Package ‘combinedevents’

October 12, 2022

Title  Calculate Scores and Marks for Track and Field Combined Events
Version  0.1.1
Description  Includes functions to calculate scores and marks for track and field combined events competitions. The functions are based on the scoring tables for combined events set forth by the International Association of Athletics Federation (2001).
License  GPL-3
BugReports  https://github.com/katie-frank/combinedevents/issues
Encoding  UTF-8
LazyData  true
RoxygenNote  7.1.1
Imports  lubridate, magrittr, rlang, stats, stringr
Suggests  knitr, rmarkdown, testthat, spelling, covr
Depends  R (>= 2.10)
Language  en-US
NeedsCompilation  no
Author  Katie Frank [aut, cre] (<https://orcid.org/0000-0002-0353-0328>)
Maintainer  Katie Frank <katiexfrank@gmail.com>
Repository  CRAN
Date/Publication  2021-02-03 22:20:02 UTC

R topics documented:

  combinedevents-package ................................................. 2
  combined_events ....................................................... 2
  combined_events_null .................................................. 3
  dec ........................................................................ 4
  marks ....................................................................... 5
  scores ....................................................................... 6
combined_events-package

combinedevents: Calculate Scores and Marks for Track and Field
Combined Events

Description

The package includes functions to calculate scores and marks for track and field combined events competitions. The functions are based on the scoring tables for combined events set forth by the International Association of Athletics Federation (2001).

Author(s)

Maintainer: Katie Frank <katiexfrank@gmail.com>

References


See Also

Useful links:
- https://katie-frank.github.io/combinedevents/
- https://github.com/katie-frank/combinedevents
- Report bugs at https://github.com/katie-frank/combinedevents/issues

combined_events

Combined events results

Description

combined_events() is a generic function used to present results of calls to scores() and marks().

Usage

combined_events(marks, scores, event_names, event, seconds, ...)

Arguments

marks a numeric vectors of marks
scores an integer vector of scores
event_names a character vector of event names
event a character string indicating the combined events competition
seconds a numeric (either 0 or 1)
... other arguments passed on to methods
combined_events_null

Value

An object of class "combined_events". The default method returns a list of that class.

See Also

scores(), marks()

combined_events_null  Combined events null results

Description

combined_events_null() is a generic function used to present results of calls to scores() and marks() where in those calls combined_event = NULL.

Usage

combined_events_null(marks, scores, event_names, seconds, ...)

Arguments

marks            a numeric vector of marks
scores           an integer vector of scores
event_names      a character vector of event names
seconds          a numeric (either 0 or 1)
...              other arguments passed on to methods

Value

An object of class combined_events_null. The default method returns a list of that class.

See Also

scores(), marks()
## Description

A dataset containing the performances of 23 athletes in the men’s decathlon at the 2016 Summer Olympics.

## Usage

dec

## Format

A data frame with 23 rows and 24 variables. The variables `100m`, `LJ`, `SP`, `HJ`, `400m`, `110mH`, `DT`, `PV`, `JT`, and `1500m` correspond to the performances of the athletes for the ten events comprising the decathlon. Those variables ending in `_p` (e.g., `100m_p`) represent the points athletes earn for their performances in each of the ten events. A full description of the 24 variables is below.

<table>
<thead>
<tr>
<th>rank</th>
<th>rank of athlete</th>
</tr>
</thead>
<tbody>
<tr>
<td>athlete</td>
<td>name of athlete</td>
</tr>
<tr>
<td>nationality</td>
<td>nationality of athlete</td>
</tr>
<tr>
<td>score_total</td>
<td>overall score</td>
</tr>
<tr>
<td>100m</td>
<td>100m result, in seconds</td>
</tr>
<tr>
<td>100m_p</td>
<td>100m points</td>
</tr>
<tr>
<td>LJ</td>
<td>long jump result, in meters</td>
</tr>
<tr>
<td>LJ_p</td>
<td>long jump points</td>
</tr>
<tr>
<td>SP</td>
<td>shot put result, in meters</td>
</tr>
<tr>
<td>SP_p</td>
<td>shot put points</td>
</tr>
<tr>
<td>HJ</td>
<td>high jump result, in meters</td>
</tr>
<tr>
<td>HJ_p</td>
<td>high jump points</td>
</tr>
<tr>
<td>400m</td>
<td>400m result, in seconds</td>
</tr>
<tr>
<td>400m_p</td>
<td>400m points</td>
</tr>
<tr>
<td>110mH</td>
<td>110m hurdles result, in seconds</td>
</tr>
<tr>
<td>110mH_p</td>
<td>110m hurdles points</td>
</tr>
<tr>
<td>DT</td>
<td>discus throw result, in meters</td>
</tr>
<tr>
<td>DT_p</td>
<td>discus throw points</td>
</tr>
<tr>
<td>PV</td>
<td>pole vault result, in meters</td>
</tr>
<tr>
<td>PV_p</td>
<td>pole vault points</td>
</tr>
<tr>
<td>JT</td>
<td>javelin throw result, in meters</td>
</tr>
<tr>
<td>JT_p</td>
<td>javelin throw points</td>
</tr>
<tr>
<td>1500m</td>
<td>1500m result, in the format mm:ss.ms</td>
</tr>
<tr>
<td>1500m_p</td>
<td>1500m points</td>
</tr>
</tbody>
</table>
marks

Source
https://en.wikipedia.org/wiki/Athletics_at_the_2016_Summer_Olympics_%2D_Men%27s_decathlon

marks

Calculate marks for track and field combined events

Description

marks() calculates marks for track and field combined events competitions.

Usage

marks(scores, gender, combined_event = NULL, seconds = FALSE)

Arguments

scores a numeric vector of track and field scores

gender gender of athlete; either "male" or "female"

combined_event an optional character string indicating the combined events competition. For gender = "male", the options are "decathlon"/"outdoor decathlon", "outdoor pentathlon", "heptathlon"/"indoor heptathlon", and "indoor pentathlon". For gender = "female", the options are "heptathlon"/"outdoor heptathlon", "decathlon"/"outdoor decathlon", and "pentathlon"/"indoor pentathlon".

If combined_event = NULL, the elements of scores must be named.

• For gender = "male", the allowed names for the elements of scores are `100m`, `LJ`, `SP`, `HJ`, `400m`, `110mH`, `DT`, `PV`, `JT`, `1500m`, `200m`, `60m`, `60mH`, and `1000m`.

• For gender = "female", the allowed names are `100m`, `LJ`, `SP`, `HJ`, `400m`, `100mH`, `DT`, `PV`, `JT`, `1500m`, `200m`, `60mH`, and `800m`.

seconds a logical; if TRUE, will return all track event marks in seconds

Details

marks() performs the opposite action of scores(): you give it the scores you want to obtain, and it gives you the marks you need to achieve those scores. For track events, marks() returns the slowest time needed to achieve the input score. Similarly, for jumping and throwing events, marks() returns the shortest distance necessary to achieve the input score.

For some events, when a score is given to marks(), the score returned may be different from the one input because some scores are not actually possible (due to rounding of track and field marks). When an impossible score is given to marks(), the function will return the closest higher score that corresponds to a mark.
Value

A list of class "combined_events" (or "combined_events_null" if combined_event = NULL) with the following fields:

- **results**: if called with non-NULL combined_event, a data frame with columns for the specified combined event containing the names of those events, mark for the resulting marks based on the input scores, and score based on the input scores. The last row of the data frame gives the total score for the specified combined events competition. If combined_event = NULL, a data frame with columns event, mark, and score.

- **marks**: the vector of marks based on the input scores for the specified combined event. If not all scores were supplied to marks(), then there will be NA values for those events with missing scores. If combined_event = NULL, the vector of marks.

- **scores**: the vector of scores for the specified combined event. If not all scores were supplied to marks(), then there will be NA values for those events with missing scores. If combined_event = NULL, the vector of scores.

- **score_total**: if called with non-NULL combined_event, an integer representing the overall score for the specified combined events competition.

- **call**: the matched call

References


Examples

```r
# Men's heptathlon
marks(scores = rep(800, 7),
     gender = "male", combined_event = "heptathlon")

# Women's pentathlon
marks(scores = c("60mH" = 981, HJ = 875, SP = 799, LJ = 956, "800m" = 1000),
      "female", "pentathlon")

# Men's events
marks(scores = c(LJ = 790, LJ = 810, HJ = 850, HJ = 900, PV = 900, PV = 915),
      "male")
```

scores

*Calculate scores for track and field combined events*

Description

scores() calculates scores for track and field combined events competitions.

Usage

scores(marks, gender, combined_event = NULL, seconds = FALSE)
Arguments

marks  a numeric or character vector of track and field marks/performances

gender  gender of athlete; either "male" or "female"

combined_event  an optional character string indicating the combined events competition. For
gender = "male", the options are "decathlon"/"outdoor decathlon", "outdoor pentathlon", "heptathlon"/"indoor heptathlon", and "indoor pentathlon".
For gender = "female", the options are "heptathlon"/"outdoor heptathlon", "decathlon"/"outdoor decathlon", and "pentathlon"/"indoor pentathlon".
If combined_event = NULL, the elements of marks must be named.

  • For gender = "male", the allowed names for the elements of marks are
    `"100m", "LJ", "SP", "HJ", "400m", "110mH", "DT", "PV", "JT", "1500m", "200m", "60m", "60mH", and "1000m".

  • For gender = "female", the allowed names are `"100m", "LJ", "SP", "HJ", "400m", "100mH", "DT", "PV", "JT", "1500m", "200m", "60mH", and "800m".

seconds  a logical; if TRUE, will return all track event marks in seconds

Value

A list of class "combined_events" (or "combined_events_null" if combined_event = NULL) with the following fields:

results  if called with non-NULL combined_event, a data frame with columns for the
specified combined event containing the names of those events, mark for the
input marks/performances, and score for the resulting scores based on those
marks. The last row of the data frame gives the total score for the specified
combined events competition. If combined_event = NULL, a data frame with
columns event, mark, and score.

marks  the vector of marks for the specified combined event. If not all marks were
supplied to scores(), then there will be NA values for those events with missing
marks. If combined_event = NULL, the vector of marks.

scores  the vector of scores based on the input marks for the specified combined event.
If not all marks were supplied to scores(), then there will be scores with NA
values for those events with missing marks. If combined_event = NULL, the
vector of scores.

score_total  if called with non-NULL combined_event, an integer representing the overall
score for the specified combined events competition

call  the matched call

References


Examples

# Men's decathlon
scores(marks = c("100m" = 11.61, LJ = 7.32, SP = 13.17, HJ = 1.9,
'400m' = 49.96, '110mH' = 15.32, DT = 38.18, PV = 4.6, JT = 58.98, '1500m' = "4:39.34"), gender = "male", combined_event = "decathlon")

# Women's heptathlon
scores(c(14.11, 1.95, 13.96, 25.61, 6.44, 45.98, "2:07.26"), "female", "heptathlon")

# Men's events
scores(c('60m' = 7.09, LJ = 7, LJ = 7.03, SP = 11.8, HJ = 2, '60mH' = 8.30, '60mH' = 9.31, PV = 4.30, '1000m' = "2:40.00"), gender = "male")
Index

* datasets
  dec, 4
combined_events, 2
combined_events_null, 3
combinedevents-package, 2
dec, 4
marks, 5
marks(), 2, 3
scores, 6
scores(), 2, 3, 5