Package 'lab2clean'

May 8, 2026

Title Automation and Standardization of Cleaning Clinical Laboratory Data

Version 2.0.0

Description Navigating the shift of clinical laboratory data from primary everyday clinical use to secondary research purposes presents a significant challenge. Given the substantial time and expertise required for lab data pre-processing and cleaning and the lack of all-in-one tools tailored for this need, we developed our algorithm 'lab2clean' as an open-source R-package. 'lab2clean' package is set to automate and standardize the intricate process of cleaning clinical laboratory results. With a keen focus on improving the data quality of laboratory result values and units, our goal is to equip researchers with a straightforward, plug-and-play tool, making it smoother for them to unlock the true potential of clinical laboratory data in clinical research and clinical machine learning (ML) model development. Functions to clean & validate result values (Version 1.0) are described in detail in 'Zayed et al. (2024)' <doi:10.1186/s12911-024-02652-7>. Functions to standardize & harmonize result units (added in Version 2.0) are described in detail in 'Zayed et al. (2025)' <doi:10.1016/j.ijmedinf.2025.106131>.

License GPL (>= 3)

Encoding UTF-8

Imports data.table, stats, utils

Suggests knitr, rmarkdown, fansi, kableExtra, printr

VignetteBuilder knitr

RoxygenNote 7.3.2

Depends R (>= 3.5)

LazyData true

NeedsCompilation no

Author Ahmed Zayed [aut, cre] (ORCID: <https://orcid.org/0000-0001-7797-1655>),
Ilias Sarikakis [aut, ctb],
Arne Janssens [aut, ctb],
Pavlos Mamouris [ctb]

Maintainer Ahmed Zayed <ahmed.zayed@kuleuven.be>

Repository CRAN

Date/Publication 2025-10-04 14:00:02 UTC

Contents

`annotable_strings` *Annotable Strings for Unit Standardization*

Description

A dataset containing commonly used strings in annotations.

Usage

```
data(annotable_strings)
```

Format

A data frame with 2679 rows and 1 variable.

Details

annotation A character string representing the string used as annotation.

clean_lab_result *Clean and Standardize Laboratory Result Values*

Description

This function is designed to clean and standardize laboratory result values. It creates two new columns "clean_result" and "scale_type" without altering the original result values. The function is part of a comprehensive R package designed for cleaning laboratory datasets.

Usage

```
clean_lab_result(
  lab_data,
  raw_result,
  locale = "NO",
  report = TRUE,
  n_records = NA
)
```

Arguments

lab_data	A data frame containing laboratory data.
raw_result	The column in ‘lab_data‘ that contains raw result values to be cleaned.
locale	A string representing the locale for the laboratory data. Defaults to "NO".
report	A report is written in the console. Defaults to "TRUE".
n_records	In case you are loading a grouped list of distinct results, then you can assign the n_records to the column that contains the frequency of each distinct result. Defaults to NA.

Details

The function undergoes the following methodology: 1. Clear Typos: Removes typographical errors and extraneous characters. 2. Handle Extra Variables: Identifies and separates extra variables from result values. 3. Detect and Assign Scale Types: Identifies and assigns the scale type using regular expressions. 4. Number Formatting: Standardizes number formats based on predefined rules and locale. 5. Mining Text Results: Identifies common words and patterns in text results.

Internal Datasets: The function uses an internal dataset; ‘common_words_languages.csv‘ which contains common words in various languages used for pattern identification in text result values.

Value

A modified ‘lab_data‘ data frame with additional columns: * ‘clean_result‘: Cleaned and standardized result values. * ‘scale_type‘: The scale type of result values (Quantitative, Ordinal, Nominal). * ‘cleaning_comments‘: Comments about the cleaning process for each record.

Note

This function is part of a larger data cleaning pipeline and should be evaluated in that context. The package framework includes functions for cleaning result values and validating quantitative results for each test identifier.

Performance of the function can be affected by the size of ‘lab_data‘. Considerations for data size or pre-processing may be needed.

Author(s)

Ahmed Zayed <ahmed.zayed@kuleuven.be>

See Also

Function 2 for result validation,

`common_words` *Data for the common words*

Description

A dataset containing data for common words.

Usage

```
data(common_words)
```

Format

A data frame with 19 rows and 9 variables.

Details

Language The name of the language.

Positive Translation of the word "Positive".

Negative Translation of the word "Negative".

Not_detected Translation of the phrase "Not detected".

High Translation of the word "High".

Low Translation of the word "Low".

Normal Translation of the word "Normal".

Sample Translation of the word "Sample".

Specimen Translation of the word "Specimen".

Function_1_dummy *Dummy Data for demonstrating function 1*

Description

A dataset containing dummy data for demonstrating function 1 ("clean_lab_result").

Usage

```
data(Function_1_dummy)
```

Format

A data frame with 87 rows and 2 variables.

Details

raw_result The raw result.

frequency The frequency of the raw result.

Function_2_dummy *Dummy Data for demonstrating function 2*

Description

A dataset containing dummy data for demonstrating function 2 ("validate_lab_result").

Usage

```
data(Function_2_dummy)
```

Format

A data frame with 86,863 rows and 5 variables.

Details

patient_id Identifier of the tested patient.

lab_datetime1 Date or datetime of the laboratory test.

loinc_code LOINC code of the laboratory test.

result_value Quantitative result value for validation.

result_unit Result unit in UCUM-compliant format.

Function_3_dummy *Dummy Data for demonstrating function 3*

Description

A dataset containing dummy data for demonstrating function 3 ("standardize_lab_unit") containing a tiny, intentionally messy collection of unit strings that exercise different techniques handled by the function.

Usage

```
data(Function_3_dummy)
```

Format

A data frame with 32 rows and 3 variables.

Details

unit_raw Raw unit string to be standardized (character)

n_records Optional frequency used to test the `n_records` argument (integer)

note Human-readable tag for the test case (character)

Function_4_dummy *Dummy Data for demonstrating function 4*

Description

A dataset containing dummy data for demonstrating function 4 ("harmonize_lab_unit") including different success (harmonized) and failure (not_harmonized) cases handled by the function.

Usage

```
data(Function_4_dummy)
```

Format

A data frame with 48 rows and 3 variables.

Details

loinc_code LOINC code of the laboratory test.

result_value Quantitative result value for validation.

result_unit Result unit in UCUM-compliant format.

`harmonize_lab_unit` *Harmonizing Laboratory Units of Measurement through Unit Conversion*

Description

This function is designed to harmonize the units found in a laboratory data set to either SI or Conventional units, converting the numeric result values in the process and (optionally) updating LOINC codes when mass–molar conversion are required.

Usage

```
harmonize_lab_unit(
  lab_data,
  loinc_code,
  result_value,
  result_unit,
  preferred_unit_system = "SI",
  report = TRUE
)
```

Arguments

`lab_data`	A data frame containing laboratory data.
`loinc_code`	The column in 'lab_data' indicating the LOINC code of the laboratory test.
`result_value`	The column in 'lab_data' with quantitative result values for conversion.
`result_unit`	The column in 'lab_data' with result units in a UCUM-valid format.
`preferred_unit_system`	A string representing the preference of the user for the unit system used for standardization. Defaults to "SI", the other option is "Conventional".
`report`	A report is written in the console. Defaults to "TRUE".

Details

The function undergoes the following methodology: 1. Extracting unit parameters (dimension & magnitude) 2. Setting reference unit (LOINC-UCUM mapping) 3. Check compatibility between reported unit and reference unit 4. Executing regular conversion 5. Executing mass<>molar conversion 6. Checking LOINC codes

Internal Datasets: The function uses an internal dataset; 'parsed_units_df' which contains 1450 parsed ucum units

Value

A modified ‘lab_data‘ data frame with additional columns (original row order preserved): * ‘harmonized_unit‘: Harmonized units according to the preferred unit system. * ‘OMOP_concept_id‘: The concept id of the harmonized unit according to the OMOP Common Data Model. * ‘new_value‘: The result value after the conversion. * ‘new_loinc_code‘: If the unit conversion led to a new loinc code (e.g. in mass-molar conversion). * ‘property_group_id‘: the code of the LOINC group (parent group ID / Group ID). * ‘cleaning_comments‘: Comments about the harmonization and conversion process for each lab result.

Note

This function is part of a larger data cleaning pipeline and should be evaluated in that context. The package framework includes functions for cleaning result values and validating quantitative results for each test identifier.

Performance of the function can be affected by the size of ‘lab_data‘. Considerations for data size or pre-processing may be needed.

Author(s)

Ahmed Zayed <ahmed.zayed@kuleuven.be>, Ilias Sarikakis <sarikakisilias@gmail.com>

See Also

Function 1 for result value cleaning, Function 2 for result validation, Function 3 for unit format standardized to UCUM,

---

`logic_rules` *Data for the logic rules*

Description

A dataset containing data for the logic rules.

Usage

```
data(logic_rules)
```

Format

A data frame with 18 rows and 4 variables.

Details

rule_id Identifier for the logic rule.

rule_index The sequence index of the rule.

rule_part The textual content of the rule part.

rule_part_type The type/category of the rule part (e.g., operator, term, value).

loinc_reference_unit_v1 *Data for the Reference Harmonized Units for LOINC Groups*

Description

A dataset mapping each LOINC codes to the reference harmonized unit of their LOINC group.

Usage

```
data(loinc_reference_unit_v1)
```

Format

A data frame with 33197 rows and 8 variables.

Details

loinc_code Contains 33,197 different LOINC codes.

unit_system The unit system (SI or conventional) of the reference unit.

reference_unit The harmonized reference unit.

OMOP_concept_id The OMOP standardized concept ID for the harmonized unit, if applicable.

mass_molar_unit The reference unit of another LOINC code from the same mass–molar group.

molecular_weight The molecular weight of the analyte, if applicable.

mass_molar_loinc The other LOINC code that shares the same mass–molar group.

property_group_id The LOINC group ID that shares the same component, property, and time aspect.

parsed_units_df *Data for the parsed UCUM units*

Description

Intermediate dataset representing parsed UCUM units and the parameters necessary for machine readability and conversion.

Usage

```
data(parsed_units_df)
```

Format

A data frame with 1439 rows and 8 variables.

Details

csCode_ Case-sensitive code.

ciCode_ Case-insensitive code.

magnitude_ Magnitude of the unit.

dim_ The dimensionality of the unit (e.g., mass/time).

cnv_ Special conversion involved (if any).

cnvPfx_ Prefix used in the special conversion.

isArbitrary_ Logical flag indicating if the unit is arbitrary.

moleExp_ Logical flag indicating if the unit includes molar expression.

---

reportable_interval *Data for the reportable interval*

Description

A dataset containing data for the reportable interval.

Usage

```
data(reportable_interval)
```

Format

A data frame with 493 rows and 4 variables.

Details

interval_loinc_code The LOINC code to which the reportable interval applies.

UCUM_unit The UCUM-compliant unit for the laboratory measurement.

low_reportable_limit The lower limit of the reportable range.

high_reportable_limit The upper limit of the reportable range.

RWD_units_to_UCUM_V2 *Data for the RWD units mapped to standard UCUM-valid units*

Description

A dataset containing RWD units mapped to standard UCUM-valid units.

Usage

```
data(RWD_units_to_UCUM_V2)
```

Format

A data frame with N rows and 3 variables:

clean_unit_lower Case-insensitive representation of invalid or inconsistent units as found in real-world data (RWD).

ucum_code The equivalent UCUM-compliant format for the given RWD unit.

source_match The source from which this mapping or match was derived.

A data frame with 5120 rows and 3 variables.

standardize_lab_unit *Clean and Standardize Formats of Laboratory Units of Measurement*

Description

This function is designed to clean and standardize formats of laboratory units of measurement. It standardizes the units' format according to the Unified Code for Units of Measure (UCUM) https://ucum.org/ucum

Usage

```
standardize_lab_unit(lab_data, raw_unit, report = TRUE, n_records = NA)
```

Arguments

lab_data	A data frame containing laboratory data.
raw_unit	The column in 'lab_data' that contains raw units to be cleaned.
report	A report is written in the console. Defaults to "TRUE".
n_records	In case you are loading a grouped list of distinct results, then you can assign the n_records to the column that contains the frequency of each distinct result. Defaults to NA.

Details

The function undergoes the following methodology: 1. Pre-processing unit srings. 2. Lookup in commom units database. 3. Check Syntax Integrity of units with no UCUM match. 4. Parsing of units which passesd checks (tokenize and classify) 5. Restructuring of parsed units (apply correction rules & final validation)

Internal Datasets: The function uses an internal dataset; 'RWD_units_to_UCUM_V2' which contains 3739 synonyms of 1448 ucum units.

Value

A modified 'lab_data' data frame with additional columns: * 'ucum_code': Cleaned and standardized units according to UCUM syntax. * 'cleaning_comments': Comments about the cleaning process for each unit.

Note

This function is part of a larger data cleaning pipeline and should be evaluated in that context. The package framework includes functions for cleaning result values and validating quantitative results for each test identifier.

Performance of the function can be affected by the size of 'lab_data'. Considerations for data size or pre-processing may be needed.

Author(s)

Ahmed Zayed <ahmed.zayed@kuleuven.be>, Ilias Sarikakis <sarikakisilias@gmail.com>

See Also

Function 1 for result value cleaning, Function 2 for result validation, Function 3 for unit format cleaning, Function 4 for unit conversion.

---

validate_lab_result *Validate Quantitative Laboratory Result Values*

Description

This function is designed to validate quantitative laboratory result values. It modifies the provided 'lab_data' dataframe in-place, adding one new column.

Usage

```
validate_lab_result(
  lab_data,
  result_value,
  result_unit,
  loinc_code,
  patient_id,
```

```
  lab_datetime,
  report = TRUE
)
```

Arguments

lab_data	A data frame containing laboratory data.
result_value	The column in 'lab_data' with quantitative result values for validation.
result_unit	The column in 'lab_data' with result units in a UCUM-valid format.
loinc_code	The column in 'lab_data' indicating the LOINC code of the laboratory test.
patient_id	The column in 'lab_data' indicating the identifier of the tested patient.
lab_datetime	The column in 'lab_data' with the date or datetime of the laboratory test.
report	A report is written in the console. Defaults to "TRUE".

Details

The function employs the following validation methodology: 1. Reportable limits check: Identifies implausible values outside reportable limits. 2. Logic rules check: Identifies values that contradict some predefined logic rules. 3. Delta limits check: Flags values with excessive change from prior results for the same test and patient.

Internal Datasets: The function uses two internal datasets included with the package: 1. 'reportable_interval': Contains information on reportable intervals. 2. 'logic_rules': Contains logic rules for validation.

Value

A modified 'lab_data' data frame with additional columns: * 'flag': specifies the flag detected in the result records that violated one or more of the validation checks

Note

This function is a component of a broader laboratory data cleaning pipeline and should be evaluated accordingly. The package's framework includes functions for cleaning result values, validating quantitative results, standardizing unit formats, performing unit conversion, and assisting in LOINC code mapping.

Concerning performance, the function's speed might be influenced by the size of 'lab_data'. Consider: * Limiting the number of records processed. * Optimize the function for larger datasets. * Implement pre-processing steps to divide the dataset chronologically.

Author(s)

Ahmed Zayed <ahmed.zayed@kuleuvne.be>, Arne Janssens <arne.janssens@kuleuven.be>

See Also

Function 1 for result value cleaning,

Index