Package ‘FSDAM’

November 20, 2020

LazyLoad yes
LazyData yes
Version 2020.11-18
Title Forward Stepwise Deep Autoencoder-Based Monotone NLDR
Maintainer Youyi Fong <youyifong@gmail.com>
Depends R (>= 3.5.0)
Suggests R.rsp, RUnit
Imports kyotil, reticulate (>= 1.10)
VignetteBuilder R.rsp
Description FS-DAM performs feature extraction through latent variables identification. Implementation is based on autoencoders with monotonicity and orthogonality constraints.
License GPL (>= 2)
NeedsCompilation no
Author Youyi Fong [cre],
Jun Xu [aut]
Repository CRAN
Date/Publication 2020-11-20 09:50:08 UTC

R topics documented:

cc.505 ................................................................. 2
fsdam ................................................................. 3
hvt505tier1 ......................................................... 4

Index 6
Select Biomarkers from the HVN 505 Correlates Analysis

Description
See reference.

Usage
data("cc.505")

Format
A data frame with 189 observations on the following 27 variables.

ptid  a character vector
trt   a numeric vector
case  a numeric vector
control a numeric vector
perprot a numeric vector
last_uninfec_immun_vst a numeric vector
racefull a numeric vector
racefulltxt a character vector
bmi   a numeric vector
bmicat a numeric vector
bmicattxt a character vector
earliest_pos_vst a numeric vector
level  a character vector
matchlevel a character vector
samplingfraction a numeric vector
vst9subcohort a numeric vector
HIVwk28preunbl a numeric vector
age   a numeric vector
racecc a character vector
bhvrisk a numeric vector
BMI   a numeric vector
stratums_vaccs a numeric vector
stratuminds a numeric vector
cd4.env.poly a numeric vector
cd8.env.poly a numeric vector
mfounders a numeric vector
wei   a numeric vector
References

fsdam  

**FS-DAM NLDR**

Description
Forward stepwise deep autoencoder-based monotone nonlinear dimension reduction.

Usage
```r
fsdam(dat, opt_numCode = ncol(dat), opt_seed = 1, opt_model = "n", opt_gpu = 0,
      opt_k = 100, opt_nEpochs = 10000,
      opt_constr = c("newpenalization", "constrained", "none"),
      opt_tuneParam = 10, opt_penfun = "mean", opt_ortho = 1, opt_earlystop = "no",
      verbose = FALSE)
```

# S3 method for class 'fsdam'
plot(x, which=c("mse", "history", "decoder.func", "scatterplot"),
     k=NULL, dim.1=NULL, dim.2=NULL, col.predict=2, ...)

Arguments
- **dat**: data frame.
- **opt_numCode**: number of components to extract
- **opt_seed**: seed for torch
- **opt_model**: n for newpenalization
- **opt_gpu**: zero-based index of gpu to be used among all gpus. If negative, then no gpu is used
- **opt_k**: number of nodes in the coding/decoding layers
- **opt_nEpochs**: number of epochs for training
- **opt_constr**: constraint string
- **opt_tuneParam**: tuning parameter for monotonicity penalty
- **opt_penfun**: penalize sum or mean
- **opt_ortho**: tuning parameter for orthogonality penalty
- **opt_earlystop**: whether to stop early
- **verbose**: whether to print progress
- **x**: fsdam object
- **which**:
k the component to plot
dim.1 index of the first variable
dim.2 index of the second variable
col.predict color of the predicted curve when which = scatterplot
... plotting arguments

Details
If the torch python package is not available, this function will stop.

To make sure the right python installation is used, run reticulate::use_python("/app/easybuild/software/Python/3.7.4-foss-2016b/bin/python") in R before running this function for the first time.

References

Examples

## Not run:
fit=fsdam(hvtn505tier1[1:100,-1], opt_numCode=2, verbose=TRUE)
fit
plot(fit,which="mse")
plot(fit,which="history")

## End(Not run)
**Format**

A data frame with 150 observations on the following 9 variables.

- **ptid** a character vector
- **CD8\_ANYVRCENV\_PolyfunctionalityScore\_score** a numeric vector
- **IgGw28\_env\_mdw** a numeric vector
- **IgGw28\_V1V2\_mdw** a numeric vector
- **IgGw28\_gp41\_mdw** a numeric vector
- **ADCP1** a numeric vector
- **R2aConSgp140CFI** a numeric vector
- **IgAw28\_env\_mdw** a numeric vector
- **IgG3w28\_env\_mdw** a numeric vector

**References**


Index

cc.505, 2
FSDAM (fsdam), 3
fsdam, 3
hvtn505tier1, 4
plot.fsdam (fsdam), 3