Package ‘transplantr’

October 14, 2022

Type Package

Title Audit and Research Functions for Transplantation

Version 0.2.0

Description A set of vectorised functions to calculate medical equations used in transplantation, focused mainly on transplantation of abdominal organs. These functions include donor and recipient risk indices as used by NHS Blood & Transplant, OPTN/UNOS and Eurotransplant, tools for quantifying HLA mismatches, functions for calculating estimated glomerular filtration rate (eGFR), a function to calculate the APRI (AST to platelet ratio) score used in initial screening of suitability to receive a transplant from a hepatitis C seropositive donor and some biochemical unit converter functions. All functions are designed to work with either US or international units. References for the equations are provided in the vignettes and function documentation.

URL https://transplantr.txtools.net,
https://github.com/johnasher/transplantr

BugReports https://github.com/johnasher/transplantr/issues

Depends R (>= 3.1.0)

Suggests dplyr, stringr, knitr, rmarkdown, testthat, spelling

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

VignetteBuilder knitr

Language en-GB

NeedsCompilation no

Author John Asher [aut, cre] (<https://orcid.org/0000-0001-8735-6453>)

Maintainer John Asher <john.asher@outlook.com>

Repository CRAN

Date/Publication 2020-02-28 16:50:03 UTC
R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>apri</td>
<td>3</td>
</tr>
<tr>
<td>bar_score</td>
<td>4</td>
</tr>
<tr>
<td>bilirubin_to_SI</td>
<td>5</td>
</tr>
<tr>
<td>bilirubin_to_US</td>
<td>5</td>
</tr>
<tr>
<td>bun_to_urea</td>
<td>6</td>
</tr>
<tr>
<td>chi2dob</td>
<td>6</td>
</tr>
<tr>
<td>ckd_epi</td>
<td>7</td>
</tr>
<tr>
<td>ckd_epi_US</td>
<td>8</td>
</tr>
<tr>
<td>cockcroft</td>
<td>9</td>
</tr>
<tr>
<td>cockcroft_US</td>
<td>10</td>
</tr>
<tr>
<td>creatinine_to_SI</td>
<td>10</td>
</tr>
<tr>
<td>creatinine_to_US</td>
<td>11</td>
</tr>
<tr>
<td>epts</td>
<td>11</td>
</tr>
<tr>
<td>epts_lookup</td>
<td>12</td>
</tr>
<tr>
<td>et_dri</td>
<td>13</td>
</tr>
<tr>
<td>hla_mm_level</td>
<td>14</td>
</tr>
<tr>
<td>hla_mm_level_str</td>
<td>14</td>
</tr>
<tr>
<td>ibw</td>
<td>15</td>
</tr>
<tr>
<td>kdpi</td>
<td>16</td>
</tr>
<tr>
<td>kdpi_lookup</td>
<td>17</td>
</tr>
<tr>
<td>kdpi_US</td>
<td>18</td>
</tr>
<tr>
<td>kidney.donors</td>
<td>19</td>
</tr>
<tr>
<td>liver.pts</td>
<td>19</td>
</tr>
<tr>
<td>liver_dri</td>
<td>20</td>
</tr>
<tr>
<td>mdrd</td>
<td>21</td>
</tr>
<tr>
<td>mdrd_US</td>
<td>22</td>
</tr>
<tr>
<td>meld</td>
<td>23</td>
</tr>
<tr>
<td>meld_na</td>
<td>24</td>
</tr>
<tr>
<td>meld_na_US</td>
<td>25</td>
</tr>
<tr>
<td>meld_US</td>
<td>25</td>
</tr>
<tr>
<td>mismatches</td>
<td>26</td>
</tr>
<tr>
<td>nankivell</td>
<td>27</td>
</tr>
<tr>
<td>nankivell_spk</td>
<td>28</td>
</tr>
<tr>
<td>nankivell_spk_US</td>
<td>29</td>
</tr>
<tr>
<td>nankivell_US</td>
<td>30</td>
</tr>
<tr>
<td>pdri</td>
<td>30</td>
</tr>
<tr>
<td>pedi_soft</td>
<td>32</td>
</tr>
<tr>
<td>peld</td>
<td>33</td>
</tr>
<tr>
<td>peld_US</td>
<td>34</td>
</tr>
<tr>
<td>p_pass</td>
<td>34</td>
</tr>
<tr>
<td>p_soft</td>
<td>36</td>
</tr>
<tr>
<td>p_soft_US</td>
<td>37</td>
</tr>
<tr>
<td>raw_epts</td>
<td>38</td>
</tr>
<tr>
<td>results</td>
<td>39</td>
</tr>
<tr>
<td>results_US</td>
<td>39</td>
</tr>
<tr>
<td>schwartz</td>
<td>40</td>
</tr>
</tbody>
</table>
Description

A vectorised function to calculate the APRI score, a predictor of hepatic fibrosis.

Usage

```r
apri(ast, plt, ast_uln = 40)
```

Arguments

- `ast` numeric vector of serum AST levels in IU/l
- `plt` numeric vector of platelet counts (10^9/l)
- `ast_uln` single number value for lab upper limit of normal for AST levels (default is 40)

Details


Value

numeric vector of APRI scores
Examples

apri(ast = 38, plt = 150, ast_uln = 40)

# if the lab upper limit of normal is 40, ast_uln can be omitted
apri(ast = 160, plt = 75)

bar_score

BAR (Balance of Risk) score in liver transplantation

Description

A vectorised function to calculate the BAR score to predict patient survival after liver transplantation using a composite of donor and recipient factors.

Usage

bar_score(Age, MELD, ReTx, LifeSupport, CIT, DonorAge)

Arguments

Age    numeric vector of recipient ages in years
MELD   numeric vector of MELD scores
ReTx   numeric vector of whether retransplant (1 = "yes", 0 = "no")
LifeSupport  numeric vector of whether on life support pre-transplant (1 = "yes", 0 = "no")
CIT    numeric vector of cold ischaemic time in hours
DonorAge numeric vector of donor ages

Details


Value

numeric vector of BAR scores

Examples

bar_score(Age = 63, MELD = 27, ReTx = 0, LifeSupport = 0, CIT = 9.5, DonorAge = 67)
**bilirubin_to_SI**  

**Bilirubin unit converter (mg/dl -> µmol/l)**

---

**Description**

A vectorised function to convert serum bilirubin levels from mg/dl to µmol/l

**Usage**

```
bilirubin_to_SI(bili)
```

**Arguments**

- `bili` numeric vector of bilirubin levels (mg/dl)

**Value**

numeric vector of bilirubin levels in µmol/l

**Examples**

```
bilirubin_to_SI(bili = 3.1)
```

---

**bilirubin_to_US**  

**Bilirubin unit converter (µmol/l -> mg/dl)**

---

**Description**

A vectorised function to convert serum bilirubin levels from µmol/l to mg/dl

**Usage**

```
bilirubin_to_US(bili)
```

**Arguments**

- `bili` numeric vector of bilirubin levels (µmol/l)

**Value**

numeric vector of bilirubin levels in mg/dl

**Examples**

```
bilirubin_to_US(bili = 54)
```
bun_to_urea  \hspace{1cm} \textit{Convert BUN to urea}

\textbf{Description}

A vectorised function to convert blood urea nitrogen (BUN) to urea. The default unit for urea is mmol/l but this can be changed to mg/dl by setting the optional units parameter to "US".

\textbf{Usage}

\begin{verbatim}
bun_to_urea(BUN, units = "SI")
\end{verbatim}

\textbf{Arguments}

\begin{itemize}
  \item \textbf{BUN} \hspace{1cm} numeric vector of blood urea nitrogen levels (mg/dl)
  \item \textbf{units} \hspace{1cm} units for urea ("SI" for mmol/l, "US" for mg/dl)
\end{itemize}

\textbf{Value}

numeric vector of urea levels

\textbf{Examples}

\begin{verbatim}
bun_to_urea(8.0)
\end{verbatim}

\chi2dob  \hspace{1cm} \textit{Convert CHI number to date of birth}

\textbf{Description}

A vectorised function to convert a Scottish CHI number to date of birth in POSIXct date format. Note that this function does not always work as not all CHI numbers correspond accurately to date of birth, and any person born before 1920 will appear to be 100 years younger unless the optional cent parameter set to TRUE. Childen born in or after 2020 will appear 100 years older unless the optional paed parameter is set to TRUE. This function requires the stringr package.

\textbf{Usage}

\begin{verbatim}
chi2dob(chi, paed = FALSE, cent = FALSE)
\end{verbatim}

\textbf{Arguments}

\begin{itemize}
  \item \textbf{chi} \hspace{1cm} vector of CHI numbers (as numeric or string)
  \item \textbf{paed} \hspace{1cm} Whether paediatric patient (TRUE/FALSE), either a vector or a single TRUE/FALSE for whole series
  \item \textbf{cent} \hspace{1cm} Whether born before 1920 (TRUE/FALSE), either a vector or a single TRUE/FALSE for whole series
\end{itemize}
Details

The paed and cent parameters can either be provided as vectors for each case, for example in a series where there are patients with dates of birth in both the 1910s and 2010s, or alternatively can be set as a single TRUE or FALSE for the whole series.

Value

a vector of POSIXct dates

Examples

# as a single numeric
chi2dob(1503541234)

# as a single character string, for a patient born in 1919
chi2dob("1108191234", cent = TRUE)

# as a mixed vector of adults and children, including one born in 1919
chi2dob(chi = c("1503541234", "1108191234", "0510141234"),
         cent = c(FALSE, TRUE, FALSE))

ckd_epi  eGFR by CKD-EPI equation

Description

A vectorised function to calculate estimated glomerular filtration rate using the CKD-EPI equation. By default the equation accepts serum creatinine in µmol/l but can be changed to mg/dl by setting the units parameter to "US". To allow for serial measurements over time, such as for transplant follow-up data, there is an optional offset = n parameter which increases the age value used in the equation by n years.

Usage

ckd_epi(creat, age, sex, ethnicity, units = "SI", offset = 0)

Arguments

creat  numeric vector of serum creatinine in µmol/l (or mg/dl if units = "US")
age  numeric vector of age in years (accepts integers or decimals)
sex  character vector of sex ("F" for female, "M" for male)
ethnicity  character vector of patient ethnicity, one of "black" or "non-black"
units  non-vectorised optional parameter for creatinine unit ("SI" for µmol/l (default), "US" for mg/dl)
offset  non-vectorised optional numeric parameter for offset in years
Details


Value

a numeric vector of eGFR values

Examples

ckd_epi_US(creat = 1.5, age = 64.3, sex = "F", ethnicity = "black")

ckd_epi_US(creat = 120, age = 45.2, sex = "M", ethnicity = "non-black")

Description

A wrapper function for the ckd_epi() vectorised function to calculate estimated glomerular filtration rate using the CKD-EPI equation, using serum creatinine in mg/dl. To allow for serial measurements over time, such as for transplant follow-up data, there is an optional offset = n parameter which increases the age value used in the equation by n years.

Usage

ckd_epi_US(creat, age, sex, ethnicity, offset = 0)

Arguments

creat numeric vector of serum creatinine in µmol/l (or mg/dl if units = "US")
age numeric vector of age in years (accepts integers or decimals)
sex character vector of sex ("F" for female, "M" for male)
etnicity character vector of patient ethnicity, one of "black" or "non-black"
offset non-vectorised optional parameter for offset in years

Value

a numeric vector of eGFR values

Examples

ckd_epi_US(creat = 1.5, age = 64.3, sex = "F", ethnicity = "black")
Description

A vectorised function to estimate creatinine clearance using the Cockcroft-Gault equation. By default this uses serum creatinine in µmol/l but can be changed to mg/dl by setting the units parameter to "US".

Usage

cockcroft(creat, age, sex, weight, units = "SI")

Arguments

creatinine levels in µmol/l (or mg/dl if units = "US")
age numeric vector of ages in years
sex character vector of sex ("F" = female, "M" = male)
weight numeric vector of weights in kilograms
units non-vectorised parameter for creatinine units ("SI" for µmol/l (default) or "US" for mg/dl)

Details


Value

numeric vector of creatinine clearances in ml/min

Examples

# calculate creatinine clearance using creatinine in µmol/l
cockcroft(creat = 88.4, age = 25, sex = "F", weight = 60)

# calculate using creatinine in mg/dl
cockcroft(creat = 1, age = 25, sex = "F", weight = 60, units = "US")
```r
cockcroft_US

*Creatinine clearance by Cockcroft-Gault equation (US units)*

**Description**

A wrapper function for cockcroft(), a vectorised function to estimate creatinine clearance using the Cockcroft-Gault equation, but using creatinine in mg/dl

**Usage**

```r
cockcroft_US(creat, age, sex, weight)
```

**Arguments**

- `creat` numeric vector of creatinine levels in mg/dl
- `age` numeric vector of ages in years
- `sex` character vector of sex ("F" = female, "M" = male)
- `weight` numeric vector of weights in kilograms

**Details**


**Value**

numeric vector of creatinine clearances in ml/min

**Examples**

```r
cockcroft_US(creat = 1, age = 25, sex = "F", weight = 60)
```

---

```r
creatinine_to_SI

*Creatinine unit converter (mg/dl -> µmol/l)*

**Description**

A vectorised function to convert serum creatinine levels from mg/dl to µmol/l

**Usage**

```r
creatinine_to_SI(creat)
```

**Arguments**

- `creat` numeric vector of creatinine levels (mg/dl)
```
creatinine_to_US

Value
numeric vector of creatinine levels in µmol/l

Examples
creatinine_to_SI(creat = 2.0)

creatinine_to_US Creatinine unit converter (µmol/l -> mg/dl)

Description
A vectorised function to convert serum creatinine levels from µmol/l to mg/dl

Usage
creatinine_to_US(creat)

Arguments
creat numeric vector of creatinine levels (µmol/l)

Value
numeric vector of creatinine levels in mg/dl

Examples
creatinine_to_US(creat = 176)

epts Estimated Post-Transplant Survival Score (EPTS)

Description
A vectorised function to calculate EPTS scores as percentiles for norm-related prediction of patient survival after adult renal transplants. This function generates the EPTN scores as percentiles using the most recent lookup table on the OPTN website published in March 2019 and using SRTR data from 2018. The table can be found at https://optn.transplant.hrsa.gov/media/2973/epts_mapping_table_2018.pdf

Usage
epts(age, dm, prev_tx, dx)
**Arguments**

- **age**
  - numeric vector of patient age in years (with decimals)
- **dm**
  - numeric vector of whether patient diabetic (1 = yes, 0 = no)
- **prev_tx**
  - numeric vector of whether patient has a previous solid organ transplant
- **dx**
  - numeric vector of duration of dialysis in years (with decimals)

**Details**

This function requires the dplyr package to be installed. References: https://optn.transplant.hrsa.gov/resources/allocation-calculators/epts-calculator/ and https://optn.transplant.hrsa.gov/media/1511/guide_to_calculating_interpreting_epts.pdf

**Value**

numeric vector of EPTS scores as percentiles

**Examples**

```r
epts(age = 23.5838467, dm = 0, prev_tx = 1, dx = 5.0814511)
epts(age = 52.8788501, dm = 0, prev_tx = 0, dx = 0)
epts(age = 22.5242984, dm = 1, prev_tx = 1, dx = 6.8747433)
```

---

**epts_lookup**

**EPTS lookup function**

**Description**

A vectorised function to convert EPTS scores to percentiles for norm-related prediction of patient survival after adult renal transplants. This calculator uses the most recent lookup table published in March 2019 and using SRTR data from 2018 from the OPTN website at https://optn.transplant.hrsa.gov/media/2973/epts_mapping_table_2018.pdf

**Usage**

```
epts_lookup(raw)
```

**Arguments**

- **raw**
  - numeric vector of raw EPTS scores

**Details**

This function requires the dplyr package to be installed.


**Value**

numeric vector of EPTS scores as percentiles
et_dri

Examples

epts_lookup(1.54) # 21

---

et_dri

Eurotransplant Donor Risk Index in Liver Transplantation (ET-DRI)

Description

A vectorised function to calculate the Eurotransplant Donor Risk Index for liver transplantation. The ET-DRI is a variant of the American DRI published by Feng et al but adapted to the European population. The American liver DRI is available using the transplantr::liver_dri() function.

Usage

et_dri(age, cod, dcd, split, share, cit, ggt, rescue)

Arguments

- age: numeric vector of patient ages in years
- cod: character string vector of donor causes of death, one of "trauma", "anoxia", "cva" or "other"
- dcd: numeric vector of whether DCD (1 = yes, 0 = no)
- split: numeric vector of whether liver split (1 = yes, 0 = no)
- share: character string vector of type of sharing, one of "local", "regional" or "national"
- cit: numeric vector of cold ischaemic times in hours
- ggt: numeric vector of last pre-transplant serum gamma-GT level in IU/l
- rescue: numeric vector of whether rescue transplant (1 = yes, 0 = no)

Details


Value

numeric vector of ET-DRI scores

Examples

et_dri(age = 39, cod = "trauma", dcd = 0, split = 0, share = "local", cit = 8, ggt = 50, rescue = 0) # 1.00

et_dri(age = 25, cod = "cva", dcd = 0, split = 0, share = "local", cit = 8, ggt = 50, rescue = 0) # 1.15
**hla_mm_level**

*HLA mismatch level*

**Description**

Vectorised function to calculate HLA mismatch level as used in UK national deceased donor kidney matching scheme.

**Usage**

\[
\text{hla_mm_level}(a, b, dr)
\]

**Arguments**

- **a** numeric vector of HLA A locus mismatches (0-2)
- **b** numeric vector of HLA B locus mismatches (0-2)
- **dr** numeric vector of HLA DR locus mismatches (0-2)

**Details**

Mismatches should be provided as numeric vectors of integers with values from 0 to 2. The HLA mismatch level is derived from UK kidney matching policies which can be found on the NHS Blood & Transplant ODT website at www.odt.nhs.uk

**Value**

numeric vector of HLA mismatch levels (1-4)

**Examples**

\[
\text{hla_mm_level}(a = 0, b = 1, dr = 1)
\]

**hla_mm_level_str**

*HLA mismatch level from string*

**Description**

Vectorised function to calculate HLA mismatch levels where the HLA A, B and DR mismatch is recorded as a string rather than as separate numeric values, e.g. "1:0:1" or "101". The function calculates the mismatch level as used in the UK national deceased donor kidney matching scheme. By default, the function assumes a single separator character is used between each of the three numbers in the mismatch; if not, set the `sep` parameter to `FALSE`. This function needs the `stringr` package to be installed.
**Usage**

```r
hla_mm_level_str(mm, sep = TRUE)
```

**Arguments**

- `mm`: character string vector of HLA mismatches, e.g. "1:1:0" or "211"
- `sep`: logical to indicate whether separator used in the HLA mismatch strings (default TRUE)

**Details**

The HLA mismatch level is derived from UK kidney matching policies which can be found on the NHS Blood & Transplant ODT website at www.odt.nhs.uk

**Value**

numeric vector of HLA mismatch levels

**Examples**

```r
# using string of HLA mismatches with colons
hla_mm_level_str("1:1:0")

# using string of HLA mismatches without separator
hla_mm_level_str("211", sep = FALSE)
```

---

**ibw**

**Ideal body weight**

**Description**

A vectorised function to calculate adult ideal body weight based on height and sex. This function assumes ideal BMI of 21.5 for females and 23 for males.

**Usage**

```r
ibw(height, sex)
```

**Arguments**

- `height`: numeric vector of heights in cm
- `sex`: character vector of sex ("F" for female or "M" for male)

**Value**

numeric vector of ideal body weights in kg
Examples

\[ \text{ibw(} \text{height} = 183, \text{ sex} = "M"\text{)} \]

\[ \text{kdpi} \]

\[ \text{US KDPI} \]

Description

Vectorised function to calculate US KDPI percentile as published by OPTN/UNOS. Please note that this function uses creatinine measured in \( \mu \text{mol/l} \) by default, but can be changed to mg/dl if the optional units parameter is set to "US".

Usage

\[ \text{kdpi}(\text{age}, \text{height}, \text{weight}, \text{eth}, \text{htn}, \text{dm}, \text{cva}, \text{creat}, \text{hcv}, \text{dcd}, \text{scaling} = 1, \text{units} = "\text{SI}"\) \]

Arguments

- **age**: numeric vector of donor ages in years
- **height**: numeric vector of donor heights in cm
- **weight**: numeric vector of donor weights in kg
- **eth**: character string vector of donor ethnicity, "black" or "non-black"
- **htn**: numeric vector of donor hypertension history (1 = yes, 0 = no)
- **dm**: numeric vector of donor diabetes history (1 = yes, 0 = no)
- **cva**: numeric vector of whether donor death due to CVA (1 = yes, 0 = no)
- **creat**: numeric vector of donor serum creatinine (\( \mu \text{mol/l} \))
- **hcv**: numeric vector of donor hepatitis C history (1 = yes, 0 = no)
- **dcd**: numeric vector of type of donor (1 = DCD, 0 = DBD)
- **scaling**: single numeric value for OPTN scaling factor (optional, defaults to 1)
- **units**: single string value to indicate creatinine units ("SI" for \( \mu \text{mol/l} \), "US" for mg/dl)

Details

The KDRI is calculated from KDRI normalised by a scaling factor based on the median KDRI in the previous year. For 2018, this was approximately 1.250609 as is published on the OPTN website. The scaling parameter in this function defaults to 1, so can be left out to calculate the KDPI without scaling.

This function requires the dplyr package to be installed.

kdpi_lookup

Value

numeric vector of US KDRI values

Examples

# with creatinine in μmol/l (units = "SI" can be omitted)
kdpi(age = 40, height = 170, weight = 80, eth = "non-black", htn = 0, dm = 0,
   cva = 0, creat = 120, hcv = 0, dcd = 0, scaling = 1.250609, units = "SI")

# with creatinine in mg/dl
kdpi(age = 40, height = 170, weight = 80, eth = "non-black", htn = 0, dm = 0,
   cva = 0, creat = 1.4, hcv = 0, dcd = 0, scaling = 1.250609, units = "US")

kdpi_lookup

US KDPI lookup function

Description

A vectorised function to convert kdri KDRI scores to KDPI percentiles. If the OPTN scaling factor
was not used when calculating the KDRI, it can be set here using the optional scaling parameter
which uses a default value of 1 (for no scaling).

Usage

kdpi_lookup(kdri, scaling = 1)

Arguments

kdri numeric vector of KDRI values
scaling optional parameter for scaling factor (default is 1)

Details

This function requires the dplyr package to be installed.

Value

numeric vector of KDPI percentiles

Examples

# if scaling factor was used when calculating KDRI
kdpi_lookup(1.25)

# if scaling factor for 2018 needs to be applied
kdpi_lookup(1.25, scaling = 1.2506957544151)
kdpi_US

US KDPI (US units)

Description

Wrapper function for the kdpi() vectorised function to calculate US KDPI percentile as published by OPTN/UNOS, using creatinine measured in mg/dl (please use the kdpi() function for µmol/l).

Usage

```r
kdpi_US(age, height, weight, eth, htn, dm, cva, creat, hcv, dcd,
        scaling = 1)
```

Arguments

- `age`: numeric vector of donor ages in years
- `height`: numeric vector of donor heights in cm
- `weight`: numeric vector of donor weights in kg
- `eth`: character string vector of donor ethnicity, "black" or "non-black"
- `htn`: numeric vector of donor hypertension history (1 = yes, 0 = no)
- `dm`: numeric vector of donor diabetes history (1 = yes, 0 = no)
- `cva`: numeric vector of whether donor death due to CVA (1 = yes, 0 = no)
- `creat`: numeric vector of donor serum creatinine (µmol/l)
- `hcv`: numeric vector of donor hepatitis C history (1 = yes, 0 = no)
- `dcd`: numeric vector of type of donor (1 = DCD, 0 = DBD)
- `scaling`: single numeric value for OPTN scaling factor (optional, defaults to 1)

Details

The KDRI is calculated from KDRI normalised by a scaling factor based on the median KDRI in the previous year. For 2018, this was approximately 1.250609 as is published on the OPTN website. The scaling parameter in this function defaults to 1, so can be left out to calculate the KDPI without scaling.

This function requires the dplyr package to be installed.


Value

numeric vector of US KDRI values
Examples

```r
# with creatinine in mg/dl
kdpi_US(age = 40, height = 170, weight = 80, eth = "non-black", htn = 0, dm = 0,
cva = 0, creat = 1.4, hcv = 0, dcd = 0, scaling = 1.250609)
```

### kidney.donors

**Simulated dataset of donors to illustrate KDRI vignette.**

#### Description

A simulated dataset of kidney donors to illustrate the KDRI vignette

#### Usage

```r
kidney.donors
```

#### Format

A data frame with 4 rows and 7 variables:

- **Donor.Age** donor age in years
- **Donor.Height** donor height in cm
- **Donor.Hypertension** donor hypertension
- **Donor.Sex** donor sex
- **Donor.CMV** donor CMV status
- **Donor.GFR** donor GFR
- **Donor.Hospital_Stay** donor hospital stay in days ...

```
liver.pts
```

**Simulated dataset to illustrate MELD calculator vignette.**

#### Description

A simulated dataset of liver patient characteristics for the MELD vignette

#### Usage

```r
liver.pts
```
Format

A data frame with 4 rows and 6 variables:

- **Patient.Age**  patient age in years
- **Patient.INR**  Patient INR
- **Patient.Bilirubin**  Patient serum bilirubin in µmol/l
- **Patient.Creatinine**  serum creatinine in µmol/l
- **Patient.Sodium**  Patient serum sodium in mmol/l
- **Patient.Dialysed**  Whether patient dialysed (1 = yes, 0 = no) ...

---

**liver_dri**  
*Liver Donor Risk Index (DRI)*

Description

A vectorised function to calculate the Liver Donor Risk Index as published by Feng and others.

Usage

```r
liver_dri(age, cod, eth, dcd, split, share, cit, height)
```

Arguments

- **age**  numeric vector of patient ages in years
- **cod**  character string vector of donor causes of death, one of "trauma", "anoxia", "cva" or "other"
- **eth**  character string vector of ethnicity, one of "black", "white" or "other"
- **dcd**  numeric vector of whether DCD (1 = yes, 0 = no)
- **split**  numeric vector of whether liver split (1 = yes, 0 = no)
- **share**  character string vector of type of sharing, one of "regional" or "national"
- **cit**  numeric vector of cold ischaemic times in hours
- **height**  numeric vector of patient heights in cm

Details


Value

numeric vector of liver DRI values
Examples

liver_dri(age = 25, cod = "trauma", eth = "white", dcd = 0, split = 0,
         share = "local", cit = 8, height = 170) # 1.00

liver_dri(age = 64, cod = "cva", eth = "white", dcd = 0, split = 0,
         share = "local", cit = 14, height = 170) # 1.88

mdrd

eGFR by abbreviated MDRD equation

Description

A vectorised function to calculate estimated glomerular filtration rate using the abbreviated (four
variable) MDRD equation. By default the equation accepts serum creatinine in µmol/l but can be
changed to mg/dl by setting the units parameter to "US". To allow for serial measurements over
time, such as for transplant follow-up data, there is an optional offset = n parameter which increases
the age value used in the equation by n years.

Usage

mdrd(creat, age, sex, ethnicity, units = "SI", offset = 0)

Arguments

creat numeric vector of serum creatinine in µmol/l (or mg/dl if units = "US")
age numeric vector of age in years (accepts integers or decimals)
sex character vector of sex ("F" for female, "M" for male)
etnicity character vector of patient ethnicity, one of "black" or "non-black"
units non-vectorised optional parameter for creatinine unit ("SI" for µmol/l (default),
        "US" for mg/dl)
offset non-vectorised optional parameter for offset in years

Details


Value

a numeric vector of eGFR values

Examples

mdrd(creat = 120, age = 45.2, sex = "M", ethnicity = "non-black")
mdrd(creat = 1.5, age = 64.3, sex = "F", ethnicity = "black", units = "US")
Description

A wrapper for the mdrd4v() vectorised function to calculate estimated glomerular filtration rate using the abbreviated (four variable) MDRD equation, but using serum creatinine in mg/dl. To allow for serial measurements over time, such as for transplant follow-up data, there is an optional offset = n parameter which increases the age value used in the equation by n years.

Usage

mdrd_US(creat, age, sex, ethnicity, offset = 0)

Arguments

- **creat**: numeric vector of serum creatinine in µmol/l (or mg/dl if units = "US")
- **age**: numeric vector of age in years (accepts integers or decimals)
- **sex**: character vector of sex ("F" for female, "M" for male)
- **ethnicity**: character vector of patient ethnicity, one of "black" or "non-black"
- **offset**: non-vectorised optional parameter for offset in years

Details


Value

a numeric vector of eGFR values

Examples

mdrd_US(creat = 1.5, age = 64.3, sex = "F", ethnicity = "black")
meld

**MELD score**

### Description

A vectorised function to calculate the MELD score using µmol/l for bilirubin and creatinine. The units can be changed to mg/dl by setting the optional units parameter to "US". If the patient is on CVVH or has been dialysed at least twice in the same week, the dialysis argument should be set to 1, which changes the creatinine level used in the formula to 4mg/dl (353µmol/l). Following UNOS guidelines, the values for INR as well bilirubin and creatinine (in mg/dl) are set to a minimum value of 1 if less than 1.0.

### Usage

meld(INR, bili, creat, dialysis, units = "SI")

### Arguments

- **INR**: numeric vector of INR
- **bili**: numeric vector of bilirubin (µmol/l)
- **creat**: numeric vector of creatinine (µmol/l)
- **dialysis**: numeric vector of whether on dialysis/CVVH (1 = yes, 2 = no)
- **units**: Units for bilirubin and creatinine ("SI" for µmol/l (default), "US" for mg/dl)

### Details


### Value

MELD score

### Examples

meld(INR = 2.0, bili = 54, creat = 170, dialysis = 0)
meld(INR = 2.0, bili = 3.1, creat = 1.9, dialysis = 0, units = "US")
**meld_na**

**MELD-Na score**

**Description**

A vectorised function to calculate the MELD-Na score, a variant of the MELD score incorporating serum sodium levels. By default, bilirubin and creatinine are in µmol/l but this can be changed to mg/dl by setting the optional units parameter to "US".

**Usage**

`meld_na(INR, bili, creat, Na, dialysis, units = "SI")`

**Arguments**

- **INR**: numeric vector of INR
- **bili**: numeric vector of bilirubin (µmol/l)
- **creat**: numeric vector of creatinine (µmol/l)
- **Na**: numeric vector of sodium (mmol/l)
- **dialysis**: whether on dialysis/CVVH (1 = yes, 0 = no)
- **units**: Units for bilirubin and creatinine ("SI" for µmol/l (default), "US" for mg/dl)

**Details**


**Value**

numeric vector of MELD-Na scores

**Examples**

`meld_na(INR = 1.8, bili = 34, creat = 176, Na = 131, dialysis = 0)`

`meld_na(INR = 1.8, bili = 2, creat = 2, Na = 131, dialysis = 0, units = "US")`
meld_na_US

MELD-Na score (US units)

Description
A wrapper for meld_na(), a vectorised function to calculate the MELD-Na score, a variant of the
MELD score incorporating serum sodium levels. In this wrapper function, bilirubin and creatinine
are in mg/dl.

Usage
meld_na_US(INR, bili, creat, Na, dialysis)

Arguments

INR        numeric vector of INR
bili       numeric vector of serum bilirubin (mg/dl)
creat      numeric vector of serum creatinine (mg/dl)
Na         numeric vector of serum sodium (mmol/l)
dialysis   whether on dialysis/CVVH (1 = yes, 0 = no)

Details
sodium concentration into MELD. Gastroenterology 2006; 130(6):1652-60.

Value
numeric vector of MELD-Na scores

Examples
meld_na_US(INR = 1.8, bili = 2, creat = 2, Na = 131, dialysis = 0)

meld_US

MELD score (US units)

Description
A wrapper for the vectorised function meld() to calculate the MELD score, but using mg/dl for
bilirubin and creatinine. If the patient is on CVVH or has been dialysed at least twice in the same
week, the dialysis argument should be set to 1, which changes the creatinine level used in the
formula to 4mg/dl. Following UNOS guidelines, the values for INR as well bilirubin and creatinine
are set to a minimum value of 1 if less than 1.0
Usage
meld_US(INR, bili, creat, dialysis)

Arguments
INR numeric vector of INR  
bili numeric vector of bilirubin in mg/dl  
creat numeric vector of creatinine in mg/dl  
dialysis numeric vector of whether on dialysis/CVVH (1 = yes, 0 = no)

Details

Value
MELD score

Examples
meld_US(INR = 2.0, bili = 2.3, creat = 1.9, dialysis = 1)

mismatches

Simulated dataset to illustrate mismatches for HLA vignette.

Description
A simulated dataset of HLA mismatches

Usage
mismatches

Format
A data frame with 4 rows and 5 variables:

HLA.A.MM HLA A mismatch  
HLA.B.MM HLA B mismatch  
HLA.DR.MM HLA DR mismatch  
HLA.MM HLA mismatch as string  
HLA.MM.s HLA mismatch as string with separators ...
A vectorised function to calculate eGFR using the Nankivell formula. By default the equation accepts serum creatinine in µmol/l but can be changed to mg/dl by setting the units parameter to "US". The Urea parameter is serum urea in mmol/l, but if the units parameter is set to "US", Blood Urea Nitrogen (BUN) in mg/dl is used instead. It has not been possible to run automated tests to confirm the accuracy of this function, so please use with caution.

Usage

nankivell(SCr, Urea, Weight, Height, Sex, Units = "SI")

Arguments

- **SCr**: numeric vector of serum creatinine in µmol/l (or mg/dl if units = "US")
- **Urea**: numeric vector of serum urea in mmol/l (or BUN in mg/dl if units = "US")
- **Weight**: numeric vector of patient weights in kilograms
- **Height**: numeric vector of patient heights in centimetres
- **Sex**: character vector of sex ("F" for female, "M" for male)
- **Units**: non-vectorised optional parameter for creatinine and urea/BUN units ("SI" for µmol/l (default), "US" for mg/dl)

Details


Value

a numeric vector of eGFR values

Examples

nankivell(SCr = 118, Urea = 13.4, Weight = 65, Height = 172, Sex = "M") # 44.55
nankivell_spk  

**Description**

A vectorised function to calculate the eGFR using the Nankivell-SPK formula. By default the equation accepts serum creatinine in µmol/l but can be changed to mg/dl by setting the units parameter to "US". To allow for serial measurements over time, such as for transplant follow-up data, there is an optional offset = n parameter which increases the age value used in the equation by n years. It has not been possible to run automated tests to confirm the accuracy of this function, so please use with caution.

**Usage**

```
nankivell_spk(SCr, Age, Sex, Weight, Height, Units = "SI", Offset = 0)
```

**Arguments**

- **SCr**: numeric vector of serum creatinine in µmol/l (or mg/dl if units = "US")
- **Age**: numeric vector of patient ages in years
- **Sex**: character vector of sex ("F" for female, "M" for male)
- **Weight**: numeric vector of patient weights in kilograms
- **Height**: numeric vector of patient heights in centimetres
- **Units**: non-vectorised optional parameter for creatinine and urea/BUN units ("SI" for µmol/l (default), "US" for mg/dl)
- **Offset**: non-vectorised optional numeric parameter for offset in years

**Details**


**Value**

numeric vector of eGFR values

**Examples**

```
nankivell_spk(SCr = 118, Age = 74, Sex = "M", Weight = 172, Height = 172) # 48.8
```
Description

A wrapper function for the nankivell_spk() vectorised function to calculate the eGFR using the Nankivell-SPK formula using serum creatinine in mg/dl. To allow for serial measurements over time, such as for transplant follow-up data, there is an optional offset = n parameter which increases the age value used in the equation by n years. It has not been possible to run automated tests to confirm the accuracy of this function, so please use with caution.

Usage

nankivell_spk_US(SCr, Age, Sex, Weight, Height, Offset = 0)

Arguments

SCr numeric vector of serum creatinine in µmol/l (or mg/dl if units = "US")
Age numeric vector of patient ages in years
Sex character vector of sex ("F" for female, "M" for male)
Weight numeric vector of patient weights in kilograms
Height numeric vector of patient heights in metres
Offset non-vectorised optional numeric parameter for offset in years

Details


Value

numeric vector of eGFR values

Examples

nankivell_spk_US(SCr = 1.33, Age = 74, Sex = "M", Weight = 172, Height = 172) # 49.0
**nankivell_US**

**eGFR using Nankivell formula (US units)**

**Description**

A wrapper function for the nankivell() vectorised function to calculate eGFR using the Nankivell formula, but using creatinine and BUN in mg/dl rather than international units. It has not been possible to run automated tests to confirm the accuracy of this function, so please use with caution.

**Usage**

```r
nankivell_US(SCr, Urea, Weight, Height, Sex)
```

**Arguments**

- **SCr**: numeric vector of serum creatinine in µmol/l (or mg/dl if units = "US")
- **Urea**: numeric vector of serum urea in mmol/l (or BUN in mg/dl if units = "US")
- **Weight**: numeric vector of patient weights in kilograms
- **Height**: numeric vector of patient heights in centimetres
- **Sex**: character vector of sex ("F" for female, "M" for male)

**Details**


**Value**

a numeric vector of eGFR values

**Examples**

```r
nankivell_US(SCr = 1.33, Urea = 13.4, Weight = 65, Height = 172, Sex = "M") # 44.55
```

**pdri**

**Pancreas donor risk index**

**Description**

A vectorised function to calculate the Pancreas Donor Risk Index as published by Axelrod et al. By default, the serum creatinine is used in mcmmol/l but this can be changed to mg/dl by setting the optional units parameter to "US"
Usage

```r
pdri(age, sex, creat, eth, bmi, height, cva, cit, dcd, intent = c("SPK", "PAK", "Other"), units = "SI")
```

Arguments

- **age**: numeric vector of ages in years
- **sex**: character vector of patient sex ("F" for female, "M" for male)
- **creat**: numeric vector of serum creatinine (µmol/l by default)
- **eth**: character vector of ethnicity, one of "asian", "black" or "other" in each case
- **bmi**: numeric vector of body mass index (BMI)
- **height**: numeric vector of heights in centimetres
- **cva**: numeric vector of whether CVA is cause of death (1 = yes, 0 = no)
- **cit**: numeric vector of cold ischaemic times in hours
- **dcd**: numeric vector of whether donor after circulatory death (1 = DCD, 0 = DBD)
- **intent**: character vector of implant intent, "PAK" for pancreas after kidney or "other"
- **units**: single character string for creatinine units: one of "SI" (for µmol/l) or "US" (for mg/dl)

Details


Value

numeric vector of pancreas donor risk index values

Examples

```r
pdri(age = 28, sex = "M", creat = 1.0, eth = "other", bmi = 24, height = 173, cva = 0, cit = 12, dcd = 0, units = "US") # 1.00

pdri(age = 45, sex = "M", creat = 88.4, eth = "other", bmi = 24, height = 173, cva = 0, cit = 12, dcd = 0, units = "SI") # 1.56
```
**pedi_soft**  

**Pedi-SOFT Score**  

**Description**  
A vectorised function to calculate the Pedi-SOFT score used to predict survival after liver transplantation in children.

**Usage**  
```r  
pedi_soft(CTVG, Weight, Dx, LifeSupport, PrevTx)  
```

**Arguments**  
- **CTVG**: numeric vector of whether cadaveric technical variant graft (1 for "yes", 0 for "no")
- **Weight**: numeric vector of recipient weight in kg
- **Dx**: numeric vector of whether on dialysis or creatinine clearance under 30 (1 for "yes", 0 for "no")
- **LifeSupport**: numeric vector of whether on life support pre-transplant (1 for "yes", 0 for "no")
- **PrevTx**: numeric vector of number of previous liver transplants

**Details**  

**Value**  
numeric vector of Pedi-SOFT scores

**Examples**  
```r  
pedi_soft(CTVG = 1, Weight = 10, Dx = 0, LifeSupport = 0, PrevTx = 0) # 4  
```
**Description**

A vectorised function to generate a PELD score for paediatric liver transplant candidates. The default unit for bilirubin is µmol/l and albumin in g/l, but these can be changed to mg/dl and g/dl respectively by setting the optional units parameter to "US".

**Usage**

```r
peld(INR, bili, albumin, listing_age, growth_failure, units = "SI")
```

**Arguments**

- **INR**: INR
- **bili**: serum biliruin (µmol/l)
- **albumin**: serum albumin (g/l)
- **listing_age**: age at the time of listing (years; integer or decimal)
- **growth_failure**: whether there is growth failure (1 = yes, 0 = no)
- **units**: units used for bilirubin ("SI" for µmol/l (default), "US" for mg/dl)

**Details**

Some labs report albumin in g/dl rather than the g/l used in this function. If units are set to "US" then g/dl is assumed and albumin should be divided by 10 if the lab reports are in g/l. If using SI units, take care to multiply the lab albumin by 10 if the lab output is in g/dl.


**Value**

numeric vector of PELD scores

**Examples**

```r
peld(INR = 2, bili = 54, albumin = 25, listing_age = 2, growth_failure = 1)
peld(INR = 2, bili = 3.1, albumin = 25, listing_age = 2, growth_failure = 1, units = "US")
```
### peld_US

**PELD score (US units)**

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A wrapper for peld(), a vectorised function to generate a PELD score for paediatric liver transplant candidates, using mg/dl as the unit for the serum bilirubin level.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>peld_US(INR, bili, albumin, listing_age, growth_failure)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Arguments</th>
</tr>
</thead>
<tbody>
<tr>
<td>INR</td>
</tr>
<tr>
<td>bili</td>
</tr>
<tr>
<td>albumin</td>
</tr>
<tr>
<td>listing_age</td>
</tr>
<tr>
<td>growth_failure</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Details</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric vector of PELD scores</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>peld_US(INR = 2, bili = 3.1, albumin = 2.5, listing_age = 2, growth_failure = 1)</td>
</tr>
</tbody>
</table>

### p_pass

**P-PASS pre-procurement pancreas suitability score**

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A vectorised function to calculate the P-PASS pre-procurement pancreas allocation suitability score used in the Eurotransplant area. The score are between 9 and 27, and in a study published by Vinkers et al. in 2008, pancreata with P-PASS score less than 17 were three times more likely to be transplanted than those with scores of 17 or more.</td>
</tr>
</tbody>
</table>
\textbf{Usage}

\begin{verbatim}
p_pass(age, bmi, icu, c.arr, Na, amylase = NULL, lipase = NULL, norad, dopam)
\end{verbatim}

\textbf{Arguments}

- \texttt{age} numeric vector of donor ages in years
- \texttt{bmi} numeric vector of donor body mass index (BMI)
- \texttt{icu} numeric vector of length of donor ICU stay in days
- \texttt{c.arr} numeric vector for duration of cardiac arrest (use 0 if no cardiac arrest)
- \texttt{Na} numeric vector of donor serum sodium in mmmol/l
- \texttt{amylase} numeric vector of donor serum amylase in IU/l (0 if not available)
- \texttt{lipase} numeric vector of donor serum lipase in IU/l (0 if not available)
- \texttt{norad} numeric vector of noradrenaline (0 if not used)
- \texttt{dopam} numeric vector of dopamine or dobutamine (0 if not used)

\textbf{Details}

At least one of amylase or lipase is needed for each case, but this function can take datasets with a mixture of amylase and lipase levels and will allocate points based on the higher points for cases when both are provided.


\textbf{Value}

numeric vector of P-PASS scores

\textbf{Examples}

- # as a single case
\begin{verbatim}
p_pass(age = 25, bmi = 19, icu = 0, c.arr = 0, Na = 135,
       amylase = 101, lipase = 120, norad = 0, dopam = 0) # 9
\end{verbatim}

- # as a vector with mixed amylase and lipase availability
\begin{verbatim}
p_pass(age = c(25, 31, 45), bmi = c(18, 22, 35), icu = c(2, 5, 10), c.arr = c(0, 4, 10),
       Na = c(135, 157, 164), amylase = c(120, NA, 400), lipase = c(155, 170, NA),
       norad = c(0, 0.02, 0.06), dopam = c(0, 5, 11)) # 9, 19, 25
\end{verbatim}

- # as a vector with all lipase values missing
\begin{verbatim}
p_pass(age = c(25, 31, 45), bmi = c(18, 22, 35), icu = c(2, 5, 10), c.arr = c(0, 4, 10),
       Na = c(135, 157, 164), amylase = c(120, 145, 400), lipase = c(NA, NA, NA),
       norad = c(0, 0.02, 0.06), dopam = c(0, 5, 11)) # 9, 19, 25
\end{verbatim}
p_soft

**P-SOFT Score**

**Description**

A vectorised function to calculate the pre-procurement component of the SOFT score used to predict patient survival after liver transplantation. The function needs the MELD score as one of its inputs - this is available using the transplantr::meld() function. The units for albumin are g/l but this is changed to g/dl if the optional Units parameter is set to "US"

**Usage**

```r
p_soft(Age, BMI, PrevTx, AbdoSurg, Albumin, Dx, ICU, Admitted, MELD, LifeSupport, Encephalopathy, PVThrombosis, Ascites, Units = "SI")
```

**Arguments**

- **Age** numeric vector of patient ages in years
- **BMI** numeric vector of patient BMI in kg/m²
- **PrevTx** numeric vector of number of previous transplants
- **AbdoSurg** numeric vector of whether previous abdominal surgery (1 = "yes", 0 = "no")
- **Albumin** numeric vector of serum albumin in g/l
- **Dx** numeric vector of whether on dialysis before transplant (1 = "yes", 0 = "no")
- **ICU** numeric vector of whether patients in intensive care unit before transplant (1 = "yes", 0 = "no")
- **Admitted** numeric vector of whether admitted to hospital pre-transplant (1 = "yes", 0 = "no")
- **MELD** numeric vector of MELD scores
- **LifeSupport** numeric vector of whether on life support pre-transplant (1 = "yes", 0 = "no")
- **Encephalopathy** numeric vector of whether encephalopathy present (1 = "yes", 0 = "no")
- **PVThrombosis** numeric vector of whether portal vein thrombosis (1 = "yes", 0 = "no")
- **Ascites** numeric vector of whether ascites pre-transplant (1 = "yes", 0 = "no")
- **Units** optional scalar for albumin units (one of "SI" for g/l, "US" for g/dl)

**Details**


**Value**

numeric vector of P-SOFT scores
Examples

```r
p_soft(Age = 65, BMI = 36, PrevTx = 2, AbdoSurg = 1, Albumin = 29, Dx = 0,
   ICU = 0, Admitted = 1, MELD = 32, LifeSupport = 0, Encephalopathy = 1,
   PVThrombosis = 1, Ascites = 1) # 37
```

---

### Description

A wrapper for the `p_soft()` vectorised function to calculate the pre-procurement component of the SOFT score used to predict patient survival after liver transplantation. The function needs the MELD score as one of its inputs - this is available using the transplantr::meld() function. The units for albumin are g/dl (rather than g/l in `p_soft()` function)

### Usage

```r
p_soft_US(Age, BMI, PrevTx, AbdoSurg, Albumin, Dx, ICU, Admitted, MELD,
   LifeSupport, Encephalopathy, PVThrombosis, Ascites)
```

### Arguments

- **Age**: numeric vector of patient ages in years
- **BMI**: numeric vector of patient BMI in kg/m2
- **PrevTx**: numeric vector of number of previous transplants
- **AbdoSurg**: numeric vector of whether previous abdominal surgery (1 = "yes", 0 = "no")
- **Albumin**: numeric vector of serum albumin in g/dl
- **Dx**: numeric vector of whether on dialysis before transplant (1 = "yes", 0 = "no")
- **ICU**: numeric vector of whether patients in intensive care unit before transplant (1 = "yes", 0 = "no")
- **Admitted**: numeric vector of whether admitted to hospital pre-transplant (1 = "yes", 0 = "no")
- **MELD**: numeric vector of MELD scores
- **LifeSupport**: numeric vector of whether on life support pre-transplant (1 = "yes", 0 = "no")
- **Encephalopathy**: numeric vector of whether encephalopathy present (1 = "yes", 0 = "no")
- **PVThrombosis**: numeric vector of whether portal vein thrombosis (1 = "yes", 0 = "no")
- **Ascites**: numeric vector of whether ascites pre-transplant (1 = "yes", 0 = "no")

### Details

Value

numeric vector of P-SOFT scores

Examples

p_soft_US(Age = 65, BMI = 36, PrevTx = 2, AbdoSurg = 1, Albumin = 2.9,
Dx = 0, ICU = 0, Admitted = 1, MELD = 32, LifeSupport = 0, Encephalopathy = 1,
PVThrombosis = 1, Ascites = 1) # 37

raw_epts  Estimated Post-Transplant Survival Score (EPTS)

Description

A vectorised function to calculate raw EPTS scores for norm-related prediction of patient survival after adult renal transplants. This function generates the raw EPTS scores which can be converted to percentiles using the lookup table on the OPTN website at https://optn.transplant.hrsa.gov/media/2973/epts_mapping_table_2018.pdf

Usage

raw_epts(age, dm, prev_tx, dx)

Arguments

age numeric vector of patient age in years (with decimals)
dm numeric vector of whether patient diabetic (1 = yes, 0 = no)
prev_tx numeric vector of whether patient has a previous solid organ transplant
dx numeric vector of duration of dialysis in years (with decimals)

Details


Value

numeric vector of raw EPTS scores

Examples

raw_epts(age = 23.5838467, dm = 0, prev_tx = 1, dx = 5.0814511) # 0.9666517
raw_epts(age = 52.8788501, dm = 0, prev_tx = 0, dx = 0) # 1.440306
raw_epts(age = 22.5242984, dm = 1, prev_tx = 1, dx = 6.8747433) # 1.868751
Simulated dataset to illustrate eGFR calculator vignette.

**Description**

A simulated dataset of patient characteristics and creatinine levels

**Usage**

results

**Format**

A data frame with 4 rows and 6 variables:

- **Creatinine**  serum creatinine in µmol/l
- **Age**  patient age in years
- **Sex**  Patient sex, M or F
- **Ethnicity**  Patient ethnicity
- **Weight**  Patient weight in kg
- **Height**  Patient height in cm ...

---

Simulated dataset to illustrate eGFR calculator vignette.

**Description**

A simulated dataset of patient characteristics and creatinine levels

**Usage**

results_US

**Format**

A data frame with 4 rows and 6 variables:

- **Creatinine**  serum creatinine in mg/dl
- **Sex**  Patient sex, M or F
- **Age**  patient age in years
- **Ethnicity**  Patient ethnicity
- **Weight**  Patient weight in kg
- **Height**  Patient height in cm ...
**Description**

A vectorised formula to calculate estimate glomerular filtration rate in children using the bedside Schwartz formula. By default this uses serum creatinine in µmol/l but this can be changed to mg/dl by setting the optional units parameter to "US".

**Usage**

schwartz(creat, height, units = "SI")

**Arguments**

- **creat**: numeric vector of creatinine levels in µmol/l (or mg/dl if units = "US")
- **height**: numeric vector of heights in cm
- **units**: non-vectorised optional parameter for creatinine unit ("SI" for µmol/l (default), "US" for mg/dl)

**Details**


**Value**

numeric vector of eGFR values

**Examples**

# calculate using creatinine in µmol/l
schwartz(creat = 64, height = 101)

# calculate using mg/dl
schwartz(creat = 0.7, height = 101, units = "US")
schwartz_US

eGFR by bedside Schwartz formula (US units)

Description
A wrapper function for the schwartz() vectorised formula to calculate estimate glomerular filtration rate in children using the bedside Schwartz formula, using serum creatinine in mg/dl. Use the schwartz() function instead for µmol/l.

Usage
schwartz_US(creat, height)

Arguments
creat     numeric vector of creatinine levels in µmol/l (or mg/dl if units = "US")
height    numeric vector of heights in cm

Details

Value
numeric vector of eGFR values

Examples
# calculate using creatinine in -mg/dl
schwartz_US(creat = 0.7, height = 101)

serial.results
Simulated dataset to illustrate serial results eGFR calculator vignette.

Description
A simulated dataset of patient characteristics and creatinine levels

Usage
serial.results
Format

A data frame with 4 rows and 6 variables:

- **Age**  patient age in years
- **Sex**  Patient sex, M or F
- **Ethnicity**  Patient ethnicity
- **Creatinine_1yr**  serum creatinine at 1 year, in µmol/l
- **Creatinine_5yr**  serum creatinine at 5 years, in µmol/l

Description

A vectorised function to calculate SOFT Scores for predicting patient survival after liver transplantation. The units for donor serum creatinine are in µmol/l and recipient serum albumin in g/l but they can be changed to mg/dl and g/dl respectively by setting the `Units` parameter to "US".

Usage

```r
soft(Age, BMI, PrevTx, AbdoSurg, Albumin, Dx, ICU, Admitted, MELD, LifeSupport, Encephalopathy, PVThrombosis, Ascites, PortalBleed, DonorAge, DonorCVA, DonorScr, National, CIT, Units = "SI")
```

Arguments

- **Age**  numeric vector of patient ages in years
- **BMI**  numeric vector of patient BMI in kg/m2
- **PrevTx**  numeric vector of number of previous transplants
- **AbdoSurg**  numeric vector of whether previous abdominal surgery (1 = "yes", 0 = "no")
- **Albumin**  numeric vector of serum albumin in g/l
- **Dx**  numeric vector of whether on dialysis before transplant (1 = "yes", 0 = "no")
- **ICU**  numeric vector of whether patients in intensive care unit before transplant (1 = "yes", 0 = "no")
- **Admitted**  numeric vector of whether admitted to hospital pre-transplant (1 = "yes", 0 = "no")
- **MELD**  numeric vector of MELD scores
- **LifeSupport**  numeric vector of whether on life support pre-transplant (1 = "yes", 0 = "no")
- **Encephalopathy**  numeric vector of whether encephalopathy present (1 = "yes", 0 = "no")
- **PVThrombosis**  numeric vector of whether portal vein thrombosis (1 = "yes", 0 = "no")
- **Ascites**  numeric vector of whether ascites pre-transplant (1 = "yes", 0 = "no")
**PortalBleed** numeric vector of whether portal bleeding in 48 hours pre-transplant (1 = "yes", 0 = "no")

**DonorAge** numeric vector of donor ages in years

**DonorCVA** numeric vector of whether donor cause of death is CVA/stroke (1 = "yes", 0 = "no")

**DonorSCr** numeric vector of donor terminal serum creatinine

**National** numeric vector of whether national allocation (1 = "yes", 0 = "no")

**CIT** numeric vector of cold ischaemic time in hours

**Units** units to use for creatinine and albumin, "SI" (default) for µmol/l and g/l, "US" for mg/dl and g/dl

### Details


### Value

numeric vector of SOFT Scores

### Examples

```r
soft(Age = 35, BMI = 20, PrevTx = 0, AbdoSurg = 1, Albumin = 30, Dx = 0,
    ICU = 0, Admitted = 0, MELD = 29, LifeSupport = 0, Encephalopathy = 1,
    PVThrombosis = 0, Ascites = 1, PortalBleed = 0, DonorAge = 44, DonorCVA = 0,
    DonorSCr = 110, National = 0, CIT = 8) # 7
```

---

### Description

A vectorised function to calculate SOFT Scores for predicting patient survival after liver transplantation when the P-SOFT score is already known. The P-SOFT Score can be calculated using the `transplantr::p_soft()` function. Alternatively, the SOFT Score can be calculated in full, including the P-SOFT parameters using the `transplantr::soft()` function. The units for donor serum creatinine are in µmol/l but can be changed to mg/dl by setting the Units parameter to "US".

### Usage

```r
soft2(PSoft, PortalBleed, DonorAge, DonorCVA, DonorSCr, National, CIT,
    Units = "SI")
```
Arguments

- **PSoft**: numeric vector of P-SOFT scores
- **PortalBleed**: numeric vector of whether portal bleeding in 48 hours pre-transplant (1 = "yes", 0 = "no")
- **DonorAge**: numeric vector of donor ages in years
- **DonorCVA**: numeric vector of whether donor cause of death is CVA/stroke (1 = "yes", 0 = "no")
- **DonorSCr**: numeric vector of donor terminal serum creatinine
- **National**: numeric vector of whether national allocation (1 = "yes", 0 = "no")
- **CIT**: numeric vector of cold ischaemic time in hours
- **Units**: units to use for creatinine, "SI" (default) for µmol/l, "US" for mg/dl

Details


Value

numeric vector of SOFT Scores

Examples

```r
soft2(PSoft = 4, PortalBleed = 0, DonorAge = 61, DonorCVA = 1, DonorSCr = 140, National = 1, CIT = 12) # 13
```

**soft2_US**

*SOFT score from P-SOFT (US units)*

Description

A wrapper using US units for the soft2() vectorised function to calculate SOFT Scores for predicting patient survival after liver transplantation when the P-SOFT score is already known. The P-SOFT Score can be calculated using the transplantr::p_soft() function. Alternatively, the SOFT Score can be calculated in full, including the P-SOFT parameters using the transplantr::soft() or transplantr::soft_US() function. The units for donor serum creatinine are in mg/dl

Usage

```r
soft2_US(PSoft, PortalBleed, DonorAge, DonorCVA, DonorSCr, National, CIT)
```
Arguments

**PSoft** numeric vector of P-SOFT scores

**PortalBleed** numeric vector of whether portal bleeding in 48 hours pre-transplant (1 = "yes", 0 = "no")

**DonorAge** numeric vector of donor ages in years

**DonorCVA** numeric vector of whether donor cause of death is CVA/stroke (1 = "yes", 0 = "no")

**DonorScr** numeric vector of donor terminal serum creatinine in mg/dl

**National** numeric vector of whether national allocation (1 = "yes", 0 = "no")

**CIT** numeric vector of cold ischaemic time in hours

Details


Value

numeric vector of SOFT Scores

Examples

```r
soft2_US(PSoft = 4, PortalBleed = 0, DonorAge = 61, DonorCVA = 1, DonorScr = 1.6, National = 1, CIT = 12) # 13
```

Description

A wrapper function using US units for the soft() vectorised function to calculate SOFT Scores for predicting patient survival after liver transplantation. The units for donor serum creatinine and recipient serum albumin in g/l.

Usage

```r
```
## Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age</td>
<td>numeric vector of patient ages in years</td>
</tr>
<tr>
<td>BMI</td>
<td>numeric vector of patient BMI in kg/m2</td>
</tr>
<tr>
<td>PrevTx</td>
<td>numeric vector of number of previous transplants</td>
</tr>
<tr>
<td>AbdoSurg</td>
<td>numeric vector of whether previous abdominal surgery (1 = &quot;yes&quot;, 0 = &quot;no&quot;)</td>
</tr>
<tr>
<td>Albumin</td>
<td>numeric vector of serum albumin in g/dl</td>
</tr>
<tr>
<td>Dx</td>
<td>numeric vector of whether on dialysis before transplant (1 = &quot;yes&quot;, 0 = &quot;no&quot;)</td>
</tr>
<tr>
<td>ICU</td>
<td>numeric vector of whether patients in intensive care unit before transplant (1 = &quot;yes&quot;, 0 = &quot;no&quot;)</td>
</tr>
<tr>
<td>Admitted</td>
<td>numeric vector of whether admitted to hospital pre-transplant (1 = &quot;yes&quot;, 0 = &quot;no&quot;)</td>
</tr>
<tr>
<td>MELD</td>
<td>numeric vector of MELD scores</td>
</tr>
<tr>
<td>LifeSupport</td>
<td>numeric vector of whether on life support pre-transplant (1 = &quot;yes&quot;, 0 = &quot;no&quot;)</td>
</tr>
<tr>
<td>Encephalopathy</td>
<td>numeric vector of whether encephalopathy present (1 = &quot;yes&quot;, 0 = &quot;no&quot;)</td>
</tr>
<tr>
<td>PVThrombosis</td>
<td>numeric vector of whether portal vein thrombosis (1 = &quot;yes&quot;, 0 = &quot;no&quot;)</td>
</tr>
<tr>
<td>Ascites</td>
<td>numeric vector of whether ascites pre-transplant (1 = &quot;yes&quot;, 0 = &quot;no&quot;)</td>
</tr>
<tr>
<td>PortalBleed</td>
<td>numeric vector of whether portal bleeding in 48 hours pre-transplant (1 = &quot;yes&quot;, 0 = &quot;no&quot;)</td>
</tr>
<tr>
<td>DonorAge</td>
<td>numeric vector of donor ages in years</td>
</tr>
<tr>
<td>DonorCVA</td>
<td>numeric vector of whether donor cause of death is CVA/stroke (1 = &quot;yes&quot;, 0 = &quot;no&quot;)</td>
</tr>
<tr>
<td>DonorScr</td>
<td>numeric vector of donor terminal serum creatinine in mg/dl</td>
</tr>
<tr>
<td>National</td>
<td>numeric vector of whether national allocation (1 = &quot;yes&quot;, 0 = &quot;no&quot;)</td>
</tr>
<tr>
<td>CIT</td>
<td>numeric vector of cold ischaemic time in hours</td>
</tr>
</tbody>
</table>

## Details


## Value

type: numeric vector of SOFT Scores

## Examples

road_US(Age = 35, BMI = 20, PrevTx = 0, AbdoSurg = 1, Albumin = 3.0, Dx = 0, ICU = 0, Admitted = 0, MELD = 29, LifeSupport = 0, Encephalopathy = 1, PVThrombosis = 0, Ascites = 1, PortalBleed = 0, DonorAge = 44, DonorCVA = 0, DonorScr = 1.2, National = 0, CIT = 8) # 7
**ukeld**

**UKELD score**

**Description**

A vectorised function to calculate the UKELD score using SI units for bilirubin and creatinine.

**Usage**

```r
ukeld(INR, bili, creat, Na, units = "SI")
```

**Arguments**

- **INR**: numeric vector of INR
- **bili**: numeric vector of bilirubin in µmol/l
- **creat**: numeric vector of creatinine in µmol/l
- **Na**: numeric vector of sodium in mmol/l
- **units**: Units for bilirubin and creatinine ("SI" for µmol/l (default), "US" for mg/dl)

**Details**


**Value**

numeric vector of UKELD scores

**Examples**

```r
ukeld(INR = 1.0, bili = 212, creat = 54, Na = 126)
```

---

**ukeld_US**

**UKELD score (US units)**

**Description**

A vectorised function to calculate the UKELD score using US units for bilirubin and creatinine.

**Usage**

```r
ukeld_US(INR, bili, creat, Na)
```
Arguments

- **INR**: numeric vector of INR
- **bili**: numeric vector of bilirubin in mg/dl
- **creat**: numeric vector of creatinine in mg/dl
- **Na**: numeric vector of sodium in mmol/l

Details


Value

UKELD score

Examples

ukeld_US(INR = 2.0, bili = 1.8, creat = 170, Na = 130)

---

 ukkdri  

*UK Kidney Donor Risk Index (NHSBT, 2019 version)*

---

Description

A vectorised function to calculate the UK Kidney Donor Risk Index as used in the new national kidney matching scheme implemented in September 2019.

Usage

ukkdri(age, height, htn, sex, cmv, gfr, hdays)

Arguments

- **age**: numeric vector of donor age in years
- **height**: numeric vector of donor height in cm
- **htn**: numeric vector of whether donor history of hypertension (1 = yes, 0 = no)
- **sex**: character vector of donor sex ("F" = female, "M" = male)
- **cmv**: numeric vector of whether donor CMV IgG positive (1 = yes, 0 = no)
- **gfr**: numeric vector of donor eGFR at time of donation
- **hdays**: numeric vector of number of days donor in hospital before donation

Details

The UK KDRI is documented in the UK kidney matching policy which can be found on the NHS Blood & Transplant ODT website at www.odt.nhs.uk
**UKKDRI**

**Value**

numeric vector of UK Kidney Donor Risk Index values (2019 version)

**Examples**

```r
ukkdri(age = 50, height = 170, htn = 1, sex = "F", cmv = 0, gfr = 90, hdays = 2)
```

---

**ukkdri_q**

*UK Kidney Donor Risk Index Quartile (2019)*

**Description**

Vectorised function to convert UKKDRI values to quartiles of risk. The function takes a numeric vector of UKKDRI values as input, and returns a vector of quartiles. By default this is also a numeric vector with values 1-4, but this can be changed to a character string vector of D1-D4 to match the nomenclature in the NHSBT ODT documentation by setting the prefix parameter to TRUE. The output can also be as a vector of factors by setting fct to TRUE (this can be combined with the prefix parameter).

**Usage**

```r
ukkdri_q(dri, prefix = FALSE, fct = FALSE)
```

**Arguments**

- `dri`: numeric vector of UKKDRI values
- `prefix`: whether to prefix results with "D" (default FALSE)
- `fct`: whether to return results as a factor (default FALSE)

**Details**

The UK KDRI quartile ranges are documented in the UK kidney matching policy which can be found on the NHS Blood & Transplant ODT website at www.odt.nhs.uk

**Value**

vector of UKKDRI quartiles

**Examples**

```r
# obtain quartile of a single value
ukkdri_q(1.01)

# factor vector of results with prefix
dri = c(0.69, 1.01, 1.36, 1.54)
ukkdri_q(dri, prefix = TRUE, fct = TRUE)
```
**ukkrri**  
*UK Kidney Recipient Risk Index (NHSBT, 2019 version)*

**Description**
A vectorised function to calculate the UK Kidney Recipient Risk Index as used in the new national kidney matching scheme implemented in September 2019.

**Usage**
```
ukkrri(age, dx, wait, dm)
```

**Arguments**
- **age**: numeric vector of patient ages in years
- **dx**: numeric vector of whether on dialysis at time of listing (1 = yes, 0 = no)
- **wait**: numeric vector of waiting time from start of dialysis
- **dm**: numeric vector of whether patient has diabetes (1 = yes, 0 = no)

**Details**
The UK KRRI is documented in the UK kidney matching policy which can be found on the NHS Blood & Transplant ODT website at www.odt.nhs.uk

**Value**
numeric vector of UK Kidney Recipient Risk Index values

**Examples**
```
ukkrri(age = 45, dx = 0, wait = 750, dm = 0)
```

---

**ukkrri_q**  
*UK Kidney Recipient Risk Index Quartile (2019)*

**Description**
Vectorised function to convert UKKRRI values to quartiles of risk. The function takes a numeric vector of UKKRRI values as input, and returns a vector of quartiles. By default this is also a numeric vector with values 1-4, but this can be changed to a character string vector of R1-R4 to match the nomenclature in the NHSBT ODT documentation by setting the prefix parameter to TRUE. The output can also be as a vector of factors by setting fct to TRUE (this can be combined with the prefix parameter).
**urea_to_bun**

**Usage**

`ukkrri_q(rri, prefix = FALSE, fct = FALSE)`

**Arguments**

- `rri`: numeric vector of UKKRRI values
- `prefix`: whether to prefix results with "R" (default FALSE)
- `fct`: whether to return results as a factor (default FALSE)

**Details**

The UK KRRI quartile ranges are documented in the UK kidney matching policy which can be found on the NHS Blood & Transplant ODT website at www.odt.nhs.uk

**Value**

vector of UKKRRI quartiles

**Examples**

```r
# obtain quartile of a single value
ukkrri_q(1.01)

# factor vector of results with prefix
rri = c(0.69, 0.75, 0.96, 1.36)
ukkrri_q(rri, prefix = TRUE, fct = TRUE)
```

---

**urea_to_bun**

*Convert urea to BUN*

**Description**

A vectorised function to convert urea to blood urea nitrogen (BUN). By default the urea is measured in mmol/l but this can be changed to mg/dl by setting the optional units parameter to "US"

**Usage**

`urea_to_bun(urea, units = "SI")`

**Arguments**

- `urea`: numeric vector of urea levels (mmol/l by default)
- `units`: units for urea ("SI" for mmol/l, "US" for mg/dl)

**Value**

numeric vector of blood urea nitrogen (BUN) levels in mg/dl
Examples

urea_to_bun(5.4)

uskdri

US Kidney Donor Risk Index

Description

Vectorised function to calculate US Kidney Donor Risk Index as published by UNOS. Please note that this function uses creatinine measured in µmol/l by default, but can be changed to mg/dl if the optional units parameter is set to "US".

Usage

uskdri(age, height, weight, eth, htn, dm, cva, creat, hcv, dcd,
       scaling = 1, units = "SI")

Arguments

age    numeric vector of donor ages in years
height numeric vector of donor heights in cm
weight numeric vector of donor weights in kg
eth    character string vector of donor ethnicity, "black" or "non-black"
htn    numeric vector of donor hypertension history (1 = yes, 0 = no)
dm     numeric vector of donor diabetes history (1 = yes, 0 = no)
cva    numeric vector of whether donor death due to CVA (1 = yes, 0 = no)
creat  numeric vector of donor serum creatinine (µmol/l)
hcv    numeric vector of donor hepatitis C history (1 = yes, 0 = no)
dcd    numeric vector of type of donor (1 = DCD, 0 = DBD)
scaling single numeric value for OPTN scaling factor (optional, defaults to 1)
units  single string value to indicate creatinine units ("SI" for µmol/l, "US" for mg/dl)

Details

The KDRI is normalised by a scaling factor based on the median KDRI in the previous year. For 2018, this was approximately 1.250609 as is published on the OPTN website. The scaling parameter in this function defaults to 1, so can be left out to calculate the KDRI without scaling.


Value

numeric vector of US KDRI values
Examples

# with creatinine in μmol/l (units = "SI" can be omitted)
uskdri(age = 40, height = 170, weight = 80, eth = "non-black", htn = 0, dm = 0,
cva = 0, creat = 120, hcv = 0, dcd = 0, scaling = 1.250609, units = "SI")

# with creatinine in mg/dl
uskdri(age = 40, height = 170, weight = 80, eth = "non-black", htn = 0, dm = 0,
cva = 0, creat = 1.4, hcv = 0, dcd = 0, scaling = 1.250609, units = "US")

uskdri_US

US Kidney Donor Risk Index (US units)

Description

Wrapper function for the uskdri() vectorised function to calculate US Kidney Donor Risk Index as published by UNOS but using mg/dl as the units for creatinine.

Usage

uskdri_US(age, height, weight, eth, htn, dm, cva, creat, hcv, dcd,
   scaling = 1)

Arguments

age numeric vector of donor ages in years
height numeric vector of donor heights in cm
weight numeric vector of donor weights in kg
eth character string vector of donor ethnicity, "black" or "non-black"
htn numeric vector of donor hypertension history (1 = yes, 0 = no)
dm numeric vector of donor diabetes history (1 = yes, 0 = no)
cva numeric vector of whether donor death due to CVA (1 = yes, 0 = no)
creat numeric vector of donor serum creatinine (mg/dl)
hcv numeric vector of donor hepatitis C history (1 = yes, 0 = no)
dcd numeric vector of type of donor (1 = DCD, 0 = DBD)
scaling single numeric value for OPTN scaling factor (optional, defaults to 1)

Details

The KDRI is normalised by a scaling factor based on the median KDRI in the previous year. For 2018, this was approximately 1.250609 as is published on the OPTN website. The scaling parameter in this function defaults to 1, so can be left out to calculate the KDRI without scaling.

Value
	numeric vector of US KDRI values

Examples

```r
uskdri(age = 40, height = 170, weight = 80, eth = "non-black", htn = 0, dm = 0,
cva = 0, creat = 1.4, hcv = 0, dcd = 0)
```

---

**walser**

*eGFR using the Walser formula*

Description

A vectorised function to calculate eGFR using the Walser formula. By default the equation accepts serum creatinine in µmol/l but can be changed to mg/dl by setting the units parameter to "US". It has not been possible to run automated tests to confirm the accuracy of this function, so please use with caution.

Usage

```r
walser(SCr, Age, Weight, Sex, Units = "SI")
```

Arguments

- **SCr**: numeric vector of serum creatinine in µmol/l (or mg/dl if units = "US")
- **Age**: numeric vector of patient ages in years
- **Weight**: numeric vector of patient weights in kilograms
- **Sex**: character vector of sex ("F" for female, "M" for male)
- **Units**: non-vectorised optional parameter for creatinine units ("SI" for µmol/l (default), "US" for mg/dl)

Details


Value

numeric vectors of eGFR values

Examples

```r
walser(SCr = 118, Age = 74, Weight = 65, Sex = "M") # 56.1
```
walser_US

Description

A wrapper function for the walser() vectorised function to calculate eGFR using the Walser formula, using serum creatinine in mg/dl. It has not been possible to run automated tests to confirm the accuracy of this function, so please use with caution.

Usage

walser_US(SCr, Age, Weight, Sex)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCr</td>
<td>numeric vector of serum creatinine in µmol/l (or mg/dl if units = &quot;US&quot;)</td>
</tr>
<tr>
<td>Age</td>
<td>numeric vector of patient ages in years</td>
</tr>
<tr>
<td>Weight</td>
<td>numeric vector of patient weights in kilograms</td>
</tr>
<tr>
<td>Sex</td>
<td>character vector of sex (&quot;F&quot; for female, &quot;M&quot; for male)</td>
</tr>
</tbody>
</table>

Details


Value

numeric vectors of eGFR values

Examples

walser_US(SCr = 1.33, Age = 74, Weight = 65, Sex = "M") # 56.3

watson_ukkdri

UK Kidney Donor Risk Index (2012 version)

Description

A vectorised function to calculate the UK Kidney Donor Risk Index as published by Watson et al. in 2012. Please note that this is not the same risk index as used in the new UK kidney matching scheme starting in September 2019.

Usage

watson_ukkdri(age, htn, weight, hdays, adrenaline)
Arguments

- **age**: numeric vector of donor ages
- **htn**: numeric vector of whether donor history of hypertension (1 = yes, 0 = no)
- **weight**: numeric vector of donor weights in kg
- **hdays**: numeric vector of donor length of hospital stay
- **adrenaline**: numeric vector of whether donor treated with adrenaline (1 = yes, 0 = no)

Details


Value

numeric vector of UK Kidney Donor Risk Index (2012) values

Examples

```
watson_ukkdri(age = 40, htn = 0, weight = 75, hdays = 0, adrenaline = 0) # 1.00
```
Index

* datasets
  kidney.donors, 19
  liver.pts, 19
  mismatches, 26
  results, 39
  results_US, 39
  serial.results, 41

apri, 3

bar_score, 4
bilirubin_to_SI, 5
bilirubin_to_US, 5
bun_to_urea, 6

chi2dob, 6
ckd_epi, 7
ckd_epi_US, 8
cockcroft, 9
cockcroft_US, 10
creatinine_to_SI, 10
creatinine_to_US, 11

epts, 11
epts_lookup, 12
et_dri, 13

hla_mm_level, 14
hla_mm_level_str, 14

ibw, 15

kdpi, 16
kdpi_lookup, 17
kdpi_US, 18
kidney.donors, 19

liver.pts, 19
liver_dri, 20

mdrd, 21

mdrd_US, 22
meld, 23
meld_na, 24
meld_na_US, 25
meld_US, 25
mismatches, 26

nankivell, 27
nankivell_spk, 28
nankivell_spk_US, 29
nankivell_US, 30

p_pass, 34
p_soft, 36
p_soft_US, 37
pdri, 30
pedi_soft, 32
peld, 33
peld_US, 34

raw_epts, 38
results, 39
results_US, 39

schwartz, 40
schwartz_US, 41
serial_results, 41
soft, 42
soft2, 43
soft2_US, 44
soft_US, 45

ukeld, 47
ukeld_US, 47
ukkdri, 48
ukkdri_q, 49
ukkrri, 50
ukkrri_q, 50
urea_to_bun, 51
uskdri, 52
uskdri_US, 53
walser, 54
walser_US, 55
watson_ukkdri, 55