxia, or others)
D_diabetes	donor history of diabetes (binary)
D_hypertension	donor history of hypertension (binary)
R_age	recipient age in years
R_retpx	recipient listed for retransplant (binary)
R_tpxyear	recipient year of transplant (continuous)
times	time points for predictions, in years
newdata	data frame with variables (requires correct variable names)

**Value**

KIDMO Score

---

kidmo_hr2rank	<i>KIDMO rank</i>
---------------	-------------------

---

**Description**

Conversion of (unscaled) hazard ratio into percentile rank.

**Usage**

```
kidmo_hr2rank(hr)
```

**Arguments**

hr                      hazard ratio

**Value**

percentile

---

kidmo_model	<i>KIDMO prediction model</i>
-------------	-------------------------------

---

**Description**

Returns KIDMO prediction model fit.

**Usage**

```
kidmo_model()
```

**Value**

model fit

---

lifeport_d2	<i>D-squared for LifePort data</i>
-------------	------------------------------------

---

**Description**

Calculate Mahalanobis distance D-squared for LifePort temperature and perfusion data.

**Usage**

```
lifeport_d2(data, type)
```

**Arguments**

data                      data frame or matrix with temperature or perfusion data  
type                      string, type of D-square either "temp" or "perf"

**Value**

data frame with D-squared and rank

---

lifeport_process	<i>Process LifePort data</i>
------------------	------------------------------

---

**Description**

Processing of LifPort data adds runtime, clock time, and smoothed time series.

**Usage**

```
lifeport_process(lpdat, window_size = 15)
```

**Arguments**

lpdat	list with data from lifeport_read()
window_size	rolling window size for filtering

**Value**

list with LifePort data

---

lifeport_read	<i>Read LifePort raw data</i>
---------------	-------------------------------

---

**Description**

Function to read LifePort binary as well as ASCII raw data files.

**Usage**

```
lifeport_read(file, format = "guess")
```

**Arguments**

file	data file with path
format	guess (default), binary or plaintxt

**Value**

list with LifePort data

---

lifeport_sumstats	<i>Summary statistics for LifePort data</i>
-------------------	---

---

**Description**

Adds summary statistics for pressure, flow, resistance, and temperature time series.

**Usage**

```
lifeport_sumstats(lpdat, ice_threshold = 3, infuse_threshold = 10)
```

**Arguments**

lpdat	list with data from lifeport_process()
ice_threshold	threshold for ice temperature in degrees Celsius
infuse_threshold	threshold for infuse temperature in degrees Celsius

**Value**

list with LifePort data

---

mean_sd	<i>Returns mean and SD as string</i>
---------	--------------------------------------

---

**Description**

Helper function for tidy formatting.

**Usage**

```
mean_sd(x, d1 = 1, d2 = 1)
```

**Arguments**

x	numeric vector
d1	number of digits
d2	number of digits

**Value**

character object

---

median_iqr	Returns median and interquartile range IQR
------------	--

---

**Description**

Helper function for tidy formatting.

**Usage**

```
median_iqr(x, d1 = 1, d2 = 1, d3 = 1, compact = FALSE)
```

**Arguments**

x	numeric vector
d1	number of digits
d2	number of digits
d3	number of digits
compact	use en dash instead of "from X to Y"

**Value**

character object

---

miss_perc	Returns count and percentage of missing data.
-----------	---

---

**Description**

Helper function for tidy formatting.

**Usage**

```
miss_perc(x, d2 = 1)
```

**Arguments**

x	vector
d2	number of digits

**Value**

character object

---

nearest	<i>Nearest element</i>
---------	------------------------

---

**Description**

Nearest element in vector for a given set of values.

**Usage**

```
nearest(y, q)
```

**Arguments**

y	vector to be searched
q	vector of values of interest

**Value**

indices of the nearest elements in y for a set of values in q

---

num2date	<i>Convert Excel numeric days to date</i>
----------	---

---

**Description**

Convert Excel days since origin to POSIXct data type (date/time).

**Usage**

```
num2date(
  days,
  origin = "1899-12-30",
  tz = "CET",
  filter = TRUE,
  pattern = "[0-9]{2}\\.[0-9]{2}\\.[0-9]{4}",
  format = "%d.%m.%Y",
  round = TRUE
)
```

**Arguments**

days	days since origin as numeric or string
origin	origin, default in excel is 1899-12-30
tz	time zone to be forced upon
filter	apply fix for dates not recognized (default is TRUE)
pattern	the pattern to find dates not recognized
format	format to convert dates not recognized, e.g. %d.%m.%Y %H:%M:%OS
round	recommended when format has no time, only date information

Value

date of the type POSIXct

---

optn_kdri	<i>OPTN KDRI</i>
-----------	------------------

---

Description

Calculates the OPTN KDRI according to the 2024 version.

Usage

```
optn_kdri(  
  D_age,  
  D_height,  
  D_weight,  
  D_hypertension,  
  D_diabetes,  
  D_CVA,  
  D_SCr,  
  D_DCD,  
  scaling = 1.40436817065005  
)
```

Arguments

D_age	donor age in years
D_height	donor height in cm
D_weight	donor weight in kg
D_hypertension	donor hypertension
D_diabetes	donor diabetes
D_CVA	donor cause of death is cardiovascular accident
D_SCr	serum creatinine in mg/dL
D_DCD	donation after cardiac death
scaling	scaling factor that is published every year by the OPTN

Details

See details under "Learn about KDPI" at <https://optn.transplant.hrsa.gov/data/allocation-calculators/kdri-calculator/>.

Value

KDRI hazard ratio



---

print.kidmo	<i>Print method for kidmo objects</i>
-------------	---------------------------------------

---

**Description**

Print method for kidmo objects

**Usage**

```
## S3 method for class 'kidmo'
print(x, ...)
```

**Arguments**

x	An object of class kidmo.
...	Additional arguments (ignored).

---

swt_colors	<i>SWT colors</i>
------------	-------------------

---

**Description**

Easy access to official SWT color scheme.

**Usage**

```
swt_colors()
```

**Value**

a SWT color object

**Examples**

```
mycolors = swt_colors()
mycolors$red.liver
```

---

swt_skeleton	<i>SWT skeleton</i>
--------------	---------------------

---

**Description**

This internal function enables a Swisstransplant Document in Quarto for RStudio projects.

**Usage**

```
swt_skeleton(path)
```

**Arguments**

path	project path
------	--------------

---

swt_style	<i>SWT theme for ggplot</i>
-----------	-----------------------------

---

### Description

This function allows you to add the SWT theme to your ggplot graphics.

### Usage

```
swt_style(
  title_size = 14,
  subtitle_size = 14,
  font_size = 10,
  grey_theme = FALSE,
  legend_position = "top"
)
```

### Arguments

title_size	font size of the title
subtitle_size	font size of the subtitle
font_size	font font size of the legend, axis text, and axis titles
grey_theme	whether to use the grey theme instead (TRUE or FALSE)
legend_position	position of the legend (top, bottom, left or right)

### Examples

```
library(ggplot2)
ggplot(mtcars, aes(wt, mpg)) +
  geom_point() +
  swt_style()
```

---

tidy_missing	<i>Tidy missing data summary</i>
--------------	----------------------------------

---

### Description

Calculates missing data for each variable in data frame.

### Usage

```
tidy_missing(df)
```

### Arguments

df	data frame with raw data
----	--------------------------

**Value**

data frame with summary data

---

tidy_pvalues	<i>Formats p-values.</i>
--------------	--------------------------

---

**Description**

Helper function for tidy formatting.

**Usage**

```
tidy_pvalues(x, compact = FALSE)
```

**Arguments**

x	numerical vector with p-values
compact	logical, no asterisks when TRUE

**Value**

formatted p-values as character vector

---

tidy_rmsfit	<i>Tidy rms model fit results</i>
-------------	-----------------------------------

---

**Description**

Shows tidy regression table with results as data frame.

**Usage**

```
tidy_rmsfit(fit, ...)
```

**Arguments**

fit	model fit from rms
...	optional arguments to summary of the rms fit object.

**Value**

formatted data.frame

---

uk_dcd_score	<i>UK DCD Risk Score</i>
--------------	--------------------------

---

**Description**

Calculates the UK DCD Risk Score that can range between 0 and 27.

**Usage**

```
uk_dcd_score(D_age, D_BMI, fWIT, CIT, R_age, R_MELD, retpx)
```

**Arguments**

D_age	donor age in years
D_BMI	donor BMI in kg/m <sup>2</sup>
fWIT	functional warm ischemia time in minutes
CIT	cold ischemia time in hours
R_age	recipient age in years
R_MELD	recipient lab MELD score
retpx	whether the aim is a retransplant

**Details**

Reference: Schlegel A, Kalisvaart M, Scalera I, et al. The UK DCD Risk Score: A new proposal to define futility in donation-after-circulatory-death liver transplantation. *J Hepatol.* 2018;68(3):456-464. doi:10.1016/j.jhep.2017.10.034

**Value**

UK DCD Risk Score

---

uk_kdri	<i>UK KDRI 2019</i>
---------	---------------------

---

**Description**

Calculates the UK KDRI version from 2019.

**Usage**

```
uk_kdri(D_age, D_height, D_hypertension, D_female, D_CMV, D_eGFR, D_days_hosp)
```

**Arguments**

D_age	donor age in years
D_height	donor height in cm
D_hypertension	donor hypertension
D_female	donor is female
D_CMV	donor cytomegalovirus positive
D_eGFR	estimated glomerular filtration rate (eGFR) in mL/min/1.73m <sup>2</sup>
D_days_hosp	days in hospital

**Details**

Reference: Kim JJ, Curtis RMK, Reynolds B, et al. The UK kidney donor risk index poorly predicts long-term transplant survival in paediatric kidney transplant recipients. Front Immunol. 2023;14:1207145. doi:10.3389/fimmu.2023.1207145

Calculator at <https://www.glasgowtransplant.com/tools/ukkdri.html>.

**Value**

UK KDRI 2019 hazard ratio

# Index

## \* datasets

exam.device, [4](#)  
exam.organ, [5](#)  
exam.sumstats, [6](#)  
exam.timeseries, [7](#)

count\_perc, [2](#)

date2num, [2](#)

egfr\_ckd\_epi, [3](#)  
egfr\_schwartz, [3](#)  
exam.device, [4](#)  
exam.organ, [5](#)  
exam.sumstats, [6](#)  
exam.timeseries, [7](#)

fmt\_hla, [7](#)

get\_days\_in\_year, [8](#)

hla\_mismatch, [8](#)  
hla\_parse, [9](#)

kidmo, [10](#)  
kidmo\_hr2rank, [10](#)  
kidmo\_model, [11](#)

lifeport\_d2, [11](#)  
lifeport\_process, [12](#)  
lifeport\_read, [12](#)  
lifeport\_sumstats, [13](#)

mean\_sd, [13](#)  
median\_iqr, [14](#)  
miss\_perc, [14](#)

nearest, [15](#)  
num2date, [15](#)

optn\_kdri, [16](#)

print.kidmo, [17](#)

swt\_colors, [17](#)  
swt\_skeleton, [17](#)

swt\_style, [18](#)

tidy\_missing, [18](#)  
tidy\_pvalues, [19](#)  
tidy\_rmsfit, [19](#)

uk\_dcd\_score, [20](#)  
uk\_kdri, [20](#)