Propensity score based data analysis using nonrandom

Susanne Stampf

2009-11-30

Abstract

For some time, propensity score based methods have been frequently applied in the analysis of data from observational studies. The propensity score is the conditional probability of a certain treatment or exposure given patient's covariates. Propensity score methods are used to eliminate baseline imbalances in covariate distributions between treatment or exposure groups and permit to estimate marginal effects.

The package `nonrandom` is a tool for a comprehensive data analysis using stratification and matching by the propensity score. Several functions are implemented, starting from the selection of the propensity score model up to estimating propensity score based treatment or exposure effects. Before estimating the propensity score, `relative.effect()` permits to investigate the extent to which a covariate is confounding the treatment or exposure effect. This measure may support the decision to include a covariate in the propensity score model. `pscore()` estimates the propensity score and provides all information about the model. Stratification and matching by the propensity score are implemented in `ps.makestrata()` and `ps.match()`, respectively. To check the balance of covariate distributions between treatment or exposure groups, `ps.balance()` tests the distributions using statistical tests or standardized differences and `dist.plot()` allows for a graphical balance check. Finally, propensity score based estimators for the treatment or exposure effect can be determined by `ps.estimate()`. It also provides a comparison to regression based estimates alternatively used.

All functions can be applied separately as well as combined. Additionally, it is possible to apply all functions repeatedly to decide which analysis strategy is the most suitable one.

There are two data examples to illustrate the application of `nonrandom`. In the first data example, quality of life is investigated in breast cancer patients in an observational treatment study of the German Breast Cancer Study Group (GBSG). The second data example deals with lower respiratory tract infections (LRTI) in infants and children in the observational study Pri.DE (Pediatric Respiratory Infection, Deutschland) in Germany.

Contents

1 Introduction

For some time, propensity score based methods have been frequently applied in the analysis of data from observational studies. In 1983, Rosenbaum and Rubin introduced the propensity score as conditional probability of receiving a certain treatment[1] given covariates [1]. In general, the propensity score is unknown and has to be estimated using an appropriate model. The selection of the correct propensity score model is often the first obstacle. Lunt et al.[2] proposed a measure estimating the extent to which a covariate is confounding the treatment effect. Covariates with a large extent are potential candidates for the inclusion in the propensity score model.

Propensity score methods are embedded in the framework of causal modeling dealing with counterfactuals [3]-[5]. Consider a pair of random variables (Y_0, Y_1), where Y_1 denotes the response of an individual if treated, and Y_0 represents the response of the same individual if not treated. The observed response is $Y = ZY_1 + (1-Z)Y_0$, and the expected values of counterfactuals $\mathbf{E}[Y_1]$ and $\mathbf{E}[Y_0]$ can be derived if an identifying assumption called 'strongly ignorable treatment assignment' (SITA) holds [1]. This assumption states, that, within subgroups defined by the propensity score, the observed response of individuals assigned to treatment $Z = 0$ has the same distribution as the unobserved response of individualss assigned to treatment $Z = 1$, if the latter had been assigned to treatment $Z = 0$. The idea of the propensity score was initiated to estimate average linear treatment effects as $\mathbf{E}[Y_1] - \mathbf{E}[Y_0]$ [1]. By now, the idea has been transferred to estimating the marginal odds ratio of response, i.e., the change in odds of response, if everybody versus nobody were treated [6]-[8].

In observational studies, covariate distributions differ generally between treatment groups and propensity score methods aim to eliminate such imbalances. There are several propensity score methods: stratification, matching and covariate adjustment by the propensity score. An further approach is the inverse probability weighting by the propensity score [9]-[11], but it is rarely used. Stratification and matching by the propensity are more popular methods since they are easy to understand. But matching by the propensity score is applied at most in medical research [12, 13].

Stratification by the propensity score groups observations such that distributions of measured covariates are sufficiently balanced in treatment groups within each stratum [1, 14]. It can be supposed that each stratum mimics a randomized sit-

[1] In the following, we only use the phrase '... conditional probability of receiving a certain treatment', i.e., we concentrate on the comparison of response in treated and untreated observations. The comparison of two treatments, e.g.,new and standard therapy are also possible. The propensity score can be also be the conditional probability of being exposed given covariates, respectively, such that the comparison of exposed and unexposed observations is of interest.

uation in which distribution of measured covariates are balanced in expectation. If then the assumption of 'SITA' holds, stratum-specific parameters can be estimated unbiasedly [1]. Those can be summed up using appropriate weights to estimate the marginal parameter of interest.

If matching by the propensity score is used, one or more untreated observations are matched to one treated observation or vice versa. Observations within matched sets have similar propensity scores whereas the similarity is often defined by a caliper, generally used as one-fifth of the standard deviation of the logit of the estimated propensity score [15]. Although matching by the propensity score has been frequently applied [12, 13], it has been shown that the dependence structure in the total matched sample is often not accounted for the estimation of the parameter of interest [16]-[18]. Approaches such as generalized linear mixed models and generalized estimation equations are appropriate to analyze data with correlated observations [19]-[23].

In the following, the application of the package `nonrandom` is demonstrated step by step introducing all implemented functions. The usage of the function is illustrated by the exemplary analysis of two data sets. First, there are data on quality of life in n = 646 breast cancer patients in an observational treatment study of the German Breast Cancer Study Group (GBSG) [24, 25]. Patients with mastectomy and lumpectomy, respectively, are compared with each other regarding the quality of life measured as a linear sum score. The second data example deals with lower respiratory tract infections (LRTI) in a population of n = 3.078 infants and children aged less than three years in the observational study Pri.DE (Pediatric Respiratory Infection, Deutschland) in Germany [26]. Here, the impact of a current infection with the respiratory syncytial virus (RSV) on the severity of LRTI is investigated [8].

2 The estimation of the propensity score

The propensity score, i.e., the conditional probability of receiving a certain treatment given observed covariates is generally unknown and has to be estimated by an appropriate model. The selection of the propensity score model is often a delicate issue [27]-[31]. A measure describing the extent to which a covariate is confounding the treatment effect on response is proposed by Lunt et al. [2] and covariates with a large impact are potential candidates for the propensity score model. This proposal is implemented in `relative.effect()`. If an appropriate propensity score model is selected, `pscore()` estimates the propensity score based on the selection.

2.1 relative.effect

An important step is to decide which covariates X_k, $k = 1, ..., K$ should be included in the propensity score model. The measure describing the extent to which a covariate X_k is confounding the effect of treatment Z on response Y is defined as a relative effect (per cent)

$$\left(\frac{\beta_{z,x_k} - \beta_z}{\beta_z}\right) \times 100$$

with the unadjusted treatment effect β_z on response Y and the treatment effect β_{z,x_k} adjusted for covariate X_k. If the response is binary, the relative effect (per cent) is defined as

$$\left(\frac{exp\{\beta_{z,x_k}\} - exp\{\beta_z\}}{exp\{\beta_z\}}\right) \times 100.$$

Therefor, $K+1$ regression models for response Y, both unadjusted and adjusted for covariates X_k, $k = 1, ..., K$, are fitted using an appropriate generalized linear regression model with a response function according to the scale of response (internal use of 'glm'). There are two options fitting a generalized linear regression model for response. Either use the argument `formula` to specify a formula, typically as '$Y \sim Z + X_1 + ... + X_K$',

```
load(stu1) ## data on quality of life

stu1.effect <-
   relative.effect(data    = stu1,
                   formula = pst~therapie+tgr+age)
```

or specify response, treatment and covariates separately by using the arguments `resp`, `treat` and `sel`.

```
load(pride) ## data dealing with LRTI

pride.effect <-
   relative.effect(data  = pride,
                   sel   = c(2:14),   ## covariates
                   resp  = 15,        ## response
                   treat = "PCR_RSV") ## exposure(!)
```

Independent of the manner of the application of `relative.effect()`, it yields a list containing information about the unadjusted treatment effect β_z, the treatment effects β_{z,x_k} adjusted separately for the covariates X_k, $K = 1, ..., K$ and their relative effects. Additionally, the names of treatment, response and selected covariates are given as well as the description of the error distribution used in the generalized linear regression models.

```
stu1.effect$unadj.treat     ## unadjusted treatment effect
[1] 1.589436

stu1.effect$adj.treat.cov   ## adjusted treatment effects
      tgr       age
1.7004732 0.7880392

stu1.effect$rel.eff.treat   ## relative effects for covariates
       tgr        age
  6.985956 -50.420198
```

2.2 pscore

If an appropriate propensity score model is selected, `pscore()` estimates the propensity score, i.e., the conditional probability of receiving a certain treatment Z given covariates X_k, $k = 1, ..., K$, using a logistic regression model

$$P(Z = 1|X_1, ..., X_K) = \frac{exp\{\alpha_0 + \alpha_1 X_1 + ... + \alpha_K X_K\}}{1 + exp\{\alpha_0 + \alpha_1 X_1 + ... + \alpha_K X_K\}}.$$

Applying `pscore()`, it is possible to specify a name for the variable including the estimated propensity score (`name.pscore`). The default is 'pscore'.

```
## STU1
stu1.ps <- pscore(data    = stu1,
                  formula = therapie~tgr+age)

## PRIDE
pride.ps <- pscore(data        = pride,
                   formula     = PCR_RSV~SEX+RSVINF+REGION+
                                 AGE+ELTATOP+EINZ+EXT,
                   name.pscore = "ps")
```

The output object is of class '"pscore"' and contains a list with information about the propensity score model.

```
## STU1
stu1.ps$name.pscore     ## name of the estimated propensity
[1] "pscore"            ## score added to data

stu1.ps$name.treat      ## name of the treatment variable
[1] "therapie"

stu1.ps$formula.pscore  ## formula of the propensity
```

```
therapie ~ tgr + age    ## score model


##PRIDE
pride.ps$name.pscore    ## name of the estimated propensity
[1] "ps"                ## score

pride.ps$name.treat
[1] "PCR_RSV"
```

Furthermore, the complete data set (`$data`), extended by the estimated propensity score labeled by `name.pscore`, the estimated individual propensity scores (`$pscore`) and the treatment variable (`$treat`) are available.

3 Propensity score methods

Observational studies frequently exhibit imbalances in covariate distributions between treatment groups. Stratification and matching methods are used to eliminate these imbalances.

3.1 `ps.makestrata` - stratification by the propensity score

Stratification by the estimated propensity score groups observations with similar or identical estimated propensity score. In `ps.makestrata()`, stratification can be done in several ways whereas stratification using quintiles of the distribution of the estimated propensity score yields a 90 per cent reduction of bias [14, 34].

The usage of `ps.makestrata()` depends on the class of the input object whereas '"data.frame"' and '"pscore"' (if `pscore()` is previously used) are permitted. No specification of the stratification variable (`stratified.by`) is needed if the input object is of class '"pscore"' (the estimated propensity score stored in '"object$pscore"' is automatically sourced), contrary to the case where the input object is a data frame.

Several options for the argument `breaks` used to define the strata, are available. The default is 'NULL', i.e., the stratification variable is factorized and each factor corresponds to one stratum:

```
stu1.strata4 <- ps.makestrata(object = stu1.ps)

stu1.strata4$intervals
[1] "0.601" "0.709" "0.824" "0.883"
```

If an integer is given in `breaks`, the number of strata w.r.t. the stratification variable is specified:

```
pride.strata.b5 <- ps.makestrata(object = pride.ps,
                                 breaks = 5,
                                 name.stratum.index = "stratum")
pride.strata.b5$intervals
[1] "[0.0619,0.168]" "(0.168,0.275]"  "(0.275,0.382]"
[4] "(0.382,0.488]" "(0.488,0.595]"
```

The argument `name.stratum.index` specifies the name of the variable including the generated stratum indices. If a numeric vector is given or an appropriate R-function is used, e.g., `quantile()`, whose values indicate the stratum bounds:

```
pride.strata5 <- ps.makestrata(object = pride.ps,
                               breaks = quantile(pride.ps$pscore,
                                                 seq(0,1,0.2)))
pride.strata5$intervals
[1] "[0.0624,0.236]" "(0.236,0.306]"  "(0.306,0.369]"
[4] "(0.369,0.431]" "(0.431,0.594]"
```

Depending on the class of the input object, `ps.makestrata()` returns an object of class '"stratified.pscore"' or '"stratified.data.frame"'. If the class of the input object is '"pscore"', the output object inherits all values from the input object. Similar to `pscore()`, the complete data set (`$data`) extended by the stratum indices labeled by `name.stratum.index` and the name of the stratification variable (`$stratified.by`) are available.

Furthermore, the individual stratum indices (`$stratum.index`) generated at least as well as the corresponding stratum intervals (`$intervals`) are stored in the output object.

```
##STU1
stu1.strata4$name.stratum.index    ## default
[1] "stratum.index"

stu1.strata4$stratified.by         ## default
[1] "pscore"


## PRIDE
pride.strata5$name.stratum.index
[1] "stratum"

pride.strata5$stratified.by
[1] "ps"
```

3.2 ps.match - matching by the propensity score

The most popular propensity score method to cope with covariate imbalances is matching by the propensity score.

One or more untreated observations will be matched to each treated observation (or vice versa) according to the estimated propensity score. Matched treated and untreated observations have similar or identical estimated propensity scores wheres the similarity is defined by a caliper, often with a maximum width of one-fifth of the standard deviation of the logit of the estimated propensity score [15].

Similar to `ps.makestrata()`, the usage of the function `ps.match()` depends both on the classes and on the number of the input objects. Allowed classes are '"data.frame"' and '"pscore"' (if `pscore()` is previously used) and one or two input objects, respectively, are permitted.

No specification of the matching variable (`matched.by`) and the treatment variable (`treat`) is needed if the input object is of class '"pscore"'. The estimated propensity score and the treatment variable stored in '"object$pscore"' and '"object$treat"' are automatically sourced. In case of one or two data frames as input objects, both arguments are needed. A second data frame (`object.control`) is necessary if the first input object is a data frame (`object`) containing only treated or only untreated observations. If the matching variable differs in both data frames given, the matching variable in the second data frame (`control.matched.by`) must be specified. Independent of the classes and the numbers of the input objects, the value of the treatment variable indicating 'treated' must be given (`who.treated`). The default is '1'.

There are some parameters to define the matching procedure: the matching ratio (`ratio`) indicating how many observations should be matched, the caliper size (`caliper`) and the corresponding scale `x`, the statement `givenTmatchingC` indicating who should be matched to whom (treated to untreated observations or vice versa) and the statement `bestmatch.first` indicating whether matching partners should be taken randomly from the pool of potential matching partners or those with the most similar estimated propensity score. Furthermore, a random number can be specified (`setseed`) to make the matching procedure reproducible.

As demonstrated in the data example 'pride', one untreated observation is matched to each treated observation (`ratio` =1) and the caliper size is set to one-fifth of the standard deviation of the logit of the estimated propensity score (`caliper`='logit', `x`=0.2). The matching variable is 'ps' indicating the estimated propensity score stored in the input object.

```
pride.match1 <- ps.match(object     = pride.ps,
                         ratio      = 1,        ## default
```

```
                          caliper    = "logit",  ## default
                          matched.by = "ps",
                          setSeed    = 38902)
```

In the data example 'stu1', the matching algorithm is switched such that two treated observations were matched to each untreated observation because fewer untreated than treated observations were observed. Furthermore, the caliper size is set to '0.5'.

```
stu1.match2 <- ps.match(object          = stu1.ps,
                        ratio           = 2,
                        caliper         = 0.5,
                        givenTmatchingC = FALSE,
                        setseed         = 39062)
Argument 'givenTmatchingC'=FALSE: Treated elements were matched to
each untreated element.
```

`ps.match()` returns an object of class '"matched.pscore"', '"matched.data.frame"' or '"matched.data.frames"' depending on the class(es) of the input object(s) and on the argument `combine.output`. The complete data set (`$data`) and a data set limited to the matched observations (`$data.matched`) are available. Both are extended by column(s) including the matching indices labeled by `name.match.index`. Furthermore the individual matching indices generated at last (`$match.index`, `$name.match.index`), the name of the matching variable (`$matched.by`) and the matching parameters (`$match.parameters`) used at last are stored in the output object. If there are two input objects and argument `combine.output` is set to 'TRUE' (default), the values 'data', 'data.matched' and 'match.index' are data frames and a vector, respectively. If `combine.output` is 'FALSE', these values are lists with entries corresponding to the input objects. If the class of the input object is '"pscore"', the output object also inherits all values from the input object.

```
## PRIDE
pride.match1$match.parameters
$caliper
[1] 0.1018815

$ratio
[1] 1

$who.treated
[1] 1

$givenTmatchingC
```

```
[1] TRUE

$bestmatch.first
[1] TRUE

pride.match1$matched.by
[1] "ps"

pride.match1$name.match.index
[1] "match.index"
```

4 The balance check for covariate distributions

Propensity score methods are used to eliminate imbalances in covariate distributions between treatment groups. An important, but often neglected issue is to check those covariate distributions after the balancing procedure (stratification or matching). Graphical checks, statistical tests and standardized differences can be used to examine covariate distributions [35]-[37].

4.1 `dist.plot` - graphical checks

`dist.plot()` offers to plot the distributions of selected covariates in the treatment groups. There are a couple of arguments to configurate the plots illustrated by means of both data examples.

The usage of `dist.plot()` depends on the class of the input object. The arguments `treat`, `stratum.index` or `match.index` have not to be specified if `ps.makestrata()` and `ps.match()` are previously used, respectively. The corresponding values stored in the input object are used. This is in contrast to the case where the input object is a data frame.

If the input object is of class '"stratified.data.frame"' or '"stratified.pscore"', the distributions of the selected covariates given in the argument `sel` are plotted, automatically separated by treatment and strata. If the class of the input object is either '"matched.data.frame"', '"matched.data.frames "' or '"matched.pscore"', the covariate distributions in the treatment groups are only illustrated in the matched data. If a comparison to the original data is desired, the argument `compare` has to be set to 'TRUE' and the graphics will be extended.

There are two different plot types which act depending on the type of covariates. The selected covariates are classified in categorical and non-categorical covariates. Whether a covariate is categorical or not is decided by means of the argument `cat.level`. The default is '10', i.e., if the covariate has more than ten different values, it is considered as non-categorical.

If the argument `plot.type` is '1' (default), bar plots are used to show frequencies for categorical and means for non-categorical covariates (see Figure 1). The covariate distributions are illustrated by means of histograms if `plot.type` is set to '2' (see Figure 2). Here, the argument `plot.levels` specifies the number of cutpoints needed to define the classes for the histogram if the covariate is non-categorical. But the classification still depends on the sructure of the covariate to be plotted such that the number of classes can differ from the specified `plot.levels`. If the covariate is non-categorical, the number of its categories are used to define the cutpoints.

```
## Figure 1
pride.plot1 <-
   dist.plot(object    = pride.strata5,
             sel       = c("REGION", "AGE"),
             plot.type = 1)                  ## default

pride.plot1$var.cat                          ## categorical
[1] "REGION"                                 ## covariates

pride.plot1$var.noncat                       ## non-categorical
[1] "AGE"                                    ## covariates


## Figure 2
pride.plot2 <-
   dist.plot(object       = pride.match1,
             sel          = c("AGE"),
             plot.type    = 2,
             compare      = TRUE,
             legend.title = "RSV infection", ## title of legend
             sub.cex      = 0.7)             ## font size of
                                             ## sub titles

pride.plot2$breaks.noncat                    ## cutpoints of the
[[1]]                                        ## histogram for
[1] 0.0 0.5 1.0 1.5 2.0 2.5 3.0              ## non-categorical
                                             ## covariate
```

Furthermore, there are three useful arguments. The argument `with.legend` is set to 'TRUE' by default, i.e., a legend is shown. If `plot.type` is set to 1, the labels of the categories (categorical covariate) or the labels of the treatment variable (non-categorical covariate) are presented in the legend. Therefore be careful to modify the argument when different covariate types are to be plotted simultaneously. If `plot.type` is set to '2', labels of the treatment variable are

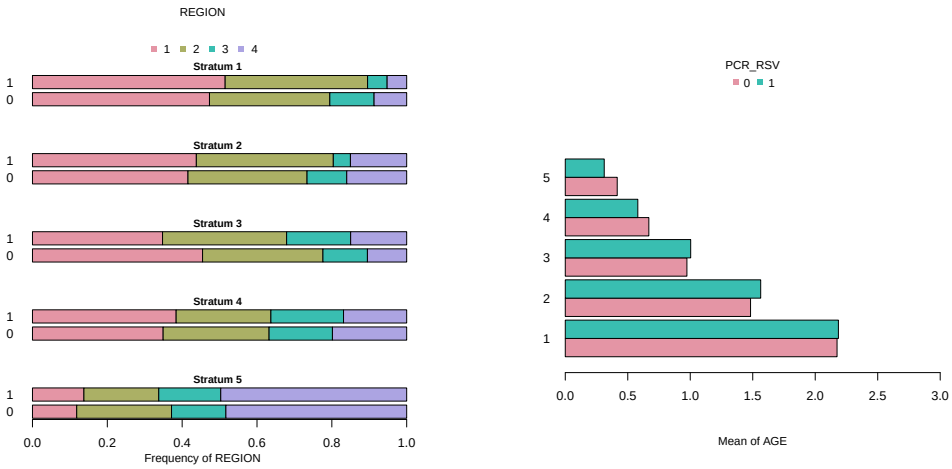

Figure 1: Frequencies of the categorical covariate 'REGION' (left) and means of the non-categorical covariate 'AGE' (right) in the stratified data set 'pride' are illustrated using standard settings

shown in the legend independent of the covariate type.

```
## Figure 3 (left)
stu1.plot1 <-
    dist.plot(object  = stu1.match2,
              sel     = c("tgr"),
              compare = TRUE,
              label.match = c("original data","matched sample"))


## Figure 3 (right)
stu1.plot2 <-
    dist.plot(object      = stu1.match2,
              sel         = c("age"),
              compare     = TRUE,
              plot.type   = 2,
              with.legend = FALSE)
```

The arguments `label.stratum` and `label.match` can be changed if the defaults set to '"Stratum"' and '"c(Original, Matched)"' are not appropriate. All other available arguments should be mainly used to modify font sizes, inner and outer margins of the plot and so on. All values which are plotted are additionally stored in the output list. The number and the manner of the list entries depend

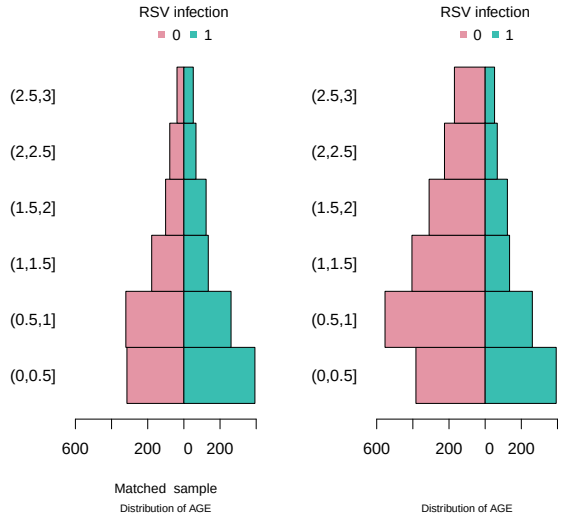

Figure 2: The distribution of the non-categorical covariate 'AGE' in the matched data set of 'pride' (left) compared to its distribution in the original data set 'pride' (right) are shown using histograms

on the type both of the covariates and of the plot. Using `plot.type`='1', the frequencies for the categorical and the means for the non-categorical covariates are stored in lists. The length of these lists is related to the number of categorical and non-categorical covariates.

```
## Figure 1
pride.plot1$frequency      ## frequencies scaled to one for
[[1]]                      ## categorical covariates
, , index = 1              ## Stratum 1

  treat
           0          1
1 0.47302905 0.51492537
2 0.32157676 0.38059701
3 0.11825726 0.05223881
4 0.08713693 0.05223881
```

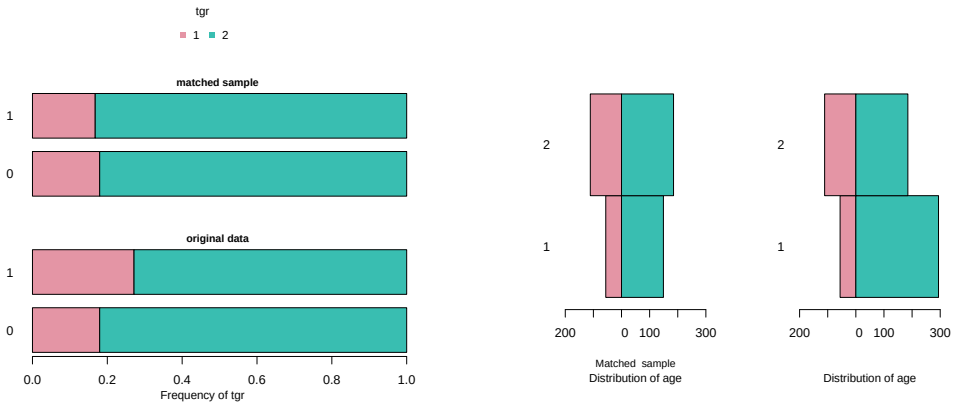

Figure 3: The distribution of both categorical covariates 'tgr' (left) and 'age' (right) in the matched data set 'stu1' are illustrated using different plot types

```
    ...                    ## ...

, , index = 5              ## Stratum 5

   treat
             0          1
  1 0.11824324 0.13750000
  2 0.25337838 0.20000000
  3 0.14527027 0.16562500
  4 0.48310811 0.49687500

pride.plot1$mean           ## means only for non-categorical
[[1]]                      ## covariates
         1        2         3         4         5
0 2.174609 1.482517 0.9733666 0.6686585 0.4156446
1 2.185500 1.563706 1.0035175 0.5804821 0.3117728
```

In case of `plot.type`='2', frequencies are stored in lists w.r.t. the lower and upper value of the treatment variable, respectively, which are indicated by `x.` and `y.` at the beginning of the value name.

```
## Figure 2
pride.plot2$x.s.noncat    ## left side in graphics
[[1]]
```

```
[[1]]$`1`   ## original data
[1] 383 554 405 310 225 170

[[1]]$`2`   ## matched data
[1] 315 321 178 101  78  38

pride.plot2$y.s.noncat    ## right side in graphics
[[1]]
[[1]]$`1`   ## original data
[1] 393 261 135 123  67  52

[[1]]$`2`   ## matched data
[1] 393 261 135 123  67  52
```

Furthermore, information about treatment (`$treatment`), the individual stratum indices (`$stratum.index`) or matching indices (`$match.index`) and the selected covariates (`$name.sel`, `$sel`) are also saved in the output list.

4.2 `ps.balance` - statistical tests and standardized differences

`ps.balance()` permits the application of statistical tests or the calculation of standardized differences to access whether covariate distributions between treatment groups are balanced. The method of standardized differences is preferred in the literature since they do not depend on the sample size [38]-[40].

Similar to the functions described above, the usage of `ps.balance()` depends on the class of the input object. If either `ps.makestrata()` or `ps.match()` are previously used, the arguments `treat`, `stratum.index` or `match.index` are not needed, contrary to the case if the input object is a data frame.

To apply classical statistical tests on data, i.e., t-test for non-categorical covariates and χ^2- test for categorical covariates (internal use of `t.test()` and `chisq.test()`), the argument `method` must be set to 'classical' (default). The argument `cat.levels` specifies whether a covariate is categorical or not (see `dist.plot()`). The tests are employed to the data both before and after the balancing procedure (stratification or matching).

```
## PRIDE
pride.balance <- ps.balance(object  = pride.strata5,
                            sel     = c(2:8),
                            method  = "classical",
                            alpha   = 5)
```

If the argmuent `method` is set to '"stand.diff"', standardized differences are cal-

culated for each selected covariate before and after the balancing procedure.

```
## STU1
stu1.balance <- ps.balance(object  = stu1.match2,
                           sel     = c("tgr","age"),
                           method  = "stand.diff",
                           alpha   =  20)
```

The value '"bal.test"' of the output object (of the same class as the input object) contains comprehensive information about the balance between treatment groups for each selected covariate. Here, the values '0' and '1' describe whether the covariate distribution is 'imbalanced' and 'balanced', respectively. In case of stratified data, the covariate distribution is considered as balanced after stratification, if each stratum-specific distribution is balanced. If there is an imbalance in at least one stratum, the covariate distribution is considered as imbalanced after stratification.

```
## STU1
stu1.balance$bal.test$balance.table
             tgr age
table.before   0   0    ## 'tgr' are imbalanced before, but balanced
table.after    1   0    ## after matching; 'age' remains imbalanced

stu1.balance$bal.test$balance.table.summary
                      before: no balance (0) before: balance (1)
after: no balance (0)                      1                    0
after: balance (1)                         1                    0

stu1.balance$bal.test$covariates.NA
character(0)

stu1.balance$bal.test$covariates.bal.before
character(0)

stu1.balance$bal.test$covariates.bal.after
[1] "tgr"


## PRIDE
pride.balance$bal.test$balance.table
             SEX ETHNO FRUEHG RSVINF HERZ REGION AGE
table.before   0     1      1      0    1      0   0
table.after    1     0      1     NA    1      0   0
```

```
pride.balance$bal.test$balance.table.summary
                      before: no balance (0) before: balance (1)
after: no balance (0)                      2                   1
after: balance (1)                         1                   2

pride.balance$bal.test$covariates.NA
[1] "RSVINF"

pride.balance$bal.test$covariates.bal.before
[1] "ETHNO"  "FRUEHG" "HERZ"

pride.balance$bal.test$covariates.bal.after
[1] "SEX"    "FRUEHG" "HERZ"
```

Additionally, the names of covariates for which either the statistical test could not be applied or standardized differences could not be calculated are saved (`$covariates.NA`). The names of balanced covariates before and after the balancing procedure are also stored in the output list (`$covariates.bal.before` and `$covariates.bal.after`).

Depending on the selected method, information about test results or standardized differences is available. If statistical tests are applied, resultant p-values for each covariate are given as a matrix (`$p.value`). Here, there is a column for each covariate and p-values from the tests in the original data can be found in the first row and those from the tests within the stratified or the matched data are placed in rows $2, ..., S$.

```
pride.balance$bal.test$p.value                    ## p-values of tests
       SEX ETHNO FRUEHG RSVINF  HERZ REGION   AGE
[1,] 0.010 0.907  0.413  0.000 0.518  0.000 0.000  ## original data
[2,] 0.296 0.632  0.223  0.647 0.766  0.058 0.830  ## stratum 1
[3,] 0.160 0.003  0.980  0.422 0.642  0.133 0.084  ##
[4,] 0.798 0.169  0.678  0.757 0.484  0.038 0.429  ## ...
[5,] 0.124 0.212  0.724     NA 0.843  0.542 0.002  ##
[6,] 0.960 0.882  0.404     NA 0.523  0.415 0.000  ## stratum 5

pride.balance$bal.test$method   ## applied tests
      SEX  ETHNO  FRUEHG  RSVINF  HERZ  REGION        AGE
    "cat"  "cat"   "cat"   "cat" "cat"   "cat"  "non-cat"


pride.balance$bal.test$alpha
[1] 5                           ## significance level
```

If standardized differences are calculated, the standardized differences (per cent),

the means and the standard deviations (SD) in the treatment groups for each covariate are given in the same manner as in value `$p.value`.

```
stu1.balance$bal.test$Means.treat.0
           tgr       age   ## means w.r.t. treatment '0'
[1,] 0.8203593 0.6646707   ## ... before matching
[2,] 0.8203593 0.6646707   ## ... after matching

stu1.balance$bal.test$Means.treat.1
           tgr       age   ## means w.r.t. treatment '1'
[1,] 0.7286013 0.3862213
[2,] 0.8323353 0.5538922

stu1.balance$bal.test$SDs.treat.0
           tgr       age   ## SDs w.r.t. treatment '0'
[1,] 0.3838879 0.4721055
[2,] 0.3838879 0.4721055

stu1.balance$bal.test$SDs.treat.1
           tgr       age   ## SDs w.r.t. treatment '1'
[1,] 0.4446813 0.4868823
[2,] 0.3735682 0.4970871

stu1.balance$bal.test$Standardized.differences
           tgr      age    ## standard differences
[1,] 22.089171 58.06462
[2,]  3.161882 22.85235

stu1.balance$bal.test$method
  tgr      age
"bin"    "bin"          ## type of covariate

stu1.balance$bal.test$alpha
[1] 20                     ## cutpoint for the decision
                           ## about balance
```

In value `$method`, the type of each covariates is stored. The significance level is also available (`$alpha`). It has to be interpreted as cutpoint at which the decision about the balance of a covariate distribution is made if standardized differences are calculated.

The check for balance of covariate distributions entails the knowledge about the correctness of the propensity score model. If the propensity score model is correctly fitted, at least covariates included in propensity score model should be sufficiently balanced after the stratification or matching. Otherwise re-modeling

of the propensity score model should be considered.

5 Propensity score based treatment effects

The estimation of the propensity score based treatment effect differs in the application of the propensity score method. Therefore, the following section is separated by the propensity score methods which can be applied.

In general, the usage of `ps.estimate()` depends on the class of the input object. If `ps.makestrata()` or `ps.match()` are previously used, the arguments `treat`, `stratum.index` or `match.index` are not needed, contrary to the case if the input object is a data frame.

5.1 Estimator based on stratification by the propensity score

If stratification is applied in data with continuous response, the marginal treatment effect based on the propensity score is estimated as a weighted sum of differences of the mean responses in treated and untreated observations over the propensity score strata. To summarize, two different kinds of weights w_s are possible. Firstly, the weights are equal to the proportion of observations in each stratum (`weights`='rr') or, secondly, the weights are related to the inverse variance of the stratum-specific treatment effect (`weights`='opt').

```
stu1.estimate <-
    ps.estimate(object    = stu1.strata4,
                resp    = "pst",           ## continuous response
                weights = "opt",
                regr    = c("tgr", "age")  ## regression model
```

In case of stratified data with binary response, both the stratified Mantel-Haenszel estimator and the estimator based on response rates [6] are used to estimate a treatment effect as an odds ratio. Both methods estimate different parameters and therefore they differ in their interpretation of the estimated odds ratio [8]. Propensity score methods are used to estimate marginal treatment effects, but only the response rates estimator fulfills the criteria for an estimator of the marginal odds ratio. It is defined as an odds ratio of marginal response probabilities, contrary to the stratified Mantel-Haenszel estimator which is a weighted sum of stratum-specific odds ratios [6, 8]. A marginal odds ratio for response describes the change in odds of response, if everybody versus nobody were treated. It is different to the conditional odds ratio, e.g., estimated by logistic regression (with the assumption of constant individual odds ratios). The

popular Mantel-Haenszel estimator stratified by the propensity score can fail to estimate both the indivdiual, conditional and the marginal odds ratio [8].

```
pride.estimate <-
   ps.estimate(object  = pride.strata5,
               resp    = "SEVERE",         ## binary response
               treat   = "PCR_RSV",
               family  = "binomial",
               adj     = c("AGE", "EXT", "KRANKSUM"),
               regr    = SEVERE ~ PCR_RSV + SEX + ETHNO + FRUEHG +
                                  HERZ + ELTATOP + REGION + AGE +
                                  TOBACCO + VOLLSTIL +EXT + EINZ +
                                  KRANKSUM,
               weights = "rr")
```

In addition to the estimation of the unadjusted propensity score based treatment effect, it is possible to adjust for residual imbalances in strata using argument `adj`. Stratum-specific treatment effects are then estimated using generalized linear models which are the same in each stratum. Furthermore, traditional regression models can be fitted using argument `regr`. There are two options to specify both arguments `adj` and `regr`. First, they can be given as formulas, typically as 'Y$\sim$Z$+X_1+...+X_K$'. Here, response Y and treatment Z must be the same as arguments `resp` and `treat` if given. Another option is to specifiy only a vector with names or integers related to the covariates in the data for which the treatment effect on response should be adjusted for in the strata.

The output object contains information about all estimates which are listed separated by the estimation procedure. Furthermore, the values depend on the type of response (continuous or binary). Regression based estimates are included in value `$lr.estimation`. Here, both the estimated conditional and marginal treatment effects and their standard errors are given. If the response is binary, the standard errors are given on the log scale. Information about the regression model is also available. In case of continuous response, the continuous and the marginal treatment effects are identical.

```
## STU1
stu1.estimate$lr.estimation   ## regression based treatment effect
$effect
[1] 0.8979454                 ## conditional treatment effect

$se
[1] 1.299290                  ## standard error of conditional effect

$regr.formula
pst ~ therapie + tgr + age    ## regression model used
```

The value `$ps.estimation` includes the crude treatment effect (`$crude`) and the estimated propensity score based treatment effects, both unadjusted (`$unadj`) and adjusted (`$adj`) if desired. Additionally, the estimated stratum-specific treatment effects, estimated standard errors (on the log scale for binary response) and the used weights per stratum are given. In case of binary response, both the estimated stratum-specific odds ratios needed for the stratified Mantel-Haenszel estimator and the estimated stratum-specific response probabilities used by the response rates estimator are also stored.

```
## PRIDE
pride.estimate$ps.estimation ## propensity score based estimates
$crude
$crude$effect                ## crude treatment effect via 'Y~Z'
[1] 1.676623

$crude$se                    ## standard error for the crude
[1] 0.07961634               ## treatment effect


$unadj
$unadj$effect.mh            ## stratified Mantel-Haenszel estimator
[1] 1.418535

$unadj$odds.str             ## stratum-specific odds ratios
       1        2        3        4        5
1.164420 1.147405 1.164530 1.671525 2.166276

$unadj$se.mh                ## standard error for the
[1] 0.08234186              ## Mantel-Haenszel estimator

$unadj$effect               ## response rates based estimator
[1] 1.361594

$unadj$se                   ## standard error on log scale for the
[1] 0.08051982              ## response rates based estimator

$unadj$p1                   ## estimated marginal response
[1] 0.6303096               ## probabilities for treatment '1'

$unadj$p0                   ## estimated marginal response
[1] 0.5559866               ## probabilities for treatment '0'

$unadj$p1.str  ## stratum-specific response prob's for treatment '1'
        1         2         3         4         5
```

```
0.4776119 0.5686275 0.5828877 0.7130802 0.8093750

$unadj$p0.str  ## stratum-specific response prob's for treatment '0'
        1         2         3         4         5
0.4398340 0.5346320 0.5454545 0.5978836 0.6621622


$adj
$adj$model                   ## adjustment within strata
SEVERE ~ PCR_RSV + AGE + EXT + KRANKSUM

$adj$effect.str              ## adjusted stratum-specific effects
[1] 1.181630 1.205805 1.111415 1.534873 2.094389

$adj$effect                  ## adjusted overall propensity score
[1] 1.425658                 ## based treatment effect

$adj$se                      ## standard error for the adjusted
[1] 0.1893357                ## propensity score based estimator


$weights
[1] "rr"

$weights.str                 ## weights per stratum
[1] 0.2001300 0.1998051 0.2001300 0.1998051 0.2001300
```

Further values in the output object contain information about the response (`$name.resp`, `$resp`), the treatment (`$name.treat`, `$treat`) and the stratum indices (`$name.stratum.index`, `$stratum.index`). The output object inherits all values from the input object as well.

5.2 Estimator based on matching by the propensity score

If matching is applied, the dependency structure of the matched sample can be accounted for in the data analysis [41]-[43]. Generalized linear mixed models are appropriate and implemented in `lmer` (package `lme4`). It is used in `ps.estimate()` for the estimation of treatment effects in data matched by the propensity score. Therefore, random intercepts for each matching set are modeled.

The data analysis of a matched sample can be done in the same way as for stratified data. The values of the output object in case of matched data differ slightly from the those based on the analysis of stratified data. There are nat-

urally no stratum-specific effect estimates and corresponding weights available, but only an estimated overall treatment effect and its estimated standard error. If the response is binary, the standard error are given on the log scale.

```
## STU1, matched sample
stu1.estimate.match <-
    ps.estimate(object = stu1.match2,
                resp   = "pst")

stu1.estimate.match$ps.estimation  ## crude effect, identical to that
$crude                             ## of analysis of stratified data
$crude$effect
[1] 1.589436

$crude$se
[1] 1.260993


$unadj
$unadj$effect
[1] 0.8732535

$unadj$se
[1] 1.317626


$adj
[1] "No adjustment"

$weights
NULL

$weights.str
NULL


## PRIDE, matched sample
pride.estimate.match <-
   ps.estimate(object  = pride.match1,
               resp    = "SEVERE",
               family  = "binomial")

pride.estimate.match$ps.estimation$unadj
$unadj$effect
```

```
[1] 1.378804

$unadj$se
[1] 0.09157813
```

As above, information about the response (`$name.resp`, `$resp`), the treatment (`$name.treat`, `$treat`)and the matching indices (`$name.match.index`, `$match.index`) are stored in the output object. The output object also inherits all values from the input object.

References

[1] PR Rosenbaum and DB Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.

[2] M Lunt, D Solomon, K Rothman, R Glynn, K Hyrich, DPM Symmons, and T Stürmer. Different methods of balancing covariates leading to different effect estimates in the presence of effect modification. *American Journal of Epidemiology*, 169(7):909–917, 2009.

[3] J Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2000.

[4] MA Hernan. A definition of causal effect for epidemiological research. *Journal of American Statistical Association*, 58:265–271, 2003.

[5] MA Hernan and JM Robins. Estimating causal effects from epidemiological data. *Journal of Epidemiology and Community Health*, 60:578–586, 2006.

[6] E Graf and M Schumacher. Letter to the editor: Comments on the performance of different propensity score methods for estimating marginal odds ratios. *Statistics in Medicine*, 27(19):3915–3917, 2008.

[7] A Forbes and S Shortreed. Letter to the editor: Inverse probability weighted estimation of the marginal odds ratio: Correspondence regarding 'The performance of different propensity score methods for estimating marginal odds ratios'. *Statistics in Medicine*, 27(26):5556–5559, 2008.

[8] S Stampf, E Graf, C Schmoor, and M Schumacher. Estimators and confidence intervals for the marginal odds ratio using logistic regression and propensity score stratification. *Statistics in Medicine*, in press, 2010.

[9] PR Rosenbaum. Model-based direct adjustment. *Journal of American Statistical Association*, 82:387–394, 1987.

[10] K Hirano and GW Imbens. Estimation of causal effects using propensity score weighting: An application to data on right heart catheterization. *Health Services & Outcomes Research Methodology*, 2:259–278, 2001.

[11] DA Freedman and RA Berk. Weighting regressions by propensity scores. *Evaluation Review*, 32:392–409, 2008.

[12] BR Shah, A Laupacis, JE Hux, and PC Austin. Propensity score methods gave similar results to traditional regression modeling in observational studies: A systematic review. *Journal of Clinical Epidemiology*, 58(6):550–559, 2005.

[13] T Stürmer, M Joshi, RJ Glynn, J Avorn, KJ Rothman, and S Schneeweiss. A review of the application of propensity score methods yielded increasing use, advantages in specific settings, but not substantially different estimates compared with conventional multivariable methods. *Journal of Clinical Epidemiology*, 59(5):437–461, 2006.

[14] PR Rosenbaum and DB Rubin. Reducing bias in observational studies using subclassification on the propensity score. *Journal of American Statistical Association*, 79(387):516–524, 1984.

[15] WG Cochran and DB Rubin. Controlling bias in observational studies: a review. *Sankhya Series A*, (35):516–524, 1973.

[16] DB Rubin and N Thomas. Characterizing the effect of matching using linear propensity score methods with normal distributions. *Biometrike*, 79(4):797–809, 1992.

[17] P Austin. Propensity-score matching in the cardiovascular surgery literature from 2004 to 2006: a systematic review and suggestions for improvement. *Journal of Thoracic and Cardiovascular Surgery*, 134:1128–1135, 2007.

[18] PC Austin. A critical appraisal of propensity-score matching in the medical literature between 1996 and 2003. *Statistics in Medicine*, 27(12):2037–2049, 2008.

[19] JA Nelder and RWM Wedderburn. Generalized linear models. *Jornal of Royal Statistical Society A*, 135(3):370–384, 1972.

[20] DR Cox and EJ Snell. *Analysis of binary data*. Chapman and Hall, London, second edition, 1989.

[21] PJ Diggle, KY Liang, and SL Zeger. *Analysis of Longitudinal Data*. Oxford University Press, Oxford, 1994.

[22] JA Hanley, A Negassa, MD deB. Edwardes, and JE Forrester. Statistical analysis of correlated data using generalized estimating equations: An orientation. *American Journal of Epidemiology*, 157(4):364–375, 2003.

[23] AJ Dobson and AG Barnett. *Introduction to Generalized Linear Models*. Chapman and Hall, London, third edition, 2008.

[24] H. F. Rauschecker, R. Sauer, A. Schauer, M. Schumacher, M. Olschewski, W. Sauerbrei, M. H. Seegenschmiedt, and C. Schmoor. Therapy of small breast cancer – four–year results of a prospective non–randomized study. *Breast Cancer Research and Treatment*, 34:1–13, 1995.

[25] Stephen Senn, Erika Graf, and Angelika Caputo. Stratification for the propensity score compared with linear regression techniques to assess the effect of treatment or exposure. *Statistics in Medicine*, 26(30):5529–5544, 2007.

[26] J Forster, G Ihorst, CH Rieger, V Stephan, HD Frank, H Gurth, R Berner, A Rohwedder, H Werchau, M Schumacher, T Tsai, and G Petersen. Prospective population-based study of viral lower respiratory tract infections in children under 3 years of age (the pri.de study). *European Journal of Pediatrics*, 163(12):709–716, 2004.

[27] C Drake. Effects of misspecification on the propensity score on estimatiors of treatment effects. *Biometrics*, 49(4):1231–1236, 1993.

[28] Katherine Huppler Hullsiek and Thomas A. Louis. Propensity score modeling strategies for the causal analysis of observational data. *Biostatistics*, 2(4):179–193, 2002.

[29] S Weitzen, KL Lapane, AY Toledano, AL Hume, and V Mor. Principles for modelling propensity scores in medical research. *Pharmacoepidemiology and Drug Safety*, 13(12):841–853, 2004.

[30] Alan M. Brookhart, Sebastian Schneeweiss, Kenneth J. Rothman, Robert J. Glynn, Jer ry Avorn, and Til Stürmer. Variable selection for propensity score models. *American Journal of Epidemiology*, 141(12):1–8, 2006.

[31] DB Rubin. The design versus the analysis of observational studies for causal effects: Parallels with the design of randomized trials. *Statistics in Medicine*, 26:20–36, 2007.

[32] PR Rosenbaum. Observational studies. *Springer Verlag*, New York, 1995.

[33] MM Joffe and PR Rosenbaum. Invited commentary: Propensity scores. *American Journal of Epidemiology*, 150:327–333, 1999.

[34] WG Cochran. The effectiveness of adjustment by subclassification in removing bias in observational studies. *Biometrics*, 24:295–313, 1968.

[35] PC Austin. Assessing balance in measured baseline covariates when using many-to-one matching on the propensity score. *Pharmacoepidemiology and drug safety*, 17(12):1218–1225, 2008.

[36] BB Hansen. Commentary: The essential role of balance tests in propensity-matched observational studies: Comments on 'a critical appraisal of propensity-score matching in the medical literature between 1996 and 2003' by peter austin, statistics in medicine. *Statistics in Medicine*, 27(12):2050–2054, 2008.

[37] PC Austin. The realibility of different propensity score methods to balance measured covariates between treated and untreated subjects in observational studies. *Medical decision making*, doi:10.1177/0272989X09341755, 2008.

[38] J Hill. Commentary: Discussion of research using propensity-score matching: Comments on 'a critical appraisal of propensity-score matching in the medical literature between 1996 and 2003' by peter austin, statistics in medicine. *Statistics in Medicine*, 27(12):2055–2061, 2008.

[39] EA Stuart. Commentary: Developing practical recommendations for the use of propensity scores: Discussion of 'a critical appraisal of propensity score matching in the medical literature between 1996 and 2003' by peter austin, statistics in medicine. *Statistics in Medicine*, 27(12):2062–2065, 2008.

[40] PC Austin. Rejoinder: Discussion of 'a critical appraisal of propensity-score matching in the medical literature between 1996 and 2003'. *Statistics in Medicine*, 27(12):2066–2069, 2008.

[41] NE Breslow and NE Day. *Statistical Methods in Cancer Research, Volume 1 - The Analysis of Case-Control Studies*. International Agency for Research on Cancer (IARC Scientific Publications No. 32), Lyon, 1980.

[42] KJ Rothman, S Greenalnd, and TL Lash. *Modern epidemiology*. Lippincott Williams & Wilkins, Philadelphia, third edition, 2008.

[43] A Agresti and Y Min. Effects and non-effects of paired identical observations in comparing proportions with binary matched-pairs data. *Statistics in Medicine*, 23:65–75, 2004.