Package 'clinTrialData'

March 17, 2026

Title Clinical Trial Example Datasets

Version 0.1.3

Description A collection of clinical trial example datasets from
multiple sources including the CDISC Pilot 01 study
(CDISC <https://www.cdisc.org/>). All datasets are provided in Parquet
format for efficient storage and can be accessed using the 'connector'
package. Designed for training, testing, prototyping, and demonstrating
clinical data analysis workflows.

Depends R (>= 4.1.0)

License Apache License (>= 2)

URL https://lovemore-gakava.github.io/clinTrialData/,
https://github.com/Lovemore-Gakava/clinTrialData

BugReports https://github.com/Lovemore-Gakava/clinTrialData/issues

Encoding UTF-8

Language en-US

RoxygenNote 7.3.3

Imports connector, httr, jsonlite, piggyback, tools

Suggests arrow, dplyr, ggplot2, testthat (>= 3.0.0), knitr, rmarkdown,
tidyr

Config/testthat/edition 3

VignetteBuilder knitr

NeedsCompilation no

Author Lovemore Gakava [aut, cre, cph]

Maintainer Lovemore Gakava <Lovemore.Gakava@gmail.com>

Repository CRAN

Date/Publication 2026-03-17 08:40:08 UTC

Contents

Index **10**

---

cache_dir   *Get the Local Cache Directory*

Description

Returns the path to the local cache directory where downloaded clinical trial datasets are stored. The location follows the platform-specific user data directory convention via `tools::R_user_dir()`.

You can delete any subdirectory here to remove a cached dataset, or clear the entire directory to free disk space.

Usage

```
cache_dir()
```

Value

A character string with the path to the cache directory.

Examples

```
cache_dir()
```

---

clinTrialData-data *Clinical Trial Datasets*

Description

The clinTrialData package contains clinical trial datasets from multiple sources, stored in Parquet format. Data is accessed using connector functions.

Available Data Sources

CDISC Pilot 01 Study:

The CDISC Pilot 01 study data includes both ADaM and SDTM domains.

ADaM datasets include:

- ADSL: Subject-Level Analysis Dataset
- ADAE: Adverse Events Analysis Dataset
- ADLBC: Laboratory Analysis Dataset (Chemistry)
- ADLBH: Laboratory Analysis Dataset (Hematology)
- ADLBHY: Laboratory Analysis Dataset (Hy's Law)
- ADQSADAS: ADAS-Cog Questionnaire Analysis Dataset
- ADQSCIBC: CIBC Questionnaire Analysis Dataset
- ADQSNPIX: NPI-X Questionnaire Analysis Dataset
- ADTTE: Time-to-Event Analysis Dataset
- ADVS: Vital Signs Analysis Dataset

SDTM datasets include:

- DM: Demographics
- AE: Adverse Events
- VS: Vital Signs
- LB: Laboratory Test Results
- And 18 additional domains (see list_data_sources() for details)

Usage

Data sources are discovered by scanning the package directory structure. List available datasets with list_data_sources().

Access data using the connection function:

```
# Connect to any data source (e.g., CDISC Pilot data)
db <- connect_clinical_data("cdisc_pilot")

# List available datasets
db$adam$list_content_cnt()

# Read a dataset
```

```
adsl <- db$adam$read_cnt("adsl")

# See all available data sources
list_data_sources()
```

Data Format

Datasets are stored in Parquet format:

- Columnar storage

- Fast reads

- Compression

- Cross-platform compatibility

Source

CDISC Pilot 01 Study Data Various clinical trial data sources

References

CDISC. Clinical Data Interchange Standards Consortium. https://www.cdisc.org/

---

connect_clinical_data   *Connect to Clinical Data by Source*

Description

Generic connection function that allows access to any data source in the package. Data sources are automatically discovered by scanning the package's example data directory structure.

Usage

```
connect_clinical_data(source = "cdisc_pilot")
```

Arguments

source   Character string specifying the data source. Use list_data_sources() to see all available options.

Value

A connectors object

Examples

```
if (interactive()) {
  # Connect to CDISC Pilot data
  db <- connect_clinical_data("cdisc_pilot")

  # List available datasets
  db$adam$list_content_cnt()

  # Read a dataset (requires the arrow package)
  if (requireNamespace("arrow", quietly = TRUE)) {
    adsl <- db$adam$read_cnt("adsl")
  }

  # List available sources
  list_data_sources()
}
```

---

dataset_info   *Inspect a Clinical Trial Dataset Without Downloading*

Description

Fetches and displays metadata for any study available in the `clinTrialData` library – without downloading the full dataset. Metadata includes the study description, available domains and datasets, subject count, version, and data source attribution.

For studies already downloaded via [download_study()](#), the metadata is read from the local cache and works offline. For studies not yet downloaded, a small JSON file (~2KB) is fetched from the GitHub Release.

Usage

```
dataset_info(source, repo = "Lovemore-Gakava/clinTrialData")
```

Arguments

source   Character string. Name of the study (e.g. `"cdisc_pilot_extended"`). Use
              [list_available_studies()](#) to see all options.

repo   GitHub repository in the form `"owner/repo"`. Defaults to the official `clinTrialData`
              release repository.

Value

Invisibly returns the metadata as a named list.

Examples

```
dataset_info("cdisc_pilot")
```

---

download_study | *Download a Clinical Trial Study Dataset*

Description

Downloads a study dataset from a GitHub Release and stores it in the local cache (see `cache_dir()`). Once downloaded, the study is available to `connect_clinical_data()` without an internet connection.

Requires the `piggyback` package.

Usage

```
download_study(
  source,
  version = "latest",
  force = FALSE,
  repo = "Lovemore-Gakava/clinTrialData"
)
```

Arguments

source	Character string. The name of the study to download (e.g. `"cdisc_pilot"`). Use `list_available_studies()` to see all options.
version	Character string. The release tag to download from. Defaults to `"latest"`, which resolves to the most recent release.
force	Logical. If `TRUE`, re-download even if the study is already cached. Defaults to `FALSE`.
repo	GitHub repository in the form `"owner/repo"`. Defaults to the official `clinTrialData` release repository.

Value

Invisibly returns the path to the cached study directory.

Examples

```
if (interactive()) {
  # Download a study not bundled with the package
  download_study("cdisc_pilot_extended")

  # Then connect as usual
  db <- connect_clinical_data("cdisc_pilot_extended")
}
```

```
list_available_studies
```

List Studies Available for Download

Description

Returns a data frame of all clinical trial studies available as GitHub Release assets, along with their local cache status. Studies marked as `cached = TRUE` are already downloaded and available for use with `connect_clinical_data()` without an internet connection.

When GitHub is unreachable, the function falls back to the last successfully fetched listing (if available) and issues a warning. The `cached` column is always recomputed from the local filesystem.

Requires the `piggyback` package.

Usage

```
list_available_studies(repo = "Lovemore-Gakava/clinTrialData")
```

Arguments

repo   GitHub repository in the form `"owner/repo"`. Defaults to the official `clinTrialData` release repository.

Value

A data frame with columns:

source Study name (pass this to `download_study()` or `connect_clinical_data()`)

version Release tag the asset belongs to

size_mb Asset size in megabytes

cached `TRUE` if the study is already in the local cache

Examples

```
if (interactive()) {
  list_available_studies()
}
```

---

list_data_sources   *List Available Clinical Data Sources*

Description

Returns information about all clinical datasets available locally – both datasets bundled with the package and any datasets previously downloaded via [download_study()](). The location column indicates whether a dataset is ″bundled″ (shipped with the package) or ″cached″ (downloaded to the user cache directory).

To see datasets available for download from GitHub, use [list_available_studies()]().

Usage

```
list_data_sources()
```

Value

A data frame with columns:

source Dataset name (pass to [connect_clinical_data()]())

description Human-readable study description

domains Comma-separated list of available data domains (e.g. ″adam, sdtm″)

format Storage format (″parquet″)

location Either ″bundled″ or ″cached″

Examples

```
list_data_sources()
```

---

remove_cnt.ConnectorLockedFS
                              Remove Content with Lock Check

Description

S3 method for remove_cnt that checks if the study folder is locked before allowing remove operations.

Usage

```
## S3 method for class 'ConnectorLockedFS'
remove_cnt(connector_object, name, ...)
```

Arguments

connector_object

> The ConnectorLockedFS object

name   The file name to remove

...   Additional arguments passed to the underlying connector

Value

Invisible connector_object

---

write_cnt.ConnectorLockedFS

Write Content with Lock Check

Description

S3 method for write_cnt that checks if the study folder is locked before allowing write operations.

Usage

```
## S3 method for class 'ConnectorLockedFS'
write_cnt(connector_object, x, name, overwrite = FALSE, ...)
```

Arguments

connector_object

> The ConnectorLockedFS object

x   The data to write

name   The file name

overwrite   Whether to overwrite existing files

...   Additional arguments passed to the underlying connector

Value

Invisible connector_object

Index