Package 'causalimages'

February 5, 2023

Title causalimages: R Package for Causal Inference with Earth Observation, Bio-medical, and Social Science Images

Version 2.0

Authors 'Connor Jerzak <cjerzak@g.harvard.edu> [aut, cre], Fredrik Johansson <fredrik.johansson@chalmers.se> [aut], Adel Daoud <adel.daoud@liu.se> [aut]'

Description
R Package for Causal Inference with Earth Observation, Bio-medical, and Social Science Images.

Depends R (>= 3.3.3)

Encoding UTF-8

LazyData true

Maintainer 'Connor Jerzak' <connor.jerzak@gmail.com>

Imports tensorflow

RoxygenNote 7.2.1

R topics documented:

`AnalyzeImageHeterogeneity` *AnalyzeImageHeterogeneity*

Description

Implements ...

Usage

```
AnalyzeImageHeterogeneity(
  obsW,
  obsY,
  X = NULL,
  imageKeys = NULL,
  transportabilityMat = NULL,
  lat = NULL,
  long = NULL,
  externalFigureKey = "",
  acquireImageRepFxn,
  acquireImageFxn_full = acquireImageRepFxn,
  TYPE = "variational_minimal",
  SimMode = F,
  nDepth_conv = 1,
  nDepth_dense = 1,
  plotResults = F,
  figuresPath = "./",
  kClust_est = 2,
  maxPoolSize = 2L,
  strides = 1L,
  nMonte_predictive = 10L,
  y_density = "normal",
  orthogonalize = F,
  compile = F,
  nMonte_variational = 5L,
  kernelWidth,
  nSGD = 400,
  nDenseWidth = 64L,
  nFilters = 7L
)
```

Arguments

`DAG`	'DAG'.

Value

A list consiting of

- Items.

References

- References here

Examples

```
#set seed
set.seed(1)

#Geneate data
x <- rnorm(100)
```

`SimulateImageSystem` *SimulateImageSystem*

Description

This function (1) generates simulated causal structures using images.

Usage

```
SimulateImageSystem(dag = NULL, ...)
```

Arguments

`dag`	*(character string)* An input DAG specifying causal structure. This input should be of the form `'i->t,i->y,t->y,....'` Currently, only one node in a DAG can be an image (this should be labeled "`i`"). The non-image nodes can have arbitrary string labels. The image can be a confounder, effect moderator, effect mediator. If the image is to be used as a moderator, use the notation, `t-i>y`.
`...`	*(optional)* In estimation mode, users input the data matrices associated with the non-image nodes of `DAG` and image node `i`. For example, if `x` is a DAG node, users must, in estimation mode, supply data to `x` in a form that can be coerced to a tensor.
`treatment`	*(character string, optional)* In estimation mode, users specify the treatment variable here. If `treatment` is specified, users must provide other data inputs to the DAG (see `...`).
`image_pool`	*(character string, optional)* The path to where analysis specific images are located. This can be specified both in simulation and estimation mode. If not specified, the simulation uses a pool of Landsat images from Nigeria.
`analysis_level`	*(character string, default is* `'scene'`*)* Defines the unit of analysis used in the simulation framework. This is ignored in estimation mode, where the unit of analysis is inferred from the data dimensions.
`control`	*(list)* A list containing control parameters in the data generating process.

Value

A list:

- In *simulation mode*, the function returns a list with as many elements as unique nodes in DAG. Each element represents the simulated data.
- In *estimation mode*,the function returns an estimated treatment effect with 95% confidence intervals.

References

- CITES

Examples

```
#set seed
set.seed(1)

# Simulation mode
#simulatedData <- causalimage('r->i, i->t, t->y, r->y')
#print(names(simulatedData))

# Estimation mode
#estimatedResults <- causalimage('r->i, i->t, t->y, r->y', y=y, r=r, y=y', treatment='t')
#print( estimatedResults )
```

Index