Package ‘usemodels’

Title   Boilerplate Code for 'Tidymodels' Analyses
Version  0.2.0
Description Code snippets to fit models using the tidymodels framework can be easily created for a given data set.
License  MIT + file LICENSE
URL     https://usemodels.tidymodels.org/,
        https://github.com/tidymodels/usemodels
BugReports https://github.com/tidymodels/usemodels/issues
Imports  cli, clipr, dplyr, purrr, recipes (>= 0.1.15), rlang, tidyr,
        tune (>= 0.1.2)
Suggests covr, modeldata, spelling, testthat
Config/Needs/website tidyverse/tidytemplate
Config/testthat/edition 3
Encoding  UTF-8
Language  en-US
RoxygenNote  7.1.2
NeedsCompilation  no
Author     Max Kuhn [aut, cre] (https://orcid.org/0000-0003-2402-136X),
           RStudio [cph]
Maintainer Max Kuhn <max@rstudio.com>
Repository  CRAN
Date/Publication  2022-02-18 22:10:02 UTC

R topics documented:

  use_glmnet .................................................. 2

Index 5
Description

These functions make suggestions for code when using a few common models. They print out code to the console that could be considered minimal syntax for their respective techniques. Each creates a prototype recipe and workflow object that can be edited or updated as the data require.

Usage

use_glmnet(
  formula,  # formula to be used in model
  data,    # data frame to use
  prefix = "glmnet",  # prefix for model
  verbose = FALSE,    # print verbose output
  tune = TRUE,        # use tuning parameters
  colors = TRUE,      # use color codes
  clipboard = FALSE   # copy code to clipboard
)

use_xgboost(
  formula,  # formula to be used in model
  data,    # data frame to use
  prefix = "xgboost",  # prefix for model
  verbose = FALSE,    # print verbose output
  tune = TRUE,        # use tuning parameters
  colors = TRUE,      # use color codes
  clipboard = FALSE   # copy code to clipboard
)

use_kknn(
  formula,  # formula to be used in model
  data,    # data frame to use
  prefix = "kknn",  # prefix for model
  verbose = FALSE,    # print verbose output
  tune = TRUE,        # use tuning parameters
  colors = TRUE,      # use color codes
  clipboard = FALSE   # copy code to clipboard
)

use_ranger(
  formula,  # formula to be used in model
  data,    # data frame to use
  prefix = "ranger",  # prefix for model
  verbose = FALSE,    # print verbose output
  tune = TRUE,        # use tuning parameters
  colors = TRUE,      # use color codes
  clipboard = FALSE   # copy code to clipboard
)
use_Glmnet

colors = TRUE,
clipboard = FALSE
)

use_earth(
  formula,
data,
  prefix = "earth",
  verbose = FALSE,
  tune = TRUE,
  colors = TRUE,
  clipboard = FALSE
)

use_cubist(
  formula,
data,
  prefix = "cubist",
  verbose = FALSE,
  tune = TRUE,
  colors = TRUE,
  clipboard = FALSE
)

use_kernlab_svm_rbf(
  formula,
data,
  prefix = "kernlab",
  verbose = FALSE,
  tune = TRUE,
  colors = TRUE,
  clipboard = FALSE
)

use_kernlab_svm_poly(
  formula,
data,
  prefix = "kernlab",
  verbose = FALSE,
  tune = TRUE,
  colors = TRUE,
  clipboard = FALSE
)

use_C5.0(
  formula,
data,
  prefix = "C5.0",
  verbose = FALSE,
  tune = TRUE,
  colors = TRUE,
  clipboard = FALSE
)
Arguments

- **formula**: A simple model formula with no in-line functions. This will be used to template the recipe object as well as determining which outcome and predictor columns will be used.
- **data**: A data frame with the columns used in the analysis.
- **prefix**: A single character string to use as a prefix for the resulting objects.
- **verbose**: A single logical that determined whether comments are added to the printed code explaining why certain lines are used.
- **tune**: A single logical that controls if code for model tuning should be printed.
- **colors**: A single logical for coloring warnings and code snippets that require the users attention (ignored when colors = FALSE)
- **clipboard**: A single logical for whether the code output should be sent to the clip board or printed in the console.

Details

Based on the columns in data, certain recipe steps printed. For example, if a model requires that qualitative predictors be converted to numeric (say, using dummy variables) then an additional `step_dummy()` is added. Otherwise that recipe step is not included in the output.

The syntax is opinionated and should not be considered the exact answer for every data analysis. It has reasonable defaults.

Value

Invisible NULL but code is printed to the console.

Examples

```r
library(modeldata)
data(ad_data)
use_glmnet(Class ~ ., data = ad_data)

data(Sacramento)
use_glmnet(price ~ ., data = Sacramento, verbose = TRUE, prefix = "sac_homes")
```
Index

call_C5.0 (use_glmnet), 2
use_cubist (use_glmnet), 2
use_earth (use_glmnet), 2
use_glmnet, 2
use_kernlab_svm_poly (use_glmnet), 2
use_kernlab_svm_rbf (use_glmnet), 2
use_kknn (use_glmnet), 2
use_ranger (use_glmnet), 2
use_xgboost (use_glmnet), 2