Package: privacyR (via r-universe)

October 21, 2026

Type Package

Title Privacy-Preserving Data Anonymization

Version 1.0.1

Description Tools for anonymizing sensitive patient and research data. Helps protect privacy while keeping data useful for analysis. Anonymizes IDs, names, dates, locations, and ages while maintaining referential integrity. Methods based on: Sweeney (2002) <doi:10.1142/S0218488502001648>, Dwork et al. (2006) <doi:10.1007/11681878_14>, El Emam et al. (2011) <doi:10.1371/journal.pone.0028071>, Fung et al. (2010) <doi:10.1145/1749603.1749605>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

Imports lubridate, digest

Suggests testthat (>= 3.0.0), knitr, rmarkdown, data.table, pkgdown

VignetteBuilder knitr

NeedsCompilation no

Author Vikrant Dev Rathore [aut, cre]

Maintainer Vikrant Dev Rathore <rathore.vikrant@gmail.com>

URL https://vikrant31.github.io/privacyR/, https://github.com/vikrant31/privacyR

Repository https://vikrant31.r-universe.dev

Date/Publication 2025-11-24 15:19:30 UTC

RemoteUrl https://github.com/vikrant31/privacyr

RemoteRef HEAD

RemoteSha add36813b52e8484cf3d146dd28a5dcb32c290a0

Contents

anonymize_age 2
anonymize_dataframe 3
anonymize_dates 5
anonymize_id 6
anonymize_locations 6
anonymize_names 7

Index **9**

`anonymize_age` *Anonymize Age by Buckets*

Description

Groups ages into buckets for privacy protection. Default uses 10-year buckets (0-9, 10-19, etc.) which are useful for research. Ages 90+ are grouped together.

Usage

```
anonymize_age(x, method = c("10year", "hipaa"), custom_buckets = NULL)
```

Arguments

`x`	A numeric vector of ages to anonymize
`method`	Character string specifying bucketing method: "10year" (default) uses 10-year buckets: 0-9, 10-19, 20-29, 30-39, 40-49, 50-59, 60-69, 70-79, 80-89, 90+ "hipaa" uses HIPAA-compliant buckets: 0-17, 18-64, 65-89, 90+
`custom_buckets`	Optional named numeric vector for custom buckets. Format: c("0-9" = 9, "10-19" = 19, "20-29" = 29, "90+" = Inf)

Value

A character vector of age buckets

Examples

```
ages <- c(25, 45, 67, 92, 15, 78)
anonymize_age(ages)  # Uses 10-year buckets by default
anonymize_age(ages, method = "hipaa")  # Use HIPAA buckets
```

anonymize_dataframe *Anonymize Patient Data in a Data Frame*

Description

Main function to anonymize patient data in a data frame or data.table. Automatically detects and anonymizes columns based on data types and naming patterns, or you can manually specify columns. Different datasets get different anonymized values for better privacy.

Usage

```
anonymize_dataframe(
  data,
  id_cols = NULL,
  name_cols = NULL,
  date_cols = NULL,
  location_cols = NULL,
  age_cols = NULL,
  auto_detect = TRUE,
  detect_by_type = TRUE,
  date_method = "shift",
  date_granularity = "month",
  location_method = "generalize",
  age_method = "10year",
  use_uuid = TRUE,
  seed = NULL,
  dataset_specific = TRUE
)
```

Arguments

data	A data frame or data.table containing patient data
id_cols	Character vector of column names containing patient IDs
name_cols	Character vector of column names containing patient names
date_cols	Character vector of column names containing dates
location_cols	Character vector of column names containing locations
age_cols	Character vector of column names containing ages
auto_detect	Logical, if TRUE (default), automatically detects columns based on data types and common naming patterns
detect_by_type	Logical, if TRUE (default), detects columns by their R data types (Date, character, etc.) in addition to name patterns
date_method	Method for date anonymization: "shift" or "round" (default: "shift"). Use "round" to enable granularity options including "month_year" (YYYYMM format).

`date_granularity`
: For date rounding (when date_method = "round"): "day", "week", "month", "month_year" (returns YYYYMM format, e.g., "202005"), "quarter", or "year" (default: "month")

`location_method`
: Method for location anonymization: "remove" or "generalize"

`age_method`
: Method for age anonymization: "10year" (default) uses 10-year buckets (0-9, 10-19, 20-29, ..., 80-89, 90+) for better research utility, or "hipaa" for HIPAA-compliant buckets (0-17, 18-64, 65-89, 90+)

`use_uuid`
: Logical, if TRUE uses short UUIDs for IDs, names, and locations instead of sequential identifiers (default: TRUE). Dates and ages are not affected.

`seed`
: An optional seed for reproducible anonymization. Different datasets will still get different anonymized values even with the same seed.

`dataset_specific`
: Logical, if TRUE (default), generates dataset-specific seeds so different datasets get different anonymized values

Value

A data frame with anonymized patient data (preserves data.table class if input was data.table)

Examples

```
# Basic usage with auto-detection
patient_data <- data.frame(
  patient_id = c("P001", "P002", "P003"),
  name = c("John Doe", "Jane Smith", "Bob Johnson"),
  dob = as.Date(c("1980-01-15", "1975-03-20", "1990-06-10")),
  location = c("New York, NY", "Los Angeles, CA", "Chicago, IL"),
  diagnosis = c("A", "B", "A")
)
anonymize_dataframe(patient_data, seed = 123)

# With month_year date granularity (YYYYMM format)
anonymize_dataframe(patient_data, date_method = "round", date_granularity = "month_year")

# Works with data.table
if (requireNamespace("data.table", quietly = TRUE)) {
  dt <- data.table::as.data.table(patient_data)
  anonymize_dataframe(dt)
}

# With UUID anonymization (default)
anonymize_dataframe(patient_data, seed = 123)

# Without UUID (sequential IDs)
anonymize_dataframe(patient_data, use_uuid = FALSE, seed = 123)
```

anonymize_dates *Anonymize Dates*

Description

Anonymizes dates by shifting them by a random offset or rounding to a specified granularity. Shifting preserves relative time differences.

Usage

```
anonymize_dates(
  x,
  method = c("shift", "round"),
  days_shift = NULL,
  granularity = "month",
  seed = NULL
)
```

Arguments

x	A vector of dates (Date, POSIXct, or character that can be coerced to Date)
method	Character string specifying anonymization method: "shift" (default) shifts all dates by a random offset, "round" rounds dates to specified granularity
days_shift	For "shift" method: number of days to shift (default: random between -365 and 365)
granularity	For "round" method: "day", "week", "month", "month_year", "quarter", or "year" (default: "month"). "month_year" returns character strings in "YYYYMM" format (e.g., "202005" for May 2020).
seed	An optional seed for reproducible anonymization

Value

A Date vector of anonymized dates (or character vector for "month_year" granularity)

Examples

```
dates <- as.Date(c("2020-01-15", "2020-03-20", "2020-06-10"))
anonymize_dates(dates, method = "shift", seed = 123)
anonymize_dates(dates, method = "round", granularity = "month")
anonymize_dates(dates, method = "round", granularity = "month_year")
```

anonymize_id *Anonymize Patient Identifiers*

Description

Replaces patient identifiers with anonymized versions while maintaining referential integrity (same IDs get the same anonymized value).

Usage

```
anonymize_id(x, prefix = "ID", seed = NULL, use_uuid = TRUE)
```

Arguments

x	A vector of identifiers to anonymize (character, numeric, or factor)
prefix	A character string to prefix anonymized IDs (default: "ID")
seed	An optional seed for reproducible anonymization
use_uuid	Logical, if TRUE uses short UUIDs instead of sequential IDs (default: TRUE).

Value

A character vector of anonymized identifiers

Examples

```
ids <- c("P001", "P002", "P003", "P001")
anonymize_id(ids)
anonymize_id(ids, prefix = "PAT", seed = 123)
anonymize_id(ids, use_uuid = FALSE, seed = 123)  # Use sequential IDs
```

anonymize_locations *Anonymize Geographic Locations*

Description

Anonymizes geographic locations by removing them or replacing with generic labels. Maintains referential integrity (same locations get the same value).

Usage

```
anonymize_locations(
  x,
  method = c("remove", "generalize"),
  prefix = "Location",
  seed = NULL,
  use_uuid = TRUE
)
```

Arguments

x	A character vector of locations to anonymize
method	Character string specifying anonymization method: "remove" (default) removes location information, "generalize" replaces with generic location labels
prefix	For "generalize" method: prefix for generic locations (default: "Location")
seed	An optional seed for reproducible anonymization
use_uuid	Logical, if TRUE uses short UUIDs instead of sequential IDs (default: TRUE). Only applies when method = "generalize".

Value

A character vector of anonymized locations

Examples

```
locations <- c("New York, NY", "Los Angeles, CA", "Chicago, IL")
anonymize_locations(locations, method = "remove")
anonymize_locations(locations, method = "generalize", seed = 123)
anonymize_locations(locations, method = "generalize",
                    use_uuid = FALSE, seed = 123)  # Use sequential IDs
```

---

anonymize_names *Anonymize Patient Names*

Description

Replaces patient names with anonymized identifiers while maintaining referential integrity (same names get the same anonymized value).

Usage

```
anonymize_names(x, prefix = "Patient", seed = NULL, use_uuid = TRUE)
```

Arguments

x	A character vector of names to anonymize
prefix	A character string to prefix anonymized names (default: "Patient")
seed	An optional seed for reproducible anonymization
use_uuid	Logical, if TRUE uses short UUIDs instead of sequential IDs (default: TRUE).

Value

A character vector of anonymized names

Examples

```
names <- c("John Doe", "Jane Smith", "Bob Johnson")
anonymize_names(names)
anonymize_names(names, prefix = "PAT", seed = 123)
anonymize_names(names, use_uuid = FALSE, seed = 123)  # Use sequential IDs
```

Index

anonymize_age, 2
anonymize_dataframe, 3
anonymize_dates, 5
anonymize_id, 6
anonymize_locations, 6
anonymize_names, 7