Conditional and Marginal Effects in a Regression Context

Ashley I Naimi

Spring 2024

Contents

1 Introduction

Let's say we're interested in estimating the average treatment effect of some binary exposure X on some outcome Y. We can estimate this effect on the difference scale:

$$\psi = E(Y^1 - Y^0)$$

On the ratio scale:

$$\psi = E(Y^1)/E(Y^0)$$

And, if the outcome is binary (assuming $Y \in [0, 1]$), on the odds ratio scale:

$$\psi = \frac{E(Y^1)}{1 - E(Y^1)} \Big/ \frac{E(Y^0)}{1 - E(Y^0)}$$

Note that these are **average** or **marginal** effects, because they represent contrasts of averages of potential outcomes, with the averages taken over all individuals in the population of interest. This are different from **conditional** average treatment effects, or CATEs. In practice, we'd often use regression models to estimate the risk difference, risk ratio, and odds ratio. However, before we discuss regression modeling, it's important to understand some numerical properties of the risk difference, risk ratio, and odds ratio as a measure of association or effect.

Cautionary Note: Communicating Risk Contrasts

Several people use different nomenclature to communicate risk differences, risk ratios, and odds ratios. Among these include "attributable risk", "excess risk", or "absolute risk increase/reduction" for the risk difference, and "relative risk" for the risk ratio or odds ratio. However, ambiguous terms such as "relative risk" should be avoided. The risk difference, risk ratio, and odds ratio are all measures of "relative risk." and thus this term should not be generally used. Additionally, as English language constructs, "attributable risk" and "excess risk" are both applicable to the risk difference and risk ratio. To avoid confusion, it's best to use the relevantly accurate terminology: i.e., risk difference, risk ratio, or odds ratio.

2 Collapsibility versus Noncollapsibility of Association Contrasts

Consider the following data adapted from Greenland (2005):

```r
# table 1 from greenland
d <- data.frame(z = c(1, 1, 1, 1, 0, 0, 0,
    0), x = c(1, 1, 0, 0, 1, 1, 0, 0), y = c(1,
    0, 1, 0, 1, 0, 1, 0), n = c(200, 50,
    150, 100, 100, 150, 50, 200))

d <- d %>%
    uncount(n)
```

These data lead to the following contingency table :

	$Z = 1$		$Z = 0$		Marginal	
	$X = 1$	$X = 0$	$X = 1$	$X = 0$	$X = 1$	$X = 0$
$Y = 1$	0.20	0.15	0.10	0.05	0.30	0.20
$Y = 0$	0.05	0.10	0.15	0.20	0.20	0.30
Risks[a]	0.80	0.60	0.40	0.20	0.60	0.40
Risk differences	0.20		0.20		0.20	
Risk ratios	1.33		2.00		1.50	
Odds ratios	2.67		2.67		2.25	

[a]Probabilities of $Y = 1$.

Figure 1: Table 1 from Greenland 2005 showing examples of collapsibility and noncollapsibility in a three-way distribution.

This Table shows some important properties of the risk difference, risk ratio, and odds ratio. First, let's ensure we understand it's elements.

The contingency tables in Table 1 show the proportions of the outcome Y stratified by Z and overall (Marginal) for each level of X. Importantly, the data in this Table was generated from a model where Z did not affect the exposure X. The only causal relations are between the exposure X and the outcome Y, and the covariate Z and the outcome Y.

These proportions are used to then compute the "risks" or the $P(Y = 1)$. From these probabilities, we can compute risk differences, risk ratios, and odds ratios.[1]

[1] Be sure you know how to compute these.

Notice how these results differ between the Z strata and overall:

- The risk difference is 0.2 across stratum and overall. This is true even though the absolute risks differ across strata and overall. This is a general phenomenon associated with the risk difference. That is, across strata of *a variable that is not causally related to or associated with the exposure* the

risk difference is constant and equal to the overall risk difference. For this reason, we say that the **risk difference is strictly collapsible.**

- The risk ratio is 1.33 for the $Z = 1$ stratum and 2.00 for $Z = 0$. Furthermore, overall, the risk ratio is 1.5. While these three risk ratios are different, they are related in an important way. Note that the weighted average of the stratum specific risk ratios equal the overall risk ratio:

$$\begin{aligned}
&\frac{P(Z=1)P(Y=1 \mid X=1, Z=1) + P(Z=0)P(Y=1 \mid X=1, Z=0)}{P(Z=1)P(Y=1 \mid X=0, Z=1) + P(Z=0)P(Y=1 \mid X=0, Z=0)} \\
&= \frac{0.5(0.80) + 0.5(0.40)}{0.5(0.60) + 0.5(0.20)} \\
&= 1.5
\end{aligned}$$

Thus, the information that is contained in the overall risk ratio is available in the stratum specific risk ratios, but must be transformed appropriately. For this reason, we say that the **risk ratio is collapsible.**

- The odds ratio is 2.67 in both strata created by Z. However, the overall odds ratio is less than the stratum specific odds ratios, with a value of 2.25. Thus, there is no weighted combination of the stratum specific estimates that yield the overall estimate. Conceptually, it is almost as though the stratum specific odds ratios are capturing something completely different from the overall odds ratio. For this reason, we say that the **odds ratio is noncollapsible.** Note also the direction: the stratum specific odds ratio will always be greater than or equalled to the overall or marginally adjusted odds ratio.

3 Noncollapsibility in a Regression Context

Noncollapsibility is one reason we have to be aware of the differences between unadjusted, conditionally adjusted, and marginally adjusted regression models. More specifically, the issue is about understanding the differences between conditionally and marginally adjusted effects.

3.1 Unadjusted Regression Models

An unadjusted regression model yielding an unadjusted effect can be implemented as:

```
mod_unadjusted <- glm(y ~ x, data = d, family = binomial("logit"))
```

We can compute the effect of interest by taking the exponent of the coefficient for the exposure from this model:

```
exp(summary(mod_unadjusted)$coefficients[2,
    1])
```

```
## [1] 2.25
```

We can also compute the marginal effect by taking the average of the predictions from this model under exposed and unexposed settings and constructing an odds ratio with them:

```
mu1 <- mean(predict(mod_unadjusted, newdata = transform(d,
    x = 1), type = "response"))
mu0 <- mean(predict(mod_unadjusted, newdata = transform(d,
    x = 0), type = "response"))


marg_OR <- (mu1/(1 - mu1))/(mu0/(1 - mu0))

marg_OR
```

```
## [1] 2.25
```

Note that, in this case where there are no other variables in the model, the conditional and marginal odds ratio are equivalent.

3.2 Conditionally Adjusted Model

We can also adjust for Z in two ways. The first is in a conditionally adjusted model. To do this, we simply add Z to the unadjusted model, and then read the coefficient for the exposure from the model:

```
mod_adjusted <- glm(y ~ x + z, data = d,
    family = binomial("logit"))

exp(summary(mod_adjusted)$coefficients[2,
    1])
```

```
## [1] 2.666667
```

Note that in this conditionally adjusted regression model, the odds ratio is 2.67, which is larger than the unadjusted coefficient. This is true even though Z is not a confounder, collider, or any other variable of causal importance. That is, simply adjusting for a variable representing "noise" in the system yields a conditionally adjusted odds ratio that is larger than the unadjusted coefficient.

This is noncollapsibility of the odds ratio.

3.3 Marginally Adjusted Model

What happens if we adjust our model for Z, and then deploy the marginally adjusted approach to compute the odds ratio?:

```
mu1 <- mean(predict(mod_adjusted, newdata = transform(d,
    x = 1), type = "response"))
mu0 <- mean(predict(mod_adjusted, newdata = transform(d,
    x = 0), type = "response"))


marg_OR <- (mu1/(1 - mu1))/(mu0/(1 - mu0))

marg_OR
```

```
## [1] 2.25
```

We can see that, even though our regression model includes the noise variable Z, the marginally adjusted odds ratio is once again 2.25.

4 Conditionally Adjusted Parametric Regression: An NHEFS Example

Let's see how some of these concepts can be important in a more realistic setting. Suppose we wanted to use the NHEFS data to estimate the confounder adjusted effect of quitting smoking on weight change. Suppose further that we were interested specifically in whether an individual's weight change was greater than the median value in the data.

```
pacman::p_load(broom, tidyverse, boot)

nhefs <- read_csv(here("data", "nhefs.csv")) %>%
    mutate(wt_delta = as.numeric(wt82_71 >
        median(wt82_71)))

#' Quick view of data
dim(nhefs)
```

```
## [1] 1394   12
```

```
names(nhefs)
```

```
##  [1] "seqn"     "qsmk"     "sex"      "age"      "income"   "sbp"
##  [7] "dbp"      "price71"  "tax71"    "race"     "wt82_71"  "wt_delta"
```

A typical approach would be to use a generalized linear model to regress the indicator of weight change against the indicator of whether the individual quit smoking:

```
#' Here, we start fitting relevant regression models to the data.
#' modelForm is a regression argument that one can use to regress the
#' outcome (wt_delta) against the exposure (qsmk) and selected confounders.

formulaVars <- "qsmk + sex + age + income + sbp + dbp + price71 + tax71 + race"
modelForm <- as.formula(paste0("wt_delta ~",
```

```
    formulaVars))
modelForm
```

```
## wt_delta ~ qsmk + sex + age + income + sbp + dbp + price71 +
##     tax71 + race
```

```
#' This model can be used to quantify a conditionally adjusted
#' odds ratio with correct standard error
modelOR <- glm(modelForm, data = nhefs, family = binomial("logit"))

summary(modelOR)
```

```
##
## Call:
## glm(formula = modelForm, family = binomial("logit"), data = nhefs)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.822736   1.178775  -0.698 0.485203
## qsmk         0.594853   0.133325   4.462 8.13e-06 ***
## sex          0.126227   0.113976   1.107 0.268084
## age         -0.033421   0.005579  -5.991 2.09e-09 ***
## income       0.027180   0.022297   1.219 0.222846
## sbp         -0.004227   0.004232  -0.999 0.317912
## dbp          0.027001   0.006978   3.870 0.000109 ***
## price71      0.121377   0.837939   0.145 0.884828
## tax71       -0.193909   0.886114  -0.219 0.826782
## race        -0.304103   0.175398  -1.734 0.082956 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1932.5  on 1393  degrees of freedom
## Residual deviance: 1828.0  on 1384  degrees of freedom
```

```
## AIC: 1848
##
## Number of Fisher Scoring iterations: 4
```

```
tidy(modelOR)[2, ]
```

```
## # A tibble: 1 x 5
##   term  estimate std.error statistic    p.value
##   <chr>    <dbl>     <dbl>     <dbl>      <dbl>
## 1 qsmk     0.595     0.133      4.46 0.00000813
```

```
exp(tidy(modelOR)[2, 2])
```

```
## # A tibble: 1 x 1
##   estimate
##      <dbl>
## 1     1.81
```

If we were interested in estimating conditionally adjusted risk differences or risk ratios for the effect of quitting smoking, we could use a similar approach with the identity link function, ordinary least squares, or Poisson regression (Zou, 2004, Naimi and Whitcomb (2020)):

```
#' This model can be used to quantify a conditionally adjusted
#' risk ratio with INCORRECT standard error
modelRR <- glm(modelForm, data = nhefs, family = poisson("log"))

summary(modelRR)
```

```
##
## Call:
## glm(formula = modelForm, family = poisson("log"), data = nhefs)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.132237   0.804153  -1.408  0.15914
```

```
## qsmk         0.259255   0.084688   3.061  0.00220 **
## sex          0.062866   0.077844   0.808  0.41933
## age         -0.016519   0.003896  -4.240 2.23e-05 ***
## income       0.015044   0.015772   0.954  0.34017
## sbp         -0.002141   0.003019  -0.709  0.47827
## dbp          0.013128   0.004776   2.749  0.00598 **
## price71      0.067070   0.566970   0.118  0.90583
## tax71       -0.111025   0.598623  -0.185  0.85286
## race        -0.152933   0.125793  -1.216  0.22408
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 966.25  on 1393  degrees of freedom
## Residual deviance: 914.46  on 1384  degrees of freedom
## AIC: 2328.5
##
## Number of Fisher Scoring iterations: 5
```

```
tidy(modelRR)[2, ]
```

```
## # A tibble: 1 x 5
##   term  estimate std.error statistic p.value
##   <chr>    <dbl>     <dbl>     <dbl>   <dbl>
## 1 qsmk     0.259    0.0847      3.06 0.00220
```

```
exp(tidy(modelRR)[2, 2])
```

```
## # A tibble: 1 x 1
##   estimate
##      <dbl>
## 1     1.30
```

```r
#' This model can be used to quantify a conditionally adjusted
#' risk difference with INCORRECT standard error
modelRD <- lm(modelForm, data = nhefs)

summary(modelRD)
```

```
##
## Call:
## lm(formula = modelForm, data = nhefs)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.85362 -0.47432  0.01569  0.46427  0.84259
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.3128875  0.2751611   1.137 0.255691
## qsmk         0.1380039  0.0305604   4.516 6.84e-06 ***
## sex          0.0291977  0.0265032   1.102 0.270797
## age         -0.0078457  0.0012766  -6.146 1.04e-09 ***
## income       0.0060288  0.0051614   1.168 0.242981
## sbp         -0.0009280  0.0009638  -0.963 0.335747
## dbp          0.0061640  0.0015845   3.890 0.000105 ***
## price71      0.0352711  0.1953759   0.181 0.856764
## tax71       -0.0525840  0.2065347  -0.255 0.799069
## race        -0.0706438  0.0404551  -1.746 0.080993 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4833 on 1384 degrees of freedom
## Multiple R-squared:  0.07222,    Adjusted R-squared:  0.06619
## F-statistic: 11.97 on 9 and 1384 DF,  p-value: < 2.2e-16
```

```
tidy(modelRD)[2, ]
```

```
## # A tibble: 1 x 5
##   term  estimate std.error statistic    p.value
##   <chr>    <dbl>     <dbl>     <dbl>      <dbl>
## 1 qsmk     0.138    0.0306      4.52 0.00000684
```

5 Marginally Adjusted Parametric Regression: An NHEFS Example

Another approach to obtaining risk differences, risk ratios, and odds ratios from GLMs that are not subject to the limitations noted above is to use marginal standardization, which is equivalent to g computation (aka the parametric g formula) when the exposure is measured at a single time point (Naimi et al., 2017). This process gives us marginally adjusted effects, and can be implemented by fitting a single logistic model, regressing the binary outcome against all confounder variables. But instead of reading the coefficients the model, one can obtain odds ratios, risk ratios, or risk differences by using this model to generate predicted risks for each individual under "exposed" and "unexposed" scenarios in the dataset. To obtain standard errors, the entire procedure must be bootstrapped.[2]

Here is some code to implement this marginal standardization in the NHEFS data:

[2] Though some forthcoming work (not by me) has developed an analytic solution to the variance estimation problem in a marginally standardized regression framework.

```
#' Regress the outcome against the confounders with interaction
ms_model <- glm(modelForm, data = nhefs,
    family = binomial("logit"))
##' Generate predictions for everyone in the sample to obtain
##' unexposed (mu0 predictions) and exposed (mu1 predictions) risks.
mu1 <- predict(ms_model, newdata = transform(nhefs,
    qsmk = 1), type = "response")
mu0 <- predict(ms_model, newdata = transform(nhefs,
    qsmk = 0), type = "response")
```

```r
#' Marginally adjusted odds ratio
marg_stand_OR <- (mean(mu1)/mean(1 - mu1))/(mean(mu0)/mean(1 -
    mu0))
#' Marginally adjusted risk ratio
marg_stand_RR <- mean(mu1)/mean(mu0)
#' Marginally adjusted risk difference
marg_stand_RD <- mean(mu1) - mean(mu0)

#' Using the bootstrap to obtain confidence intervals for the marginally adjusted
#' risk ratio and risk difference.
bootfunc <- function(data, index) {
    boot_dat <- data[index, ]
    ms_model <- glm(modelForm, data = boot_dat,
        family = binomial("logit"))
    mu1 <- predict(ms_model, newdata = transform(boot_dat,
        qsmk = 1), type = "response")
    mu0 <- predict(ms_model, newdata = transform(boot_dat,
        qsmk = 0), type = "response")

    marg_stand_OR_ <- (mean(mu1)/mean(1 -
        mu1))/(mean(mu0)/mean(1 - mu0))
    marg_stand_RR_ <- mean(mu1)/mean(mu0)
    marg_stand_RD_ <- mean(mu1) - mean(mu0)
    res <- c(marg_stand_RD_, marg_stand_RR_,
        marg_stand_OR_)
    return(res)
}

#' Run the boot function. Set a seed to obtain reproducibility
set.seed(123)
boot_res <- boot(nhefs, bootfunc, R = 2000)

boot_RD <- boot.ci(boot_res, index = 1)
boot_RR <- boot.ci(boot_res, index = 2)
```

```
boot_OR <- boot.ci(boot_res, index = 3)
```

```
marg_stand_OR
```

```
## [1] 1.746213
```

```
marg_stand_RR
```

```
## [1] 1.295328
```

```
marg_stand_RD
```

```
## [1] 0.1377615
```

```
boot_RD
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 2000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boot_res, index = 1)
##
## Intervals :
## Level      Normal              Basic
## 95%   ( 0.0793,  0.1966 )   ( 0.0802,  0.1976 )
##
## Level     Percentile            BCa
## 95%   ( 0.0779,  0.1953 )   ( 0.0779,  0.1952 )
## Calculations and Intervals on Original Scale
```

```
boot_RR
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 2000 bootstrap replicates
```

```
##
## CALL :
## boot.ci(boot.out = boot_res, index = 2)
##
## Intervals :
## Level      Normal              Basic
## 95%   ( 1.158,  1.431 )   ( 1.156,  1.428 )
##
## Level     Percentile            BCa
## 95%   ( 1.163,  1.435 )   ( 1.163,  1.435 )
## Calculations and Intervals on Original Scale
```

```
boot_OR
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 2000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boot_res, index = 3)
##
## Intervals :
## Level      Normal              Basic
## 95%   ( 1.302,  2.162 )   ( 1.266,  2.125 )
##
## Level     Percentile            BCa
## 95%   ( 1.367,  2.227 )   ( 1.367,  2.224 )
## Calculations and Intervals on Original Scale
```

This marginal standardization approach yields an estimate of the average treatment effect under the required identifiability assumptions. However, it assumes a constant effect of qsmk on weight change across levels of all of the other variables in the model. This constant effect assumption might be true, but if one wanted to account for potential interactions between the exposure and all of the confounders in the model, there is an easy way. We call this the "stratified modeling approach."

This stratified modeling approach avoids the exposure effect homogeneity

assumption across levels of all the confounders. In effect, the approach fits a separate model for each exposure stratum. To obtain predictions under the "exposed" scenario, we use the model fit to the exposed individuals to generate predicted outcomes in the entire sample. To obtain predictions under the "unexposed" scenario, we repeat the same procedure, but with the model fit among the unexposed. One can then average the risks obtained under each exposure scenario, and take their difference and ratio to obtain the risk differences and ratios of interest.

```
#' Marginal Standardization
##' To avoid assuming no interaction between
##' quitting smoking and any of the other variables
##' in the model, we subset modeling among
##' exposed/unexposed. This code removes qsmk from the model,
##' which will allow us to regress the outcome
##' against the confounders among the exposed and
##' the unexposed separately. Doing so will allow us
##' to account for any potential exposure-covariate interactions
##' that may be present.
formulaVars <- "sex + age + income + sbp + dbp + price71 + tax71 + race"
modelForm <- as.formula(paste0("wt_delta ~",
    formulaVars))
modelForm
```

```
## wt_delta ~ sex + age + income + sbp + dbp + price71 + tax71 +
##     race
```

```
#' Regress the outcome against the confounders
#' among the unexposed (model0) and then among the exposed (model1)
model0 <- glm(modelForm, data = subset(nhefs,
    qsmk == 0), family = binomial("logit"))
model1 <- glm(modelForm, data = subset(nhefs,
    qsmk == 1), family = binomial("logit"))
##' Generate predictions for everyone in the sample using the model fit to only the
##' unexposed (mu0 predictions) and only the exposed (mu1 predictions).
```

```r
mu1 <- predict(model1, newdata = nhefs, type = "response")
mu0 <- predict(model0, newdata = nhefs, type = "response")

#' Marginally adjusted odds ratio
marg_stand_OR <- (mean(mu1)/mean(1 - mu1))/(mean(mu0)/mean(1 -
    mu0))
#' Marginally adjusted risk ratio
marg_stand_RR <- mean(mu1)/mean(mu0)
#' Marginally adjusted risk difference
marg_stand_RD <- mean(mu1) - mean(mu0)

#' Using the bootstrap to obtain confidence intervals for the marginally adjusted
#' risk ratio and risk difference.
bootfunc <- function(data, index) {
    boot_dat <- data[index, ]
    model0 <- glm(modelForm, data = subset(boot_dat,
        qsmk == 0), family = binomial("logit"))
    model1 <- glm(modelForm, data = subset(boot_dat,
        qsmk == 1), family = binomial("logit"))
    mu1 <- predict(model1, newdata = boot_dat,
        type = "response")
    mu0 <- predict(model0, newdata = boot_dat,
        type = "response")

    marg_stand_OR_ <- (mean(mu1)/mean(1 -
        mu1))/(mean(mu0)/mean(1 - mu0))
    marg_stand_RR_ <- mean(mu1)/mean(mu0)
    marg_stand_RD_ <- mean(mu1) - mean(mu0)
    res <- c(marg_stand_RD_, marg_stand_RR_,
        marg_stand_OR_)
    return(res)
}

#' Run the boot function. Set a seed to obtain reproducibility
```

```r
set.seed(123)
boot_res <- boot(nhefs, bootfunc, R = 2000)

boot_RD <- boot.ci(boot_res, index = 1)
boot_RR <- boot.ci(boot_res, index = 2)
boot_OR <- boot.ci(boot_res, index = 3)

marg_stand_OR
```

```
## [1] 1.757907
```

```r
marg_stand_RR
```

```
## [1] 1.298577
```

```r
marg_stand_RD
```

```
## [1] 0.139347
```

```r
boot_RD
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 2000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boot_res, index = 1)
##
## Intervals :
## Level      Normal              Basic
## 95%   ( 0.0794,  0.1995 )   ( 0.0791,  0.1996 )
##
## Level     Percentile            BCa
## 95%   ( 0.0791,  0.1996 )   ( 0.0790,  0.1995 )
## Calculations and Intervals on Original Scale
```

```
boot_RR
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 2000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boot_res, index = 2)
##
## Intervals :
## Level      Normal              Basic
## 95%   ( 1.159,  1.436 )   ( 1.156,  1.432 )
##
## Level     Percentile            BCa
## 95%   ( 1.165,  1.442 )   ( 1.164,  1.440 )
## Calculations and Intervals on Original Scale
```

```
boot_OR
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 2000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boot_res, index = 3)
##
## Intervals :
## Level      Normal              Basic
## 95%   ( 1.297,  2.187 )   ( 1.247,  2.142 )
##
## Level     Percentile            BCa
## 95%   ( 1.374,  2.269 )   ( 1.372,  2.261 )
## Calculations and Intervals on Original Scale
```

When predicted risks are estimated using a logistic model, relying on marginal standardization will not result in probability estimates outside the bounds [0, 1]. And because the robust variance estimator is not required,

model-based standardization will not be as affected by small sample sizes. However, the bootstrap is more computationally demanding than alternative variance estimators, which may pose problems in larger datasets.

References

Sander Greenland. Collapsibility. In Mitchell H. Gail and Jacques Bénichou, editors, *Encyclopedia of Epidemiologic Methods*. John Wiley & Sons, Ltd, 2005.

Ashley I Naimi and Brian W Whitcomb. Estimating risk ratios and risk differences using regression. *American Journal of Epidemiology*, 189(6):508–510, 2020.

Ashley I Naimi, Stephen R Cole, and Edward H Kennedy. An Introduction to G Methods. *Int J Epidemiol*, 46(2):756–62, 2017.

Guangyong Zou. A modified poisson regression approach to prospective studies with binary data. *Am J Epidemiol*, 159(7):702–706, Apr 2004.