

Package ‘latexSymb’

May 8, 2026

Type Package

Title R Functions for Readable LaTeX Mathematical Expressions

Version 1.0.0

Date 2025-06-09

Author Nicolas Escobar [aut, cre]

Maintainer Nicolas Escobar <nescoba@iu.edu>

Description Build complex 'LaTeX' mathematical expressions using intuitive 'R' functions. Replace error-prone 'LaTeX' syntax with readable, modular functions that make mathematical typesetting straightforward and maintainable.

License GPL (>= 3)

Encoding UTF-8

VignetteBuilder knitr, rmarkdown

Depends R (>= 4.2.0), purrr

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

URL <https://nicoesve.github.io/latexSymb/>

BugReports <https://github.com/nicoesve/latexSymb/issues>

RoxygenNote 7.3.2

NeedsCompilation no

Repository CRAN

Date/Publication 2025-06-17 05:00:03 UTC

Contents

+	2
at	3
common	4
il	6
lsymb	7
pths	8
Sum	8

+	<i>Arithmetic</i>
---	-------------------

Description

There are natural interpretations for doing arithmetic operations on objects of class `latexSymb`. Namely, their output is another object of that class, constructed using the `repr` of the arguments and the corresponding symbol for the operation. These functions implement the elementary ones.

Usage

```
a + b
a - b
a / b
a * b
a ^ b
under(a,b)
```

Arguments

a	An object that can be passed to <code>as.character</code>
b	An object that can be passed to <code>as.character</code>

Value

An object of class `latex_symb` whose `repr` is:

- For `+` and `-`, the concatenation of `a`'s `repr`, the corresponding arithmetic symbol, and `b`'s `repr`.
- For `*`, the concatenation of the `repr`, with a space between.
- For `/`, `a` and `b`'s `repr` inside `\frac{}{}`.
- For `^`, `a`'s `repr`, a caret and `b`'s `repr` in braces.
- For `under`, `a`'s `repr`, an underscore and `b`'s `repr` in braces

Examples

```
a <- lsymb("\\alpha")
b <- lsymb("\\beta")
a+b
a-b
a*b
a/b
a^b
under(a,b)
```

Description

These functions are used to create operations on functions in LaTeX. They represent evaluation, pullback, pushforward, derivatives and limits of functions.

Usage

```
at(f, var)
pback(f)
pfow(f)
dd(f, var)
pp(f, var)
lim(f, var, to = lsymb("\\infty"))
```

Arguments

f	latex_symb object representing a function
var	latex_symb object representing a variable
to	latex_symb object representing the limit of the function

Value

An object of class `latex_symb` whose `repr` is the LaTeX code for the operation applied to the function and the variable.

Examples

```
f <- lsymb("f")
x <- lsymb("x")
at(f, x)
pback(f)
pfow(f)
dd(f, x)
pp(f, x)
lim(f, x)
```

common

Common latex_symb objects

Description

A collection of common latex_symb objects.

Format

An RData file containing:

i latex_symb object whose repr is "i"
j latex_symb object whose repr is "j"
k latex_symb object whose repr is "k"
l latex_symb object whose repr is "l"
m latex_symb object whose repr is "m"
n latex_symb object whose repr is "n"
x latex_symb object whose repr is "x"
y latex_symb object whose repr is "y"
z latex_symb object whose repr is "z"
f latex_symb object whose repr is "f"
g latex_symb object whose repr is "g"
h latex_symb object whose repr is "h"
al latex_symb object whose repr is "\\alpha"
be latex_symb object whose repr is "\\beta"
ga latex_symb object whose repr is "\\gamma"
de latex_symb object whose repr is "\\delta"
ep latex_symb object whose repr is "\\epsilon"
ze latex_symb object whose repr is "\\zeta"
et latex_symb object whose repr is "\\eta"
th latex_symb object whose repr is "\\theta"
io latex_symb object whose repr is "\\iota"
ka latex_symb object whose repr is "\\kappa"
la latex_symb object whose repr is "\\lambda"
mu latex_symb object whose repr is "\\mu"
nu latex_symb object whose repr is "\\nu"
xi latex_symb object whose repr is "\\xi"
om latex_symb object whose repr is "\\omicron"
pi.l latex_symb object whose repr is "\\pi"

rh latex_symb object whose repr is "\rho"
si latex_symb object whose repr is "\sigma"
ta latex_symb object whose repr is "\tau"
up latex_symb object whose repr is "\upsilon"
ph latex_symb object whose repr is "\phi"
ch latex_symb object whose repr is "\chi"
ps latex_symb object whose repr is "\psi"
om latex_symb object whose repr is "\omega"
des latex_symb object whose repr is ":"
eq latex_symb object whose repr is "="
neq latex_symb object whose repr is "\neq"
lt latex_symb object whose repr is "<"
gt latex_symb object whose repr is ">"
leq latex_symb object whose repr is "\leq"
geq latex_symb object whose repr is "\geq"
bgs latex_symb object whose repr is "\in"
mapsto latex_symb object whose repr is "\mapsto"
to latex_symb object whose repr is "\rightarrow"
ldots latex_symb object whose repr is "\ldots"
Reals latex_symb object whose repr is "\mathbb{R}"
Nats latex_symb object whose repr is "\mathbb{N}"
Ints latex_symb object whose repr is "\mathbb{Z}"
Rats latex_symb object whose repr is "\mathbb{Q}"
Comps latex_symb object whose repr is "\mathbb{C}"
indic latex_symb object whose repr is "\mathbb{1}"
infty latex_symb object whose repr is "\infty"
comma latex_symb object whose repr is ","
endl latex_symb object whose repr is "\\\\"
thus latex_symb object whose repr is "\\\rightarrow"
minus latex_symb object whose repr is "-"
plus latex_symb object whose repr is "+"
times latex_symb object whose repr is "\times"
quad latex_symb object whose repr is "\quad"
ruler latex_symb object whose repr is "&"

 il

 LaTeX *Environments*

Description

Equations and symbols in LaTeX can be either inline or on their own. `il` wraps expressions for the former, `lenv` for the latter.

Usage

```
il(x)
lenv(name, rows)
```

Arguments

<code>x</code>	An object of class <code>latex_symb</code>
<code>name</code>	The name of the LaTeX environment. For instance, <code>align</code> or <code>gather</code>
<code>.</code>	
<code>rows</code>	A list of objects that can be passed to <code>as.character</code> .

Value

- For `il`, `x`'s repr surrounded by dollar signs.
- For `lenv`, a multiline string whose lines are: 1. A `\begin` statement for `name`; 2. The character representation of each row; 3. An `\end` statement for `name`

Examples

```
a1 <- lsymb("\\alpha")
be <- lsymb("\\beta")
il(a1)
lenv("align",
     c(
       lsymb(a1^2 - be^2, "&=", 0, "\\\\"),
       lsymb(pths(a1 - be)*pths(a1 + be), "&=", 0)
     )
)
```

`lsymb`*Create, print and turn to string objects of class latex_symb*

Description

The class `latex_symb` is simply a wrapper for a string with LaTeX code. `lsymb` creates the wrapper, `as.character` and `print` extract the string.

Usage

```
lsymb(...)  
## S3 method for class 'latex_symb'  
print(x, ...)  
## S3 method for class 'latex_symb'  
as.character(x, ...)
```

Arguments

<code>...</code>	Objects that can be passed to <code>as.character</code> .
<code>x</code>	An object of class <code>latex_symb</code>

Value

- `lsymb` returns an object of class `latex_symb`. It is an S3 class, whose objects are lists with a single component called `repr`. `repr` is the LaTeX code for the object, which is obtained by pasting the character representations of all the arguments.
- `print.lsymb` passes the `repr` of its first argument, plus additional arguments, to `print`. The returned value is whatever `print` returns.
- `as.character.lsymb` passes the `repr` of its first argument, plus additional arguments, to `as.character`. The returned value is whatever `as.character` returns.

Examples

```
a1 <- lsymb("\\alpha")  
print(a1)  
as.character(a1)
```

pths

LaTeX *Enclosings*

Description

It is cumbersome to have to write `left` and `right` every time a grouping is used in LaTeX. These functions take care of that.

Usage

```
pths(x)
br(x)
sqbr(x)
ang(x)
```

Arguments

`x` An object that can be passed to `as.character`.

Value

An object of class `latex_symb` whose `repr` is `x`'s `repr` enclosed by the corresponding symbols.

Examples

```
a1 <- lsymb("\\alpha")
pths(a1)
br(a1)
sqbr(a1)
ang(a1)
```

Sum

Cumulative Operators

Description

These functions are used to create cumulative operators in LaTeX. They take care of the `sum`, `prod` and `int` functions.

Usage

```
Sum(f, from = lsymb(""), to = lsymb(""))
Prod(f, from = lsymb(""), to = lsymb(""))
Int(f, meas = lsymb("dx"), from = lsymb(""), to = lsymb(""))
```

Arguments

<code>f</code>	An expression to be summed, multiplied or integrated.
<code>from</code>	The lower limit of the sum, product or integral.
<code>to</code>	The upper limit of the sum, product or integral.
<code>meas</code>	The measure of the integral.

Value

An object of class `latex_symb` whose `repr` is the LaTeX code for the cumulative operator concatenated with the limits and the expression.

Examples

```
i <- lsymb("i")
n <- lsymb("n")
x <- lsymb("x")
f <- function(x) lsymb("f") * pths(x)
Sum(i, from = 1, to = n)
Prod(i, from = 1, to = n)
Int(f(x), from = 0, to = 1)
```

Index

* (+), 2

+, 2

- (+), 2

/ (+), 2

^ (+), 2

ang (pths), 8

as.character.latex_symb (lsymb), 7

at, 3

br (pths), 8

common, 4

dd (at), 3

il, 6

Int (Sum), 8

lenv (il), 6

lim (at), 3

lsymb, 7

pback (at), 3

pfow (at), 3

pp (at), 3

print.latex_symb (lsymb), 7

Prod (Sum), 8

pths, 8

sqbr (pths), 8

Sum, 8

under (+), 2