Package ‘ura’

July 12, 2024

Type    Package
Title   Monitoring Rater Reliability
Version 1.0.1

Description Provides researchers with a simple set of diagnostic tools for monitoring the progress and reliability of raters conducting content coding tasks. Goehring (2024) [https://bengoehring.github.io/improving-content-analysis-tools-for-working-with-undergraduate-research-assistants.pdf] argues that supervisors---especially supervisors of small teams---should utilize computational tools to monitor reliability in real time. As such, this package provides easy-to-use functions for calculating inter-rater reliability statistics and measuring the reliability of one coder compared to the rest of the team.

License MIT + file LICENSE
Encoding UTF-8
LazyData true
RoxygenNote 7.3.2
Depends R (>= 2.10)
Imports dplyr, irr, magrittr, rlang (>= 0.4.11), tibble, tidyr
Suggests roxygen2, stringr, testthat (>= 3.0.0)
Config/testthat/edition 3

URL https://github.com/bengoehring/ura

BugReports https://github.com/bengoehring/ura/issues

NeedsCompilation no

Author Benjamin Goehring [aut, cre] (https://orcid.org/0000-0001-8450-6065)
Maintainer Benjamin Goehring <bengoehr@umich.edu>

Repository CRAN
Date/Publication 2024-07-12 12:30:02 UTC
**Contents**

anxiety .................................................. 2
diagnoses .................................................. 2
int_return dbl coded .................................... 3
irr_stats .................................................. 4
rater_agreement ......................................... 5
index ....................................................... 7

---

**anxiety**

*Anxiety ratings*

Simulated data from three raters rating the anxiety of 20 individuals. The codings range from 1 (no anxiety) to 6 (extremely anxious). The data are forked directly from the `irr` package, with the only difference being the shape of the dataset.

**Usage**

anxiety

**Format**

```r
## 'anxiety' A data frame with 60 rows and 3 columns:
subject_id  The subject being screened for anxiety (numeric).
rater_id    The rater evaluating the subject for anxiety (numeric).
anxiety_level The level of anxiety observed in the subject by the rater (numeric).
```

**Source**

<https://cran.r-project.org/package=irr>

---

**diagnoses**

*Psychiatric diagnoses of patients*

Data from Fleiss (1971) concerning the psychiatric conditions of thirty patients as evaluated by six raters. The data are forked directly from the `irr` package, with the only difference being the shape of the dataset.

**Usage**

diagnoses
int_return_dbl_coded

Format

## ‘diagnoses’ A data frame with 180 rows and 3 columns:

- **patient_id** The patient being screened for a psychiatric condition (numeric).
- **rater_id** The rater evaluating the patient for a psychiatric condition (numeric).
- **diagnosis** The psychiatric diagnosis of the patient (factor).

Source


References


Description

int_return_dbl_coded An internal function to return the subjects double-coded by the raters. It runs a number of checks along the way.

Usage

```r
int_return_dbl_coded(
  in_object_name,
  in_rater_column,
  in_subject_column,
  in_coding_column
)
```

Arguments

- **in_object_name** A dataframe or tibble containing raters’ codings. Each row should contain the assigned coding from a given rater-subject.
- **in_rater_column** The name of the column containing the raters’ names as a string.
- **in_subject_column** The name of the column containing the names of the subjects being coded as a string.
- **in_coding_column** The name of the column containing the codings assigned by the raters as a string.
**irr_stats**

**Author(s)**
Benjamin Goehring <bengoehr@umich.edu>

**Description**

*irr_stats* calculates a variety of IRR statistics.

**Usage**

```r
irr_stats(
  object_name,  # A dataframe or tibble containing raters' codings. Each row should contain the assigned coding from a given rater-subject.
  rater_column,  # The name of the column containing the raters' names as a string.
  subject_column,  # The name of the column containing the names of the subjects being coded as a string.
  coding_column,  # The name of the column containing the codings assigned by the raters as a string.
  round_digits = 2,  # The number of decimals to round the IRR values by. The default is 2.
  stats_to_include = c("Percentage agreement", "Krippendorf's Alpha")  # The IRR statistics to include in the output. Currently only supports percent agreement and Krippendorf's Alpha. See the documentation of the *irr* package for more information about specific IRR statistics.
)
```

**Arguments**

- **object_name**: A dataframe or tibble containing raters' codings. Each row should contain the assigned coding from a given rater-subject.
- **rater_column**: The name of the column containing the raters' names as a string.
- **subject_column**: The name of the column containing the names of the subjects being coded as a string.
- **coding_column**: The name of the column containing the codings assigned by the raters as a string.
- **round_digits**: The number of decimals to round the IRR values by. The default is 2.
- **stats_to_include**: The IRR statistics to include in the output. Currently only supports percent agreement and Krippendorf's Alpha. See the documentation of the *irr* package for more information about specific IRR statistics.

**Value**

A tibble containing the IRR statistic, the statistic's value, and the number of subjects used to calculate the statistic.

**Author(s)**
Benjamin Goehring <bengoehr@umich.edu>
**Examples**

```r
# Return IRR statistics for the diagnoses dataset:
irr_stats(diagnoses,
    rater_column = 'rater_id',
    subject_column = 'patient_id',
    coding_column = 'diagnosis')

# And IRR statistics for the anxiety dataset:
irr_stats(anxiety,
    rater_column = 'rater_id',
    subject_column = 'subject_id',
    coding_column = 'anxiety_level')
```

**Description**

rater_agreement calculates the percent agreement between each rater and the other raters who coded the same subjects.

**Usage**

```r
rater_agreement(object_name, rater_column, subject_column, coding_column)
```

**Arguments**

- **object_name**: A dataframe or tibble containing raters’ codings. Each row should contain the assigned coding from a given rater-subject.
- **rater_column**: The name of the column containing the raters’ names as a string.
- **subject_column**: The name of the column containing the names of the subjects being coded as a string.
- **coding_column**: The name of the column containing the codings assigned by the raters as a string.

**Value**

A tibble where each row notes the percent agreement between rater i and all other raters who coded the same subjects (percent_agree). The n_multi_coded column notes how many subjects have been coded by rater i that have also been coded by other raters (i.e., the denominator for the percent_agree value).

**Author(s)**

Benjamin Goehring <bengoehr@umich.edu>
Examples

# Example data: 3 raters assigning binary values to 10 subjects
example_data <- tibble::tribble(
  ~rater,~subject,~coding,
  1,1,1,
  1,2,0,
  1,3,1,
  1,4,0,
  2,3,1,
  2,9,0,
  2,10,1,
  2,4,1,
  2,5,1,
  2,6,1,
  3,5,1,
  3,6,1,
  3,7,1,
  3,8,1,
)

# Find percent agreement by rater
rater_agreement(example_data,
  rater_column = 'rater',
  subject_column = 'subject',
  coding_column = 'coding')
Index

* datasets
  anxiety, 2
  diagnoses, 2

anxiety, 2

diagnoses, 2

int_return_dbl_coded, 3
irr_stats, 4

rater_agreement, 5