

# Package ‘schtools’

May 12, 2022

**Title** Schloss Lab Tools for Reproducible Microbiome Research

**Version** 0.2.1

**Date** 2022-05-11

**Description** A collection of useful functions and example code created and used by the Schloss Lab for reproducible microbiome research. Perform common tasks like read files created by `mothur` <<https://mothur.org/>>, tidy up your microbiome data, and format R Markdown documents for publication. See the website <<http://www.schlosslab.org/schtools/>> for more information, documentation, and examples.

**License** MIT + file LICENSE

**URL** <http://www.schlosslab.org/schtools/>,  
<https://github.com/SchlossLab/schtools>

**BugReports** <https://github.com/SchlossLab/schtools/issues>

**Imports** dplyr, formatR, glue, rlang, stringr, tidyr

**Suggests** covr, cowplot, ggplot2, ggtext, here, hexSticker, knitr,  
purrr, rmarkdown, showtext, testthat

**VignetteBuilder** knitr

**Encoding** UTF-8

**RoxygenNote** 7.1.1

**NeedsCompilation** no

**Author** Kelly Sovacool [aut, cre] (<<https://orcid.org/0000-0003-3283-829X>>),  
Nick Lesniak [aut] (<<https://orcid.org/0000-0001-9359-5194>>),  
Patrick Schloss [aut],  
Sarah Lucas [ctb] (<<https://orcid.org/0000-0003-1676-5801>>),  
Courtney Armour [ctb] (<<https://orcid.org/0000-0002-5250-1224>>),  
Jacqueline Moltzau [ctb] (<<https://orcid.org/0000-0003-1398-5980>>),  
Andrew Hannah [ctb],  
Nielson Baxter [ctb],  
Alyxandria Schubert [ctb],  
Kathryn Iverson [ctb]

**Maintainer** Kelly Sovacool <sovacool@umich.edu>

Repository CRAN

Date/Publication 2022-05-11 23:20:02 UTC

## R topics documented:

close_enough . . . . .	2
format_number . . . . .	3
inline_hook . . . . .	4
is_nearly_whole . . . . .	4
is_nondesc . . . . .	5
load_deps . . . . .	6
parse_tax . . . . .	6
paste_oxford_list . . . . .	7
read_dist . . . . .	8
read_tax . . . . .	8
schtools . . . . .	9
set_knitr_opts . . . . .	9

<b>Index</b>	<b>10</b>
--------------	-----------

---

close_enough	<i>Check whether two numeric vectors are close enough for gov't work.</i>
--------------	---

---

### Description

This is like `dplyr::near()` except with much less precision.

### Usage

```
close_enough(x, y, tol = 10^-3)
```

### Arguments

x	a numeric vector
y	another numeric vector
tol	tolerance (default: $10^{-3}$ .)

### Value

TRUE if all numbers are near enough within the tolerance, otherwise FALSE

### Author(s)

Kelly Sovacool <sovacool@umich.edu>



---

inline_hook	<i>Inline hook for knitr to paste human-readable numbers and nice lists.</i>
-------------	--

---

**Description**

Inline hook for knitr to paste human-readable numbers and nice lists.

**Usage**

```
inline_hook(x)
```

**Arguments**

x	just about anything
---	---------------------

**Value**

a string where each element in x is separated by a comma and numbers are in a human-readable format.

**Author(s)**

Kelly Sovacool <sovacool@umich.edu>

Pat Schloss <pschloss@umich.edu>

**Examples**

```
inline_hook(c(1.2993992, 0.03, 1000))  
inline_hook(c("cats", "dogs"))
```

---

is_nearly_whole	<i>Checks whether a number is near to a whole number</i>
-----------------	--

---

**Description**

Checks whether a number is near to a whole number

**Usage**

```
is_nearly_whole(x)
```

**Arguments**

x	a numeric
---	-----------

**Value**

TRUE or FALSE

**Author(s)**

Kelly Sovacool <sovacool@umich.edu>

**Examples**

```
is_nearly_whole(.Machine$double.eps^0.5)
is_nearly_whole(.Machine$double.eps^0.6)
is_nearly_whole(1)
```

---

is\_nondesc

*Check whether all elements given are sorted in non-descending order*

---

**Description**

Check whether all elements given are sorted in non-descending order

**Usage**

```
is_nondesc(...)
```

**Arguments**

... anything!

**Value**

TRUE if the elements are sorted in non-descending order, otherwise FALSE

**Author(s)**

Kelly Sovacool <sovacool@umich.edu>

**Examples**

```
is_nondesc(1, 2, 3)
is_nondesc(c(1, 2), 3)
is_nondesc(6, 4, 1)
is_nondesc("a", "b", "c")
is_nondesc(c("z", "y"))
```

---

load_deps	<i>Install &amp; load packages</i>
-----------	------------------------------------

---

**Description**

Install & load packages

**Usage**

```
load_deps(...)
```

**Arguments**

... package names to install & load

**Author(s)**

Kelly Sovacool <sovacool@umich.edu>

---

parse_tax	<i>Convert taxonomy strings into dataframe of labels based on taxonomic classification</i>
-----------	--

---

**Description**

Convert taxonomy strings into dataframe of labels based on taxonomic classification

**Usage**

```
parse_tax(dat)
```

**Arguments**

dat dataframe from mothur taxonomy file with columns OTU, Size, and Taxonomy

**Value**

a wide dataframe with taxonomic labels

**Author(s)**

Nick Lesniak, <nlesniak@umich.edu>

**Examples**

```
taxonomy_filepath <- system.file("extdata",
  "test.taxonomy",
  package = "schtools"
)
taxonomy_tbl <- read_tax(taxonomy_filepath)
head(taxonomy_tbl)
```

---

paste\_oxford\_list      *Create a prose string from a list or vector*

---

**Description**

The word 'and' is inserted before the last element and an Oxford comma is used.

**Usage**

```
paste_oxford_list(x)
```

**Arguments**

x                    a list or vector

**Value**

a string where each element in x is separated by a comma

**Author(s)**

Pat Schloss <pschloss@umich.edu>

Kelly Sovacool <sovacool@umich.edu>

**Examples**

```
paste_oxford_list(1:3)
paste_oxford_list(c("cats", "dogs", "turtles"))
```

---

read_dist	<i>Read in lower left triangular matrix from file</i>
-----------	---

---

**Description**

Read in lower left triangular matrix from file

**Usage**

```
read_dist(dist_filename)
```

**Arguments**

dist\_filename filename of lower left triangular matrix (.dist)

**Value**

distance matrix as a tibble

**Author(s)**

Nick Lesniak, <nlesniak@umich.edu>

**Examples**

```
dist_filepath <- system.file("extdata",  
  "sample.final.thetayc.0.03.lt.ave.dist",  
  package = "schtools"  
)  
dist_tbl <- read_dist(dist_filepath)  
head(dist_tbl)
```

---

read_tax	<i>Read in a taxonomy file and parse it to a wide dataframe</i>
----------	---

---

**Description**

Read in a taxonomy file and parse it to a wide dataframe

**Usage**

```
read_tax(taxonomy_filename, sep = "\t")
```

**Arguments**

taxonomy\_filename  
filename of taxonomy file

sep  
Character that separates fields of the taxonomy file. (Default: \t).



**Value**

dataframe of taxonomic labels, formatted by `parse_tax()`

**Author(s)**

Nick Lesniak, <nlesniak@umich.edu>

Kelly Sovacool, <sovacool@umich.edu>

**Examples**

```
taxonomy_filepath <- system.file("extdata",
  "test.taxonomy",
  package = "schtools"
)
taxonomy_tbl <- read_tax(taxonomy_filepath)
head(taxonomy_tbl)
```

---

schtools

*schtools: Schloss Lab Tools for Reproducible Microbiome Research*

---

**Description**

A collection of useful functions and example code created and used by the Schloss Lab for reproducible microbiome research. Perform common tasks like read files created by `mothur`, tidy up your microbiome data, and format `rmarkdown` documents for publication. See the [website](#) for more information, documentation, and examples.

---

set\_knitr\_opts

*Set knitr chunk options & inline hook*

---

**Description**

Call this function in the setup chunk of your R Markdown files.

**Usage**

```
set_knitr_opts()
```

**Author(s)**

Pat Schloss <pschloss@umich.edu>

Kelly Sovacool <sovacool@umich.edu>

# Index

`close_enough`, 2

`format_number`, 3

`inline_hook`, 4

`is_nearly_whole`, 4

`is_nondesc`, 5

`load_deps`, 6

`parse_tax`, 6

`paste_oxford_list`, 7

`read_dist`, 8

`read_tax`, 8

`schttools`, 9

`set_knitr_opts`, 9