Untitled

Dehshini

2023-12-01

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
#import dataset as nepal
nepal = read_csv("nepal621_v2.csv")
```

```
## Rows: 27121 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (4): sex, age, trt, status
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
#View(nepal)

#Create a table that displays the numbers of deaths and numbers of survivors for the
#vitamin A and control groups separately for the six age-by-sex strata.
nepal %>%
  group_by(trt, sex, age) %>%
  summarize(N_Alive = sum(status=="Alive"),
            Perc_Alive = round(N_Alive/n(),4)*100,
            N_Died = sum(status=="Died"),
            Perc_Died = round(N_Died/n(),4)*100,
            Total=n())
```

```
## `summarise()` has grouped output by 'trt', 'sex'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 12 x 8
## # Groups:   trt, sex [4]
##    trt     sex    age    N_Alive Perc_Alive N_Died Perc_Died Total
##    <chr>   <chr>  <chr>    <int>      <dbl>  <int>     <dbl> <int>
##  1 Placebo Female 1 to 2    2615       97.3     72      2.68  2687
##  2 Placebo Female 3 to 4    2542       99.0     25      0.97  2567
##  3 Placebo Female <1        1219       94.6     69      5.36  1288
```

```
##  4 Placebo Male   1 to 2    2770       98.3     47      1.67  2817
##  5 Placebo Male   3 to 4    2677       99.0     26      0.96  2703
##  6 Placebo Male   <1        1276       96.2     51      3.84  1327
##  7 Vit A   Female 1 to 2    2724       98.1     52      1.87  2776
##  8 Vit A   Female 3 to 4    2529       99.4     15      0.59  2544
##  9 Vit A   Female <1        1291       96.0     54      4.01  1345
## 10 Vit A   Male   1 to 2    2837       98.6     40      1.39  2877
## 11 Vit A   Male   3 to 4    2752       99.6     12      0.43  2764
## 12 Vit A   Male   <1        1366       95.8     60      4.21  1426
```

```
#proportion of children who died in the vitamin A group and in the control group
nepal %>%
  group_by(trt) %>%
  summarize(N_Alive = sum(status=="Alive"),
            Perc_Alive = round(N_Alive/n(),4)*100,
            N_Died = sum(status=="Died"),
            Perc_Died = round(N_Died/n(),4)*100,
            Total=n())
```

```
## # A tibble: 2 x 6
##   trt     N_Alive Perc_Alive N_Died Perc_Died Total
##   <chr>     <int>      <dbl>  <int>     <dbl> <int>
## 1 Placebo   13099       97.8    290      2.17 13389
## 2 Vit A     13499       98.3    233       1.7 13732
```

```
#Calculate a 95% confidence interval for each mortality rate
nepal %>%
  group_by(trt) %>%
  summarize(N_Alive = sum(status=="Alive"),
            p_Alive = N_Alive/n(),
            N_Died = sum(status=="Died"),
            p_Died = N_Died/n(),
            Total = n(),
            se_Died = sqrt(p_Died *(1-p_Died)/Total),
            CI_L = p_Died - 1.96*se_Died,
            CI_U = p_Died + 1.96*se_Died)
```

```
## # A tibble: 2 x 9
##   trt     N_Alive p_Alive N_Died p_Died Total se_Died   CI_L   CI_U
##   <chr>     <int>   <dbl>  <int>  <dbl> <int>   <dbl>  <dbl>  <dbl>
## 1 Placebo   13099   0.978    290 0.0217 13389 0.00126 0.0192 0.0241
## 2 Vit A     13499   0.983    233 0.0170 13732 0.00110 0.0148 0.0191
```

```
#C.I by hand
p.1 = 0.0217 # sample proportion of dead for placebo group
n.1 =  13389 # sample size for placebo group
p.2 = 0.0170 # sample proportion of dead for vitA group
n.2 = 13732 # sample size for vitA group

#diff between placebo and vitA
diff = p.1 - p.2
```

```r
# standard error
se = sqrt(p.1*(1-p.1)/n.1 + p.2*(1-p.2)/n.2)

# confidence interval
LL = diff - 1.96*se
UL = diff + 1.96*se

#confidence interval by age, sex and trt
nepal %>%
  group_by(sex, age) %>%
  summarize(N_Plac = sum(trt=="Placebo"),
            p_Plac = sum(status=="Died" & trt=="Placebo")/N_Plac,
            N_VitA = sum(trt=="Vit A"),
            p_VitA = sum(status=="Died" & trt=="Vit A")/N_VitA,
            diff = p_Plac - p_VitA,
            se = sqrt(p_Plac*(1 - p_Plac)/N_Plac + p_VitA*(1 - p_VitA)/N_VitA),
            CI_L = diff - 1.96*se,
            CI_U = diff + 1.96*se)
```

```
## `summarise()` has grouped output by 'sex'. You can override using the `.groups`
## argument.
```

```
## # A tibble: 6 x 10
## # Groups:   sex [2]
##   sex    age    N_Plac  p_Plac N_VitA  p_VitA     diff      se      CI_L    CI_U
##   <chr>  <chr>   <int>   <dbl>  <int>   <dbl>    <dbl>   <dbl>     <dbl>   <dbl>
## 1 Female 1 to 2   2687 0.0268    2776 0.0187   0.00806 0.00404  0.000144 0.0160
## 2 Female 3 to 4   2567 0.00974   2544 0.00590  0.00384 0.00246 -0.000983 0.00867
## 3 Female <1       1288 0.0536    1345 0.0401   0.0134  0.00825 -0.00274  0.0296
## 4 Male   1 to 2   2817 0.0167    2877 0.0139   0.00278 0.00325 -0.00360  0.00916
## 5 Male   3 to 4   2703 0.00962   2764 0.00434  0.00528 0.00226  0.000856 0.00970
## 6 Male   <1       1327 0.0384    1426 0.0421  -0.00364 0.00749 -0.0183   0.0110
```

```r
#save the above as a new dataframe
dataForCIplot = nepal %>%
  group_by(sex, age) %>%
  summarize(N_Plac = sum(trt=="Placebo"),
            p_Plac = sum(status=="Died" & trt=="Placebo")/N_Plac,
            N_VitA = sum(trt=="Vit A"),
            p_VitA = sum(status=="Died" & trt=="Vit A")/N_VitA,
            diff = p_Plac - p_VitA,
            se = sqrt(p_Plac*(1 - p_Plac)/N_Plac + p_VitA*(1 - p_VitA)/N_VitA),
            CI_L = diff - 1.96*se,
            CI_U = diff + 1.96*se)
```

```
## `summarise()` has grouped output by 'sex'. You can override using the `.groups`
## argument.
```

```r
#concatenate the overall values(diff, UL, LL, se) into new lists with the grouped data
diffg = c(dataForCIplot$diff, 0.0047)
LLg = c(dataForCIplot$CI_L, 0.00142)
ULg = c(dataForCIplot$CI_U, 0.00798)
```

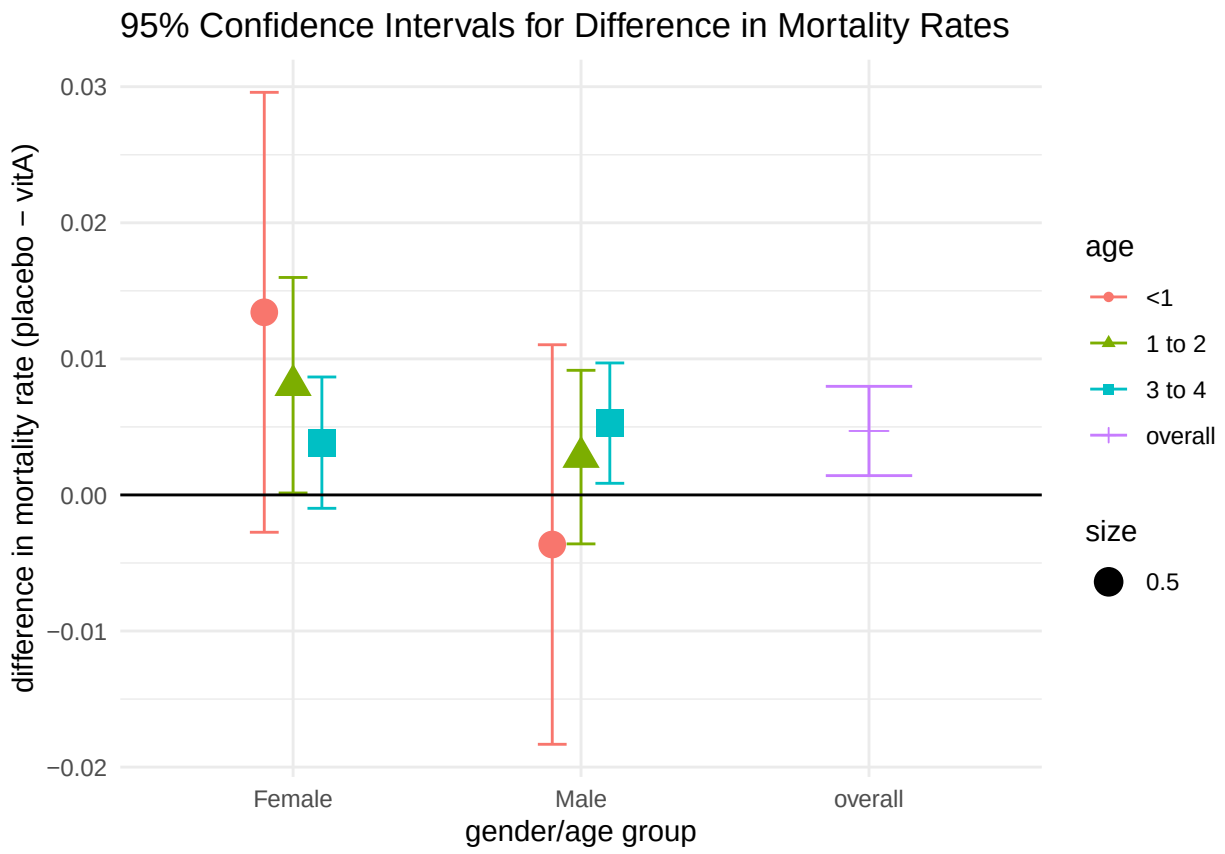
```r
se_g = c(dataForCIplot$se, se)

#create a new dataframe for the plot using the lists created
df = data_frame(sex=c(dataForCIplot$sex, "overall"),
                age=c(dataForCIplot$age, "overall"),
                diff=diffg,
                se=se_g,
                LL = LLg,
                UL = ULg)
```

```
## Warning: `data_frame()` was deprecated in tibble 1.1.0.
## i Please use `tibble()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```r
#plot: create error plots
g2 = ggplot(df, aes(x=sex , y=diff, ymin=LL, ymax=UL))

g2 + geom_errorbar(aes(color=age), position = position_dodge(0.3), width = 0.3) +
  geom_point(aes(color=age, size=0.5, shape=age), position = position_dodge(0.3)) +
  geom_hline(aes(yintercept=0), color="black") +
  xlab("gender/age group") +
  ylab("difference in mortality rate (placebo - vitA)") +
  theme_minimal()+
  labs(title = "95% Confidence Intervals for Difference in Mortality Rates")
```




```
#fit a model on the data
model1 = glm(factor(status) ~ trt, data=nepal, family=binomial(link="identity"))
summary(model1)
```

```
##
## Call:
## glm(formula = factor(status) ~ trt, family = binomial(link = "identity"),
##     data = nepal)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.021660   0.001258  17.217  < 2e-16 ***
## trtVit A    -0.004692   0.001673  -2.805  0.00503 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5166.0  on 27120  degrees of freedom
## Residual deviance: 5158.1  on 27119  degrees of freedom
## AIC: 5162.1
##
## Number of Fisher Scoring iterations: 2
```

```r
confint(model1)
```

```
## Waiting for profiling to be done...
```

```
##                    2.5 %       97.5 %
## (Intercept)  0.019284720  0.024217501
## trtVit A    -0.007988047 -0.001420949
```

```r
#second part

#Create two age groups.
nepal = nepal %>%
  mutate(agegp = ifelse(age == "3 to 4", "3+ years", "<3 years"))

#calculate odds
nepal %>%
  group_by(agegp, trt) %>%
  summarize(N_Alive = sum(status=="Alive"),
            N_Died = sum(status=="Died"),
            Odds = N_Died/N_Alive)
```

```
## `summarise()` has grouped output by 'agegp'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 4 x 5
## # Groups:   agegp [2]
##   agegp    trt     N_Alive N_Died    Odds
##   <chr>    <chr>     <int>  <int>   <dbl>
## 1 3+ years Placebo    5219     51 0.00977
## 2 3+ years Vit A      5281     27 0.00511
## 3 <3 years Placebo    7880    239 0.0303
## 4 <3 years Vit A      8218    206 0.0251
```

```r
#calculate OR of death for plac vs vitA
nepal %>%
  group_by(agegp) %>%
  summarize(N_Alive_P = sum(status=="Alive" & trt=="Placebo"),
            N_Died_P = sum(status=="Died" & trt=="Placebo"),
            N_Alive_V = sum(status=="Alive" & trt=="Vit A"),
            N_Died_V = sum(status=="Died" & trt=="Vit A"),
            OR = (N_Died_P/N_Alive_P)/(N_Died_V/N_Alive_V),
            se = sqrt(1/N_Alive_P + 1/N_Died_P + 1/N_Alive_V + 1/N_Died_V),
            CI_L = exp(log(OR)-1.96*se),
            CI_U = exp(log(OR)+1.96*se))
```

```
## # A tibble: 2 x 9
##   agegp    N_Alive_P N_Died_P N_Alive_V N_Died_V    OR     se  CI_L  CI_U
##   <chr>        <int>    <int>     <int>    <int> <dbl>  <dbl> <dbl> <dbl>
## 1 3+ years      5219       51      5281       27  1.91 0.239   1.20  3.05
## 2 <3 years      7880      239      8218      206  1.21 0.0964  1.00  1.46
```

```r
#create a dataframe for the low age group
nepal621.lowage = nepal %>% filter(agegp == "<3 years")


#fit a model for the lowage group
model2 = glm(factor(status) ~ factor(trt, levels = c("Vit A", "Placebo")), data=nepal621.lowage,
             family=binomial(link="logit"))
summary(model2) # This summary is on the logOR scale
```

```
##
## Call:
## glm(formula = factor(status) ~ factor(trt, levels = c("Vit A",
##     "Placebo")), family = binomial(link = "logit"), data = nepal621.lowage)
##
## Coefficients:
##                                                 Estimate Std. Error z value
## (Intercept)                                     -3.68621    0.07054 -52.257
## factor(trt, levels = c("Vit A", "Placebo"))Placebo  0.19059    0.09637   1.978
##                                                 Pr(>|z|)
## (Intercept)                                       <2e-16 ***
## factor(trt, levels = c("Vit A", "Placebo"))Placebo    0.048 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4095.8  on 16542  degrees of freedom
## Residual deviance: 4091.9  on 16541  degrees of freedom
## AIC: 4095.9
##
## Number of Fisher Scoring iterations: 6
```

```r
exp(model2$coefficients) # We exponentiate to get on the OR scale
```

```
##                                       (Intercept)
##                                        0.02506693
## factor(trt, levels = c("Vit A", "Placebo"))Placebo
##                                        1.20995885
```

```r
exp(confint(model2))
```

```
## Waiting for profiling to be done...
```

```
##                                                     2.5 %     97.5 %
## (Intercept)                                     0.02176241 0.02869846
## factor(trt, levels = c("Vit A", "Placebo"))Placebo 1.00199628 1.46230798
```

```r
#create data frame for the high age group
nepal621.highage = nepal %>% filter(agegp == "3+ years")

#fit a model for the high age group
```

```
model3 = glm(factor(status) ~ factor(trt, levels=c("Vit A", "Placebo")), data=nepal621.highage,
             family=binomial(link="logit"))
summary(model3)
```

```
##
## Call:
## glm(formula = factor(status) ~ factor(trt, levels = c("Vit A",
##     "Placebo")), family = binomial(link = "logit"), data = nepal621.highage)
##
## Coefficients:
##                                                 Estimate Std. Error z value
## (Intercept)                                      -5.2760     0.1929 -27.345
## factor(trt, levels = c("Vit A", "Placebo"))Placebo   0.6478     0.2388   2.713
##                                                 Pr(>|z|)
## (Intercept)                                      < 2e-16 ***
## factor(trt, levels = c("Vit A", "Placebo"))Placebo  0.00667 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 921.36  on 10577  degrees of freedom
## Residual deviance: 913.62  on 10576  degrees of freedom
## AIC: 917.62
##
## Number of Fisher Scoring iterations: 8
```

```
exp(model3$coefficients)
```

```
##                                       (Intercept)
##                                       0.005112668
## factor(trt, levels = c("Vit A", "Placebo"))Placebo
##                                       1.911328266
```

```
exp(confint(model3))
```

```
## Waiting for profiling to be done...
```

```
##                                                       2.5 %      97.5 %
## (Intercept)                                      0.003415677 0.007298643
## factor(trt, levels = c("Vit A", "Placebo"))Placebo 1.207731642 3.093089503
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.