rcme: Recounting Crime with Measurement Error - workshop

Brunton-Smith, I., Pina-Sánchez, J., Buil-Gil, D., and Cernat, A

February 06, 2024

Introduction

In this workshop you will get practical experience of using our package - **rcme**: Recounting Crime with Measurement Error - to assess the sensitivity of regression results using recorded crime data. No prior knowledge of R (and RStudio) is required as the workshop proceeds in a step-by-step fashion. However, we have assumed a working knowledge of regression methods.

More details about the package and worked examples can be found in Pina-Sánchez et al., (2023a) and on RecountingCrime website https://recountingcrime.wordpress.com.

Before we begin: Introducing R and RStudio

R is an open source software package that can be used for a very wide range of statistical analyses. You can obtain and install it for free, with versions available for PCs, Macs and Linux. To find out what is available, go to the Comprehensive R Archive Network (CRAN) at http://cran.r-project.org/. Being free is not necessarily a good reason to use R. However, R is also a well developed, documented and supported (by an extensive user community) data analysis software. It is widely used in research, both academic and commercial.

RStudio is a free user-interface for R that makes basic data analysis much more straightforward and user-friendly. In particular, it allows you to see your data, outputs and user commands simultaneously. It can be downloaded from https://rstudio.com/products/rstudio/.

RStudio is not required for this workshop, however it can make data analysis easier. You can find out more about how to use it at https://education.rstudio.com/learn/beginner/.

R is command-line driven. That is, the user types a command that the software interprets and responds to. This may mean that R initially feels a bit daunting, however once you know the commands it is usually much faster to type them than to work through a series of menu options. A log or script of the commands can also be saved for use on another occasion or for sharing with others. All of the R code in the worksheet will be identified in separate boxes that look like this:

```
Example of R code
```

When you see code in one of these boxes you should type the code into the R Console (or a script file). You can also copy the text directly from your browser to save typing!

All the R output that you will get in the R console will be identified in boxes that look like this:

```
## [1] "Example of R output"
```

Downloading and installing rcme

R **packages** are user written programs that vastly increase the capabilities of R, enabling you to conduct almost any form of statistical analysis, as well as create interactive webpages, draw maps, and scrape websites. We have designed **rcme**: Recounting Crime with Measurement Error as an R Package that can be downloaded directly from github. R packages need to first be downloaded onto your computer and saved with R. R refers to this process as installing the package. Because our package is a work in progress, we also need to install `devtools`.

```
install.packages("devtools")
devtools::install_github("RecountingCrime/rcme")
```

Installing a package makes it available for use by R. However, in order to actually use the package it must also be loaded into the current workspace. This is done with the `library()` command.

```
library(rcme)
```

As a reminder, the first time you use a package it must first be downloaded and installed into R using the `install.packages()` command. The downloaded package must then be loaded into the current workspace using the command `library()`. If you come back to RStudio at a later date you do not need to re-install the packages, but you ALWAYS need to load them in using `library()`.

IMPORTANT: It is easy to forget to correctly load packages, particularly if R restarts for some reason. Usually, the main cause of errors when trying to complete these workshops is that one or more packages have not correctly loaded, so it is often a good idea to check this first when things go wrong. You can quickly see which packages you have loaded into your current R session by consulting the Packages window in the bottom right quadrant of RStudio. Loaded packages will have a tick next to them.

Example 1: Violent crime and disorder across Local Authorities

In our first example we will examine the effect of violent crime on disorder across Local Authorities in England and Wales. The data is from a sample of 250 Local Authorities with the included variables simulated to broadly match the data reported in Pina-Sanchez et al. (2023a). The data is included with the **rcme** package and we can view the top few rows by typing:

```
head(crime_disorder)
```

```
##   violent_crime white_british unemployment median_age    disorder
## 1     17.954613     -1.875242  -0.74880766 -0.4089145  0.64666711
## 2     12.177885      2.201486   0.03518811  2.1158555 -0.07387513
## 3      8.141439      1.610787  -0.14918213  1.8627676  1.21494115
## 4     25.822089     -1.074690  -0.32459869 -1.3888129 -0.21920971
## 5     22.655406     -2.114620   0.48229187 -2.2554911  0.92906244
## 6     16.560196     -0.521898  -0.41359409 -1.2104559 -0.69439539
##   log_violent_crime
## 1          2.887847
## 2          2.499622
## 3          2.096967
## 4          3.251230
## 5          3.120399
## 6          2.807002
```

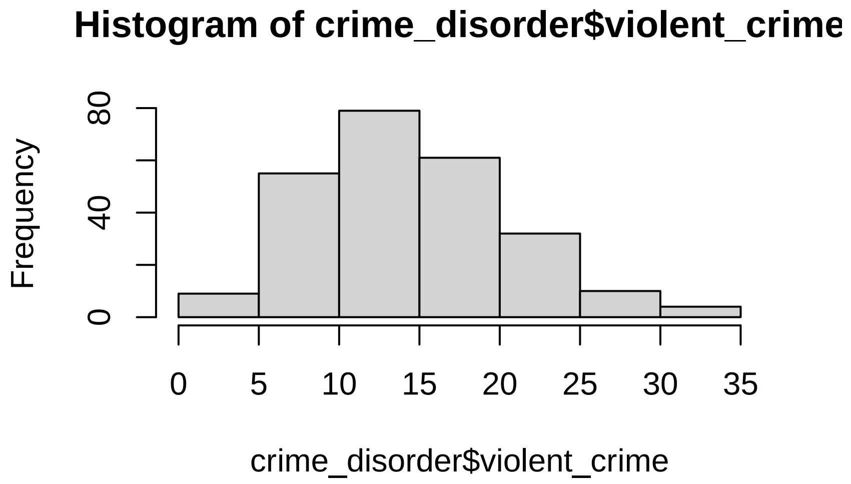
Let's start by taking a summary look at the raw data. Here we can see a mean violent crime rate of 14.5 violent crimes per 1,000 in each Local Authority with a minimum of 4 and maximum of 33.3. The remaining variables have all been standardised with a mean of 0 and standard deviation of 1. These detail the size of the white british population in each LA, the level of unemployment, median age and extent of disorder. Disorder is measured as the area weighted average score from local resident's assessments of the extent of disorderly behaviour with higher scores corresponding to areas with higher levels of disorder.

```
summary(crime_disorder)
```

```
##  violent_crime    white_british       unemployment        median_age
##  Min.   : 4.216   Min.   :-2.85200   Min.   :-2.21630   Min.   :-2.48795
##  1st Qu.: 9.814   1st Qu.:-0.68914   1st Qu.:-0.68375   1st Qu.:-0.71623
##  Median :13.765   Median : 0.07608   Median :-0.05838   Median : 0.02947
##  Mean   :14.529   Mean   : 0.00000   Mean   : 0.00000   Mean   : 0.00000
##  3rd Qu.:18.706   3rd Qu.: 0.69612   3rd Qu.: 0.67639   3rd Qu.: 0.69951
##  Max.   :33.320   Max.   : 2.65133   Max.   : 2.62725   Max.   : 3.09363
##     disorder        log_violent_crime
##  Min.   :-3.1834   Min.   :1.439
##  1st Qu.:-0.6609   1st Qu.:2.284
##  Median : 0.0817   Median :2.622
##  Mean   : 0.0000   Mean   :2.581
##  3rd Qu.: 0.6679   3rd Qu.:2.929
##  Max.   : 3.0735   Max.   :3.506
```

Requesting a histogram of the level of violent crime reveals it is approximately normally distributed.

```
hist(crime_disorder$violent_crime)
```



We start our analysis by estimating a linear regression model exploring the effect of violent crime on levels of neighbourhood disorder, whilst also controlling for levels of unemployment, the percentage of residents that are White British, and the median age. We will save the model results in the object `naive.1` and request the full model output using `summary()`.

```
naive.1 <- lm(disorder ~ violent_crime + white_british + unemployment + median_age,
              data = crime_disorder)
summary(naive.1)
```

```
##
## Call:
## lm(formula = disorder ~ violent_crime + white_british + unemployment +
##     median_age, data = crime_disorder)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.62375 -0.58247  0.00079  0.53063  2.23531
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -0.40192    0.18042  -2.228  0.02681 *
## violent_crime  0.02766    0.01184   2.336  0.02028 *
## white_british -0.08747    0.08334  -1.050  0.29496
## unemployment   0.20937    0.06768   3.093  0.00221 **
## median_age    -0.17909    0.09000  -1.990  0.04770 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8603 on 245 degrees of freedom
## Multiple R-squared:  0.2718, Adjusted R-squared:  0.2599
## F-statistic: 22.86 on 4 and 245 DF,  p-value: 4.613e-16
```

Here we observe the expected positive effect of violent crime on disorder. The effect is statistically significant, but modest in size. Areas of higher unemployment are also identified as having higher disorder, while the older and whiter the area, the lower the level of disorder (although these latter effects do not reach standard levels of statistical significance).

To examine whether the observed effect of violent crime on disorder is robust to the measurement error mechanisms affecting police recorded rates of violent crime (e.g. when violent crime is an independent variable), we use the function `rcme_ind()`. Users must specify the model formula followed by details of the dataset being used. Next we identify the crime variable as our key predictor of interest (`focal_variable`). Users are then required to include values for the expected recording rate (`R`), as well as (optionally) the correlation between the measurement error and focal variable (`D`). For now we will ignore `D`, but we will return to this in our second example.

```
me.1 <- rcme_ind(  #Change when function changed - e.g. logging etc.
  formula = "disorder ~ violent_crime + white_british + unemployment + median_age",
  data = crime_disorder,
  focal_variable = "violent_crime",
  R = c(0.31, 0.46, 0.67))
```

In this example we have selected expected recording rates that broadly match official estimates for the proportion of reported incidents of violence recorded by the police (0.67), the proportion of crimes experienced that are reported to the police as estimated by the crime survey for england and wales (0.46), and the implied overall recording rate (0.67*0.56 = 0.31).

rcme saves the the measurement error adjusted estimate of the effect of violent crime on levels of disorder for all requested values of the expected recorded crime rate. These can be straightforwardly viewed in tabular form, along with the original estimate.

```
me.1
```

```
## $sim_result
##      R D log_var rr focal_variable    SE
## 1 0.31 1   FALSE  1          0.009 0.004
## 2 0.46 1   FALSE  1          0.013 0.005
## 3 0.67 1   FALSE  1          0.019 0.008
##
## $naive
##
## Call:
## lm(formula = paste0(outcome, " ~ ", paste0(c(paste0(focal_variable,
##     collapse = ""), predictors[!predictors %in% focal_variable]),
##     collapse = " + ")), data = data)
##
## Coefficients:
##   (Intercept)  violent_crime  white_british   unemployment     median_age
##      -0.40192        0.02766       -0.08747        0.20937       -0.17909
##
##
## $focal_variable
## [1] "violent_crime"
```

The results included in the columns headed `focal_variable` and `SE` are the error adjusted estimates of the effect of violent crime on disorder for each of the different expected recorded crime rates. These results imply that in the presence of substantial undercounting (a recorded crime rate of 31%) the positive association between violent crime and disorder is severly attenuated. When the focus is solely on under-recording (where R = 0.67), or under-reporting (R = 0.46) the attenuation effect is still present (albeit smaller in magnitude), but the results remain statistically significant.

Importantly, Pina-Sanchez et al (2023b) recommend logging the crime variable in most situations to mitigate some of the more adverse impacts of the multiplicative measurement error form that is expected to affect crime rates. We can see the effect this has on our example by re-estimating the native model with logged crime and then repeating rcme_ind. Note that when requesting logged crime be use in RCME this must be done with the additional command `log_var = T`, not including the `log()` wrapper in the formula.

```
naive.1log <- lm(disorder ~ log(violent_crime) + white_british + unemployment +
                   median_age, data = crime_disorder)
summary(naive.1log)
```

```
##
## Call:
## lm(formula = disorder ~ log(violent_crime) + white_british +
##     unemployment + median_age, data = crime_disorder)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.66107 -0.58347 -0.00198  0.52401  2.19310
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -1.02838    0.40467  -2.541  0.01166 *
## log(violent_crime)  0.39849    0.15539   2.564  0.01093 *
## white_british      -0.08915    0.08241  -1.082  0.28044
## unemployment        0.21015    0.06697   3.138  0.00191 **
## median_age         -0.17004    0.09018  -1.886  0.06054 .
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.8584 on 245 degrees of freedom
## Multiple R-squared:  0.275,  Adjusted R-squared:  0.2632
## F-statistic: 23.23 on 4 and 245 DF,  p-value: 2.705e-16
```

```
me.1log <- rcme_ind(  #Change when function changed - e.g. logging etc.
  formula = "disorder ~ violent_crime + white_british + unemployment + median_age",
  data = crime_disorder,
  focal_variable = "violent_crime",
  R = c(0.31, 0.46, 0.67),
  log_var = T)
me.1log
```

```
## $sim_result
##      R D log_var rr focal_variable    SE
## 1 0.31 1    TRUE  1          0.398 0.155
## 2 0.46 1    TRUE  1          0.398 0.155
## 3 0.67 1    TRUE  1          0.398 0.155
##
## $naive
##
## Call:
## lm(formula = paste0(outcome, " ~ ", paste0(c(paste0("log(", focal_variable,
##     ")", collapse = ""), predictors[!predictors %in% focal_variable]),
##     collapse = " + ")), data = data)
##
## Coefficients:
##        (Intercept)  log(violent_crime)       white_british        unemployment
##           -1.02838             0.39849            -0.08915             0.21015
##         median_age
##           -0.17004
##
##
## $focal_variable
## [1] "violent_crime"
```

Here we confirm the positive association between violent crime and disorder in the naive model. The change in magnitude reflects the change to a log-scale, meaning that a 10% increase in the crime rate is associated with a statistically significant yet modest (0.4 standard deviations) increase in disorder. Importantly, using rcme_ind we can see that this observed effect is robust to under reporting.

Example 2: Criminal Damage across London

In our second example, we consider adjustments when recorded crime is taken as the outcome variable, and allow for the possibility of differential errors. To do so we examine the effect of collective efficacy on levels of crime across London. The data, `crime_damage`, is from a sample of 250 Middle Layer Super Output Areas (MSOA) in London with the included variables simulated to broadly match the data reported in Pina-Sanchez et al. (2023a)

```r
head(crime_damage)
```

```
##   collective_efficacy unemployment median_age white_british damage_crime
## 1         -1.28442575   0.05379451 -0.4016502    -0.2027723    3.9094352
## 2         -1.03925828   1.18204531 -1.1662788    -1.2573142    3.0694682
## 3          0.51861659  -0.81635704  2.0705378     0.8522283    0.4778192
## 4          0.67044665  -0.98987457  1.5649438     0.3077032    2.2011857
## 5          0.58813673   0.18282236 -0.1707982    -0.2032584    2.8591986
## 6         -0.03851433  -0.47728529  1.1711697    -0.5604057    1.2604280
##   log_damage_crime
## 1        1.3633929
## 2        1.1215043
## 3       -0.7385229
## 4        0.7889962
## 5        1.0505414
## 6        0.2314514
```

Examining the raw data we can see a mean criminal damage rate of 2.9 violent crimes per 1,000 in each MSOA. Collective efficacy (measured using Metropolitan Police Public Attitudes Survey data) is a combination of social cohesion and neighbourhood informal social control derived by aggregating assessments to the area level, with higher scores representing areas where residents would be more likely to intervene in the presence of disorder and crime. The remaining variables are the same as in the first example (albeit measured in each MSOA rather than Local Authority) and have again been standardised with a mean of 0.

```r
summary(crime_damage)
```

```
##  collective_efficacy  unemployment        median_age        white_british     
##  Min.   :-3.10657    Min.   :-2.26315   Min.   :-2.47902   Min.   :-2.59402  
##  1st Qu.:-0.61599    1st Qu.:-0.71732   1st Qu.:-0.69115   1st Qu.:-0.73960  
##  Median : 0.01054    Median :-0.07106   Median :-0.02687   Median : 0.08378  
##  Mean   : 0.00000    Mean   : 0.00000   Mean   : 0.00000   Mean   : 0.00000  
##  3rd Qu.: 0.69989    3rd Qu.: 0.69868   3rd Qu.: 0.62289   3rd Qu.: 0.72843  
##  Max.   : 2.40057    Max.   : 2.55476   Max.   : 2.49421   Max.   : 2.51350  
##   damage_crime    log_damage_crime 
##  Min.   :0.4778   Min.   :-0.7385  
##  1st Qu.:2.0265   1st Qu.: 0.7063  
##  Median :2.8164   Median : 1.0355  
##  Mean   :2.8682   Mean   : 0.9546  
##  3rd Qu.:3.6483   3rd Qu.: 1.2943  
##  Max.   :6.4906   Max.   : 1.8704  
```

As before, we start by estimating our outcome model. Here we include our variable of interest (collective efficacy) as well as controls for the area unemployment rate, proprtion white british, and age structure of the area.

```r
naive.2 <- lm(damage_crime ~ collective_efficacy + unemployment + white_british +
                median_age, data = crime_damage)
summary(naive.2)
```

```
##
## Call:
## lm(formula = damage_crime ~ collective_efficacy + unemployment +
##     white_british + median_age, data = crime_damage)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.7415 -0.6522 -0.0311  0.7086  2.3915
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          2.86820    0.06223  46.092  < 2e-16 ***
## collective_efficacy -0.27791    0.08343  -3.331 0.000999 ***
## unemployment         0.26966    0.09678   2.786 0.005749 **
## white_british        0.35622    0.08754   4.069 6.37e-05 ***
## median_age          -0.40489    0.08947  -4.526 9.39e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9839 on 245 degrees of freedom
## Multiple R-squared:  0.2973, Adjusted R-squared:  0.2859
## F-statistic: 25.92 on 4 and 245 DF,  p-value: < 2.2e-16
```

Our focus is on the effect of collective efficacy on police recorded criminal damage, where we observe the expected negative association. Areas that are higher in collective efficacy generally experience lower levels of criminal damage. Criminal damage is more prevalent in areas with higher unemployment rates and where a larger share of the population are white, but lower in areas characterised by an older population.

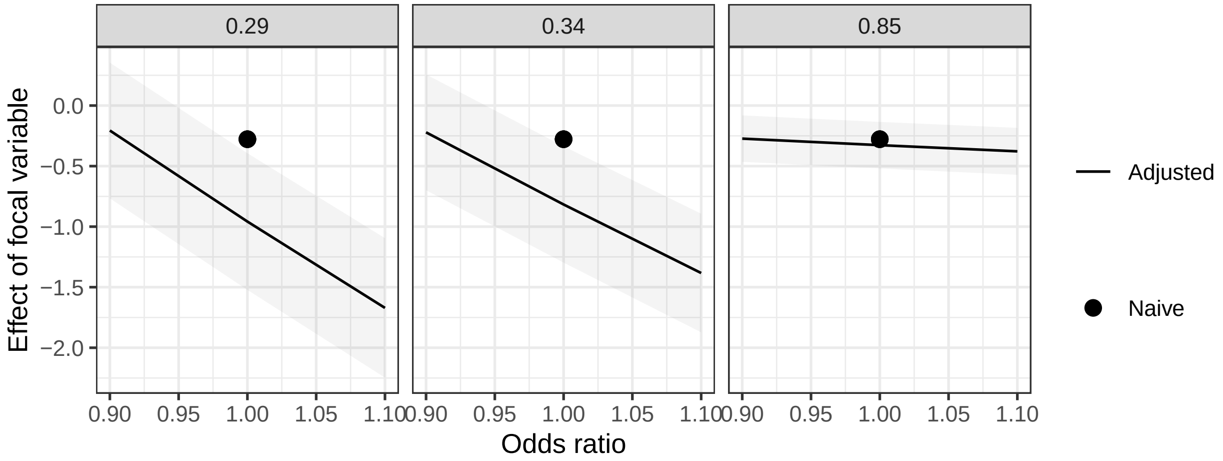
When crime is the outcome variable we use the functon `rcme_out()`. This time we specify `collective_efficacy` as the `focal_variable` since we are interested in understanding the extent that the estimated relationship with recorded crime may be biased as a result of measurement error in crime. For the case of criminal damage, reporting (34%) and recording (86%) rates differ markedly from violent crime (ONS, 2020; Her Majesty Inspectorate of Constabulary, 2014: 65). However, the overall counting rate, after considering the share of crimes reported that are recorded, is remarkably similar at 28.9% (0.86*0.34). We therefore retain a similar minimum expeced recording rate, `R`, of 0.29 , selecting additional values of 0.34 and 0.86.

To incorporate differential errors, we also need to provide a plausible range of estimates for the association, `D`, between the systematic error in police recorded violent crime rates and our focal predictor (collective efficacy). This should be included as an odds ratio, with values below 1 reflecting a negative association between the systematic errors and collective efficacy, and values above 1 representing a positive association. Selection of appropriate values for this association is complex, with Pina-Sanchez et al. (2022b) setting out a one plausible strategy for generating concrete estimates of this association using a combination of survey data and census statistics. However, in the absence of a plausible alternative, we can simply select values of 0.9, 1, and 1.1 to explore the sensitivity of our main result to modest positive or negative associations.

```r
me.2 <- rcme_out( #Update when package finished - logging etc.
  formula = "damage_crime ~ collective_efficacy + unemployment + white_british
  + median_age",
  data = crime_damage,
  focal_variable = "collective_efficacy",
  R = c(0.29, 0.34, 0.85),
  D = c(0.9, 1, 1.1),
  log_var=F)
```

We could explore the raw data in tabular format like the first example. However, with the added complexity of including differential errors it is generally easier to examine the measurement error effect visually. This can be done with the command `rcme_sim_plot()`.

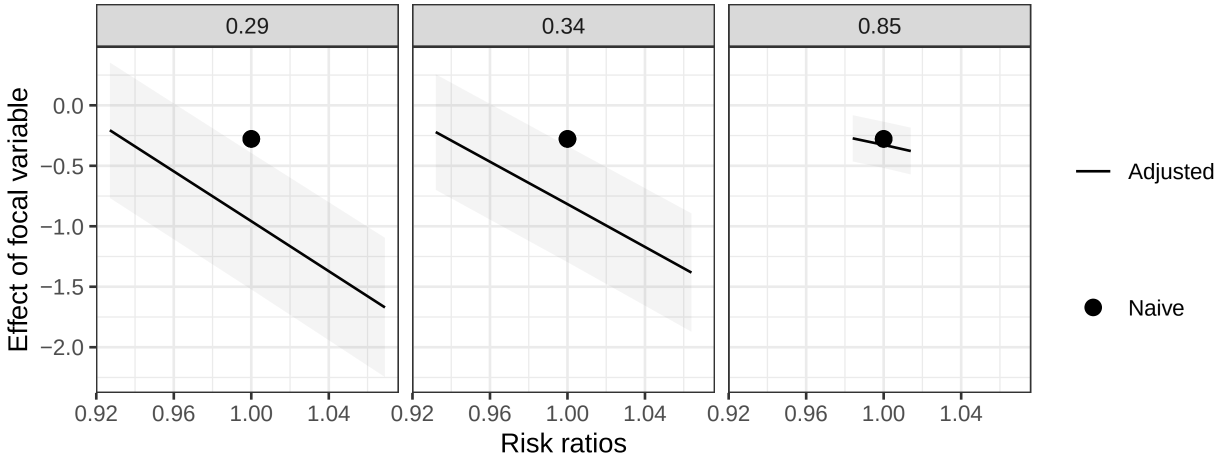
```
rcme_sim_plot(me.2)
```



The three panels show the range of possible values for the association between collective efficacy and criminal damage (represented on the vertical axis), as the association between between collective efficacy and measurement error changes from positive to negative (an odds ratio of 1 means no association) for the three counting rates considered (0.29, 0.34 and 0.86). Here we find that the association between collective efficacy and criminal damage is highly likely to be negative, with all three graphs showing the adjusted value of the effect of collective efficacy remains below zero. This is true across all values of the differential errors. If we assume that there is no differential error (an odds ratio of 1) our sensitivity analysis suggests that the naive estimate of the effect of collective efficacy is likely to be an underestimate of the true value, particularly if the assumed recording rate is low. But the effect of measurement error may be even more complex if we are willing to assume that there is differential error. Here, when the recording rate is low, the expected coefficient estimate is also highly sensitive to the presence of differential error. The effect of collective efficacy might be even more negative than the native model estimate if the association of the error with collective efficacy is positive. Conversely, if the association with collective efficacy is negative, the gap between the naive estimate and the possible true estimate gets smaller. However, if we believe the expected recording rate is high (the rightmost graph), then differential errors appear less pronounced.

We can also display the results using risk ratios by including the command (`rr = T`) in our call to rcme_sim_plot. Note that whilst odds ratios have a common scaling across all values of the recording rate, risk ratios are always interpreted as *proportional* to the recording rate. As a result, the range of values for each of the plots may be differnt. For a full discussion of risk ratios and their interpretation, see https://statisticsbyjim.com/probability/relative-risk/.

```
rcme_sim_plot(me.2, rr = T)
```

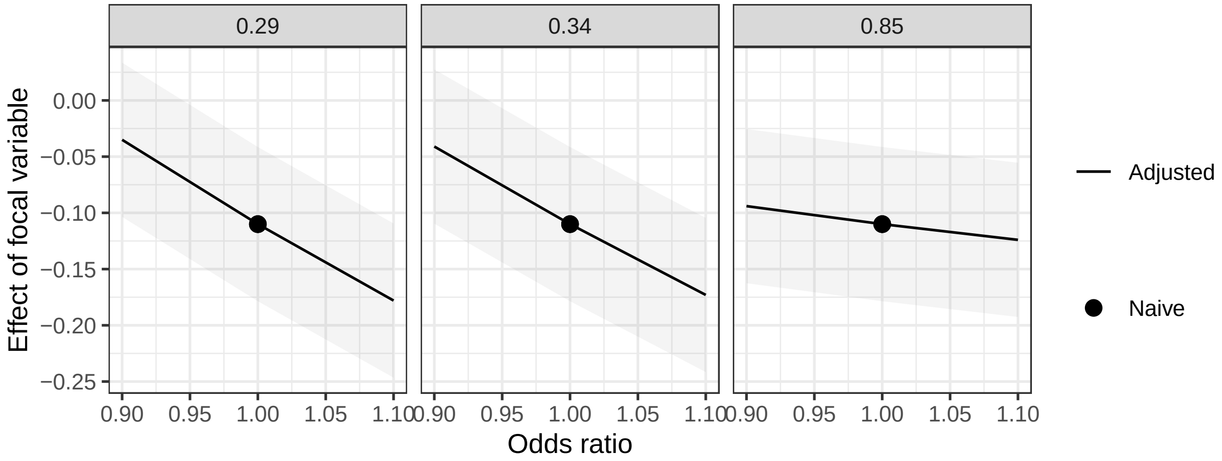


And once again, we log the crime variable as this is expected to mitigate some of the more adverse impacts of the multiplicative measurement error form present in crime rates.

```
naive.2log <- lm(log(damage_crime) ~ collective_efficacy + unemployment + white_british +
                    median_age, data = crime_damage)
summary(naive.2log)
```

```
##
## Call:
## lm(formula = log(damage_crime) ~ collective_efficacy + unemployment +
##     white_british + median_age, data = crime_damage)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.47667 -0.19247  0.05156  0.29677  0.69409
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          0.95460    0.02606  36.637  < 2e-16 ***
## collective_efficacy -0.11004    0.03493  -3.150 0.001835 **
## unemployment         0.08025    0.04052   1.980 0.048786 *
## white_british        0.13076    0.03666   3.567 0.000434 ***
## median_age          -0.17582    0.03746  -4.693 4.47e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.412 on 245 degrees of freedom
## Multiple R-squared:  0.2694, Adjusted R-squared:  0.2575
## F-statistic: 22.59 on 4 and 245 DF,  p-value: 6.758e-16
```

```
me.2log <- rcme_out( #Update when package finished - logging etc.
  formula = "damage_crime ~ collective_efficacy + unemployment + white_british
  + median_age",
  data = crime_damage,
  focal_variable = "collective_efficacy",
  R = c(0.29, 0.34, 0.85),
  D = c(0.9, 1, 1.1),
  log_var=T)
rcme_sim_plot(me.2log)
```



References

Pina-Sánchez, J., Brunton-Smith, I., Buil-Gil, D., and Cernat, A. (2023a) 'Exploring the Impact of Measurement Error in Police Recorded Crime Rates through Sensitivity Analysis'. Crime Science. 12(14).

Pina-Sánchez, J., Buil-Gil, D., Brunton-Smith, I., and Cernat, A. (2023b) 'The Impact of Measurement Error in Models Using Police Recorded Crime Rates'. Journal of Quantitative Criminology. 39: 975-1002.