Metafor for LLM Numerical Data Extraction (Case Study)

Hye Sun Yun

2024-04-15

We manually extract the relevant information and then plug the values into Metafor's escalc function. Then, we fit random-effects model (fixed effects) and create a forest plot for visualization.

Metafor's escalc

Reference: https://wviechtb.github.io/metafor/reference/escalc.html

Function to calculate various effect sizes or outcome measures (and the corresponding sampling variances) that are commonly used in meta-analyses.

```
escalc(measure, ai, bi, ci, di, n1i, n2i, x1i, x2i, t1i, t2i,
       m1i, m2i, sd1i, sd2i, xi, mi, ri, ti, fi, pi, sdi, r2i, ni, yi, vi, sei,
       data, slab, subset, include,
       add=1/2, to="only0", drop00=FALSE, vtype="LS",
       var.names=c("yi","vi"), add.measure=FALSE,
       append=TRUE, replace=TRUE, digits, ...)
```

Metafor's rma

Reference: https://wviechtb.github.io/metafor/reference/rma.uni.html

Function to fit meta-analytic equal-, fixed-, and random-effects models and (mixed-effects) meta-regression models using a linear (mixed-effects) model framework. In this work, we do a standard fixed-effects models.

```
rma(yi, vi, sei, weights, ai, bi, ci, di, n1i, n2i, x1i, x2i, t1i, t2i,
        m1i, m2i, sd1i, sd2i, xi, mi, ri, ti, fi, pi, sdi, r2i, ni, mods, scale,
        measure="GEN", intercept=TRUE, data, slab, subset,
        add=1/2, to="only0", drop00=FALSE, vtype="LS",
        method="REML", weighted=TRUE, test="z",
        level=95, btt, att, tau2, verbose=FALSE, digits, control, ...)
```

Metafor's forest

Reference: https://wviechtb.github.io/metafor/reference/forest.html

Function to create forest plots.

```
forest(x, vi, sei, ci.lb, ci.ub,
       annotate=TRUE, showweights=FALSE, header=FALSE,
       xlim, alim, olim, ylim, at, steps=5,
       level=95, refline=0, digits=2L, width,
       xlab, slab, ilab, ilab.xpos, ilab.pos,
       order, subset, transf, atransf, targs, rows,
       efac=1, pch, psize, plim=c(0.5,1.5), col,
       shade, colshade, lty, fonts, cex, cex.lab, cex.axis, ...)
```

Import package

Import relevant package:

```
# install.packages("metafor")
# load metafor package
library(metafor)
```

```
## Warning: package 'metafor' was built under R version 4.3.2
```

Case Study

We calculate the log odds ratios, fit meta-analytic fixed-effects model, and create the forest plots for cochrane reference meta anlayses and outputs from two LLMs (GPT-4 and Mistral 7B Instruct).

Cochrane

```
### Get the data
dat <- read.csv("files/cochrane_binary_outcomes.csv")
```

```
## Warning in read.table(file = file, header = header, sep = sep, quote = quote, :
## incomplete final line found by readTableHeader on
## 'files/cochrane_binary_outcomes.csv'
```

```
### calculate log odds ratios and corresponding sampling variances (and use
### the 'slab' argument to store study labels as part of the data frame)
dat <- escalc(measure="OR", ai=ai, n1i=n1i, ci=ci, n2i=n2i, data=dat,
              slab=paste(author, year, sep=", "), drop00=TRUE)
dat
```

```
##
##    author year  ai  n1i  ci  n2i      yi     vi
## 1 WHO STC 2021 285 2743 289 2708 -0.0299 0.0078
## 2 Spinner 2020   3  193   4  200 -0.2566 0.5937
## 3  Beigel 2020  59  541  77  521 -0.3484 0.0343
## 4    Wang 2020  22  158  10   78  0.0953 0.1675
```

```
### fit random-effects model (fixed effects)
res <- rma(yi, vi, data=dat, method = "FE")
res
```

```
##
## Fixed-Effects Model (k = 4)
##
## I^2 (total heterogeneity / total variability):   0.00%
## H^2 (total variability / sampling variability):  0.89
##
## Test for Heterogeneity:
## Q(df = 3) = 2.6575, p-val = 0.4475
##
## Model Results:
##
## estimate      se     zval    pval    ci.lb   ci.ub
##  -0.0840  0.0778  -1.0794  0.2804  -0.2365  0.0685
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
### estimated average odds ratio (and 95% CI/PI)
pred <- predict(res, transf=exp, digits=2)
pred
```

```
##
##  pred ci.lb ci.ub
##  0.92  0.79  1.07
```

```r
############################################################################

### need the rounded estimate and CI bounds further below
pred <- fmtx(c(pred$pred, pred$ci.lb, pred$ci.ub), digits=2)

### total number of studies
k <- nrow(dat)

### get the weights and format them as will be used in the forest plot
weights <- paste0(fmtx(weights(res), digits=1), "%")
weights[weights == "NA%"] <- ""

### adjust the margins
par(mar=c(13.8,0,1.3,0), mgp=c(2,0.2,0), tcl=-0.2)

pdf(width=8.8,height=3.2,file="forest_plots/cochrane_forest_plot.pdf")

### forest plot with extra annotations
sav <- forest(res, atransf=exp, at=log(c(0.01, 0.10, 1, 10, 100)), xlim=c(-30,11),
       xlab="", efac=c(0,4), textpos=c(-30,-4.7), lty=c(1,1,0), refline=NA,
       ilab=cbind(ai, n1i, ci, n2i, weights),
       ilab.xpos=c(-20.6,-18.6,-16.1,-14.1,-10.8), ilab.pos=2,
       cex=0.78, header=c("Study"), mlab="")

### add horizontal line at the top
segments(sav$xlim[1], k+1, sav$xlim[2]-5, k+1, lwd=0.8)

### add vertical reference line at 0
segments(0, -2, 0, k+1, lwd=0.8)

### now we add a bunch of text; since some of the text falls outside of the
### plot region, we set xpd=NA so nothing gets clipped
par(xpd=NA)

### adjust cex as used in the forest plot and use a bold font
par(cex=sav$cex, font=2)

text(sav$ilab.xpos, k+2, pos=2, c("Events","Total","Events","Total", "Weight"))
text(c(-19.1,-15.1), k+3, pos=2, c("Remdesivir","Control"))
text(0, k+3, "Odds ratio, 95% CI")

### use a non-bold font for the rest of the text
par(cex=sav$cex, font=1)

text(c(sav$xlim[1],sav$ilab.xpos[c(2,4,5)]), -1, pos=c(4,2,2,2,2),
     c("Total (95% CI)", sum(dat$n1i), sum(dat$n2i), "100.0%"))
```

```
### add 'Favors remdesivir'/'Favors control' text below the x-axis
text(log(c(0.01, 100)), -3, c("Favors remdesivir","Favors control"), pos=c(4,2), offset=-1)

### add text for total events
text(sav$ilab.xpos[c(1,3)], -1, c(sum(dat$ai),sum(dat$ci)), pos=2)

dev.off
```

```
## function (which = dev.cur())
## {
##     if (which == 1)
##         stop("cannot shut down device 1 (the null device)")
##     .External(C_devoff, as.integer(which))
##     dev.cur()
## }
## <bytecode: 0x7fd8844be958>
## <environment: namespace:grDevices>
```

GPT-4

```
### Get the data
dat <- read.csv("files/gpt4_binary_outcomes.csv")
```

```
## Warning in read.table(file = file, header = header, sep = sep, quote = quote, :
## incomplete final line found by readTableHeader on
## 'files/gpt4_binary_outcomes.csv'
```

```
dat
```

```
##    author year  ai  n1i  ci  n2i
## 1 WHO STC 2021 301 2743 303 2708
## 2 Spinner 2020   5  396   4  200
## 3  Beigel 2020  59  541  77  521
## 4    Wang 2020  22  158  10   78
```

```
### calculate log odds ratios and corresponding sampling variances (and use
### the 'slab' argument to store study labels as part of the data frame)
dat <- escalc(measure="OR", ai=ai, n1i=n1i, ci=ci, n2i=n2i, data=dat,
              slab=paste(author, year, sep=", "), drop00=TRUE)
dat
```

```
##
##    author year  ai  n1i  ci  n2i      yi     vi
## 1 WHO STC 2021 301 2743 303 2708 -0.0219 0.0074
## 2 Spinner 2020   5  396   4  200 -0.4674 0.4577
## 3  Beigel 2020  59  541  77  521 -0.3484 0.0343
## 4    Wang 2020  22  158  10   78  0.0953 0.1675
```

```
### fit random-effects model (fixed effects)
res <- rma(yi, vi, data=dat, method = "FE")
res
```

```
##
## Fixed-Effects Model (k = 4)
##
## I^2 (total heterogeneity / total variability):   2.19%
## H^2 (total variability / sampling variability):  1.02
##
## Test for Heterogeneity:
## Q(df = 3) = 3.0670, p-val = 0.3814
##
## Model Results:
##
## estimate      se     zval    pval    ci.lb   ci.ub
##  -0.0790  0.0763  -1.0351  0.3006  -0.2286  0.0706
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
### estimated average odds ratio (and 95% CI/PI)
pred <- predict(res, transf=exp, digits=2)
pred
```

```
##
##  pred ci.lb ci.ub
##  0.92  0.80  1.07
```

```r
############################################################################

### need the rounded estimate and CI bounds further below
pred <- fmtx(c(pred$pred, pred$ci.lb, pred$ci.ub), digits=2)

### total number of studies
k <- nrow(dat)

### get the weights and format them as will be used in the forest plot
weights <- paste0(fmtx(weights(res), digits=1), "%")
weights[weights == "NA%"] <- ""

### adjust the margins
par(mar=c(13.8,0,1.3,0), mgp=c(2,0.2,0), tcl=-0.2)

pdf(width=8.8,height=3.2,file="forest_plots/gpt4_forest_plot.pdf")

### forest plot with extra annotations
sav <- forest(res, atransf=exp, at=log(c(0.01, 0.10, 1, 10, 100)), xlim=c(-30,11),
       xlab="", efac=c(0,4), textpos=c(-30,-4.7), lty=c(1,1,0), refline=NA,
       ilab=cbind(ai, n1i, ci, n2i, weights),
       ilab.xpos=c(-20.6,-18.6,-16.1,-14.1,-10.8), ilab.pos=2,
       cex=0.78, header=c("Study"), mlab="")

### add horizontal line at the top
segments(sav$xlim[1], k+1, sav$xlim[2]-5, k+1, lwd=0.8)

### add vertical reference line at 0
segments(0, -2, 0, k+1, lwd=0.8)

### now we add a bunch of text; since some of the text falls outside of the
### plot region, we set xpd=NA so nothing gets clipped
par(xpd=NA)
```

```
### adjust cex as used in the forest plot and use a bold font
par(cex=sav$cex, font=2)

text(sav$ilab.xpos, k+2, pos=2, c("Events","Total","Events","Total", "Weight"))
text(c(-19.1,-15.1), k+3, pos=2, c("Remdesivir","Control"))
text(0, k+3, "Odds ratio, 95% CI")

### use a non-bold font for the rest of the text
par(cex=sav$cex, font=1)

text(c(sav$xlim[1],sav$ilab.xpos[c(2,4,5)]), -1, pos=c(4,2,2,2,2),
     c("Total (95% CI)", sum(dat$n1i), sum(dat$n2i), "100.0%"))

### add 'Favors remdesivir'/'Favors control' text below the x-axis
text(log(c(0.01, 100)), -3, c("Favors remdesivir","Favors control"), pos=c(4,2), offset=-1)

### add text for total events
text(sav$ilab.xpos[c(1,3)], -1, c(sum(dat$ai),sum(dat$ci)), pos=2)

dev.off
```

```
## function (which = dev.cur())
## {
##     if (which == 1)
##         stop("cannot shut down device 1 (the null device)")
##     .External(C_devoff, as.integer(which))
##     dev.cur()
## }
## <bytecode: 0x7fd8844be958>
## <environment: namespace:grDevices>
```

Mistral Instruct 7B

```
### Get the data
dat <- read.csv("files/mistral7B_binary_outcomes.csv")
```

```
## Warning in read.table(file = file, header = header, sep = sep, quote = quote, :
## incomplete final line found by readTableHeader on
## 'files/mistral7B_binary_outcomes.csv'
```

```
dat
```

```
##    author year  ai  n1i  ci  n2i
## 1 WHO STC 2021 301 2743 303 2708
## 2 Spinner 2020   2  197   4  200
## 3  Beigel 2020  59  541  77  521
## 4    Wang 2020  22  158  10   78
```

```
### calculate log odds ratios and corresponding sampling variances (and use
### the 'slab' argument to store study labels as part of the data frame)
dat <- escalc(measure="OR", ai=ai, n1i=n1i, ci=ci, n2i=n2i, data=dat,
              slab=paste(author, year, sep=", "), drop00=TRUE)
dat
```

```
##
##    author year  ai  n1i  ci  n2i      yi     vi
## 1 WHO STC 2021 301 2743 303 2708 -0.0219 0.0074
## 2 Spinner 2020   2  197   4  200 -0.6880 0.7602
## 3  Beigel 2020  59  541  77  521 -0.3484 0.0343
## 4    Wang 2020  22  158  10   78  0.0953 0.1675
```

```
### fit random-effects model (fixed effects)
res <- rma(yi, vi, data=dat, method = "FE")
res
```

```
##
## Fixed-Effects Model (k = 4)
##
## I^2 (total heterogeneity / total variability):   6.98%
## H^2 (total variability / sampling variability):  1.08
##
## Test for Heterogeneity:
## Q(df = 3) = 3.2252, p-val = 0.3582
##
## Model Results:
##
## estimate      se     zval    pval    ci.lb   ci.ub
##  -0.0787  0.0765  -1.0288  0.3036  -0.2287  0.0713
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
### estimated average odds ratio (and 95% CI/PI)
pred <- predict(res, transf=exp, digits=2)
pred
```

```
##
##  pred ci.lb ci.ub
##  0.92  0.80  1.07
```

```
############################################################################

### need the rounded estimate and CI bounds further below
pred <- fmtx(c(pred$pred, pred$ci.lb, pred$ci.ub), digits=2)

### total number of studies
k <- nrow(dat)

### get the weights and format them as will be used in the forest plot
weights <- paste0(fmtx(weights(res), digits=1), "%")
weights[weights == "NA%"] <- ""

### adjust the margins
par(mar=c(13.8,0,1.3,0), mgp=c(2,0.2,0), tcl=-0.2)

pdf(width=8.8,height=3.2,file="forest_plots/mistral_forest_plot.pdf")

### forest plot with extra annotations
sav <- forest(res, atransf=exp, at=log(c(0.01, 0.10, 1, 10, 100)), xlim=c(-30,11),
       xlab="", efac=c(0,4), textpos=c(-30,-4.7), lty=c(1,1,0), refline=NA,
       ilab=cbind(ai, n1i, ci, n2i, weights),
       ilab.xpos=c(-20.6,-18.6,-16.1,-14.1,-10.8), ilab.pos=2,
```

```r
       cex=0.78, header=c("Study"), mlab="")

### add horizontal line at the top
segments(sav$xlim[1], k+1, sav$xlim[2]-5, k+1, lwd=0.8)

### add vertical reference line at 0
segments(0, -2, 0, k+1, lwd=0.8)

### now we add a bunch of text; since some of the text falls outside of the
### plot region, we set xpd=NA so nothing gets clipped
par(xpd=NA)

### adjust cex as used in the forest plot and use a bold font
par(cex=sav$cex, font=2)

text(sav$ilab.xpos, k+2, pos=2, c("Events","Total","Events","Total", "Weight"))
text(c(-19.1,-15.1), k+3, pos=2, c("Remdesivir","Control"))
text(0, k+3, "Odds ratio, 95% CI")

### use a non-bold font for the rest of the text
par(cex=sav$cex, font=1)

text(c(sav$xlim[1],sav$ilab.xpos[c(2,4,5)]), -1, pos=c(4,2,2,2,2),
     c("Total (95% CI)", sum(dat$n1i), sum(dat$n2i), "100.0%"))

### add 'Favors remdesivir'/'Favors control' text below the x-axis
text(log(c(0.01, 100)), -3, c("Favors remdesivir","Favors control"), pos=c(4,2), offset=-1)

### add text for total events
text(sav$ilab.xpos[c(1,3)], -1, c(sum(dat$ai),sum(dat$ci)), pos=2)

dev.off
```

```
## function (which = dev.cur())
## {
##     if (which == 1)
##         stop("cannot shut down device 1 (the null device)")
##     .External(C_devoff, as.integer(which))
##     dev.cur()
## }
## <bytecode: 0x7fd8844be958>
## <environment: namespace:grDevices>
```