

Calculating missing values

Spatially targeted systematic review

McEwen Khundi

London School of Hygiene & Tropical Medicine, London, UK

Malawi-Liverpool-Wellcome Clinical Research Programme, Blantyre, Malawi

March 11, 2020

Table of Contents:

Contents

1. Background	1
2. Set-up	1
3. Define functions	1
4. Goswami2012 paper HIV	2
5. Goswami2012 paper TB	2
6. Fatima2016 paper TB	3
7. Barreto2015 Leprosy	3
6. Srivastava 2009 paper Malaria	4
8. Morris 2018 Malaria	4
X. Reproducibility	5

1. Background

2. Set-up

Load all required packages for analysis.

3. Define functions

```
prop_CI_to_n <- function(prop,c1,c2){  
  n = round((prop*(1-prop))/(((c2-c1)/4)^2))  
}  
  
prop_std_error <- function(prop,n){  
  std_er = ((prop*(1-prop)/n))^(1/2)  
}  
  
prop_to_CI <- function(prop, prop_stderror){  
  low_ci = prop - 2*prop_stderror  
  upper_ci = prop + 2*prop_stderror  
  list(low_ci,upper_ci)  
}  
  
poisson_to_prop <- function(x1,x2){  
  prop = x1/(x1 + x2)  
}  
  
relative_riskratio <- function(a,b,c,d){
```

```

rr = (a/(a+b))/(c/(c+d))
}
relative_riskratio_stderror <- function(a,b,c,d){
  rr_stder = sqrt((1/a) + (1/c) - (1/(a+b)) - (1/(c+d)))
}

rr_risk_CI <- function(rr,rr_stder){
  lowerCI = exp(log(rr)-1.96*rr_stder)
  upperCI = exp(log(rr) + 1.96*rr_stder)

  list(lowerCI,upperCI)
}

```

4. Goswami2012 paper HIV

```

N_total = prop_CI_to_n(0.004,0.005,0.003)
n_cases = round(N_total*0.004)
glue::glue("Total number of cases N:{N_total} ", "Number of positive cases n:{n_cases}")

## Total number of cases N:15936 Number of positive cases n:64
#Calculation for RISK Ratios and RR
rr = relative_riskratio(a=8,b=(240-8),c=64, d=(15936-64))
rr_stder = relative_riskratio_stderror(a=8,b=(240-8),c=64, d=(15936-64))

lowCI_upperCI = rr_risk_CI(rr,rr_stder)

glue::glue("The risk ratio:{rr}, test statistic:{rr/rr_stder}")

## The risk ratio:8.3, test statistic:22.4738739251008
glue::glue("The risk ratio:{rr}, 95%CI lowerCI:{lowCI_upperCI[[1]]}, upperCI:{lowCI_upperCI[[2]]}")

## The risk ratio:8.3, 95%CI lowerCI:4.0244681799129, upperCI:17.1177897104136

```

5. Goswami2012 paper TB

```

N_total = prop_CI_to_n(0.06,0.066,0.056)
n_cases = round(N_total*0.06)
glue::glue("Total number of cases N:{N_total} ", "Number of positive cases n:{n_cases}")

## Total number of cases N:9024 Number of positive cases n:541
#Calculation for RISK Ratios and RR
rr = relative_riskratio(a=36,b=(234-36),c=541, d=(9024-541))
rr_stder = relative_riskratio_stderror(a=36,b=(234-36),c=541, d=(9024-541))

lowCI_upperCI = rr_risk_CI(rr,rr_stder)

glue::glue("The risk ratio:{rr}, test statistic:{rr/rr_stder}")

## The risk ratio:2.56618797099389, test statistic:16.1520465679158
glue::glue("The risk ratio:{rr}, 95%CI lowerCI:{lowCI_upperCI[[1]]}, upperCI:{lowCI_upperCI[[2]]}")

## The risk ratio:2.56618797099389, 95%CI lowerCI:1.87953176897249, upperCI:3.50370278980375

```

6. Fatima2016 paper TB

```
prop = poisson_to_prop(100384,108341)
prop

## [1] 0.480939

prop_std = prop_std_error(prop, (100384 +108341))
prop_std

## [1] 0.001093621

glue::glue("The prop:{prop} prop std error: {prop_std} the test statistic {(0.5-prop)/prop_std}")

## The prop:0.480939034614924 prop std error: 0.00109362130915856 the test statistic 17.4292190774353

low_upper_ci = prop_to_CI(prop,prop_std)

glue::glue("The prop: {1/(prop)-1}, lowerCI: {1/(low_upper_ci[[1]])-1}, upperCI: {1/(low_upper_ci[[2]])}")

## The prop: 1.07926562001913, lowerCI: 1.08876502755124, upperCI: 1.0698522252335
```

7. Barreto2015 Leprosy

```
#Start with the hotspot children
prop = 11/134

prop_std = prop_std_error(prop, 134)
prop_std

## [1] 0.02371328

glue::glue("The prop:{prop} prop std error: {prop_std}")

## The prop:0.082089552238806 prop std error: 0.0237132838979372

low_upper_ci = prop_to_CI(prop,prop_std)

glue::glue("The prop%: {100*prop}, lowerCI%: {100*low_upper_ci[[1]]}, upperCI%: {100*low_upper_ci[[2]]}")

## The prop%: 8.2089552238806, lowerCI%: 3.46629844429316, upperCI%: 12.951612003468

#Second with the randomly selected children children
prop = 63/1592

prop_std = prop_std_error(prop, 1592)
prop_std

## [1] 0.004886067

glue::glue("The prop:{prop} prop std error: {prop_std}")

## The prop:0.039572864321608 prop std error: 0.00488606704957417

low_upper_ci = prop_to_CI(prop,prop_std)

glue::glue("The prop%: {100*prop}, lowerCI%: {100*low_upper_ci[[1]]}, upperCI%: {100*low_upper_ci[[2]]}")
```

```
## The prop%: 3.9572864321608, lowerCI%: 2.98007302224597, upperCI%: 4.93449984207564
#Calculation for RISK Ratios and RR
rr = relative_riskratio(a=11,b=(134-11),c=63, d=(1592-63))
rr_stder = relative_riskratio_stderror(a=11,b=(134-11),c=63, d=(1592-63))

lowCI_upperCI = rr_risk_CI(rr,rr_stder)

glue::glue("The risk ratio:{rr}, test statistic:{rr/rr_stder}")

## The risk ratio:2.07438995498697, test statistic:6.60314768838653
glue::glue("The risk ratio:{rr}, 95%CI lowerCI:{lowCI_upperCI[[1]]}, upperCI:{lowCI_upperCI[[2]]}")

## The risk ratio:2.07438995498697, 95%CI lowerCI:1.12067346820729, upperCI:3.83973905640364
```

6. Srivastava 2009 paper Malaria

```
prop = poisson_to_prop(90829,96042)
prop

## [1] 0.4860519
prop_std = prop_std_error(prop, (90829 +96042))
prop_std

## [1] 0.001156192
glue::glue("The prop:{prop} prop std error: {prop_std} the test statistic {(0.5-prop)/prop_std}")

## The prop:0.486051875357867 prop std error: 0.00115619210527176 the test statistic 12.0638469840221
low_upper_ci = prop_to_CI(prop,prop_std)

glue::glue("The prop: {1/(prop)-1}, lowerCI: {1/(low_upper_ci[[1]])-1}, upperCI: {1/(low_upper_ci[[2]])-1}")

## The prop: 1.05739356372965, lowerCI: 1.06722837043595, upperCI: 1.04765189181482
```

8. Morris 2018 Malaria

```
#Start with the control clusters
prop = 52/12307

prop_std = prop_std_error(prop, 12307)
prop_std

## [1] 0.0005846959
glue::glue("The prop:{prop} prop std error: {prop_std}")

## The prop:0.00422523766961892 prop std error: 0.000584695873375013
low_upper_ci = prop_to_CI(prop,prop_std)

glue::glue("The prop per 1000: {1000*prop}, lowerCI per 1000: {1000*low_upper_ci[[1]]}, upperCI per 1000: {1000*low_upper_ci[[2]]}")

## The prop per 1000: 4.22523766961892, lowerCI per 1000: 3.05584592286889, upperCI per 1000: 5.3946294
```

```

#The intervention clusters
prop = 43/10944

prop_std = prop_std_error(prop, 10944)
prop_std

## [1] 0.0005980029

glue::glue("The prop:{prop} prop std error: {prop_std}")

## The prop:0.00392909356725146 prop std error: 0.000598002873774306

low_upper_ci = prop_to_CI(prop,prop_std)

glue::glue("The prop per 1000: {1000*prop}, lowerCI per 1000: {1000*low_upper_ci[[1]]}, upperCI per 1000: {1000*low_upper_ci[[2]]}")

## The prop per 1000: 3.92909356725146, lowerCI per 1000: 2.73308781970285, upperCI per 1000: 5.1250993

```

X. Reproducibility

This reproduction of the analysis was run by:

keyName	value
sysname	Windows
release	10 x64
version	build 17134
nodename	LAPT2176
machine	x86-64
login	lsh1805281
user	lsh1805281
effective_user	lsh1805281

Analysis was run at **2020-03-11 16:04:02**, and using the following Session Info:

```

R version 3.6.1 (2019-07-05)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 10 x64 (build 17134)

```

Matrix products: default

```

locale:
[1] LC_COLLATE=English_United Kingdom.1252
[2] LC_CTYPE=English_United Kingdom.1252
[3] LC_MONETARY=English_United Kingdom.1252
[4] LC_NUMERIC=C
[5] LC_TIME=English_United Kingdom.1252

```

attached base packages:

```

[1] stats      graphics  grDevices  utils      datasets  methods   base

```

other attached packages:

```

[1] knitr_1.23          pmthemes_0.0.0.9000 forcats_0.4.0
[4] stringr_1.4.0       dplyr_0.8.99.9000  purrr_0.3.3
[7] readr_1.3.1         tidyr_1.0.2        tibble_2.99.99.9014

```

[10] ggplot2_3.2.1 tidyverse_1.3.0

loaded via a namespace (and not attached):

[1] tidyselect_1.0.0.9000	xfun_0.8	haven_2.2.0
[4] lattice_0.20-38	colorspace_1.4-1	vctrs_0.2.99.9005
[7] generics_0.0.2	htmltools_0.3.6	yaml_2.2.0
[10] rlang_0.4.4	pillar_1.4.3.9000	withr_2.1.2
[13] glue_1.3.1	DBI_1.0.0	dbplyr_1.4.2
[16] modelr_0.1.5	readxl_1.3.1	lifecycle_0.1.0
[19] munsell_0.5.0	gtable_0.3.0	cellranger_1.1.0
[22] rvest_0.3.5	evaluate_0.14	fansi_0.4.0
[25] highr_0.8	broom_0.5.5	Rcpp_1.0.2
[28] backports_1.1.4	scales_1.0.0	jsonlite_1.6
[31] lobstr_1.1.1	fs_1.3.1	hms_0.5.3
[34] digest_0.6.20	stringi_1.4.3	grid_3.6.1
[37] cli_2.0.2	tools_3.6.1	magrittr_1.5
[40] lazyeval_0.2.2	crayon_1.3.4	pkgconfig_2.0.2
[43] zeallot_0.1.0	xml2_1.2.2	reprex_0.3.0
[46] lubridate_1.7.4	assertthat_0.2.1	rmarkdown_1.14
[49] httr_1.4.1	rstudioapi_0.11	R6_2.4.0
[52] nlme_3.1-140	compiler_3.6.1	