

# Package ‘tidycat’

June 5, 2020

**Type** Package

**Title** Expand Tidy Output for Categorical Parameter Estimates

**Version** 0.1.1

**Maintainer** Guy J. Abel <g.j.abel@gmail.com>

**URL** <https://github.com/guyabel/tidycat>

**BugReports** <https://github.com/guyabel/tidycat/issues>

**Description** Create additional rows and columns on broom::tidy() output to allow for easier control on categorical parameter estimates.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.0

**Imports** magrittr, utils, tidyr, tibble, dplyr, stringr, stats, forcats

**Suggests** broom, ggplot2, ggforce, knitr, rmarkdown

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Guy J. Abel [aut, cre] (<<https://orcid.org/0000-0002-4893-5687>>)

**Repository** CRAN

**Date/Publication** 2020-06-05 15:40:03 UTC

## R topics documented:

factor_regex . . . . .	2
tidy_categorical . . . . .	2

<b>Index</b>	<b>6</b>
--------------	----------

---

factor_regex	<i>Generate Regular Expression to Detect Factors</i>
--------------	--

---

**Description**

Primarily developed for use within [tidycat::tidy\\_categorical\(\)](#)

**Usage**

```
factor_regex(m, at_start = TRUE)
```

**Arguments**

m	A model object, created using a function such as <a href="#">stats::lm()</a>
at_start	Logical indicating whether or not to include ^ in the regular expression to begin search at start of string

**Value**

A character string for use as a regular expression.

**Author(s)**

Guy J. Abel

**Examples**

```
m0 <- lm(formula = mpg ~ disp + as.factor(am)*as.factor(vs), data = mtcars)
factor_regex(m = m0)
```

---

tidy_categorical	<i>Expand broom::tidy() Outputs for Categorical Parameter Estimates</i>
------------------	---

---

**Description**

Create additional columns in a tidy model output (such as [broom::tidy.lm\(\)](#)) to allow for easier control when plotting categorical parameter estimates.

**Usage**

```
tidy_categorical(
  d = NULL,
  m = NULL,
  include_reference = TRUE,
  reference_label = "Baseline Category",
  non_reference_label = paste0("Non-", reference_label),
  exponentiate = FALSE,
  n_level = FALSE
)
```

**Arguments**

d	A data frame <code>tibble::tibble()</code> output from <code>broom::tidy.lm()</code> ; with one row for each term in the regression, including column term
m	A model object, created using a function such as <code>lm()</code>
include_reference	Logical indicating to include additional rows in output for reference categories, obtained from <code>dummy.coef()</code> . Defaults to TRUE
reference_label	Character string. When used will create an additional column in output with labels to indicate if terms correspond to reference categories.
non_reference_label	Character string. When <code>reference_label</code> is used will be in output to indicate if terms not corresponding to reference categories.
exponentiate	Logical indicating whether or not the results in <code>broom::tidy.lm()</code> are exponentiated. Defaults to FALSE.
n_level	Logical indicating whether or not to include a column <code>n_level</code> for the number of observations per category. Defaults to FALSE.

**Value**

Expanded `tibble::tibble()` from the version passed to `d` including additional columns:

variable	The name of the variable that the regression term belongs to.
level	The level of the categorical variable that the regression term belongs to. Will be an the term name for numeric variables.
effect	The type of term (main or interaction)
reference	The type of term (reference or non-reference) with label passed from <code>reference_label</code> . If <code>reference_label</code> is set NULL will not be created.
n_level	The the number of observations per category. If <code>n_level</code> is set NULL (default) will not be created.

In addition, extra rows will be added, if `include_reference` is set to FALSE for the reference categories, obtained from `dummy.coef()`

**Author(s)**

Guy J. Abel

**See Also**[broom::tidy.lm\(\)](#)**Examples**

```

# strip ordering in factors (currently ordered factor not supported)
library(dplyr)
library(broom)

m0 <- esoph %>%
  mutate_if(is.factor, ~factor(., ordered = FALSE)) %>%
  glm(cbind(ncases, ncontrols) ~ agegp + tobgp * alcgp, data = .,
      family = binomial())
# tidy
tidy(m0)

# add further columns to tidy output to help manage categorical variables
m0 %>%
  tidy() %>%
  tidy_categorical(m = m0, include_reference = FALSE)

# include reference categories and column to indicate the additional terms
m0 %>%
  tidy() %>%
  tidy_categorical(m = m0)

# coefficient plots
d0 <- m0 %>%
  tidy(conf.int = TRUE) %>%
  tidy_categorical(m = m0) %>%
  # drop the intercept term
  slice(-1)
d0

# typical coefficient plot
library(ggplot2)
library(tidyr)
ggplot(data = d0 %>% drop_na(),
       mapping = aes(x = term, y = estimate,
                    ymin = conf.low, ymax = conf.high)) +
  coord_flip() +
  geom_hline(yintercept = 0, linetype = "dashed") +
  geom_pointrange()

# enhanced coefficient plot using additional columns from tidy_categorical and ggforce::facet_row()
library(ggforce)
ggplot(data = d0,
       mapping = aes(x = level, colour = reference,

```

```
      y = estimate, ymin = conf.low, ymax = conf.high)) +  
facet_row(facets = vars(variable), scales = "free_x", space = "free") +  
geom_hline(yintercept = 0, linetype = "dashed") +  
geom_pointrange() +  
theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

# Index

`broom::tidy.lm()`, 2–4

`dummy.coef()`, 3

`factor_regex`, 2

`lm()`, 3

`stats::lm()`, 2

`tibble::tibble()`, 3

`tidy_categorical`, 2

`tidycat::tidy_categorical()`, 2