Code for Quantitative Bias Analysis

Jeremy Brown

Contents

Chapter 1

About

This site contains code for the applied examples in the article "*Quantifying possible bias in clinical and epidemiological studies using quantitative bias analysis*" in addition to illustrative code to conduct alternative quantitative bias analyses.

Chapter 2

Applied examples

In "*Applying quantitative bias analysis in clinical and epidemiological studies using quantitative bias analysis*" we present three applied examples of bias formulas, bounding methods, and probabilistic bias analysis. Code to generate results for these applied examples is presented below.

2.1 Bias formulas for selection bias

In a cohort study of pregnant women investigating the association between maternal lithium use, relative to non-use, and cardiac malformations in live-born infants, a covariate-adjusted risk ratio was estimated of 1.65 (95% CI, 1.02-2.68).[6] Live-born infants only were included in the study and as such there was potential for selection bias if there were differences in termination probabilities of foetuses with cardiac malformations by exposure group.

Given that the outcome was rare, and therefore odds ratios and risk ratios are approximately equivalent, bias formulas using odds ratios were applied.[4]

$$OR_{BiasAdjusted} = OR_{Observed}\frac{S_{01}S_{10}}{S_{00}S_{11}}$$

Values for the bias parameters, selection probabilities by exposure and outcome status, were specified based on the literature. The selection probability of unexposed without cardiac malformations was assumed to be 0.8 (i.e. 20% probability of termination). The selection probability of unexposed with cardiac malformation was varied from 0.5 to 0.7. (30-50% probability of termination). Selection probabilities of exposed were defined by outcome status relative to unexposed (0% to -10%).

```r
library(tidyverse)
library(ggplot2)

# Define observed odds ratio
observed_rr <- 1.65

# Define bias parameters
S00 <- 0.8
S01 <- c(0.5, 0.525, 0.55, 0.575, 0.6, 0.625, 0.65, 0.675, 0.7)
differences <- c(0, -0.05, -0.1)

# Define function to calculate bias-adjusted risk ratio
calc_bias_adj_or <- function(or, s01, s10, s00, s11) {
  bias_adj_or <- or * (s01*s10)/(s00*s11)
  return(bias_adj_or)
}

# Calculate bias-adjusted estimate for different values of bias parameters
results <- NULL
for (s01 in S01) {
  for (diff in differences) {
    bias_adj_rr <- calc_bias_adj_or(observed_rr, s01, S00 + diff, S00, s01 + diff)
    results_row <- tibble_row(bias_adj_rr=bias_adj_rr, s01=s01, diff=as.character(diff)
    results <- bind_rows(results, results_row)
  }
}

# Tidy label for difference in selection probabilities
results <- results %>%
  mutate(diff=factor(diff, levels=c("0", "-0.05", "-0.1"), labels=c("0", "-5%", "-10%")

# Plot figure of bias-adjusted estimates
ggplot(data = results, aes(x=s01, y=bias_adj_rr, colour=diff)) +
  geom_line() +
  theme_minimal() +
  ylim(1.4, 2) +
  scale_x_reverse() +
  xlab("Selection probability of unexposed with cardiac malformations") +
  ylab("Bias-adjusted risk ratio") +
  guides(colour=guide_legend(title="Difference in selection\nprobability of exposed")) +
  theme(legend.title=element_text(size=10))
```

The bias-adjusted risk ratios ranged from 1.65 to 1.80 (Figure 2.1), indicating robustness of the point estimate to selection bias under the given assumptions. We can likewise calculate bias-adjusted estimates for the lower bound of the

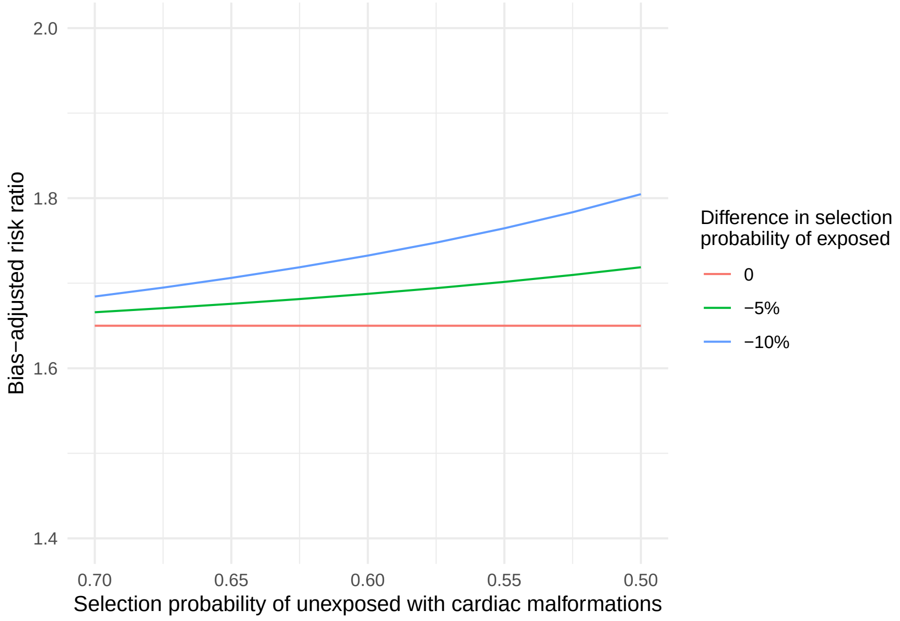
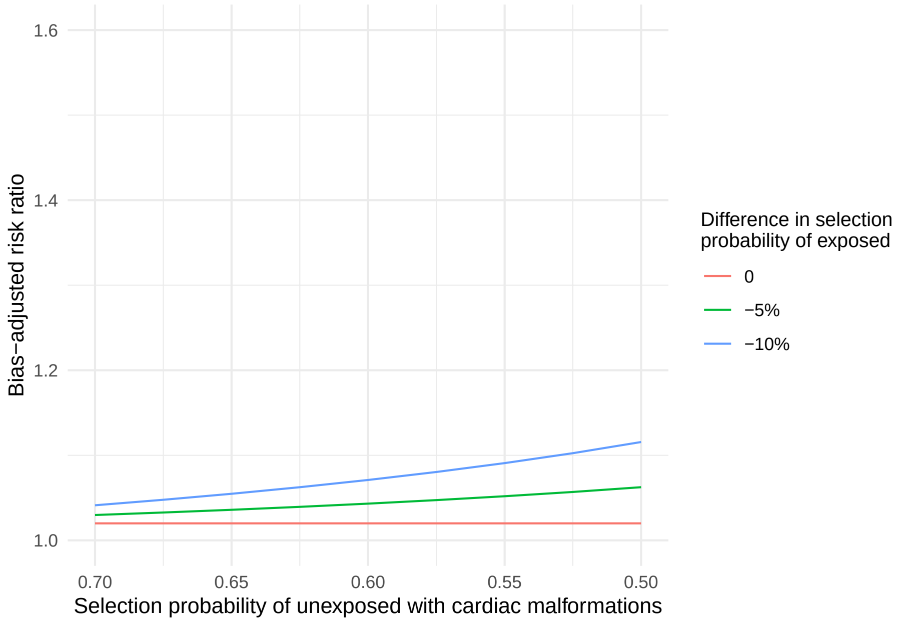


Figure 2.1: Bias-adjusted risk ratio for different assumed selection probabilities

confidence interval.

\begin{figure}



{

}

\caption{Bias-adjusted risk ratio for lower bound of 95% confidence interval for different assumed selection probabilities} \end{figure}

2.2 E-values for unmeasured confounding

In a cohort study conducted in electronic health records investigating the association between proton pump inhibitors, relative to H2 receptor antagonists, and all-cause mortality, investigators found that individuals were at higher risk of death (covariate-adjusted hazard ratio [HR] 1.38, 95% CI 1.33-1.44).[1] However, it was considered that there may be unmeasured confounding due to differences in frailty between individuals prescribed proton pump inhibitors. The prevalence of the unmeasured confounder was not known in either exposure group, and therefore rather than use bias formulas, an E-value was calculated.[2]

Given the outcome was rare, the E-value method can be applied to the hazard ratio.

$$\text{E-value} = RR_{Obs} + \sqrt{RR_{Obs}(RR_{Obs} - 1)}$$

We can use the EValue package to calculate E-Values.

```r
# load EValue package and ggplotify
library(EValue)

#Calculate E-values
evalues.HR(est=1.38, lo=1.33, hi=1.44, rare=TRUE) %>%  knitr::kable()
```

	point	lower	upper
RR	1.380000	1.330000	1.44
E-values	2.104155	1.992495	NA

And we can use *bias_plot* from the *EValue* package to display an E-value plot.

```r
# Generate E-value plot for point estimate
bias_plot(1.38, xmax=9)
```

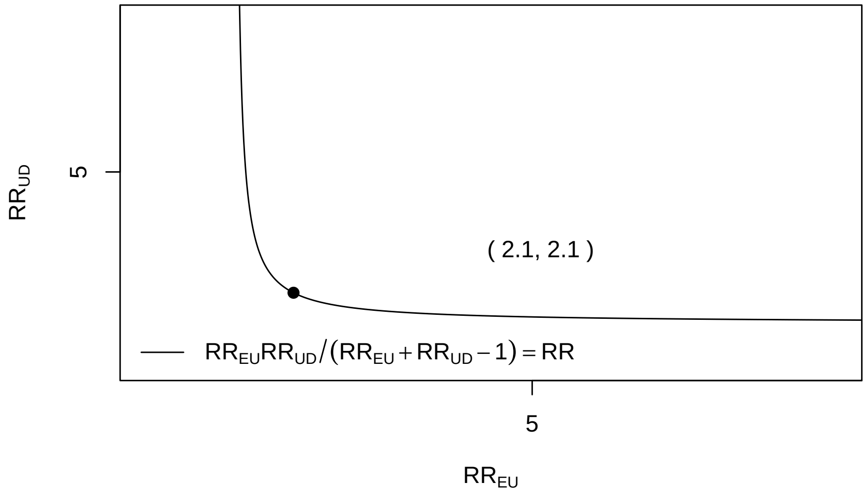


Figure 2.2: E-value plot for point estimate

The E-value for the point estimate was 2.10 and for the lower bound of the point estimate was 1.99. This represents the minimum strength of association that an unmeasured confounder would need to have with either exposure or outcome to reduce the hazard ratio to the null (i.e. 1). An unmeasured confounder with strength of association with exposure and outcome below the line in the plot could not possibly explain, on its own, the observed association.

Risk ratios between frailty and mortality >2 have been observed in the literature, and as such we could not rule out unmeasured confounding as a possible explanation for findings based on the E-value. However, as we did not specify prevalence of an unmeasured confounder, we cannot say whether such confounding was likely to account for the observed association. There may also have been additional unmeasured or partially measured confounders contributing to the observed association.

2.3 Probabilistic bias analysis for misclassification

In a cohort study of pregnant women conducted using insurance claims data, the observed covariate-adjusted risk ratio for the association between antidepressant use and congenital cardiac defects, was 1.02 (95% CI, 0.90-1.15).[5] Some misclassification of the outcome, congenital cardiac malformation was anticipated. Therefore, probabilistic bias analysis was carried out.[3] Code is not provided for this analysis, which was carrier out using SAS and participant record-level data (see sensmac macro for a SAS program to conduct this analysis).

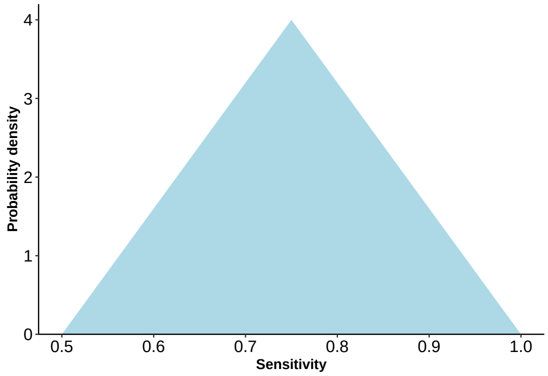


Figure 2.3: Triangular distribution for sensitivity

Positive predictive values estimated in a validation study were used to specify distributions of values for the bias parameters of sensitivity and specificity. Investigators chose triangular distributions for the positive and negative predictive value (Figure 2.3).

Values were repeatedly sampled at random 1,000 times from these distributions and these sampled values were used to calculate a distribution of bias-adjusted estimates. The median bias-adjusted estimate was 1.06 with 95% simulation interval 0.92-1.22.

Chapter 3

Additional example applications

We will present here examples of quantitative bias analysis using simulated participant-level data for a cohort study with a binary treatment, binary outcome, and binary confounder.

```
# load packages
require(tidyverse)
require(ggplot2)
library(broom)
library(janitor)
```

3.1 Selection bias

In the simulated data we have a binary treatment a, binary confounder x, outcome y, and a binary indicator for selection into the study s.

```
sim_data <- read_csv("data/simulated_data.csv") %>% select(x,a,y,s)
sim_data  %>% head(5)
```

```
## # A tibble: 5 x 4
##       x     a     y     s
##   <dbl> <dbl> <dbl> <dbl>
## 1     0     0     0     1
## 2     0     0     0     1
## 3     0     0     0     1
## 4     0     0     0     1
```

Table 3.1: Estimated odds ratio

estimate	conf.low	conf.high
2.040021	1.942229	2.142647

```
## 5      0      0      0      1
```

The data was generated such that the causal odds ratio between treatment and outcome was 2. However, in a given sample the estimate may differ due to random error. If we observed a random sample from the target population we could unbiasedly estimate the odds ratio using logistic regression.

```r
# fit logistic regression model
lgr_model <- glm(y ~ x + a, data=sim_data, family="binomial")

# tidy model outputs
or <- lgr_model %>%
  tidy(exponentiate=TRUE, conf.int=TRUE) %>%
  filter(term == "a") %>%
  select(estimate, conf.low, conf.high)

# output as table
or %>%
  knitr::kable(caption = "Estimated odds ratio")
```

However, if we consider that selection into the study was dependent on exposure and outcome and that we only observed a selected subsample of the target population, then the estimated odds ratio is biased.

```r
# restrict data to selected subsample
selected_data <- sim_data %>% filter(s == 1)

# fit logistic regression model
lgr_model_selected <- glm(y ~ x + a, data=selected_data, family="binomial")

# tidy model outputs
or_selected <- lgr_model_selected %>%
  tidy(exponentiate=TRUE, conf.int=TRUE) %>%
  filter(term == "a") %>%
  select(estimate, conf.low, conf.high)

# output as table
or_selected %>%
  knitr::kable(caption = "Estimated odds ratio in selected sample")
```

Table 3.2: Estimated odds ratio in selected sample

estimate	conf.low	conf.high
1.577843	1.494616	1.665406

Table 3.3: Bias-adjusted odds ratio

estimate	conf.low	conf.high
2.028656	1.92165	2.141236

3.1.1 Bias formulas

One option is to apply bias formulas for the odds ratio.

$$OR_{BiasAdjusted} = OR_{Observed}\frac{S_{01}S_{10}}{S_{00}S_{11}}$$

Given that the data was simulated we know the selection probabilities ($S11 = 0.7$, $S01 = 1$, $S10 = 0.9$, $S00 = 1$) and can directly plug them in to estimate a bias-adjusted odds ratio. However, in practice we will not know these probabilities and will typically specify a range of values, or for probabilistic bias analysis a distribution of values.

```r
# define bias parameters
S11 <- 0.7
S01 <- 1
S10 <- 0.9
S00 <- 1

# apply bias formula
bias_adjusted_or <- or_selected %>%
  mutate(across(c(estimate, conf.low, conf.high), ~ .x * (S01*S10)/(S11*S00)))

# output as table
bias_adjusted_or %>%
  knitr::kable(caption = "Bias-adjusted odds ratio")
```

3.1.2 Weighting

Alternatively, we can weight the individual records by the inverse probability of selection and use bootstapping to calculate a confidence interval

```
library(boot)

# Add weights
selected_data_with_weights <- selected_data %>%
  mutate(prob_select = a*y*S11 + (1-a)*y*S01 + a*(1-y)*S10 + (1-a)*(1-y)*S00) %>%
  mutate(inverse_prob = 1/prob_select)

# define function to estimate weighted odds ratio (needed for bootstrap function)
calculate_weighted_or <- function(weighted_data, i) {
  weighted_lgr <- glm(y ~ x + a, family="binomial", data=weighted_data[i,], weights=in
  weighted_or <- coef(weighted_lgr)[["a"]] %>% exp()
  return(weighted_or)
}

# set seed of random number generator to ensure reproducibility
set.seed(747)

# bootstrap calculation of confidence intervals
bootstrap_estimates <- boot(selected_data_with_weights, calculate_weighted_or, R=1000)

# calculate bias-adjusted point estimate using entire selected subsample
point_estimate <- calculate_weighted_or(selected_data_with_weights, 1:nrow(selected_da

# calculate percentile bootstrap confidence interval
conf_int <- quantile(bootstrap_estimates$t, c(0.025, 0.975))

# output bias-adjusted estimate
tibble(estimate=point_estimate, conf.low=conf_int[[1]], conf.high=conf_int[[2]]) %>%
  knitr::kable()
```

If we expect selection probabilities to differ within levels of covariates then we can specify different selection probabilities for different strata of covariates.

3.2 Misclassification

We will now consider some quantitative bias analysis methods for misclassification of a binary outcome. Similar approaches can be applied for misclassification of a binary exposure.

With differential misclassification of the outcome, m_y, the odds ratio is biased.

```
# load data
misclassified_data <- read_csv("data/simulated_data.csv") %>% select(x,a,m_y,s)
```

Table 3.4: Estimated odds ratio with misclassified outcome

estimate	conf.low	conf.high
1.737071	1.650733	1.827756

```r
# fit logistic regression model with misclassified outcome
lgr_model_misclassified <- glm(m_y ~ x + a, data=misclassified_data, family="binomial")

# tidy model outputs
or_misclassified <- lgr_model_misclassified %>%
  tidy(exponentiate=TRUE, conf.int=TRUE) %>%
  filter(term == "a") %>%
  select(estimate, conf.low, conf.high)

# output as table
or_misclassified %>%
  knitr::kable(caption = "Estimated odds ratio with misclassified outcome")
```

3.2.1 Bias formulas

Bias formulas for misclassification typically apply to 2x2 tables or 2x2 tables stratified by covariates and require us to specify the bias parameters of sensitivity and specificity. Given that the data was simulated, we know that sensitivity and specificity among the treated were 80% and 99%, and that sensitivity and specificity among the unexposed were 100%. In practice, we do not know these values, but can estimate them using validation studies and specify a range or, for probabilistic bias analysis, distribution of plausible values.

	A.1	A.0
Y*=1	a	b
Y*=0	c	d

```r
# create 2x2 table stratified by confounder
two_by_two <- misclassified_data %>% tabyl(m_y,a, x) %>% adorn_title()

# 2x2 among those without the confounder
print("Without confounder")
```

```
## [1] "Without confounder"
```

```
two_by_two[[1]]
```

```
##          a
##  m_y     0     1
##    0 57291 18078
##    1  2982  1720
```

```
# 2x2 among those with the confounder
print("With confounder")
```

```
## [1] "With confounder"
```

```
two_by_two[[2]]
```

```
##         a
##  m_y    0    1
##    0 9138 8503
##    1  932 1356
```

```
# specify bias parameters
sensitivity_a0 <-  1
sensitivity_a1 <- 0.8
specificity_a0 <- 1
specificity_a1 <- 0.99

# define function to correct 2x2 table
correct_two_by_two <- function(a, b, c, d, sensitivity_a0, sensitivity_a1, specificity_

}

# extract values from 2x2 tables
a_x0 <- two_by_two[[1]][3,3] %>% as.integer()
b_x0 <- two_by_two[[1]][2,3] %>% as.integer()
c_x0 <- two_by_two[[1]][3,2] %>% as.integer()
d_x0 <- two_by_two[[1]][3,3] %>% as.integer()
a_x1 <- two_by_two[[2]][2,2] %>% as.integer()
b_x1 <- two_by_two[[2]][2,3] %>% as.integer()
c_x1 <- two_by_two[[2]][3,2] %>% as.integer()
d_x1 <- two_by_two[[2]][3,3] %>% as.integer()

# correct 2x2 table
correct_two_by_two(a_x0, b_x0, c_x0, d_x0, sensitivity_a0, sensitivity_a1, specificity_
## NULL
```

```
# calculate
```

3.2.2 Record-level correction

3.2.3 Probabalistic bias analysis

3.3 Unmeasured confounding

Bibliography

[1] Jeremy P Brown, John R Tazare, Elizabeth Williamson, Kathryn E Mansfield, Stephen J Evans, Laurie A Tomlinson, Krishnan Bhaskaran, Liam Smeeth, Kevin Wing, and Ian J Douglas. Proton pump inhibitors and risk of all-cause and cause-specific mortality: A cohort study. *British journal of clinical pharmacology*, 87(8):3150–3161, 2021.

[2] Peng Ding and Tyler J VanderWeele. Sensitivity analysis without assumptions. *Epidemiology (Cambridge, Mass.)*, 27(3):368, 2016.

[3] Matthew P Fox, Timothy L Lash, and Sander Greenland. A method to automate probabilistic sensitivity analyses of misclassified binary variables. *International journal of epidemiology*, 34(6):1370–1376, 2005.

[4] Sander Greenland. Basic methods for sensitivity analysis of biases. *International journal of epidemiology*, 25(6):1107–1116, 1996.

[5] Krista F Huybrechts, Kristin Palmsten, Jerry Avorn, Lee S Cohen, Lewis B Holmes, Jessica M Franklin, Helen Mogun, Raisa Levin, Mary Kowal, Soko Setoguchi, et al. Antidepressant use in pregnancy and the risk of cardiac defects. *New England Journal of Medicine*, 370(25):2397–2407, 2014.

[6] Elisabetta Patorno, Krista F Huybrechts, Brian T Bateman, Jacqueline M Cohen, Rishi J Desai, Helen Mogun, Lee S Cohen, and Sonia Hernandez-Diaz. Lithium use in pregnancy and the risk of cardiac malformations. *New England Journal of Medicine*, 376(23):2245–2254, 2017.