# Package 'injurytools'

November 28, 2025

**Title** A Toolkit for Sports Injury and Illness Data Analysis

**Version** 2.0.0

**Description** Sports Injury Data analysis aims to identify and describe the magnitude of the injury problem, and to gain more insights (e.g. determine potential risk factors) by statistical modelling approaches. The 'injurytools' package provides standardized routines and utilities that simplify such analyses. It offers functions for data preparation, informative visualizations and descriptive and model-based analyses.

**License** MIT + file LICENSE

**URL** https://github.com/lzumeta/injurytools, https://lzumeta.github.io/injurytools/

**BugReports** https://github.com/lzumeta/injurytools/issues

**Depends** R (>= 4.1.0)

**Imports** checkmate, dplyr, forcats, ggplot2, lubridate, metR, purrr, rlang, scales, stats, stringr, tibble, tidyr, tidyselect, withr

**Suggests** coxme, grid, gridExtra, kableExtra, knitr, lme4, MASS, pscl, RColorBrewer, rmarkdown, spelling, survival, survminer, testthat (>= 3.0.0)

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**Encoding** UTF-8

**Language** en-US

**LazyData** true

**RoxygenNote** 7.3.2

**NeedsCompilation** no

**Author** Lore Zumeta Olaskoaga [aut, cre] (ORCID: <https://orcid.org/0000-0001-6141-1469>), Dae-Jin Lee [ctb] (ORCID: <https://orcid.org/0000-0002-8995-8535>)

**Maintainer** Lore Zumeta Olaskoaga <lorezumeta@gmail.com>

**Repository** CRAN

**Date/Publication** 2025-11-28 13:00:17 UTC

# Contents



---

`calc_burden` *Calculate case burden rate*

---

## Description

Calculate the case burden rate of a sports-related health problem (e.g. disease, injury) in a cohort.

## Usage

```
calc_burden(
  injd,
  by = NULL,
  overall = TRUE,
  method = c("poisson", "negbin", "zinfpois", "zinfnb"),
  se = TRUE,
  conf_level = 0.95,
  scale = TRUE,
  quiet = FALSE
)
```

### Arguments

| | |
|---|---|
| injd | injd **S3** object (see prepare_all()). |
| by | Character specifying the name of the column according to which compute summary statistics. It should refer to a (categorical) variable that describes a grouping factor (e.g. "type of case or injury", "injury location", "sports club"). Optional, defaults to NULL. |
| overall | Logical, whether to calculate overall (for all the cohort) or athlete-wise summary statistic (i.e. number of cases per cohort of per athlete). Defaults to TRUE. |
| method | Method to estimate the incidence (burden) rate. One of "poisson", "negbin", "zinfpois" or "zinfnb"; that stand for Poisson method, negative binomial method, zero-inflated Poisson and zero-inflated negative binomial. |
| se | Logical, whether to calculate the confidence interval related to the rate. |
| conf_level | Confidence level (defaults to 0.95). |
| scale | Logical, whether to transform the incidence and burden rates output according to the unit of exposure (defaults to TRUE). |
| quiet | Logical, whether or not to silence the warning messages (defaults to FALSE). |

### Value

The case burden rate. Either a numeric value (if overall TRUE) or a data frame indicating the case burden rate per athlete.

### References

Bahr R., Clarsen B., & Ekstrand J. (2018). Why we should focus on the burden of injuries and illnesses, not just their incidence. *British Journal of Sports Medicine*, 52(16), 1018–1021. doi:10.1136/bjsports2017098160

Waldén M., Mountjoy M., McCall A., Serner A., Massey A., Tol J. L., ... & Andersen T. E. (2023). Football-specific extension of the IOC consensus statement: methods for recording and reporting of epidemiological data on injury and illness in sport 2020. *British journal of sports medicine*.

### Examples

```
calc_burden(injd)
calc_burden(injd, overall = FALSE)
calc_burden(injd, by = "injury_type")
```

---

## calc_exposure *Calculate the exposure time*

---

### Description

Calculate the time of exposure that each athlete, or the entire cohort of athletes, has been at risk for a sport-related health problem.

### Usage

```
calc_exposure(injd, by = NULL, overall = TRUE, quiet = FALSE)
```

### Arguments

| | |
|---|---|
| injd | injd **S3** object (see prepare_all()). |
| by | Character specifying the name of the column according to which compute summary statistics. It should refer to a (categorical) variable that describes a grouping factor (e.g. "type of case or injury", "injury location", "sports club"). Optional, defaults to NULL. |
| overall | Logical, whether to calculate overall (for all the cohort) or athlete-wise summary statistic (i.e. number of cases per cohort of per athlete). Defaults to TRUE. |
| quiet | Logical, whether or not to silence the warning messages (defaults to FALSE). |

### Value

The total exposure time. Either a numeric value (if overall TRUE) or a data frame indicating the total exposure time for each athlete.

### Examples

```
calc_exposure(injd)
calc_exposure(injd, overall = FALSE)
calc_exposure(injd, by = "injury_type")
```

---

## calc_incidence *Calculate case incidence rate*

---

### Description

Calculate the case incidence rate of a sports-related health problem (e.g. disease, injury) in a cohort.

### Usage

```
calc_incidence(
  injd,
  by = NULL,
  overall = TRUE,
  method = c("poisson", "negbin", "zinfpois", "zinfnb"),
  se = TRUE,
  conf_level = 0.95,
  scale = TRUE,
  quiet = FALSE
)
```

## Arguments

| | |
|---|---|
| injd | injd **S3** object (see prepare_all()). |
| by | Character specifying the name of the column according to which compute summary statistics. It should refer to a (categorical) variable that describes a grouping factor (e.g. "type of case or injury", "injury location", "sports club"). Optional, defaults to NULL. |
| overall | Logical, whether to calculate overall (for all the cohort) or athlete-wise summary statistic (i.e. number of cases per cohort of per athlete). Defaults to TRUE. |
| method | Method to estimate the incidence (burden) rate. One of "poisson", "negbin", "zinfpois" or "zinfnb"; that stand for Poisson method, negative binomial method, zero-inflated Poisson and zero-inflated negative binomial. |
| se | Logical, whether to calculate the confidence interval related to the rate. |
| conf_level | Confidence level (defaults to 0.95). |
| scale | Logical, whether to transform the incidence and burden rates output according to the unit of exposure (defaults to TRUE). |
| quiet | Logical, whether or not to silence the warning messages (defaults to FALSE). |

## Value

The case incidence rate. Either a numeric value (if overall TRUE) or a data frame indicating the case incidence rate per athlete.

## References

Bahr R., Clarsen B., & Ekstrand J. (2018). Why we should focus on the burden of injuries and illnesses, not just their incidence. *British Journal of Sports Medicine*, 52(16), 1018–1021. doi:10.1136/bjsports2017098160

Waldén M., Mountjoy M., McCall A., Serner A., Massey A., Tol J. L., ... & Andersen T. E. (2023). Football-specific extension of the IOC consensus statement: methods for recording and reporting of epidemiological data on injury and illness in sport 2020. *British journal of sports medicine*.

## Examples

```
calc_incidence(injd)
calc_incidence(injd, overall = FALSE)
calc_incidence(injd, by = "injury_type")
calc_incidence(injd, by = "injury_type", scale = FALSE)
```

## calc_iqr_dayslost *Calculate the interquartile range days lost*

### Description

Calculate the interquartile range of the days lost due to a sports-related health problem (e.g. disease, injury) in a cohort.

### Usage

```
calc_iqr_dayslost(injd, by = NULL, overall = TRUE)
```

### Arguments

| | |
|---|---|
| injd | injd **S3** object (see prepare_all()). |
| by | Character specifying the name of the column according to which compute summary statistics. It should refer to a (categorical) variable that describes a grouping factor (e.g. "type of case or injury", "injury location", "sports club"). Optional, defaults to NULL. |
| overall | Logical, whether to calculate overall (for all the cohort) or athlete-wise summary statistic (i.e. number of cases per cohort of per athlete). Defaults to TRUE. |

### Value

The interquartile range of the days lost. Either a numeric value (if overall TRUE) or a data frame indicating the interquartile range of the days lost per athlete.

### Examples

```
calc_iqr_dayslost(injd)
calc_iqr_dayslost(injd, overall = FALSE)
calc_iqr_dayslost(injd, by = "injury_type")
```

## calc_mean_dayslost *Calculate the mean days lost*

### Description

Calculate the mean of the days lost due to a sports-related health problem (e.g. disease, injury) in a cohort.

### Usage

```
calc_mean_dayslost(injd, by = NULL, overall = TRUE)
```

### Arguments

| | |
|---|---|
| injd | injd **S3** object (see prepare_all()). |
| by | Character specifying the name of the column according to which compute summary statistics. It should refer to a (categorical) variable that describes a grouping factor (e.g. "type of case or injury", "injury location", "sports club"). Optional, defaults to NULL. |
| overall | Logical, whether to calculate overall (for all the cohort) or athlete-wise summary statistic (i.e. number of cases per cohort of per athlete). Defaults to TRUE. |

### Value

The mean of the days lost. Either a numeric value (if overall TRUE) or a data frame indicating the mean days lost per athlete.

### Examples

```
calc_mean_dayslost(injd)
calc_mean_dayslost(injd, overall = FALSE)
calc_mean_dayslost(injd, by = "injury_type")
```

---

## calc_median_dayslost *Calculate the median days lost*

---

### Description

Calculate the median of the days lost due to a sports-related health problem (e.g. disease, injury).

### Usage

```
calc_median_dayslost(injd, by = NULL, overall = TRUE)
```

### Arguments

| | |
|---|---|
| injd | injd **S3** object (see prepare_all()). |
| by | Character specifying the name of the column according to which compute summary statistics. It should refer to a (categorical) variable that describes a grouping factor (e.g. "type of case or injury", "injury location", "sports club"). Optional, defaults to NULL. |
| overall | Logical, whether to calculate overall (for all the cohort) or athlete-wise summary statistic (i.e. number of cases per cohort of per athlete). Defaults to TRUE. |

### Value

The median of the days lost. Either a numeric value (if overall TRUE) or a data frame indicating the median days lost per athlete.

## Examples

```
calc_median_dayslost(injd)
calc_median_dayslost(injd, overall = FALSE)
calc_median_dayslost(injd, by = "injury_type")
```

---

`calc_ncases` *Calculate number of cases*

---

## Description

Calculate the number of sports-related cases (e.g. injuries) that occurred in a cohort during a period.

## Usage

```
calc_ncases(injd, by = NULL, overall = TRUE)
```

## Arguments

| | |
|---|---|
| `injd` | `injd` **S3** object (see `prepare_all()`). |
| `by` | Character specifying the name of the column according to which compute summary statistics. It should refer to a (categorical) variable that describes a grouping factor (e.g. "type of case or injury", "injury location", "sports club"). Optional, defaults to `NULL`. |
| `overall` | Logical, whether to calculate overall (for all the cohort) or athlete-wise summary statistic (i.e. number of cases per cohort of per athlete). Defaults to `TRUE`. |

## Value

The number of cases. Either a numeric value (if overall `TRUE`) or a data frame indicating the number of cases per athlete.

## Examples

```
calc_ncases(injd)
calc_ncases(injd, overall = FALSE)
calc_ncases(injd, by = "injury_type")
```

## calc_ndayslost *Calculate number of days lost*

### Description

Calculate the number of days lost due to a sports-related health problem (e.g. injuries) in a cohort during a period.

### Usage

```
calc_ndayslost(injd, by = NULL, overall = TRUE)
```

### Arguments

| | |
|---|---|
| injd | injd **S3** object (see prepare_all()). |
| by | Character specifying the name of the column according to which compute summary statistics. It should refer to a (categorical) variable that describes a grouping factor (e.g. "type of case or injury", "injury location", "sports club"). Optional, defaults to NULL. |
| overall | Logical, whether to calculate overall (for all the cohort) or athlete-wise summary statistic (i.e. number of cases per cohort of per athlete). Defaults to TRUE. |

### Value

The number of days lost. Either a numeric value (if overall TRUE) or a data frame indicating the number of cases per athlete.

### Examples

```
calc_ndayslost(injd)
calc_ndayslost(injd, overall = FALSE)
calc_ndayslost(injd, by = "injury_type")
```

## calc_prevalence *Calculate prevalence proportion*

### Description

Calculate the prevalence proportion of injured athletes and the proportion of non-injured (available) athletes in the cohort, on a monthly or season basis. Further information on the type of injury may be specified so that the injury-specific prevalences are reported according to this variable.

### Usage

```
calc_prevalence(injd, time_period = c("monthly", "season"), by = NULL)
```

## Arguments

| | |
|---|---|
| injd | Prepared data. An injd object. |
| time_period | Character. One of "monthly" or "season", specifying the periodicity according to which to calculate the proportions of available and injured athletes. |
| by | Character specifying the name of the column on the basis of which to classify the injuries and calculate proportions of the injured athletes. Defaults to NULL. |

## Value

A data frame containing one row for each combination of season, month (optionally) and injury type (if by not specified, then this variable has two categories: *Available* and *Injured*). Plus, three more columns, specifying the proportion of athletes (prop) satisfying the corresponding row's combination of values, i.e. prevalence, how many athletes were injured at that moment with the type of injury of the corresponding row (n), over how many athletes were at that time in the cohort (n_athlete). See Note section.

## Note

If by is specified (and not NULL), it may happen that an athlete in one month suffers two different types of injuries. For example, a muscle and a ligament injury. In this case, this two injuries contribute to the proportions of muscle and ligament injuries for that month, resulting in an overall proportion that exceeds 100%. Besides, the athletes in Available category are those that did not suffer any injury in that moment (season-month), that is, they were healthy all the time that the period lasted.

## References

Bahr R, Clarsen B, Derman W, et al. International Olympic Committee consensus statement: methods for recording and reporting of epidemiological data on injury and illness in sport 2020 (including STROBE Extension for Sport Injury and Illness Surveillance (STROBE-SIIS)) *British Journal of Sports Medicine* 2020; 54:372-389.

Nielsen RO, Debes-Kristensen K, Hulme A, et al. Are prevalence measures better than incidence measures in sports injury research? *British Journal of Sports Medicine* 2019; 54:396-397.

## Examples

```
df_exposures <- prepare_exp(raw_df_exposures, person_id = "player_name",
                            date = "year", time_expo = "minutes_played")
df_injuries  <- prepare_inj(raw_df_injuries, person_id = "player_name",
                            date_injured = "from", date_recovered = "until")
injd         <- prepare_all(data_exposures = df_exposures,
                            data_injuries  = df_injuries,
                            exp_unit = "matches_minutes")


calc_prevalence(injd, time_period = "monthly", by = "injury_type")
calc_prevalence(injd, time_period = "monthly")
calc_prevalence(injd, time_period = "season", by = "injury_type")
calc_prevalence(injd, time_period = "season")
```

`calc_summary` *Calculate summary statistics*

## Description

Calculate epidemiological summary statistics such as case (e.g. injury) incidence and case burden (see Bahr et al. 2018), including total number of cases, number of days lost due to this event, total time of exposure etc., by means of a (widely used) Poisson method, negative binomial, zero-inflated poisson or zero-inflated negative binomial, on a athlete and overall basis.

## Usage

```
calc_summary(
  injd,
  by = NULL,
  overall = TRUE,
  method = c("poisson", "negbin", "zinfpois", "zinfnb"),
  conf_level = 0.95,
  scale = TRUE,
  quiet = FALSE
)
```

## Arguments

| | |
|---|---|
| `injd` | `injd` **S3** object (see `prepare_all()`). |
| `by` | Character specifying the name of the column according to which compute summary statistics. It should refer to a (categorical) variable that describes a grouping factor (e.g. "type of case or injury", "injury location", "sports club"). Optional, defaults to `NULL`. |
| `overall` | Logical, whether to calculate overall (for all the cohort) or athlete-wise summary statistic (i.e. number of cases per cohort of per athlete). Defaults to `TRUE`. |
| `method` | Method to estimate the incidence (burden) rate. One of "poisson", "negbin", "zinfpois" or "zinfnb"; that stand for Poisson method, negative binomial method, zero-inflated Poisson and zero-inflated negative binomial. |
| `conf_level` | Confidence level (defaults to 0.95). |
| `scale` | Logical, whether to transform the incidence and burden rates output according to the unit of exposure (defaults to `TRUE`). |
| `quiet` | Logical, whether or not to silence the warning messages (defaults to `FALSE`). |

## Value

A data frame comprising of overall or athlete-wise epidemiological summary statistics, that it's made up of the following columns:

- `totalexpo`: total exposure that the athlete has been under risk of suffering a sports-related health problem.

- ncases: number of sports-related health problems suffered by the athlete or overall in the team/cohort over the given period specified by the injd data frame.
- ndayslost: number of days lost by the athlete or overall in the team/cohort due to the sports-related health problem over the given period specified by the injd data frame.
- mean_dayslost: average of number of days lost (i.e. ndayslost) athlete-wise or overall in the team/cohort.
- median_dayslost: median of number of days lost (i.e. ndayslost) athlete-wise or overall in the team/cohort.
- qt25_dayslost and qt75_dayslost: interquartile range of number of days lost (i.e. ndayslost) athlete-wise or overall in the team/cohort.
- incidence: case incidence rate, number of cases per unit of exposure.
- burden: case burden rate, number of days lost per unit of exposure.
- incidence_sd and burden_sd: estimated standard deviation, by the specified method argument, of case incidence (incidence) and case burden (burden).
- incidence_lower and burden_lower: lower bound of, for example, 95% confidence interval (if conf_level = 0.95) of case incidence (incidence) and case burden (burden).
- incidence_upper and burden_upper: the same (as above item) applies but for the upper bound.

Apart from this column names, they may further include these other columns depending on the user's specifications to the function:

- by: only if it is specified as an argument to function.
- percent_ncases: percentage (%) of number of cases of that type relative to all types of cases (if by specified).
- percent_dayslost: percentage (%) of number of days lost because of cases of that type relative to the total number of days lost because of all types of cases (if by specified).

## References

Bahr R., Clarsen B., & Ekstrand J. (2018). Why we should focus on the burden of injuries and illnesses, not just their incidence. *British Journal of Sports Medicine*, 52(16), 1018–1021. doi:10.1136/bjsports2017098160

Waldén M., Mountjoy M., McCall A., Serner A., Massey A., Tol J. L., ... & Andersen T. E. (2023). Football-specific extension of the IOC consensus statement: methods for recording and reporting of epidemiological data on injury and illness in sport 2020. *British journal of sports medicine*.

## Examples

```
calc_summary(injd)
calc_summary(injd, overall = FALSE)
calc_summary(injd, by = "injury_type")
calc_summary(injd, by = "injury_type", overall = FALSE)
```

cut_injd *Cut the range of the follow-up*

## Description

Given an `injd` object, cut the range of the time period such that the limits of the observed dates, first and last observed dates, are `date0` and `datef`, respectively. It is possible to specify just one date, i.e. the two dates of the range do not necessarily have to be entered. See Note section.

## Usage

```
cut_injd(injd, date0, datef)
```

## Arguments

| | |
|---|---|
| `injd` | Prepared data, an `injd` object. |
| `date0` | Starting date of class Date or numeric. If `numeric`, it should refer to a year (e.g. `date` = 2018). Optional. |
| `datef` | Ending date. Same class as `date0`. Optional. |

## Value

An `injd` object with a shorter follow-up period.

## Note

Be aware that by modifying the follow-up period of the cohort, the study design is being altered. This function should not be used, unless there is no strong argument supporting it. And in that case, it should be used with caution.

## Examples

```
# Prepare data

df_injuries <- prepare_inj(
  df_injuries0   = raw_df_injuries,
  person_id      = "player_name",
  date_injured   = "from",
  date_recovered = "until"
)

df_exposures <- prepare_exp(
  df_exposures0 = raw_df_exposures,
  person_id     = "player_name",
  date          = "year",
  time_expo     = "minutes_played"
)
```

```
injd <- prepare_all(
  data_exposures = df_exposures,
  data_injuries  = df_injuries,
  exp_unit       = "matches_minutes"
)


cut_injd(injd, date0 = 2018)
```

---

`date2season` *Get the season*

---

## Description

Get the season given the date.

## Usage

```
date2season(date)
```

## Arguments

| | |
|---|---|
| `date` | A vector of class Date or integer/numeric. If it is `integer`/`numeric`, it should refer to the year in which the season started (e.g. `date` = 2015 to refer to the 2015/2016 season) |

## Value

Character specifying the respective competition season given the date. The season (output) follows this pattern: "2005/2006".

## Examples

```
date <- Sys.Date()
date2season(date)
```

## get_data_exposures *Extract exposures data frame*

### Description

Extract exposures data frame from the injd object.

### Usage

```
get_data_exposures(injd)
```

### Arguments

| | |
|---|---|
| injd | injd **S3** object (see prepare_all()). |

### Value

The exposure data frame containing the necessary columns: "person_id", "date" and "time_expo".

### Examples

```
get_data_exposures(injd)
```

## get_data_followup *Extract follow-up data frame*

### Description

Extract follow-up data frame from the injd object.

### Usage

```
get_data_followup(injd)
```

### Arguments

| | |
|---|---|
| injd | injd **S3** object (see prepare_all()). |

### Value

The follow-up data frame containing the necessary columns: "person_id", "t0" and "tf".

### Examples

```
get_data_followup(injd)
```

## get_data_injuries *Extract injury/illness data frame*

### Description

Extract injury/illness data frame from the injd object.

### Usage

```
get_data_injuries(injd)
```

### Arguments

| | |
|---|---|
| injd | injd **S3** object (see prepare_all()). |

### Value

The injury/illness data frame containing the necessary columns: "person_id", "date_injured" and "date_recovered".

### Examples

```
get_data_injuries(injd)
```

## gg_photo *Plot injuries and illnesses over the follow-up period*

### Description

Given an injd **S3** object it plots an overview of the injuries and illnesses suffered by each player/athlete in the cohort during the follow-up. Each subject timeline is depicted horizontally where the red cross indicates the exact injury or illness date, the blue circle the recovery date and the bold black line indicates the duration of the injury (time-loss) or illness.

### Usage

```
gg_photo(injd, title = NULL, fix = FALSE, by_date = "1 months")
```

### Arguments

| | |
|---|---|
| injd | Prepared data. An injd object. |
| title | Text for the main title. |
| fix | A logical value indicating whether to limit the range of date (x scale) to the maximum observed exposure date or not to limit the x scale, regardless some recovery dates might be longer than the maximum observed exposure date. |
| by_date | increment of the date sequence at which x-axis tick-marks are to drawn. An argument to be passed to base::seq.Date(). |

## Value

A ggplot object (to which optionally more layers can be added).

## Examples

```
df_exposures <- prepare_exp(raw_df_exposures, person_id = "player_name",
                            date = "year", time_expo = "minutes_played")
df_injuries  <- prepare_inj(raw_df_injuries, person_id = "player_name",
                            date_injured = "from", date_recovered = "until")
injd         <- prepare_all(data_exposures = df_exposures,
                            data_injuries  = df_injuries,
                            exp_unit = "minutes")


gg_photo(injd, title = "Injury Overview", by_date = "1 years")
```

---

# gg_prevalence *Plot bar plots representing players' prevalence*

---

## Description

Plot the proportions of available and injured players in the cohort, on a monthly or season basis, by a bar plot. Further information on the type of injury may be specified so that the injured players proportions are disaggregated and reported according to this variable.

## Usage

```
gg_prevalence(
  injd,
  time_period = c("monthly", "season"),
  by = NULL,
  line_mean = FALSE,
  title = NULL
)
```

## Arguments

| | |
|---|---|
| injd | Prepared data, an injd object. |
| time_period | Character. One of "monthly" or "season", specifying the periodicity according to which to calculate the proportions of available and injured athletes. |
| by | Character specifying the name of the column on the basis of which to classify the injuries and calculate proportions of the injured athletes. Defaults to NULL. |
| line_mean | Logical (defaults to FALSE) whether to add a horizontal line indicating the mean prevalence over the period. |
| title | Text for the main title. |

## Value

A ggplot object (to which optionally more layers can be added).

## Examples

```
df_exposures <- prepare_exp(raw_df_exposures, person_id = "player_name",
                            date = "year", time_expo = "minutes_played")
df_injuries  <- prepare_inj(raw_df_injuries, person_id = "player_name",
                            date_injured = "from", date_recovered = "until")
injd         <- prepare_all(data_exposures = df_exposures,
                            data_injuries  = df_injuries,
                            exp_unit = "matches_minutes")



library(ggplot2)
our_palette <- c("red3", rev(RColorBrewer::brewer.pal(5, "Reds")), "seagreen3")
gg_prevalence(injd, time_period = "monthly",
              title = "Monthly prevalence of sports injuries") +
  scale_fill_manual(values = our_palette)
gg_prevalence(injd, time_period = "monthly",
              title = "Monthly prevalence of sports injuries",
              line_mean = TRUE) +
  scale_fill_manual(values = our_palette)
gg_prevalence(injd, time_period = "monthly", by = "injury_type",
              title = "Monthly prevalence of each type of sports injury") +
  scale_fill_manual(values = our_palette)
gg_prevalence(injd, time_period = "monthly", by = "injury_type",
              title = "Monthly prevalence of each type of sports injury",
              line_mean = TRUE) +
  scale_fill_manual(values = our_palette)
```

---

# gg_rank *Plot athlete's health problem incidence or burden ranking*

---

## Description

A bar chart that shows athlete-wise summary statistics, either case incidence or injury burden, ranked in descending order.

## Usage

```
gg_rank(
  injd,
  by = NULL,
  summary_stat = c("incidence", "burden", "ncases", "ndayslost"),
  line_overall = FALSE,
  title = NULL
)
```

## Arguments

| | |
|---|---|
| injd | injd **S3** object (see prepare_all()). |
| by | Character specifying the name of the column according to which compute summary statistics. It should refer to a (categorical) variable that describes a grouping factor (e.g. "type of case or injury", "injury location", "sports club"). Optional, defaults to NULL. |
| summary_stat | A character value indicating whether to plot case incidence's (case's) or injury burden's (days losts') ranking. One of "incidence" ("ncases") or "burden" ("ndayslost"), respectively. |
| line_overall | Logical, whether to draw a vertical red line indicating the overall incidence or burden. Defaults to FALSE. |
| title | Text for the main title. |

## Value

A ggplot object (to which optionally more layers can be added).

## Examples

```
df_exposures <- prepare_exp(raw_df_exposures, person_id = "player_name",
                            date = "year", time_expo = "minutes_played")
df_injuries  <- prepare_inj(raw_df_injuries, person_id = "player_name",
                            date_injured = "from", date_recovered = "until")
injd         <- prepare_all(data_exposures = df_exposures,
                            data_injuries  = df_injuries,
                            exp_unit = "matches_minutes")



p1 <- gg_rank(injd, summary_stat = "incidence",
              line_overall = TRUE,
              title = "Overall injury incidence per player") +
  ggplot2::ylab(NULL)
p2 <- gg_rank(injd, summary_stat = "burden",
              line_overall = TRUE,
              title = "Overall injury burden per player") +
  ggplot2::ylab(NULL)

# install.packages("gridExtra")
# library(gridExtra)
if (require("gridExtra")) {
  gridExtra::grid.arrange(p1, p2, nrow = 1)
}
```

gg_riskmatrix *Plot risk matrices*

## Description

Depict risk matrix plots, a graph in which the case (e.g. injury) incidence (frequency) is plotted against the average days lost per case (consequence). The point estimate of case incidence together with its confidence interval is plotted, according to the method specified. On the y-axis, the mean time-loss per case together with ± IQR (days) is plotted. The number shown inside the point and the point size itself, report the case burden (days lost per athlete-exposure time), the bigger the size the greater the burden. See References section.

## Usage

```
gg_riskmatrix(
  injd,
  by = NULL,
  method = c("poisson", "negbin", "zinfpois", "zinfnb"),
  add_contour = TRUE,
  title = NULL,
  xlab = "Incidence (injuries per _)",
  ylab = "Mean time-loss (days) per injury",
  errh_height = 1,
  errv_width = 0.05,
  cont_max_x = NULL,
  cont_max_y = NULL,
  ...
)
```

## Arguments

| | |
|---|---|
| injd | injd **S3** object (see prepare_all()). |
| by | Character specifying the name of the column. A (categorical) variable referring to the "type of case" (e.g. "type of injury" muscular/articular/others or overuse/not-overuse etc.) according to which visualize epidemiological summary statistics (optional, defaults to NULL). |
| method | Method to estimate the incidence (burden) rate. One of "poisson", "negbin", "zinfpois" or "zinfnb"; that stand for Poisson method, negative binomial method, zero-inflated Poisson and zero-inflated negative binomial. |
| add_contour | Logical, whether or not to add contour lines of the product between case incidence and mean severity (i.e. 'incidence x average time-loss'), which leads to case burden (defaults to TRUE). |
| title | Text for the main title passed to ggplot2::ggtitle(). |
| xlab | x-axis label to be passed to ggplot2::xlab(). |
| ylab | y-axis label to be passed to ggplot2::ylab(). |

errh_height — Set the height of the horizontal interval whiskers; the height argument for ggplot2::geom_errorbar().

errv_width — Set the width of the vertical interval whiskers; the width argument for ggplot2::geom_errorbar().

cont_max_x, cont_max_y — Numerical (optional) values indicating the maximum on the x-axis and y-axis, respectively, to be reached by the contour.

... — Other arguments passed on to ggplot2::geom_contour() and metR::geom_text_contour(). These are often aesthetics like bins = 15 or breaks = 10.

## Value

A ggplot object (to which optionally more layers can be added).

## References

Bahr R, Clarsen B, Derman W, et al. International Olympic Committee consensus statement: methods for recording and reporting of epidemiological data on injury and illness in sport 2020 (including STROBE Extension for Sport Injury and Illness Surveillance (STROBE-SIIS)) *British Journal of Sports Medicine* 2020; 54:372-389.

Fuller C. W. (2018). Injury Risk (Burden), Risk Matrices and Risk Contours in Team Sports: A Review of Principles, Practices and Problems.*Sports Medicine*, 48(7), 1597–1606. doi:10.1007/s4027901809135

## Examples

```
df_exposures <- prepare_exp(raw_df_exposures, person_id = "player_name",
                            date = "year", time_expo = "minutes_played")
df_injuries  <- prepare_inj(raw_df_injuries, person_id = "player_name",
                            date_injured = "from", date_recovered = "until")
injd         <- prepare_all(data_exposures = df_exposures,
                            data_injuries  = df_injuries,
                            exp_unit = "matches_minutes")



gg_riskmatrix(injd)
gg_riskmatrix(injd, by = "injury_type", title = "Risk matrix")
```

---

# injd *Example of an* injd *object*

---

## Description

An injd object (**S3**), called injd, to showcase what this object is like and also to save computation time in some help files provided by the package. The result of applying prepare_all() to *raw_df_exposures* (prepare_exp(raw_df_exposures, ...)) and *raw_df_injuries* (prepare_inj(raw_df_injuries, ...)).

## Usage

```
injd
```

## Format

The main data frame in `injd` gathers information of 28 players and has 108 rows and 19 columns:

**person_id** Player identifier (factor)

**t0** Follow-up period of the corresponding player, i.e. player's first observed date, same value for each player (Date)

**tf** Follow-up period of the corresponding player, i.e. player's last observed date, same value for each player (Date)

**date_injured** Date of injury of the corresponding observation (if any). Otherwise NA (Date)

**date_recovered** Date of recovery of the corresponding observation (if any). Otherwise NA (Date)

**tstart** Beginning date of the corresponding interval in which the observation has been at risk of injury (Date)

**tstop** Ending date of the corresponding interval in which the observation has been at risk of injury (Date)

**tstart_minPlay** Beginning time. Minutes played in matches until the start of this interval in which the observation has been at risk of injury (numeric)

**tstop_minPlay** Ending time. Minutes played in matches until the finish of this interval in which the observation has been at risk of injury (numeric)

**status** injury (event) indicator (numeric)

**enum** an integer indicating the recurrence number, i.e. the $k$-th injury (event), at which the observation is at risk

**days_lost** Number of days lost due to injury (numeric)

**player_id** Identification number of the football player (factor)

**season** Season to which this player's entry corresponds (factor)

**games_lost** Number of matches lost due to injury (numeric)

**injury** Injury specification as it appears in https://www.transfermarkt.com, if any; otherwise NA (character)

**injury_acl** Whether it is Anterior Cruciate Ligament (ACL) injury or not (NO_ACL); if the interval corresponds to an injury, NA otherwise (factor)

**injury_type** A five level categorical variable indicating the type of injury, whether Bone, Concussion, Ligament, Muscle or Unknown; if any, NA otherwise (factor)

**injury_severity** A four level categorical variable indicating the severity of the injury (if any), whether Minor (<7 days lost), Moderate ([7, 28) days lost), Severe ([28, 84) days lost) or Very_severe (>=84 days lost); NA otherwise (factor)

## Details

It consists of a data frame plus 4 other attributes: a character specifying the unit of exposure (`unit_exposure`); and 3 (auxiliary) data frames: `follow_up`, `data_exposures` and `data_injuries`.

## is_injd *Check if an object is of class* injd

### Description

Check if an object x is of class injd.

### Usage

```
is_injd(x)
```

### Arguments

| | |
|---|---|
| x | any R object. |

### Value

A logical value: TRUE if x inherits from injd class, FALSE otherwise.

## prepare_data *Prepare data in a standardized format*

### Description

These are the data preprocessing functions provided by the injurytools package, which involve:

1. setting **exposure** and **injury and illness data** in a standardized format and
2. integrating both sources of data into an adequate data structure.

prepare_inj() and prepare_exp() set standardized names and proper classes to the (key) columns in injury/illness and exposure data, respectively. prepare_all() integrates both, standardized injury and exposure data sets, and convert them into an injd **S3** object that has an adequate structure for further statistical analyses. See the Prepare Sports Injury Data vignette for details.

### Usage

```
prepare_inj(
  df_injuries0,
  person_id = "person_id",
  date_injured = "date_injured",
  date_recovered = "date_recovered"
)

prepare_exp(
  df_exposures0,
  person_id = "person_id",
```

```
  date = "date",
  time_expo = "time_expo"
)

prepare_all(
  data_exposures,
  data_injuries,
  exp_unit = c("minutes", "hours", "days", "matches_num", "matches_minutes",
    "activity_days", "seasons")
)
```

## Arguments

| | |
|---|---|
| `df_injuries0` | A data frame containing injury or illness information, with columns referring to the athlete name/id, date of injury/illness and date of recovery (as minimal data). |
| `person_id` | Character referring to the column name storing sportsperson (player, athlete) identification information. |
| `date_injured` | Character referring to the column name where the information about the date of injury or illness is stored. |
| `date_recovered` | Character referring to the column name where the information about the date of recovery is stored. |
| `df_exposures0` | A data frame containing exposure information, with columns referring to the sportsperson's name/id, date of exposure and the total time of exposure of the corresponding data entry (as minimal data). |
| `date` | Character referring to the column name where the exposure date information is stored. Besides, the column must be of class Date or integer/numeric. If it is `integer`/`numeric`, it should refer to the year in which the season started (e.g. `date` = 2015 to refer to the 2015/2016 season). |
| `time_expo` | Character referring to the column name where the information about the time of exposure in that corresponding date is stored. |
| `data_exposures` | Exposure data frame with standardized column names, in the same fashion that `prepare_exp()` returns. |
| `data_injuries` | Injury data frame with standardized column names, in the same fashion that `prepare_inj()` returns. |
| `exp_unit` | Character defining the unit of exposure time ("minutes" the default). |

## Value

`prepare_inj()` returns a data frame in which the **key columns** in injury/illness data are standardized and have a proper format.

`prepare_exp()` returns a data frame in which the **key columns** in exposure data are standardized and have a proper format.

`prepare_all()` returns the `injd` **S3** object that contains all the necessary information and a proper data structure to perform further statistical analyses (e.g. calculate injury summary statistics, visualize injury data).

- If exp_unit is "minutes" (the default), the columns tstart_min and tstop_min are created which specify the time to event (injury) values, the starting and stopping time of the interval, respectively. That is the training time in minutes, that the sportsperson has been at risk, until an injury/illness (or censorship) has occurred. For other choices, tstart_x and tstop_x are also created according to the exp_unit indicated (x, one of: min, h, match, minPlay, d, acd or s). These columns will be useful for survival analysis routines. See Note section.
- It also creates days_lost column based on the difference between date_recovered and date_injured in days. And if it does exist (in the raw data) it overrides.

## Note

Depending on the unit of exposure, tstart_x and tstop_x columns might have same values (e.g. if exp_unit = "matches_num" and the player has not played any match between the corresponding period of time). Please be aware of this before performing any survival analysis related task.

## Examples

```
df_injuries <- prepare_inj(df_injuries0   = raw_df_injuries,
                           person_id      = "player_name",
                           date_injured   = "from",
                           date_recovered = "until")

df_exposures <- prepare_exp(df_exposures0 = raw_df_exposures,
                            person_id     = "player_name",
                            date          = "year",
                            time_expo     = "minutes_played")


injd <- prepare_all(data_exposures = df_exposures,
                    data_injuries  = df_injuries,
                    exp_unit = "matches_minutes")
head(injd)
class(injd)
str(injd, 1)
```

---

# raw_df_exposures *Minimal example of exposure data*

---

## Description

An example of a player exposure data set that contains minimum required exposure information as well as other player- and match-related variables. It includes Liverpool Football Club male's first team players' exposure data, exposure measured as (number or minutes of) matches played, over two consecutive seasons, 2017-2018 and 2018-2019. Each row refers to player-season. These data have been scrapped from https://www.transfermarkt.com/ website using self-defined **R** code with rvest and xml2 packages.

## Usage

```
raw_df_exposures
```

## Format

A data frame with 42 rows corresponding to 28 football players and 16 variables:

**player_name** Name of the football player (factor)

**player_id** Identification number of the football player (factor)

**season** Season to which this player's entry corresponds (factor)

**year** Year in which each season started (numeric)

**matches_played** Matches played by the player in each season (numeric)

**minutes_played** Minutes played by the player in each season (numeric)

**liga** Name of the ligue where the player played in each season (factor)

**club_name** Name of the club to which the player belongs in each season (factor)

**club_id** Identification number of the club to which the player belongs in each season (factor)

**age** Age of the player in each season (numeric)

**height** Height of the player in m (numeric)

**place** Place of birth of each player (character)

**citizenship** Citizenship of the player (factor)

**position** Position of the player on the pitch (factor)

**foot** Dominant leg of the player. One of both, left or right (factor)

**goals** Number of goals scored by the player in that season (numeric)

**assists** Number of assists provided by the player in that season (numerical)

**yellows** Number of the yellow cards received by the player in that season (numeric)

**reds** Number of the red cards received by the player in that season (numeric)

## Note

This data frame is provided for illustrative purposes. We warn that they might not be accurate, there might be a mismatch and non-completeness with what actually occurred. As such, its use cannot be recommended for epidemiological research (see also Hoenig et al., 2022).

## Source

https://www.transfermarkt.com/

## References

Hoenig, T., Edouard, P., Krause, M., Malhan, D., Relógio, A., Junge, A., & Hollander, K. (2022). Analysis of more than 20,000 injuries in European professional football by using a citizen science-based approach: An opportunity for epidemiological research?. *Journal of science and medicine in sport*, 25(4), 300-305.

raw_df_injuries *Minimal example of injury data*

## Description

An example of an injury data set containing minimum required injury information as well as other further injury-related variables. It includes Liverpool Football Club male's first team players' injury data. Each row refers to player-injury. These data have been scrapped from https://www.transfermarkt.com/ website using self-defined **R** code with `rvest` and `xml2` packages.

## Usage

```
raw_df_injuries
```

## Format

A data frame with 82 rows corresponding to 23 players and 11 variables:

**player_name** Name of the football player (factor)

**player_id** Identification number of the football player (factor)

**season** Season to which this player's entry corresponds (factor)

**from** Date of the injury of each data entry (Date)

**until** Date of the recovery of each data entry (Date)

**days_lost** Number of days lost due to injury (numeric)

**games_lost** Number of matches lost due to injury (numeric)

**injury** Injury specification as it appears in https://www.transfermarkt.com (character)

**injury_acl** Whether it is Anterior Cruciate Ligament (ACL) injury or not (NO_ACL)

**injury_type** A five level categorical variable indicating the type of injury, whether Bone, Concussion, Ligament, Muscle or Unknown; if any, NA otherwise (factor)

**injury_severity** A four level categorical variable indicating the severity of the injury (if any), whether Minor (<7 days lost), Moderate ([7, 28) days lost), Severe ([28, 84) days lost) or Very_severe (>=84 days lost); NA otherwise (factor)

## Note

This data frame is provided for illustrative purposes. We warn that they might not be accurate, there might be a mismatch and non-completeness with what actually occurred. As such, its use cannot be recommended for epidemiological research (see also Hoenig et al., 2022).

## Source

https://www.transfermarkt.com/

## References

Hoenig, T., Edouard, P., Krause, M., Malhan, D., Relógio, A., Junge, A., & Hollander, K. (2022). Analysis of more than 20,000 injuries in European professional football by using a citizen science-based approach: An opportunity for epidemiological research?. *Journal of science and medicine in sport*, 25(4), 300-305.

---

`season2year` *Get the year*

---

## Description

Get the year given the season.

## Usage

```
season2year(season)
```

## Arguments

| | |
|---|---|
| `season` | Character/factor specifying the season. It should follow the pattern "xxxx/yyyy", e.g. "2005/2006". |

## Value

Given the season, it returns the year (in `numeric`) in which the season started.

## Examples

```
season <- "2022/2023"
season2year(season)
```

# Index