Package ‘blindspiker’

February 2, 2026

Title Laboratory Blind Spike Sample Analyses

Version 0.2.0

Description A blind spike program provides samples to a laboratory in order to perform quality control (QC) checks. The samples provided are of a known quantity to the tester. The laboratory is typically uninformed of that the sample provided is a QC sample.

License GPL-3

Encoding UTF-8

Imports dplyr, magrittr, binGroup, ggplot2, tidyr, gt

RoxygenNote 7.3.3

Suggests knitr, rmarkdown

VignetteBuilder knitr

URL `https://github.com/markhogue/blindspiker`

BugReports `https://github.com/markhogue/blindspiker/issues`

Depends R (>= 3.6)

NeedsCompilation no

Author Mark Hogue [aut, cre]

Maintainer Mark Hogue <`mark.hogue.chp@gmail.com`>

Repository CRAN

Date/Publication 2026-02-02 10:40:06 UTC

Contents

`bs_prep_and_analysis` *Load data associated with the spiked samples and laboratory results from comma separated variable text files.*

Description

This function loads spike data and the laboratory results, then processes it so that the ‘sample_ID‘ ties the spike value and the results together, using the ‘left_join‘ function from the ‘dplyr‘ package.

Usage

```
bs_prep_and_analysis(spike_data, lab_data)
```

Arguments

`spike_data`	name of the loaded dataset (no quotes) or the name of the file, with path, containing the spike values, in quotes. Example forms, my_spike_data, or "C:/my_directory/my_spike_data.csv". This file column headings must be as identified in the Details section.
`lab_data`	name of the loaded dataset (no quotes) or the name of the file, with path, containing the laboratory results. Example forms, my_lab_data, or "C:/my_directory/my_lab_data.csv". This file column headings must be as identified in the Details section.

Details

False negative results are flagged for laboratory results below the detection level in the analysis of a sample spiked above the detection level. False positives are flagged for laboratory results above the detection when the analyte was not spiked. Error rates are computed with the ‘table_false‘ function.

To use this function, first set up spike value data in a .csv file (in any column order) matching the column headers of the following values:

Required for spike data:

- ‘sample_ID‘ unique identifier, character or numeric
- ‘analyte‘ character data
- ‘spike_value‘ numeric value
- ‘spike_units‘ character data
- ‘submission_date‘ character data that will be converted to date in format YYYY-MM-DD (for example 1999-12-31)

Optional for spike data:

- ‘sv_unc‘ numeric, the uncertainty of the spike value. Default = 0
- ‘sv_k‘ the coverage factor for the spike value uncertainty. Default = 2
- ‘provider lab‘ character name of laboratory providing spiked samples

Required for laboratory results:

- ‘sample_ID‘ must match spike ‘sample_ID‘
- ‘analyte‘ must match spike ‘analyte‘
- ‘result‘ numeric value
- ‘units‘ must match ‘spike_units‘
- ‘result_date‘
- ‘det_lvl‘ numeric detection level
- ‘unc‘ numeric uncertainty of the laboratory result

#’ Optional for lab data:

- ‘k‘ the coverage factor for the result uncertainty. Default = 2

Note that the two data sets (spike values and laboratory results) will be combined by ‘sample_ID‘, and also by ‘analyte‘ if present in both sets. (If the laboratory data includes a non-zero result for an analyte not present in the spike data, that would indicate a false positive.)

Use this function to load the spike data and establish a data frame named ‘bs_df‘. All plotting and data analysis functions will default to look for this data set.

Value

data frame containing all needed data to be used in subsequent functions.

Examples

```
example_spike_data <- system.file("extdata", "spikevals.csv", package = "blindspiker")
example_lab_data <- system.file("extdata", "labvals.csv", package = "blindspiker")
example_df <- bs_prep_and_analysis(spike_data = example_spike_data, lab_data = example_lab_data)
```

`plot_qq` *quantile-quantile plot*

Description

QQ plots by isotope

Usage

```
plot_qq(select_analyte, dat = bs_df)
```

Arguments

`select_analyte`	the selected analyte for this run chart
`dat`	data frame with all data needed as described in ‘bs_prep_and_analysis‘.Default is ‘bs_df‘.

Value

quantile-quantile plot of laboratory results of spiked samples

Examples

```
example_spike_data <- system.file('extdata', 'spikevals.csv', package = 'blindspiker')
example_lab_data <- system.file('extdata', 'labvals.csv', package = 'blindspiker')
example_df <- bs_prep_and_analysis(spike_data = example_spike_data, lab_data = example_lab_data)
plot_qq(select_analyte = 'unknownium', dat = example_df)
```

`plot_run` *Plot Run Chart*

Description

‘plot_run()‘ produces a Run Chart of the selected analyte. The analyte is selected from the analyte set provided. Results are plotted with error bars (uncertainty with coverage factor of 2) when the result is greater than the detection level. On the ratio version, the uncertainties for the results and spike values are combined as the square root of the sums of the relative uncertainties squared. When original results are plotted, the spike values are shown with a small salmon-colored "+".

Usage

```
plot_run(
  select_analyte,
  dat = bs_df,
  version = "original",
  log = "n",
  removal_notification = "n"
)
```

Arguments

`select_analyte`	the selected analyte for this run chart
`dat`	data frame with all data needed as described in ‘bs_prep_and_analysis‘. Default is ‘bs_df‘.
`version`	The run chart is either shown with ‘original‘ units, default, or with the result shown as a ‘ratio‘ to the spike value.
`log`	Set log = "y" to make the y-axis a log scale - original version only. Default is "n".
`removal_notification`	provide a list of results, by sample_ID that were removed from the current run plot because results were less than or equal to zero. @return run plot of laboratory analyses of spiked samples

Examples

```
example_spike_data <- system.file('extdata', 'spikevals.csv', package = 'blindspiker')
example_lab_data <- system.file('extdata', 'labvals.csv', package = 'blindspiker')
example_df <- bs_prep_and_analysis(spike_data = example_spike_data, lab_data = example_lab_data)
plot_run(select_analyte = 'unknownium', dat = example_df)
```

`plot_tat` *Plot turnaround time*

Description

Plot time for laboratory analysis by date (result date from lab - spiked sample submitted date)

Usage

```
plot_tat(select_analyte, dat = bs_df, target_days = 60)
```

Arguments

`select_analyte`	the selected analyte for this run chart
`dat`	data frame with all data needed as described in ‘bs_prep_and_analysis‘. Default is ‘bs_df‘.
`target_days`	The target turnaround time in days. Default = 60.

Value

turn-around-time plot

Examples

```
example_spike_data <- system.file('extdata', 'spikevals.csv', package = 'blindspiker')
example_lab_data <- system.file('extdata', 'labvals.csv', package = 'blindspiker')
example_df <- bs_prep_and_analysis(spike_data = example_spike_data, lab_data = example_lab_data)
plot_tat(select_analyte = 'unknownium', dat = example_df, target_days = 60)
```

`spike_combos` *Count combinations of spiked samples*

Description

A summary of spiked samples is provided based on combinations of interest.

Usage

```
spike_combos(analytes, dat = bs_df)
```

Arguments

`analytes`	a vector of analytes of interest
`dat`	data frame with all data needed as described in ‘bs_prep_and_analysis‘. Default is ‘bs_df‘.

Value

table of combinations of analytes in blind spikes

Examples

```
example_spike_data <- system.file('extdata', 'spikevals.csv', package = 'blindspiker')
example_lab_data <- system.file('extdata', 'labvals.csv', package = 'blindspiker')
example_df <- bs_prep_and_analysis(spike_data = example_spike_data, lab_data = example_lab_data)
spike_combos(analytes = c('unknownium', 'Sr-90'), dat = example_df)
```

`table_false` *Make table of false positives and false negatives in laboratory results*

Description

Make a table with estimated confidence intervals for false negatives and false positives for an analyte in the data set.

Usage

```
table_false(select_analyte, dat = bs_df)
```

Arguments

`select_analyte`	the selected analyte for this table
`dat`	data frame with all data needed as described in ‘bs_prep_and_analysis‘.

Details

False negatives are the number of laboratory results that missed a spiked value. For the false negative rate, the numerator is the number of laboratory results less than detection level for spiked samples. The denominator is the number or spiked samples.

False positives are the number of laboratory results above detection level when the analyte identified by the laboratory was not in the spiked sample. For the false positive rate, the numerator is the number of false positives. The denominator is the number of false positives plus number of true negatives.

The total error rate is the total number of laboratory results with either false negative or false positive results divided by the total number of laboratory results.

Value

table of false positive and false negative results

Examples

```
example_spike_data <- system.file('extdata', 'spikevals.csv', package = 'blindspiker')
example_lab_data <- system.file('extdata', 'labvals.csv', package = 'blindspiker')
example_df <- bs_prep_and_analysis(spike_data = example_spike_data, lab_data = example_lab_data)
table_false(select_analyte = "Sr-90", dat = example_df)
```

`table_spike` *Make table summarizing the blind spikes.*

Description

Make a table showing how many blind spike samples contained one or more analytes.

Usage

```
table_spike(dat = bs_df)
```

Arguments

`dat`	data frame with all data needed as described in ‘bs_prep_and_analysis‘.

Value

table of all analytes in blind spike samples

See Also

For combinations of analytes spiked in a single sample, see ‘spike_combos‘.

Examples

```
example_spike_data <- system.file('extdata', 'spikevals.csv', package = 'blindspiker')
example_lab_data <- system.file('extdata', 'labvals.csv', package = 'blindspiker')
example_df <- bs_prep_and_analysis(spike_data = example_spike_data, lab_data = example_lab_data)
table_spike(dat = example_df)
```

Index