# Package: WMAP (via r-universe)

November 8, 2024

## Contents

---

balancing.weights          *Compute balancing weights using FLEXOR or other methods*

---

### Description

This function calculates balancing weights based on the specified pseudo-population method.

### Usage

```
balancing.weights(
  S,
  Z,
  X,
  method,
  naturalGroupProp,
  num.random = 40,
  gammaMin = 0.001,
  gammaMax = (1 - 0.001),
  seed = NULL,
  verbose = TRUE
)
```

### Arguments

| | |
|---|---|
| S | Vector of factor levels representing the study memberships. Takes values in $\{1, ..., J\}$. |
| Z | Vector of factor levels representing the group memberships. Takes values in $\{1, ..., K\}$. |
| X | Covariate matrix of $N$ rows and $p$ columns. |
| method | Pseudo-population method, i.e., weighting method. Take values in FLEXOR, IC, or IGO. |
| naturalGroupProp | |
| | Relevant only for FLEXOR method: a fixed user-specified probability vector $\theta$. |
| num.random | Relevant only for FLEXOR method: number of random starting points of $\gamma$ in the two-step iterative procedure. Default is 40. |
| gammaMin | Relevant only for FLEXOR method: Lower bound for each $\gamma_s$ in the two-step iterative procedure. Default is 0.001. |
| gammaMax | Relevant only for FLEXOR method: Upper bound for each $\gamma_s$ in the two-step iterative procedure. Default is 0.999. |
| seed | Seed for random number generation. Default is NULL. |
| verbose | Logical; Relevant only for FLEXOR method: if TRUE (default), displays progress messages during computation to the console. Set to FALSE to suppress these messages. |

## Value

An S3 list object with the following components:

**wt.v** $N$ empirically normalized sample weights.

**percentESS** Percentage sample effective sample size (ESS) for the pseudo-population.

## Examples

```
data(demo)
balancing.weights(S, Z, X, method = "IC", naturalGroupProp)
```

---

causal.estimate                  *Estimate causal effects using FLEXOR or other methods*

---

## Description

This function estimates causal effects based on the specified pseudo-population method. The FLEXOR method involves an iterative two-step procedure.

## Usage

```
causal.estimate(
  S,
  Z,
  X,
  Y,
  B = 100,
  method,
  naturalGroupProp = NULL,
  num.random = 40,
  gammaMin = 0.001,
  gammaMax = (1 - 0.001),
  seed = NULL,
  verbose = TRUE
)
```

## Arguments

| | |
|---|---|
| S | Vector of factor levels representing the study memberships. Takes values in {1, ..., J}. |
| Z | Vector of factor levels representing the group memberships. Takes values in {1, ..., K}. |
| X | Covariate matrix of $N$ rows and $p$ columns. |
| Y | Matrix of $L$ outcomes, with dimensions $N \times L$. |
| B | Number of bootstrap samples for variance estimation. Default is 100. |

| method | Pseudo-population method, i.e., weighting method. Take values in FLEXOR, IC, or IGO. |
|---|---|
| naturalGroupProp | |
| | Relevant only for FLEXOR method: a fixed user-specified probability vector $\theta$. |
| num.random | Relevant only for FLEXOR method: number of random starting points of $\gamma$ in the two-step iterative procedure. Default is 40. |
| gammaMin | Relevant only for FLEXOR method: Lower bound for each $\gamma_s$ in the two-step iterative procedure. Default is 0.001. |
| gammaMax | Relevant only for FLEXOR method: Upper bound for each $\gamma_s$ in the two-step iterative procedure. Default is 0.999. |
| seed | Seed for random number generation. Default is NULL. |
| verbose | Logical; if TRUE (default), displays progress messages during computation to the console. Set to FALSE to suppress these messages. |

**Value**

An S3 list object with the following components:

**percentESS** Percentage sample effective sample size (ESS) of the pseudo-population.

**moments.ar** An array of dimension $3 \times K \times L$, containing:

- Estimated means, standard deviations (SDs), and medians (dimension 1),
- For $K$ groups (dimension 2),
- And $L$ counterfactual outcomes (dimension 3).

**otherFeatures.v** Estimated mean group differences for $L$ outcomes.

**collatedMoments.ar** An array of dimension $3 \times K \times L \times B$, containing:

- moments.ar of the $b$th bootstrap sample (dimensions 1–3),
- For $B$ bootstrap samples (dimension 4).

**collatedOtherFeatures.mt** A matrix of dimension $L \times B$ containing:

- otherFeatures.v of the $b$th bootstrap sample (dimension 1),
- For $B$ bootstrap samples (dimension 2).

**collatedESS** A vector of length $B$ containing percentage sample ESS for $B$ bootstrap samples.

**method** Pseudo-population method, i.e., weighting method.

**Examples**

```
data(demo)
set.seed(1)
causal.estimate(S, Z, X, Y, B = 5, method = "IC", naturalGroupProp)
```

## demo — *Demo Dataset*

### Description

A dataset containing example data for demonstration purposes.

### Usage

```
data(demo)
```

### Format

An rda object, with 450 observations and the following variables:

**S** A vector of factor levels, representing the study memberships.

**Z** A vector of factor levels, representing the group memberships.

**X** A covariate matrix.

**Y** An outcome matrix.

**naturalGroupProp** The relative group prevalences of the larger natural population. Necessary only for FLEXOR weights; it should be skipped for IC and IGO weights.

**groupNames** Disease subtype names "IDC" or "ILC"

### Details

Demo Dataset

### Examples

```
data(demo)
```

## plot.causal_estimates — *Plot method for objects of class 'causal_estimates'*

### Description

Plot method for objects of class 'causal_estimates'

### Usage

```
## S3 method for class 'causal_estimates'
plot(x, ...)
```

**Arguments**

| | |
|---|---|
| x | An object of class 'causal_estimates'. |
| ... | Additional arguments including: |

> **y_limit** The y-axis range. Default is `c(0,50)`.
>
> **color** The boxplot color. Default is "red".

**Value**

A boxplot of percent sample ESS for a specific weighting method (`FLEXOR`, `IC`, or `IGO`)

---

summary.balancing_weights

*Summary method for objects of class 'balancing_weights'*

---

**Description**

Summary method for objects of class 'balancing_weights'

**Usage**

```
## S3 method for class 'balancing_weights'
summary(object, ...)
```

**Arguments**

| | |
|---|---|
| object | An object of class 'balancing_weights' |
| ... | Additional arguments affecting the summary produced (so far no additional arguments are needed, so leave blank). |

**Value**

Printed summary of the 'balancing_weights' object, including:

**Weight length** The total number of weights.

**Weight distribution** Statistical summary of weight values.

**percentESS** Percentage sample effective sample size (ESS) for the pseudo-population.

**Examples**

```
data(demo)
output1 <- balancing.weights(S, Z, X, method = "IC", naturalGroupProp)
summary(output1)
```

---

summary.causal_estimates

*Summary method for objects of class 'causal_estimates'*

---

## Description

Summary method for objects of class 'causal_estimates'

## Usage

```
## S3 method for class 'causal_estimates'
summary(object, ...)
```

## Arguments

object          An object of class 'causal_estimates'

...             Additional arguments affecting the summary produced (so far no additional ar-
                guments are needed, so leave blank).

## Value

Printed summary of the 'causal_estimates' object, including:

**Percentage sample ESS** Percentage sample effective sample size (ESS) for the pseudo-population.

**Mean differences with 95% CI** The mean differences between two groups with their correspond-
    ing 95% confidence intervals.

**Sigma ratios with 95% CI** The ratios of standard deviations between two groups with their corre-
    sponding 95% confidence intervals.

## Examples

```
data(demo)
set.seed(1)
output2 <- causal.estimate(S, Z, X, Y, B = 5, method = "IC", naturalGroupProp)
summary(output2)
```

# Index