Package ‘stddiff.spark’

January 15, 2026

Title Calculate the Standardized Difference for Numeric, Binary and Category Variables in Apache Spark

Version 1.0

Description Provides functions to compute standardized differences for numeric, binary, and categorical variables on Apache Spark DataFrames using 'sparklyr'. The implementation mirrors the methods used in the 'stddiff' package but operates on distributed data. See Zhicheng Du, Yuantao Hao (2022) <doi:10.32614/CRAN.package.stddiff> for reference.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 4.1.0)

Imports dplyr (>= 1.1.0), tidyr (>= 1.3.0), sparklyr (>= 1.8.0)

Suggests stddiff (>= 2.1.0), testthat (>= 3.0.0), withr

Config/testthat/edition 3

SystemRequirements Apache Spark (tested with 3.4.4)

URL https://github.com/alicja-januszkiewicz/stddiff.spark

BugReports https://github.com/alicja-januszkiewicz/stddiff.spark/issues

NeedsCompilation no

Author Alicja Januszkiewicz [aut, cre, cph]

Maintainer Alicja Januszkiewicz <cran.alicja.januszkiewicz@gmail.com>

Repository CRAN

Date/Publication 2026-01-15 17:50:01 UTC

Contents

stddiff.binary 2
stddiff.category 3
stddiff.numeric 4

Index **6**

stddiff.binary *Compute Standardized Differences for Binary Variables (Spark)*

Description

Calculates standardized differences for binary variables using a Spark DataFrame. Equivalent to `stddiff::stddiff.binary` but operates on Spark data.

Usage

```
stddiff.binary(data, gcol, vcol, verbose = FALSE)
```

Arguments

`data`	A Spark DataFrame (`tbl_spark`) containing the variables.
`gcol`	Integer; column index of the binary grouping variable (e.g., treatment vs control).
`vcol`	Integer vector; column indices of the binary variables to analyze.
`verbose`	Logical; if TRUE, prints progress messages. Default is FALSE.

Details

Variables are encoded using lexicographic ordering since Spark does not have factor types. The first level alphabetically becomes 0, the second becomes 1.

The standardized difference is computed as:

$$d = \frac{|p_t - p_c|}{\sqrt{(p_t(1 - p_t) + p_c(1 - p_c))/2}}$$

Value

A numeric matrix with one row per variable and columns:

- `p.c`: Proportion in control group (first level alphabetically)
- `p.t`: Proportion in treatment group (second level alphabetically)
- `missing.c`: Number of missing values in control group
- `missing.t`: Number of missing values in treatment group
- `stddiff`: Standardized difference
- `stddiff.l`: Lower bound of 95% confidence interval
- `stddiff.u`: Upper bound of 95% confidence interval

See Also

`stddiff.category`, `stddiff.numeric`

Examples

```
sc <- sparklyr::spark_connect(master = ”local”)

spark_df <- sparklyr::copy_to(sc, mtcars)

result <- stddiff.binary(
  data = spark_df,
  gcol = 9,   # column index of grouping variable
  vcol = c(8) # columns of binary variables
)

sparklyr::spark_disconnect(sc)
```

`stddiff.category` *Compute Standardized Differences for Categorical Variables (Spark)*

Description

Calculates standardized differences for categorical variables using a Spark DataFrame. Equivalent to `stddiff::stddiff.category` but operates on Spark data.

Usage

```
stddiff.category(data, gcol, vcol, verbose = FALSE)
```

Arguments

`data` A Spark DataFrame (`tbl_spark`) containing the variables.

`gcol` Integer; column index of the binary grouping variable.

`vcol` Integer vector; column indices of the categorical variables to analyze.

`verbose` Logical; if TRUE, prints progress messages. Default is FALSE.

Details

For categorical variables with K levels, the standardized difference is computed using a multivariate approach that accounts for all K-1 levels simultaneously (excluding the reference level). Category levels are sorted lexicographically; the first level alphabetically serves as the reference.

Value

A numeric matrix with one row per category level and columns:

- `p.c`: Proportion in control group
- `p.t`: Proportion in treatment group
- `missing.c`: Number of missing values in control group (first row only)
- `missing.t`: Number of missing values in treatment group (first row only)

- `stddiff`: Standardized difference (first row only)
- `stddiff.l`: Lower CI bound (first row only)
- `stddiff.u`: Upper CI bound (first row only)

Row names are formatted as "variable_name level_name".

See Also

`stddiff.binary`, `stddiff.numeric`

Examples

```
sc <- sparklyr::spark_connect(master = "local")

spark_df <- sparklyr::copy_to(sc, as.data.frame(Titanic))

result <- stddiff.category(
  data = spark_df,
  gcol = 4,   # column index of grouping variable
  vcol = c(1) # columns of categorical variables
)

sparklyr::spark_disconnect(sc)
```

`stddiff.numeric` *Compute Standardized Differences for Numeric Variables (Spark)*

Description

Calculates standardized differences for continuous numeric variables using a Spark DataFrame. Equivalent to `stddiff::stddiff.numeric` but operates on Spark data.

Usage

```
stddiff.numeric(data, gcol, vcol, verbose = FALSE)
```

Arguments

`data`	A Spark DataFrame (`tbl_spark`) containing the variables.
`gcol`	Integer; column index of the binary grouping variable.
`vcol`	Integer vector; column indices of the numeric variables to analyze.
`verbose`	Logical; if TRUE, prints progress messages. Default is FALSE.

Details

The standardized difference for continuous variables is computed as:

$$d = \frac{|\bar{x}_t - \bar{x}_c|}{\sqrt{(s_t^2 + s_c^2)/2}}$$

where $\bar{x}$ represents means and s^2 represents variances.

This is equivalent to Cohen’s d with pooled standard deviation.

Value

A numeric matrix with one row per variable and columns:

- `mean.c`: Mean in control group
- `sd.c`: Standard deviation in control group
- `mean.t`: Mean in treatment group
- `sd.t`: Standard deviation in treatment group
- `missing.c`: Number of missing values in control group
- `missing.t`: Number of missing values in treatment group
- `stddiff`: Standardized difference
- `stddiff.l`: Lower bound of 95% confidence interval
- `stddiff.u`: Upper bound of 95% confidence interval

See Also

`stddiff.binary`, `stddiff.category`

Examples

```
sc <- sparklyr::spark_connect(master = "local")

spark_df <- sparklyr::copy_to(sc, mtcars)

result <- stddiff.numeric(
  data = spark_df,
  gcol = 8,          # column index of grouping variable
  vcol = c(1, 2, 5)  # columns of numeric variables
)

sparklyr::spark_disconnect(sc)
```

Index

stddiff.binary, *2*, *4*, *5*
stddiff.category, *2*, 3, *5*
stddiff.numeric, *2*, *4*, 4