# Package 'observationalBlocks'

April 10, 2026

**Type** Package

**Title** Block Designs for Observational Studies

**Version** 1.0.0

**Description** Creates block designs of fixed size J with at least one treated and control unit per block. Blocks larger than pairs better distinguish effects caused by a treatment from unmeasured confounding in assignment of individuals to treatment. Somewhat counterintuitively, blocks larger than pairs can use more units while attaining better covariate balance and block homogeneity. A forthcoming manuscript by Brumberg and Rosenbaum details the design.

**License** GPL-2

**Encoding** UTF-8

**Imports** iTOS, lpSolve, stats

**Suggests** DOS2, sensitivity2x2xk, sensitivitymv, weightedRank, xtable, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Depends** R (>= 3.5.0)

**NeedsCompilation** no

**RoxygenNote** 7.3.2

**LazyData** true

**Author** Katherine Brumberg [aut, cre] (ORCID: <https://orcid.org/0000-0002-5193-6250>), Paul Rosenbaum [aut]

**Maintainer** Katherine Brumberg <kbrum@umich.edu>

**Repository** CRAN

**Date/Publication** 2026-04-10 09:50:02 UTC

## Contents

---

`balEq` *Assess covariate balance and homogeneity in matched sample*

---

## Description

Computes balance diagnostics for a specified covariate in the output of `blockMatch`. Compares treated vs control means before and after matching, standardized differences, and within-block homogeneity.

## Usage

```
balEq(vname, o, detail = FALSE)
```

## Arguments

| | |
|---|---|
| `vname` | Character string naming the variable to assess (must be a column in both `o$m` and `o$all`). |
| `o` | A list containing `m` and `all`, as returned by `blockMatch`: `m` is the matched sample and `all` is the full data frame (with `z` and `matched`). |
| `detail` | Logical. If `FALSE` (default), returns the balance matrix. If `TRUE`, returns a list with `balance`, `y` (variable by block), `z` (treatment by block), and `d` (within-block differences). |

## Value

If `detail = FALSE`, a 1-row matrix with columns:

**T-before, C-before** Mean for treated and control before matching

**T-after, C-after** Equally weighted averages of within-block treated or control means after matching

**dif.before, dif.after** Raw difference (T mean - C mean) before and after

**sdif.before, sdif.after** Standardized difference of means before and after; for comparability, both use the pooled standard deviation of `vname` in the full sample before matching, where the pooling equally weights the treated and control groups

**med, q9** Median and 90th percentile of within-block means of pairwise absolute differences

**pct0** Percent of blocks with within-block mean pairwise difference of 0

If `detail = TRUE`, a list with `balance` (that matrix), `y`, `z`, and `d`.

## Examples

```
#' data(Hpylori)
df <- Hpylori[sample(1:nrow(Hpylori), 1000), ]
pr <- glm(hepaA ~ age + female, data = df, family = binomial)$fitted
cochran <- cumsum(c(0, .07, .18, .25, .25, .18, .07))
df$prc <- as.integer(cut(pr, stats::quantile(pr, cochran), include.lowest = TRUE))
df$z <- df$hepaA
bd <- basicDistance(df, near = df$female)
out <- blockMatch(df, cost = bd$cost, J = 4, ratio = 4)
balEq("age", out)
```

---

`basicDistance` *Compute distance matrix for matching*

---

## Description

Compute distance matrix for matching

## Usage

```
basicDistance(
  dat,
  xm = NULL,
  near = NULL,
  xinteger = NULL,
  prc.penalty = 1000,
  near.penalty = 100,
  integer.penalty = 20,
  compute_distance = TRUE
)
```

## Arguments

| | |
|---|---|
| `dat` | A data frame with `N` rows containing at least columns `z` and `prc`. **Treatment** `z`: binary with treated = 1, control = 0 (numeric or logical, not a factor). **Stratum** `prc`: numeric (typically integer labels). Many distinct values (e.g. over 50) can make matching slow or unstable; a warning is issued in that case. If `dat` has a column `id`, it is renamed to `Previous.id` and a new `id` column is added (row indices `1:N`). |
| `xm` | A numeric matrix or data frame with `N` rows, or `NULL`. Covariates for robust Mahalanobis distance; for a covariate with K>2 nominal levels, recode as K-1 binary variables as opposed to one numeric variable for better performance. |
| `near` | A numeric vector of length `N`, or a numeric matrix or data frame with `N` rows, or `NULL`. Each column is one nominal covariate (coded numerically) for near-exact matching. If `near` is a matrix, each column can have its own penalty via `near.penalty`. |

`xinteger` A numeric vector of length `N`, or a numeric matrix or data frame with `N` rows, or `NULL`. Integer-ordered covariates for near-fine balancing (adjacent-category imbalance is cheaper than distant). If `xinteger` is a matrix, each column can have its own penalty via `integer.penalty`.

`prc.penalty` A single finite positive number: penalty for propensity score stratum (`prc`) mismatches in the distance.

`near.penalty` Nonnegative penalties for `near`, finite. If individuals differ on their values of a covariate from `near`, then the distance between them is increased by adding `near.penalty`. If `near` is a **vector**: must be a single value (length 1). If `near` is a **matrix**: either one value (length 1), reused for every column, or a vector of length `ncol(near)` giving one penalty per column. A penalty of 0 skips that column.

`integer.penalty`
Nonnegative penalties for `xinteger`, finite. If individuals differ on a covariate from `xinteger` by `dif` in absolute value, the distance between them is increased by adding `dif * integer.penalty`. If `xinteger` is a **vector**: must be a single value (length 1). If `xinteger` is a **matrix**: either one value (length 1), reused for every column, or a vector of length `ncol(xinteger)` giving one penalty per column. A penalty of 0 skips that column.

`compute_distance`
If `TRUE` (default), build the cost matrix. If `FALSE`, only augment `dat` with `id`, check `z` and `prc` columns, and return `cost = NULL` (e.g. before passing a separate cost matrix to `blockMatch`).

## Details

This function borrows much of its functionality from the package 'iTOS'. Documentation for 'iTOS' functions `addNearExact`, `addinteger`, `addMahal` could prove helpful.

## Value

A list with components:

`dat` The input data frame with column `id` added (and `z`, `prc` coerced in place where applicable).

`cost` The cost/distance matrix for matching (rows = treated, cols = control), or `NULL` if `compute_distance = FALSE`.

## Examples

```
#' data(Hpylori)
df <- Hpylori[sample(1:nrow(Hpylori), 1000), ]
pr <- glm(hepaA ~ age + female, data = df, family = binomial)$fitted
cochran <- cumsum(c(0, .07, .18, .25, .25, .18, .07))
df$prc <- as.integer(cut(pr, stats::quantile(pr, cochran), include.lowest = TRUE))
df$z <- df$hepaA
bd <- basicDistance(df, near = df$female)
```

`blockMatch` *Block matching within propensity score strata*

## Description

Creates blocks of fixed size J with at least one control and one treated. Within each stratum, the function chooses a matching strategy based on the treated-to-control ratio: direct matching when one group dominates, or a two-stage seed-and-add approach when groups are more balanced.

## Usage

```
blockMatch(dat, cost, J = 4, ratio = 4, solver = "rlemon", rseed = 12345)
```

## Arguments

| | |
|---|---|
| `dat` | A data frame with `N` rows containing at least columns `z` and `prc`. **Treatment** `z`: binary with treated = 1, control = 0 (numeric or logical, not a factor). **Stratum** `prc`: numeric (typically integer labels). Many distinct values (e.g. over 50) can make matching slow or unstable; a warning is issued in that case. If `dat` has a column `id`, it is renamed to `Previous.id` and a new `id` column is added (row indices `1:N`). |
| `cost` | Distance matrix: one row per treated unit and one column per control, with `rownames` and `colnames` set to unit ids (row indices of `dat`, `1:N`). Often `basicDistance(...)$cost`. |
| `J` | Target number of individuals per matched block. Each block has at least one control and at least one treated. |
| `ratio` | Minimum matching ratio, greater than or equal to `J - 1`. Matching in fixed ratio occurs when the larger group is larger than the smaller group by at least this factor. Otherwise, blocks are allowed to have varying ratios of treated to control units. |
| `solver` | Either `"rlemon"` or `"rrelaxiv"`. The rlemon solver is automatically available without special installation. The rrelaxiv solver requires a special installation as detailed at https://github.com/josherrickson/rrelaxiv. |
| `rseed` | Single finite number. Fix `rseed` if you want to replicate the match or vary `rseed` to compare different random samples. |

## Value

A list with components:

| | |
|---|---|
| `m` | A data frame of the matched sample, with columns `mset` (matched set ID), `type` (factor: "seed", "add", or "single", indicating whether the unit was included in a seed match, added to a seed match, or included as part of a single stage match for strata with highly imbalanced treatment-control ratios), plus all columns from `dat`. |
| `all` | The full `dat` with an added `matched` logical column indicating who was matched. |

## References

Cochran, W. G. (1968). The effectiveness of adjustment by subclassification in removing bias in observational studies. *Biometrics*, 24(2), 295–313.

## Examples

```
data(Hpylori)
df <- Hpylori[sample(1:nrow(Hpylori), 1000), ]
pr <- glm(hepaA ~ age + female, data = df, family = binomial)$fitted
cochran <- cumsum(c(0, .07, .18, .25, .25, .18, .07))
df$prc <- as.integer(cut(pr, stats::quantile(pr, cochran), include.lowest = TRUE))
df$z <- df$hepaA
bd <- basicDistance(df, near = df$female)
out <- blockMatch(df, cost = bd$cost, J = 4, ratio = 4)
table(out$all$matched, out$all$hepaA)
```

---

# blockSizes *Maximum number of blocks of size J from treated and control counts*

---

## Description

Solves an integer program when there are `nt` treated and `nc` control units. The smaller group is exhausted (all of those units are placed in blocks). Subject to that, the linear program maximizes units from the larger group.

## Usage

```
blockSizes(nt, nc, J)
```

## Arguments

| | |
|---|---|
| `nt` | Number of treated units. |
| `nc` | Number of control units. |
| `J` | Block size (number of units per matched block). |

## Details

This function reproduces some calculations in Section 4 of the forthcoming paper “Constructing Observational Block Designs When the Propensity Score Exhibits Limited Overlap" by Brumberg and Rosenbaum.

If either `nt` or `nc` is 0, or if `nt + nc < J`, a warning is issued and the function returns a degenerate result with zero blocks and zero counts.

## Value

A list with components:

| | |
|---|---|
| `detail` | Named vector with `blocks` (total number of blocks), `treated`, and `control` units used. |
| `counts` | Named integer vector of length `J-1`: number of blocks with 1, 2, ..., J-1 treated units. |

## Examples

```
blockSizes(nt = 2, nc = 10, J = 5)
blockSizes(nt = 10, nc = 2, J = 5)
blockSizes(nt = 6, nc = 6, J = 5)
```

---

# Hpylori *Evidence of Fecal-Oral Transmission of Helicobacter Pylori*

---

## Description

Motivated by the study by Bui et al. (2016), these data from NHANES 1999-2000 concern evidence about the possible fecal-oral transmission of Helicobacter Pylori.

## Usage

```
data(Hpylori)
```

## Format

A data frame with observations (age >= 3, complete cases on key variables) on the following 11 variables.

`SEQN` NHANES id number

`female` 1 if female, 0 if male

`age` Age in years

`education` Education level. Ordered factor with levels `<9` < `9-11` < `HS/GED` < `SomeCol` < `College` < `Age<20`

`income` Family income relative to poverty. Ordered factor with levels `<2`, `>=2`, `Missing`

`black` 1 if black, 0 otherwise

`hispanic` 1 if hispanic, 0 otherwise

`born` Country of birth. Ordered factor with levels `US` < `Mexico` < `Other`

`peopleroom1` 1 if people per room > 1, 0 otherwise

`hepaA` Hepatitis A antibody, 1 if positive, 0 if negative

`helioBP` Helicobacter pylori.

## Details

Does oral consumption of fecal matter – perhaps because someone prepared food without washing their hands – cause infection with Helicobacter Pylori, a type of bacteria that infects the stomach and may cause peptic ulcers or gastric cancer? It is difficult to study this question, because there is no record of incidents in which small amounts of fecal matter were ingested. It is known that hepatitis A virus is mostly transmitted by the fecal-oral route. Following prior studies, Bui et al. (2016) used antibodies for hepatitis A as an indicator of a higher level of ingestion of fecal matter, and examined its relationship with Helicobacter pylori, adjusting for possible confounders, such as age, country of birth, or a crowded home.

## Source

NHANES, US National Health and Nutrition Examination Survey, 1999-2000. https://wwwn.cdc.gov/nchs/nhanes/

## References

Bui, D., Brown, H. E., Harris, R. B. and Oren, E. (2016) Serologic evidence for fecal–oral transmission of Helicobacter pylori. The American Journal of Tropical Medicine and Hygiene, 94(1), 82–88. doi:10.4269/ajtmh.150297 https://pmc.ncbi.nlm.nih.gov/articles/PMC4710451/

## Examples

```
data(Hpylori)
boxplot(Hpylori$helioBP ~ Hpylori$hepaA)
```

# Index