Package: banffIT (via r-universe)

October 28, 2026

Type Package

Title Automated Standardized Assignment of the Banff Classification

Version 2.0.0

Maintainer Guillaume Fabre <guijoseph.fabre@gmail.com>

Description Assigns standardized diagnoses using the Banff Classification (Category 1 to 6 diagnoses, including Acute and Chronic active T-cell mediated rejection as well as Active, Chronic active, and Chronic antibody mediated rejection). The main function considers a minimal dataset containing biopsies information in a specific format (described by a data dictionary), verifies its content and format (based on the data dictionary), assigns diagnoses, and creates a summary report. The package is developed on the reference guide to the Banff classification of renal allograft pathology Roufosse C, Simmonds N, Clahsen-van Groningen M, et al. A (2018) <doi:10.1097/TP.0000000000002366>. The full description of the Banff classification is available at <https://banfffoundation.org/>.

License GPL-3

Depends R (>= 3.4)

URL https://github.com/PersonalizedTransplantCare/banffIT

BugReports https://github.com/PersonalizedTransplantCare/banffIT/issues

Imports dplyr (>= 1.1.0), rlang, tools, utils, crayon, stringr, fs, lubridate, tidyr, fabR, madshapR (>= 2.0.0)

Suggests knitr, rmarkdown, usethis

VignetteBuilder knitr

Encoding UTF-8

Language en-US

RoxygenNote 7.2.3

NeedsCompilation no

Author Guillaume Fabre [aut, cre] (ORCID:
<https://orcid.org/0000-0002-0124-9970>), Ruth Sapir-Pichhadze
[fnd, cph, ctb], Jan Ulrich Becker [ctb], Samuel El Bouzaïdi
Tiali [ctb], Jérôme Laforme [ctb], Tina Wey [ctb], Edden
Gitelman [ctb]

Config/pak/sysreqs
cmake git make libgit2-dev libicu-dev libuv1-dev libssl-dev libx11-dev zlib1g-dev

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

Date/Publication 2026-10-28 14:02:00 UTC

RemoteUrl https://github.com/cranhaven/cranhaven.r-universe.dev

RemoteRef package/banffIT

RemoteSha a35026274b0d3875d596f332195584ec531aed6b

RemoteSubdir banffIT

Contents

add_diagnoses 2
banff_dataset_evaluate 3
banff_launcher 4
banffIT_website 5
calculate_adequacy 6
dataset_cat_as_labels 6
dataset_summarize 7
get_banff_dictionary 7
get_banff_example 8
get_banff_template 9
get_banff_version 9

Index **11**

`add_diagnoses` *Add diagnoses to the input dataset*

Description

This function takes a dataset and returns a diagnosis for each observation. For the function to run, the dataset must not contain any errors that `banff_launcher()` would have detected. Please prefer using `banff_launcher()` to run additional tests.

Usage

```
add_diagnoses(banff_dataset, version = NULL)
```

Arguments

banff_dataset	A tibble object.
version	A character string referring the version of Banff classification. The most recent classification is the default. Options are "2022" (default), "2017".

Value

A tibble object that contains additional variables with the diagnoses results.

Examples

```
{

banff_dataset <- get_banff_template()
add_diagnoses(banff_dataset)

}
```

---

banff_dataset_evaluate

Evaluates the format and content of the input dataset

Description

This function takes a dataset and evaluates its format and content based on the accepted format specified in the data dictionary. It applies a series of checks to make sure the dataset is ready to be processed by the add_diagnoses() function which assigns diagnoses to each observation of the dataset. The function evaluates whether:

- The input file is a dataset
- All mandatory variables are present in the dataset
- Missing values (NA) are present in variables where they are not allowed
- Data types are correct.
- The combination of ID, center, and biopsy date is unique
- There are duplicated variable in the dataset
- Dates are valid
- Content values follow the category values as specified in the data dictionary
- Constraints specified in the data dictionary are respected

Usage

```
banff_dataset_evaluate(banff_dataset, version = NULL)
```

Arguments

banff_dataset	A tibble object.
version	A character string referring the version of Banff classification. The most recent classification is the default. Options are "2022" (default), "2017".

Value

A list of tibble objects giving information on the assessment of the dataset.

Examples

```
{

banff_dataset <- get_banff_template()
banff_dataset_evaluate(banff_dataset)

}
```

---

banff_launcher *Launch the program*

Description

This function takes a path string identifying the input file path. The function internally runs a series of tests that assess the input dataset. If any of these tests fails, the user gets information allowing them to correct the input dataset and rerun the process. Once all tests pass, the dataset is given as an output with a diagnosis for each observation (using the function add_diagnoses() internally). The output dataset, along with its associated labels (label:en by default) are provided to the user in an Excel format file accessible in the output_folder specified. The output dataset comes with a report that summarizes information about variable distributions and descriptive statistics.

Usage

```
banff_launcher(
  input_file,
  output_folder,
  language = "label:en",
  version = NULL,
  option_filter,
  detail = FALSE
)
```

Arguments

input_file	A character string identifying the path of the input file (must be a CSV or a one sheet Excel file) to be processed.
output_folder	A character string identifying the folder path where the output files will be saved.
language	Optional argument allowing the user to get the diagnoses in a specific language. Options are "label:en" (default), "label:fr", "label:de", "label:sp", "label:nl", "label:jp", "label:in".
version	A character string referring the version of Banff classification. The most recent classification is the default. Options are "2022" (default), "2017".
option_filter	Optional argument allowing the user to filter the dataset using dplyr::filter() syntax. The variable used to filter must exist.
detail	Optional argument indicating whether the output should include temporary variables generated in the process or not. FALSE by default.

Value

Nothing to be returned. The function generates a folder (if not already existing) where Excel files are stored.

Examples

```
{

version <- get_banff_version()
input_file <- system.file("extdata",
   paste0(version,"/banff_template.xlsx"), package = "banffIT")
banff_launcher(input_file, output_folder = tempdir())

}
```

---

banffIT_website *Call the online documentation*

Description

This function sends the user to the online documentation for the package, which includes a description of the latest version of the package, vignettes, user guides, and a reference list of functions and help pages.

Usage

```
banffIT_website()
```

Value

Nothing to be returned. The function opens a web page.

Examples

```
{

banffIT_website()

}
```

`calculate_adequacy` *Calculate adequacy of each biopsy from glomeruli and arteries variables*

Description

This function calculates adequacy of each biopsy (i.e., each observation) based on glomeruli and arteries variables (if provided).

Usage

```
calculate_adequacy(banff_dataset)
```

Arguments

`banff_dataset`	A tibble object.

Value

A tibble object with two variables: the calculated adequacy (adequacy_calculated) and the adequacy specified in input (adequacy_input).

Examples

```
{

banff_dataset <- get_banff_template()
calculate_adequacy(banff_dataset)

}
```

`dataset_cat_as_labels` *Objects exported from other packages*

Description

These objects are imported from other packages. Follow the links below to see their documentation.

madshapR `dataset_cat_as_labels`

dataset_summarize *Objects exported from other packages*

Description

These objects are imported from other packages. Follow the links below to see their documentation.

madshapR dataset_summarize

get_banff_dictionary *Get the Banff data dictionary*

Description

This function gets the data dictionary used to control the consistency of the input dataset.

Usage

```
get_banff_dictionary(
  version = NULL,
  which = NULL,
  language = "label:en",
  detail = FALSE
)
```

Arguments

version	A character string referring the version of Banff classification. The most recent classification is the default. Options are "2022" (default), "2017".
which	Indicates which variables to get from the Banff data dictionary. If NULL both input and output variables are provided.*
language	Optional input allowing the user to get the diagnoses in a specific language. Options are "label:en" (default), "label:fr", "label:de", "label:sp", "label:nl", "label:jp", "label:in".
detail	Optional argument indicating whether the data dictionary should include temporary variables generated in the process or not. FALSE by default.

Value

A list of tibbles representing meta data used in the process. The metadata are the list of variables used, and their associated categories, if any.

Examples

```
{

  get_banff_dictionary()

}
```

`get_banff_example` *Get a dataset example*

Description

This function gets the dataset used in the vignette as an example.

Usage

```
get_banff_example(version = NULL)
```

Arguments

`version`	A character string referring the version of Banff classification. The most recent classification is the default. Options are "2022" (default), "2017".

Value

A tibble representing the dataset used in the the vignette as an example.

Examples

```
{

  get_banff_example()

}
```

get_banff_template *Get a dataset template*

Description

This function gets the empty dataset with variables that are mandatory in the process.

Usage

```
get_banff_template(version = NULL)
```

Arguments

version	A character string referring the version of Banff classification. The most recent classification is the default. Options are "2022" (default), "2017".

Value

A tibble representing the empty dataset.

Examples

```
{

  get_banff_template()

}
```

get_banff_version *Extract version of the Banff classification*

Description

This internal function extracts for user input the version of the Banff classification.

Usage

```
get_banff_version(version = NULL)
```

Arguments

version	A character string referring the version of Banff classification. The most recent classification is the default. Options are "2022" (default), "2017".

Value

A character string which is the version of Banff classification.

Examples

```
{

get_banff_version()

}
```

Index

∗ **imported**
dataset_cat_as_labels, 6
dataset_summarize, 7

add_diagnoses, 2
add_diagnoses(), *3, 4*

banff_dataset_evaluate, 3
banff_launcher, 4
banff_launcher(), *2*
banffIT_website, 5

calculate_adequacy, 6

dataset_cat_as_labels, 6, *6*
dataset_summarize, 7, *7*
dplyr::filter(), *5*

get_banff_dictionary, 7
get_banff_example, 8
get_banff_template, 9
get_banff_version, 9