Package ‘tidyformula’

February 17, 2023

Title Build Formulas Using Tidy Selection Helpers

Version 0.1.0

Description Provides the function 'tidyformula()', which translates formulas containing 'tidyselect'-style selection helpers. It expands these helpers by evaluating 'dplyr::select()' with the relevant selection helper and a supplied data frame. The package contains methods for traversing abstract syntax trees from Wickham, Hadley (2019) <doi:10.1201/9781351201315>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.2

Imports dplyr, purrr, rlang, stats

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Damian Pavlyshyn [aut, cre] (<`https://orcid.org/0000-0001-6929-878X`>)

Maintainer Damian Pavlyshyn <`damian.pavlyshyn@gmail.com`>

Repository CRAN

Date/Publication 2023-02-17 10:00:02 UTC

R topics documented:

`tidyformula` *Build a formula using* `tidyselect`*-style selection helpers*

Description

`tidyformula()` translates formulas containing `tidyselect`-style selection helpers, expanding these helpers by evaluating `dplyr::select()` with the relevant selection helper and a supplied data frame.

When the selection helper appears as the first argument of a function, that function is distributed across the sum of the selected variables.

Usage

```
tidyformula(
  formula,
  data,
  select_helpers = .select_helpers,
  nodistribute = c("+", "-", "*", "^"),
  env = rlang::caller_env()
)
```

Arguments

`formula`	An object of class `formula`. Can contain selection helpers to be expanded.
`data`	A data frame whose column names should be used for selection
`select_helpers`	A character vector. The names of selection helpers to be matched and substituted.
`nodistribute`	A character vector. Functions with these names are not distributed over selection helpers.
`env`	The environment to associate with the result.

Value

An object of class `formula`, which is a translation of the argument `formula` in which the selection helpers are replaced with the corresponding variables of `data`.

Examples

```
df <- data.frame(
  x1 = rnorm(5),
  x2 = rnorm(5),
  x3 = rnorm(5),
  y  = rnorm(5)
)

tidyformula(y ~ num_range("x", 1:2) + z, data = df)
```

```
#> y ~ x1 + x2 + z
#> <environment: 0x000001e0d7d53910>

tidyformula(y ~ poly(starts_with("x"), 3), data = df)
#> y ~ poly(x1, 3) + poly(x2, 3) + poly(x3, 3)
#> <environment: 0x000001e0d7d53910>

tidyformula( ~ everything() * contains("x"), data = df)
#> ~(x1 + x2 + x3 + y) * (x1 + x2 + x3)
#> <environment: 0x000001e0d7d53910>
```

Interaction operators are typically not distributed, but this behaviour can be changed.

```
tidyformula(y ~ starts_with("x")^2, data = df)
#> y ~ (x1 + x2 + x3)^2
#> <environment: 0x000001e0d7d53910>

tidyformula(y ~ starts_with("x")^2, data = df, nodistribute = c("+", "-"))
#> y ~ x1^2 + x2^2 + x3^2
#> <environment: 0x000001e0d7d53910>
```

See Also

`dplyr::select()`, `tidyselect::language` for documentation of selection helpers.

Index