Description

clusteredinterference implements the estimators from Barkley et al. (2017) *Causal Inference from Observational Studies with Clustered Interference* <arXiv:1711.04834>. These are inverse probability of treatment-weighted estimators for estimating the causal effects on outcomes of population-level treatment policies.

Details

These estimators are implemented in one function, policyFX. A healthy amount of documentation can be found in the vignette included in this package: vignette("estimate-policyFX").

makeTargetGrid

Creates the grid of all estimands to estimate.

Usage

makeTargetGrid(alphas, small_grid = FALSE)

Arguments

alphas A numeric vector for the probabilities corresponding to the policies of interest. Each entry must be between 0 and 1.
small_grid if TRUE then only estimates some policies. Default is FALSE.
policyFX

Estimate Causal Effects of Population Treatment Policies Assuming Clustered Interference

Description

This function implements the estimators from Barkley et al. (2017) for estimating causal effects ("FX") of treatment policies from an observational study when clustered interference is assumed. Clustered interference is also often known as "partial" interference. For the manuscript introducing the methods in clusteredinterference, see: URL https://arxiv.org/abs/1711.04834

Usage

policyFX(data, formula, alphas, k_samps = NULL, ..., verbose = FALSE, root_options = NULL, nAGQ = 2, return_matrices = FALSE)

Arguments

data A data.frame (not a tibble). Columns of factor types are not recommended and will sometimes throw (defensive) errors.

formula The formula defines the different components of the method. The components are specified by outcome | treatment ~ f(covariates) + (1|cluster_id) | cluster_id. The middle component is passed to glmer, so treatment ~ f(covariates) + (1|cluster_id) specifies the model form for the propensity score (i.e., treatment) model. See Details.

alphas A numeric vector for the probabilities corresponding to the policies of interest. Each entry must be between 0 and 1.

k_samps The maximum number of vectors to evaluate to estimate the counterfactual probabilities (i.e., $\omega(A, N, \alpha)$). Setting to 0 avoids approximation at the cost of increased computation time. Recommended to set <= 5.

... The dots argument. The user may supply their own target_grid through the dots argument. The target_grid can be made through exported function makeTargetGrid

verbose A Boolean on whether to print output to stderr. Defaults to FALSE.

root_options These are passed to multiroot function.

nAGQ This is the number of Adaptive Gaussian Quadrature points used in the glmer model fitting computation. Defaults to 2. It is recommended to use more than 1.

return_matrices A Boolean on whether to return the "bread" and "meat" matrices in the sandwich variance. Defaults to FALSE.
Details

These estimators are based on inverse probability-weighting by the propensity score for treatment (IPW) to estimate causal effects of counterfactual policies of interest (i.e., "policy effects") when clustered interference is assumed. The policies of interest correspond to counterfactual scenarios in which treatment may be correlated within clusters.

This method estimates causal contrasts of these policies by estimating the counterfactual treatment probabilities; taking the correlation structures into account requires heavy computational resources, so the user should be patient.

The modeling formula for the propensity score (i.e., treatment) model is specified via the formula formal argument. An example of a model logit-linear fixed effects would be $Y \mid A \sim X_1 + X_2 + (1 \mid \text{cluster}_{ID}) \mid \text{cluster}_{ID}$. A similar model that also includes an interaction term is $Y \mid A \sim X_1 + X_2 + X_1:X_2 + (1 \mid \text{cluster}_{ID})$.

Value

A list object including:

1. estimates: A tidy data.frame with columns estimand, estimate, var, se, LCI and UCI for 95% CI’s, and more information.
2. parameters: An untidy list of the point estimates of all (target and nuisance) parameters.
3. variance_matrices: When return_matrices is TRUE this is a list object for the "bread" and "meat" matrices in the sandwich variance calculations for each estimand. Otherwise, it is a list object with length 0.
4. propensity_scores: The estimated propensity scores for each cluster.
5. model: The treatment model object.
6. formula: The full formula argument provided, after coercion to a Formula object.

Author(s)

Brian G. Barkley, <BarkleyBG@unc.edu>.

Some of the plumbing functions for estimating the sandwich variance matrix were adapted from Bradley Saul’s geex package.

Some of the plumbing functions for the logistic mixed model likelihood were adapted from Bradley Saul’s inference package.

Brian G. Barkley

References


See Also

Please see the main package vignette at vignette("estimate-policyFX"). It describes the necessary arguments, as well as some extra functionality.

Examples

```r
## Not run:
data("toy_data", "clusteredinterference")
causal_fx <- policyfx(
data = toy_data,
formula = Outcome | Treatment ~ Age + Distance + (1 | Cluster_ID) | Cluster_ID,
alphas = c(0.3, 0.5),
k_samps = 1,
verbose = FALSE
)
## End(Not run)
```

---

**print.policyFX**

*Prints a summary of the estimates from a policyFX object*

**Description**

Prints a summary of the estimates from a policyFX object.

**Usage**

```r
## S3 method for class 'policyFX'
print(x, ...)
```

**Arguments**

- `x` object of class "policyFX"
- `...` User may specify integer `nrows`. 
Summary object: prints a summary of a policyFX object

## Description
Prints a summary of a policyFX object

## Usage
```r
# S3 method for class 'policyFX'
summary(object, ...)
```

## Arguments
- `object`: object of class "policyFX"
- `...`: User may specify integer `nrows`.

## Author(s)
Brian G. Barkley

---

**Toy data**

A "toy" dataset for illustrating the estimator

## Description
A dataset containing simulated data for outcome, treatment status, cluster membership, and two pretreatment covariates. There are 22 clusters that have 3 individuals each, and 8 clusters that have 4 individuals each.

## Usage
toy_data

## Format
A data frame with 98 rows and 5 variables:

- **Outcome**: Individual outcome status observed at the end of follow-up. E.g., infection status
- **Treatment**: Individual treatment status observed at the end of follow-up. E.g., vaccination status
- **Cluster_ID**: Unique identifier for the cluster of which the individual is a member
- **Age**: Individual’s age, in years
- **Distance**: Individual’s distance to river, in kilometers
Index

*Topic **datasets**
  toy_data, 6

clusteredinterference, 2
clusteredinterference-package
  (clusteredinterference), 2

Formula, 4

`glmer`, 3

makeTargetGrid, 2, 3
`multiroot`, 3

policyFX, 2, 3
print.policyFX, 5

summary.policyFX, 6

toy_data, 6