Misclassification Activity
JSM 2023 Short Course
(with solutions)

Paul Gustafson

August 6, 2023

Recall this example (slide 128)

```r
data.tbl <- matrix(c(45,94,257,945),
  dimnames = list(c("CHD+", "CHD-"),c("Resin+", "Resin-")),
  nrow = 2, byrow = TRUE)
```

```r
data.tbl
```

```
##      Resin+ Resin-
## CHD+     45     94
## CHD-    257    945
```

Naive analysis presuming correct exposure classification

Inference for exposure-disease odds-ratio

```
logOR.hat <- sum(c(1,-1,-1,1)*log(as.vector(data.tbl)))

logOR.SE <- sqrt(sum(1/as.vector(data.tbl)))

exp(logOR.hat + c(0, -1.96, 1.96)*logOR.SE)
```

```
## [1] 1.76 1.20 2.58
```

Assuming nondifferential exposure misclassification with 90% sensitivity and 80% specificity

Recall from slides 138, 139:

```
require(episensr)

ft <- misclassification(data.tbl,
 type="exposure", bias_parms=c(0.8, 0.8, 0.9, 0.9))

# point and interval estimation of OR
ft$adj.measures[2,]
```

```
##          2.5% 97.5%
##  2.42   1.37  4.26
```

Activity A

Check you can reproduce one of the **differential** classification adjustments on slide 135 (i.e., one of the off-diagonal table entries).

For instance, try presuming 90% specificity for all subjects, but sensitivity of 90% for controls, compared to 80% for cases.

Might help:

```
help(misclassification)
```

Activity A: Solution

```r
ft <- misclassification(data.tbl,
 type="exposure", bias_parms=c(0.8, 0.9, 0.9, 0.9))

# point and interval estimation of OR
ft$adj.measures[2,]
```

```
##         2.5% 97.5%
##  2.83   1.61  4.98
```

Activity B: Uncertainty about misclassification rates.

Say the investigator is confident that the misclassification is nondifferential.

Has 85% sensitivity and 85% specificity as "best guesses."

But thinks either guess could be off by as much as five percentage points.

Can you look at

```
help(probsens)
```

And provide an appropriate analysis?

HINT: First example in the help gives a template.

HINT: For simplicity, maybe "triangular" or "uniform" instead of "trapezoidal"

Activity B - Solution

```
ft <- probsens(data.tbl,
 type="exposure", reps=5000,
 seca.parms = list("triangular", c(0.80, 0.90, 0.85)),
 spca.parms = list("triangular", c(0.80, 0.90, 0.85)))

# OR inference
rownames(ft$adj)
```

```
## [1] "           Relative Risk -- systematic error:"
## [2] "              Odds Ratio -- systematic error:"
## [3] "Relative Risk -- systematic and random error:"
## [4] "   Odds Ratio -- systematic and random error:"
```

```
ft$adj.measures[4,]
```

```
##             Median  2.5th percentile 97.5th percentile
##               3.35              2.04              6.80
```

Activity C - Role of data

We have useful heuristics in statistics, such as the primal role of $\sqrt{n}$.

If I want interval estimates *twice* as narrow, I likely need about *four times* as much data.

Repeat Activity B, but with four times as much data. (Simplest to just keep cell *proportions* fixed in the 2 by 2 data table).

Reflect on what you find.

Activity C - Solution

```
ft.more <- probsens(4*data.tbl,
 type="exposure", reps=5000,
 seca.parms = list("triangular", c(0.80, 0.90, 0.85)),
 spca.parms = list("triangular", c(0.80, 0.90, 0.85)))

# OR inference
ft.more$adj.measures[4,]
```

```
##             Median  2.5th percentile 97.5th percentile
##               3.29              2.32              6.51
```

Activity C - Solution, continued

Reduction in interval width:

```
# ratio of CI on log-OR
(log(ft.more$adj[4,3])- log(ft.more$adj[4,2])) /
(log(ft$adj[4,3])      - log(ft$adj[4,2]))
```

```
## [1] 0.857
```

Not the usual payoff for quadrupling sample size!

Majority contributor to uncertainty: imprecise knowledge of the exposure classification characteristics (sens and spec).