Package ‘eff2’

September 30, 2021

Type Package

Title Efficient Least Squares for Total Causal Effects

Version 1.0.1

Author Richard Guo [aut, cre] (<https://orcid.org/0000-0002-2081-7398>)

Maintainer Richard Guo <ricguo@uw.edu>

Description Estimate a total causal effect from observational data under linearity and causal sufficiency. The observational data is supposed to be generated from a linear structural equation model (SEM) with independent and additive noise. The underlying causal DAG associated the SEM is required to be known up to a maximally oriented partially directed graph (MPDAG), which is a general class of graphs consisting of both directed and undirected edges, including CPDAGs (i.e., essential graphs) and DAGs. Such graphs are usually obtained with structure learning algorithms with added background knowledge. The program is able to estimate every identified effect, including single and multiple treatment variables. Moreover, the resulting estimate has the minimal asymptotic covariance (and hence shortest confidence intervals) among all estimators that are based on the sample covariance.

License MIT + file LICENSE

URL https://github.com/richardkwo/eff2

BugReports https://github.com/richardkwo/eff2/issues

Depends R (>= 3.5.0)

Imports pcalg (>= 2.6), RBGL, igraph

Suggests knitr, rmarkdown, testthat, qgraph

VignetteBuilder knitr

RoxygenNote 7.1.2

Encoding UTF-8

LazyData true

NeedsCompilation no

Repository CRAN

Date/Publication 2021-09-30 20:40:02 UTC

R topics documented:

eff2 2
estimateEffect 2
ex1 4
isIdentified 4

Index **6**

`eff2` *eff2: efficient least squares for estimating total causal effects*

Description

Estimate a total causal effect from observational data under linearity and causal sufficiency. The observational data is supposed to be generated from a linear structural equation model (SEM) with independent and additive noise. The underlying causal DAG associated the SEM is required to be known up to a maximally oriented partially directed graph (MPDAG), which is a general class of graphs consisting of both directed and undirected edges, including CPDAGs (i.e., essential graphs) and DAGs. Such graphs are usually obtained with structure learning algorithms with added background knowledge. The program is able to estimate every identified effect, including single and multiple treatment variables. Moreover, the resulting estimate has the minimal asymptotic covariance (and hence shortest confidence intervals) among all estimators that are based on the sample covariance.

Details

Use `estimateEffect` to estimate a total effect.

Use `isIdentified` to determine if a total effect can be identified.

`estimateEffect` *Estimate the total causal effect*

Description

Estimate the total causal effect of x on y with iterated least squares. The resulting estimate has the minimal asymptotic covariance among all least squares estimators.

Usage

```
estimateEffect(data, x, y, amat, bootstrap = FALSE)
```

Arguments

data	a data frame consisting of iid observational data
x	(integer) positions of treatment variables in the adjacency matrix; can be a singleton (single treatment) or a vector (multiple treatments)
y	(integer) position of the outcome variable in the adjacency matrix
amat	adjacency matrix representing a DAG, CPDAG or MPDAG
bootstrap	If TRUE, will estimate the standard error covariance with bootstrap (default: FALSE)

Details

Adjacency matrix amat represents the graphical information of the underlying causal DAG (directed acyclic graph). The causal DAG should be contained by the graph represented by amat, which can be a DAG, CPDAG (essential graph), or more generally, an MPDAG (maximally oriented partially directed acyclic graph).

Matrix amat is coded with the convention of amatType:

- amat[i,j]=0 and amat[j,i]=1 means i->j
- amat[i,j]=1 and amat[j,i]=0 means i<-j
- amat[i,j]=1 and amat[j,i]=1 means i--j
- amat[i,j]=0 and amat[j,i]=0 means i j

amat can be learned from observational data with a structure learning algorithm; see pc, ges and LINGAM. Additional background knowledge can also be incorporated with addBgKnowledge.

Value

A vector of the same length as x. If bootstrap=TRUE, return a list of (effect,se.cov).

See Also

isIdentified is called for determining if an effect can be identified. See also adjustment, ida, and jointIda for other estimators.

Examples

```
data("ex1")
result <- estimateEffect(ex1$data, c(5,3), 7, ex1$amat.cpdag, bootstrap=TRUE)
print(result$effect)
print(result$effect - 1.96 * sqrt(diag(result$se.cov)))
print(result$effect + 1.96 * sqrt(diag(result$se.cov)))
# compare with truth
print(ex1$true.effects)

## Not run:
# throws an error because the effect is not identified
estimateEffect(ex1$data, 3, 7, ex1$amat.cpdag)

## End(Not run)
```

`ex1` *An example of 10 variables simulated from a linear SEM*

Description

An example of 10 variables simulated from a linear SEM

Usage

```
ex1
```

Format

A list containing:

x treatment variables

y outcome variable

true.effects the true total effect of x on y

B the coefficient matrix of the SEM

amat.dag the adjacency matrix of the causal DAG

amat.cpdag the adjacency matrix of the CPDAG of the causal DAG, representing the Markov equivalence class of the DAG.

data 500 iid samples generated under student-t errors

`isIdentified` *Check if a total causal effect is identified*

Description

The total causal effect from x to y is identified if and only if there is no possibly causal path from x to y that starts with an undirected edge.

Usage

```
isIdentified(amat, x, y, type = "pdag")
```

Arguments

`amat`	adjacency matrix. See `estimateEffect` for its coding.
`x`	(integer) positions of treatment variables in the adjacency matrix
`y`	(integer) positions of outcome variables in the adjacency matrix
`type`	string specifying the type of graph of `amat`. It can be DAG (`type='dag'`) or MPDAG/CPDAG (`type='pdag'`).

Value

TRUE if identified, FALSE if not.

References

Emilija Perkovic. Identifying causal effects in maximally oriented partially directed acyclic graphs. In *Uncertainty in Artificial Intelligence (UAI)*, 2020.

See Also

estimateEffect

Examples

```
data("ex1")
# identified
isIdentified(ex1$amat.cpdag, c(3, 5), 7)
# not identified
isIdentified(ex1$amat.cpdag, 3, 7)
isIdentified(ex1$amat.cpdag, c(3, 5), 10)
```

Index

* **datasets**
    ex1, 4

addBgKnowledge, *3*
adjustment, *3*
amatType, *3*

eff2, 2
estimateEffect, *2*, 2, *4, 5*
ex1, 4

ges, *3*

ida, *3*
isIdentified, *2, 3*, 4

jointIda, *3*

LINGAM, *3*

pc, *3*