Package ‘ModelMatrixModel’

October 12, 2022

Type Package
Title Create Model Matrix and Save the Transforming Parameters
Version 0.1.0
Author Xinyong Tian
Maintainer Xinyong Tian <xinyongtian@gmail.com>
Description The model.matrix() function in R is convenient for transforming training dataset for modeling. But it does not save any parameter used in transformation, so it is hard to apply the same transformation to test dataset or new dataset. This package is created to solve the problem.
License GPL-3
Encoding UTF-8
LazyData false
RoxygenNote 7.1.1
Imports Matrix
Suggests rmarkdown, knitr
VignetteBuilder knitr
NeedsCompilation no
Repository CRAN
Date/Publication 2021-08-17 08:00:10 UTC

R topics documented:

ModelMatrixModel ........................................ 2
predict.ModelMatrixModel ............................... 3

Index 4
ModelMatrixModel

Description

This function transforms a data.frame to matrix (or sparse matrix) based on a r formula. The mean different from model.matrix() function is that it outputs a class stored with the transformed matrix, as well as the transforming parameters which can be applied to new data.

Usage

ModelMatrixModel(
  rformula,
  data,
  sparse = TRUE,
  center = FALSE,
  scale = FALSE,
  remove_1st_dummy = FALSE,
  verbose = FALSE
)

Arguments

rformula  a formula, e.g. formula("~ 1+x1+x2"),"~ 1+x1+x2",or ~ 1+x1+x2 . Note the interpreting of the formula might be different slightly from model.matrix function. In model.matrix(), intercept column will be included in output matrix with or without "1" in the formula. But in ModelMatrixModel(), intercept column will be included in output matrix only when "1" is present. Moreover "0" or "." in the formula will be ignored.

data      a data.frame.
sparse    boolean, if TRUE return a sparse matrix, i.e. a "dgCMatrix" class.
center    boolean, if center the output.
scale     boolean, if scale the output.
remove_1st_dummy boolean, if remove the first dummy variable in one hot key transformation.
verbose   boolean, if print out progress.

Details

see vignettes.

Value

A ModelMatrixModel class, which includes the transformed matrix and the transforming parameters.
Examples

```r
library(ModelMatrixModel)
traindf = data.frame(x1 = sample(LETTERS[1:5], replace = TRUE, 20),
                     x2 = rnorm(20, 100, 5),
                     y = rnorm(20, 10, 2))
mm = ModelMatrixModel(~x1+x2, traindf, remove_1st_dummy = FALSE)
data.frame(as.matrix(head(mm$x, 2)))
```

predict.ModelMatrixModel

**predict() function**

**Description**

This function transforms new data based on transforming parameters from a ModelMatrixModel object.

**Usage**

```r
## S3 method for class 'ModelMatrixModel'
predict(object, data, handleInvalid = "keep", verbose = FALSE, ...)
```

**Arguments**

- `object`: a ModelMatrixModel object.
- `data`: a data.frame.
- `handleInvalid`: a string, 'keep' or 'error'. In dummy variable transformation, if categorical variable has a factor level that is unseen before, 'keep' will keep the record, output dummy variables will be all zero.
- `verbose`: boolean, if print out progress.
- `...`: other parameters.

**Value**

A ModelMatrixModel class, which includes the transformed matrix and the necessary transforming parameters copied from input object.

**Examples**

```r
library(ModelMatrixModel)
traindf = data.frame(x1 = sample(LETTERS[1:5], replace = TRUE, 20),
                     x2 = rnorm(20, 100, 5),
                     y = rnorm(20, 10, 2))
nmm = ModelMatrixModel(~x1+x2, traindf, remove_1st_dummy = FALSE)
nmm_pred = predict(nmm, newdf)
data.frame(as.matrix(head(nmm_pred$x, 2)))
```
Index

ModelMatrixModel, 2
predict.ModelMatrixModel, 3