

SQUEAC Attack

Mark Myatt

32 July 2018

Contents

1	Calculate ‘k’ for single coverage estimate	2
2	Calculate r.out (recovering cases NOT in the program)	4
3	Calculate Prior modes (priorMode) from priorAlpha and priorBeta with their standard errors (priorModeSE)	6
4	Calculate appropriate likelihood numerators (likeN) denominators (likeD)	8
5	Make summary data.frame	10
6	Test for prior-likelihood conflict	11
7	How common are prior-likelihood conflicts?	12
8	How are the prior and likelihood modes related?	13
9	Scatterplot	14
10	How precise is the likelihood estimate alone	15
10.1	Relative precision	15

1 Calculate ‘k’ for single coverage estimate

Correction factor **k** is the ratio of the mean length of an untreated episode to the mean length of a CMAM treatment episode.

$$k = \frac{\text{Mean length of an untreated episode}}{\text{Mean length of a successful treatment episode}}$$

Mean length of an untreated episode for SAM or MAM can be assumed as 7.5 months based on [Garenne et al. \[2009\]](#). Mean length of a successful treatment episode can be estimated from routine programming monitoring data by calculating the median length of stay.

This can be implemented in R as follows:

```
medianLOS <- median(work$medianLOS, na.rm = TRUE)

k <- ((7.5 * 30.44) / 7) / work$medianLOS
```

The **median of the median length of stay** of the surveys in the dataset is 7.

The **k** values for each of the surveys in the dataset are¹:

##	[1]	NA	NA	NA	NA	NA	NA	NA
##	[8]	NA	NA	NA	NA	NA	NA	NA
##	[15]	NA	NA	NA	NA	4.076786	NA	NA
##	[22]	NA	NA	4.076786	4.659184	6.522857	NA	4.076786
##	[29]	8.153571	3.623810	4.076786	NA	3.623810	NA	5.435714
##	[36]	5.435714	NA	NA	3.623810	NA	3.623810	NA
##	[43]	NA	NA	NA	NA	3.261429	3.261429	4.076786
##	[50]	3.623810	NA	4.076786	4.076786	2.964935	NA	2.717857
##	[57]	8.153571	4.076786	4.076786	3.623810	NA	NA	8.153571
##	[64]	NA	NA	5.435714	5.435714	6.522857	8.153571	6.522857
##	[71]	NA	5.435714	6.522857	5.435714	3.623810	6.522857	8.153571
##	[78]	4.659184	NA	6.522857	NA	6.522857	NA	5.435714
##	[85]	3.623810	4.659184	4.076786	NA	5.435714	NA	5.435714
##	[92]	2.964935	4.076786	10.871429	2.717857	NA	NA	NA

¹NA values are for surveys that don't report a median length of stay.

##	[99]	NA	3.261429	4.076786	4.659184	4.076786	4.659184	5.435714
##	[106]	NA	NA	NA	NA	NA	NA	4.659184
##	[113]	NA	3.623810	NA	NA	3.623810	NA	NA
##	[120]	4.659184	2.717857	4.659184	NA	4.076786	4.659184	3.623810
##	[127]	6.522857	5.435714	8.153571	5.435714	5.435714	5.435714	4.659184
##	[134]	6.522857	6.522857	6.522857	5.435714	NA	6.522857	4.659184
##	[141]	4.659184	NA	5.435714	NA	5.435714	5.435714	NA
##	[148]	5.435714	4.076786	5.435714	4.076786	4.076786	NA	4.659184
##	[155]	8.153571	8.153571	NA	8.153571	NA	5.435714	4.659184
##	[162]	6.522857	8.153571	8.153571	8.153571	NA	8.153571	8.153571
##	[169]	NA	NA	5.435714	NA	NA	NA	8.153571
##	[176]	6.522857	6.522857	3.261429	6.522857	5.435714	8.153571	4.659184
##	[183]	5.435714	4.076786	4.659184	4.076786	NA	4.659184	NA
##	[190]	6.522857	10.871429	6.522857	5.435714	4.659184	8.153571	5.435714
##	[197]	6.522857	6.522857	5.435714	5.435714	3.623810	NA	3.623810
##	[204]	5.435714	6.522857	4.659184	3.261429	2.964935	3.261429	3.261429
##	[211]	3.623810	5.435714	3.623810	5.435714	4.076786	5.435714	4.076786
##	[218]	5.435714	NA	NA	4.076786	4.659184	4.659184	3.623810
##	[225]	4.659184	NA	NA	NA	3.623810	5.435714	NA
##	[232]	4.659184	5.435714	NA	2.717857	NA	6.522857	6.522857
##	[239]	6.522857	6.522857	NA	4.076786	NA	8.153571	3.623810
##	[246]	5.435714	NA	2.508791	2.174286	NA	4.076786	2.964935
##	[253]	4.659184	3.623810	NA	NA	5.435714	2.508791	5.435714
##	[260]	4.076786	NA	NA	3.623810	4.659184	NA	6.522857
##	[267]	5.435714	3.261429	4.659184	5.435714	5.435714	3.623810	NA
##	[274]	4.659184	4.076786	NA	4.659184	5.435714	5.435714	6.522857
##	[281]	2.964935	4.659184	5.435714	NA	NA		

2 Calculate r.out (recovering cases NOT in the program)

Using the calculated k values in previous section, number of recovering cases NOT in the program is calculated using the following formula:

$$R_{out} \approx \frac{1}{k} \times \left(R_{in} \times \frac{C_{in} + C_{out} + 1}{C_{in} + 1} - R_{in} \right)$$

where :

k = correction factor

C_{in} = current SAM cases in the program

C_{out} = current SAM cases not in the program

R_{in} = recovering SAM cases in the program

This can be implemented in R as follows:

```
r.out <- floor((1 / k) *  
              (work$r.in * ((work$c.in + work$c.out + 1) /  
                           (work$c.in + 1)) - work$r.in))
```

The resulting vector of `r.out` values are:

```
## [1] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA  
## [18] NA 4 NA NA NA NA 10 1 7 NA NA 2 NA 5 NA NA NA  
## [35] NA NA NA NA NA NA 2 NA NA NA NA NA 3 NA 5 2 NA  
## [52] NA 3 22 NA NA 2 21 4 0 NA NA NA NA NA NA 10 0  
## [69] NA NA NA 9 6 6 0 21 NA 4 NA NA NA 1 NA 2 9  
## [86] NA NA NA 0 NA 3 NA 4 NA NA NA NA NA NA NA NA 3  
## [103] 15 0 1 NA NA NA NA NA NA 3 NA 13 NA NA 3 NA NA  
## [120] 1 5 2 NA NA NA NA 6 14 NA 4 NA NA NA 8 4 NA  
## [137] 2 NA 16 NA 11 NA NA NA 8 8 NA 1 NA 30 21 21 NA  
## [154] 2 NA 5 NA 5 NA NA NA NA 9 NA 8 NA NA NA NA NA  
## [171] 7 NA NA NA 2 2 1 15 6 NA 0 NA NA 3 10 5 NA  
## [188] 11 NA 2 6 2 1 2 10 14 1 8 NA NA NA NA NA 7
```

## [205]	1	5	179	12	3	4	6	6	0	4	2	5	6	1	NA	NA	0
## [222]	NA	NA	3	12	NA	NA	NA	2	21	NA	NA	0	NA	11	NA	0	NA
## [239]	NA	NA	NA	0	NA	NA	1	NA	NA	12	16	NA	NA	7	0	NA	NA
## [256]	NA	0	7	NA	NA	NA	NA	16	11	NA	5	NA	5	2	0	1	6
## [273]	NA	11	10	NA	0	10	0	0	2	5	4	NA	NA				

3 Calculate Prior modes (priorMode) from priorAlpha and priorBeta with their standard errors (priorModeSE)

The Prior mode can be calculated from Prior α and Prior β as shown in the following formula.

$$mode_{Prior} = \frac{\alpha_{Prior} - 1}{\alpha_{Prior} + \beta_{Prior} - 2}$$

This can be implemented in R as follows:

```
priorN <- work$priorAlpha - 1  
  
priorD <- work$priorAlpha + work$priorBeta - 2  
  
priorMode <- priorN / priorD
```

which results in:

Table 1: Prior numerator, Prior denominator and Prior mode (first 20 records)

$Prior_{numerator}$	$Prior_{denominator}$	$Prior_{mode}$
13.50	28.40	0.48
19.70	39.00	0.51
11.00	21.40	0.51
18.40	33.60	0.55
7.20	26.10	0.28
9.50	24.10	0.39
8.70	20.60	0.42
17.20	34.40	0.50
9.10	19.00	0.48
18.50	33.50	0.55
23.50	43.00	0.55
24.90	53.10	0.47
19.00	38.80	0.49
14.20	34.00	0.42
7.70	17.50	0.44
32.10	64.50	0.50
19.70	39.10	0.50
21.40	36.70	0.58
19.80	34.70	0.57
10.21	30.03	0.34

4 Calculate appropriate likelihood numerators (likeN) denominators (likeD)

The likelihood mode can be calculated depending on the coverage estimator to assess: *point coverage* or *period coverage*.

$$\text{Point coverage} = \frac{C_{in}}{C_{in} + C_{out}}$$

$$\text{Period coverage} = \frac{C_{in} + R_{in}}{C_{in} + C_{out} + R_{in} + R_{out}}$$

where :

C_{in} = current SAM cases in the program

C_{out} = current SAM cases not in the program

R_{in} = recovering SAM cases in the program

R_{out} = recovering SAM cases not in the program

This can be implemented in R as follows:

```
likeN <- ifelse(work$coverType == "point", work$c.in, work$c.in + work$r.in)

likeD <- ifelse(work$coverType == "point", work$c.in + work$c.out,
  ifelse(work$coverType == "period",
    work$c.in + work$r.in + work$c.out,
    work$c.in + work$r.in + work$c.out + r.out))

likeMode <- likeN / likeD
```

which results in:

Table 2: Likelihood numerator, likelihood denominator and likelihood mode

$Likelihood_{numerator}$	$Likelihood_{denominator}$	$Likelihood_{mode}$
3	23	0.13
2	5	0.40
20	37	0.54
2	3	0.67
8	28	0.29
8	13	0.62
18	37	0.49
4	11	0.36
2	4	0.50
5	6	0.83
25	40	0.62
NA	NA	NA
32	64	0.50
25	88	0.28
20	100	0.20
92	121	0.76
17	25	0.68
4	8	0.50
38	59	0.64
4	32	0.12

5 Make summary data.frame

```
results <- data.frame(priorN, priorD, priorMode, likeN, likeD, likeMode)

results <- results[!is.na(results$priorMode) & !is.na(results$likeMode), ]
```

The resulting data.frame is:

Table 3: Summary data.frame (first 30 records)

$Prior_{numerator}$	$Prior_{denominator}$	$Prior_{mode}$	$Likelihood_{numerator}$	$Likelihood_{denominator}$	$Likelihood_{mode}$
13.50	28.40	0.4753521	3	23	0.1304348
19.70	39.00	0.5051282	2	5	0.4000000
11.00	21.40	0.5140187	20	37	0.5405405
18.40	33.60	0.5476190	2	3	0.6666667
7.20	26.10	0.2758621	8	28	0.2857143
9.50	24.10	0.3941909	8	13	0.6153846
8.70	20.60	0.4223301	18	37	0.4864865
17.20	34.40	0.5000000	4	11	0.3636364
9.10	19.00	0.4789474	2	4	0.5000000
18.50	33.50	0.5522388	5	6	0.8333333
23.50	43.00	0.5465116	25	40	0.6250000
19.00	38.80	0.4896907	32	64	0.5000000
14.20	34.00	0.4176471	25	88	0.2840909
7.70	17.50	0.4400000	20	100	0.2000000
32.10	64.50	0.4976744	92	121	0.7603306
19.70	39.10	0.5038363	17	25	0.6800000
21.40	36.70	0.5831063	4	8	0.5000000
19.80	34.70	0.5706052	38	59	0.6440678
10.21	30.03	0.3399933	4	32	0.1250000
12.40	32.50	0.3815385	7	22	0.3181818
24.40	55.40	0.4404332	40	79	0.5063291
18.40	40.70	0.4520885	34	88	0.3863636
19.80	35.50	0.5577465	60	114	0.5263158
2.20	9.10	0.2417582	32	87	0.3678161
13.20	32.03	0.4121136	15	41	0.3658537
27.50	34.00	0.8088235	71	86	0.8255814
20.20	51.00	0.3960784	17	45	0.3777778
11.60	42.60	0.2723005	10	29	0.3448276
10.00	26.00	0.3846154	4	42	0.0952381
23.70	55.30	0.4285714	20	45	0.4444444

6 Test for prior-likelihood conflict

```
for(i in 1:nrow(results)) {
  # Make a two-by-two table
  tab <- matrix(c(results$priorN[i], results$priorD[i] - results$priorN[i],
    results$likeN[i], results$likeD[i] - results$likeN[i]),
    nrow = 2, byrow = TRUE)
  # Fisher test (works with expected numbers < 5)
  results$p[i] <- round(fisher.test(round(tab))$p.value, 4)
}

results$PLC <- ifelse(results$p < 0.05, TRUE, FALSE)
```

Table 4: Summary data.frame with prior-likelihood conflict variable (first 30 records)

<i>Prior</i> _{numerator}	<i>Prior</i> _{denominator}	<i>Prior</i> _{mode}	<i>Likelihood</i> _{numerator}	<i>Likelihood</i> _{denominator}	<i>Likelihood</i> _{mode}	p-value	PLC
13.50	28.40	0.4753521	3	23	0.1304348	0.0086	TRUE
19.70	39.00	0.5051282	2	5	0.4000000	1.0000	FALSE
11.00	21.40	0.5140187	20	37	0.5405405	1.0000	FALSE
18.40	33.60	0.5476190	2	3	0.6666667	1.0000	FALSE
7.20	26.10	0.2758621	8	28	0.2857143	1.0000	FALSE
9.50	24.10	0.3941909	8	13	0.6153846	0.3071	FALSE
8.70	20.60	0.4223301	18	37	0.4864865	0.7863	FALSE
17.20	34.40	0.5000000	4	11	0.3636364	0.5030	FALSE
9.10	19.00	0.4789474	2	4	0.5000000	1.0000	FALSE
18.50	33.50	0.5522388	5	6	0.8333333	0.3703	FALSE
23.50	43.00	0.5465116	25	40	0.6250000	0.5112	FALSE
19.00	38.80	0.4896907	32	64	0.5000000	1.0000	FALSE
14.20	34.00	0.4176471	25	88	0.2840909	0.1978	FALSE
7.70	17.50	0.4400000	20	100	0.2000000	0.0353	TRUE
32.10	64.50	0.4976744	92	121	0.7603306	0.0005	TRUE
19.70	39.10	0.5038363	17	25	0.6800000	0.2069	FALSE
21.40	36.70	0.5831063	4	8	0.5000000	0.7095	FALSE
19.80	34.70	0.5706052	38	59	0.6440678	0.5164	FALSE
10.21	30.03	0.3399933	4	32	0.1250000	0.0699	FALSE
12.40	32.50	0.3815385	7	22	0.3181818	0.7753	FALSE
24.40	55.40	0.4404332	40	79	0.5063291	0.4836	FALSE
18.40	40.70	0.4520885	34	88	0.3863636	0.5620	FALSE
19.80	35.50	0.5577465	60	114	0.5263158	0.8488	FALSE
2.20	9.10	0.2417582	32	87	0.3678161	0.4848	FALSE
13.20	32.03	0.4121136	15	41	0.3658537	0.8101	FALSE
27.50	34.00	0.8088235	71	86	0.8255814	1.0000	FALSE
20.20	51.00	0.3960784	17	45	0.3777778	1.0000	FALSE
11.60	42.60	0.2723005	10	29	0.3448276	0.6077	FALSE
10.00	26.00	0.3846154	4	42	0.0952381	0.0060	TRUE
23.70	55.30	0.4285714	20	45	0.4444444	1.0000	FALSE

7 How common are prior-likelihood conflicts?

```
table(results$PLC)
```

```
##  
## FALSE TRUE  
##    222    16
```

```
round(prop.table(table(results$PLC)) * 100, 2)
```

```
##  
## FALSE TRUE  
## 93.28  6.72
```

8 How are the prior and likelihood modes related?

```
## Errors (difference)
error <- results$priorMode * 100 - results$likeMode * 100
summary(error)

##      Min.   1st Qu.   Median     Mean   3rd Qu.    Max.
## -39.8738  -7.8556  -0.9688  -0.8143   7.1055  34.4917

hist(error,
      breaks = 16,
      xlab = "Prior mode (%) - Likelihood mode (%)",
      ylab = "Number of assessments",
      main = "")
```

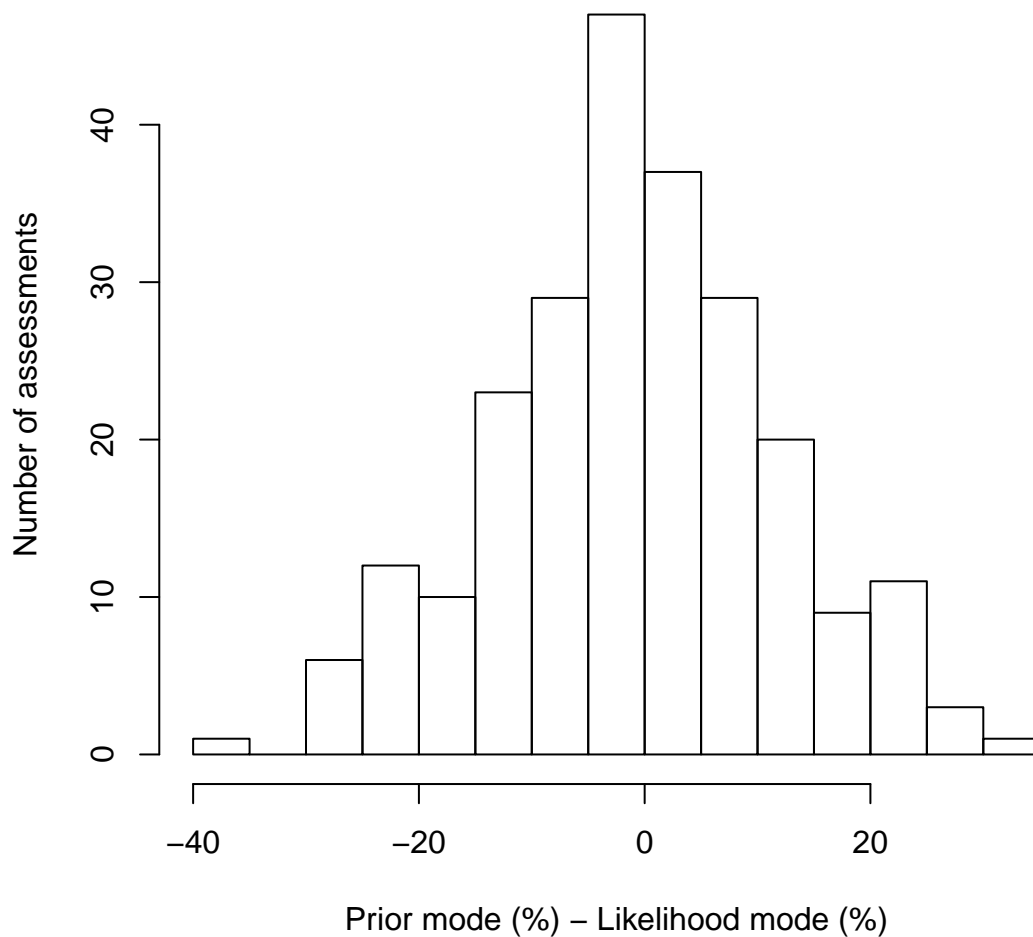


Figure 1: Prior-Likelihood Relationship

9 Scatterplot

```
plot(results$priorMode * 100,
      results$likeMode * 100,
      xlim = c(0, 100), ylim = c(0, 100),
      xlab = "Prior mode (%)",
      ylab = "Likelihood mode (%)",
      pch = ifelse(results$PLC, 19, 1),
      frame.plot = FALSE)
abline(a = 0, b = 1, lty = 2)
text(100, 15, "Prior mode > Likelihood mode", pos = 2, cex = 0.8)
text( 0, 85, "Prior mode < Likelihood mode", pos = 4, cex = 0.8)
lines(lowess(results$priorMode * 100, results$likeMode * 100, f = 2/3))
```

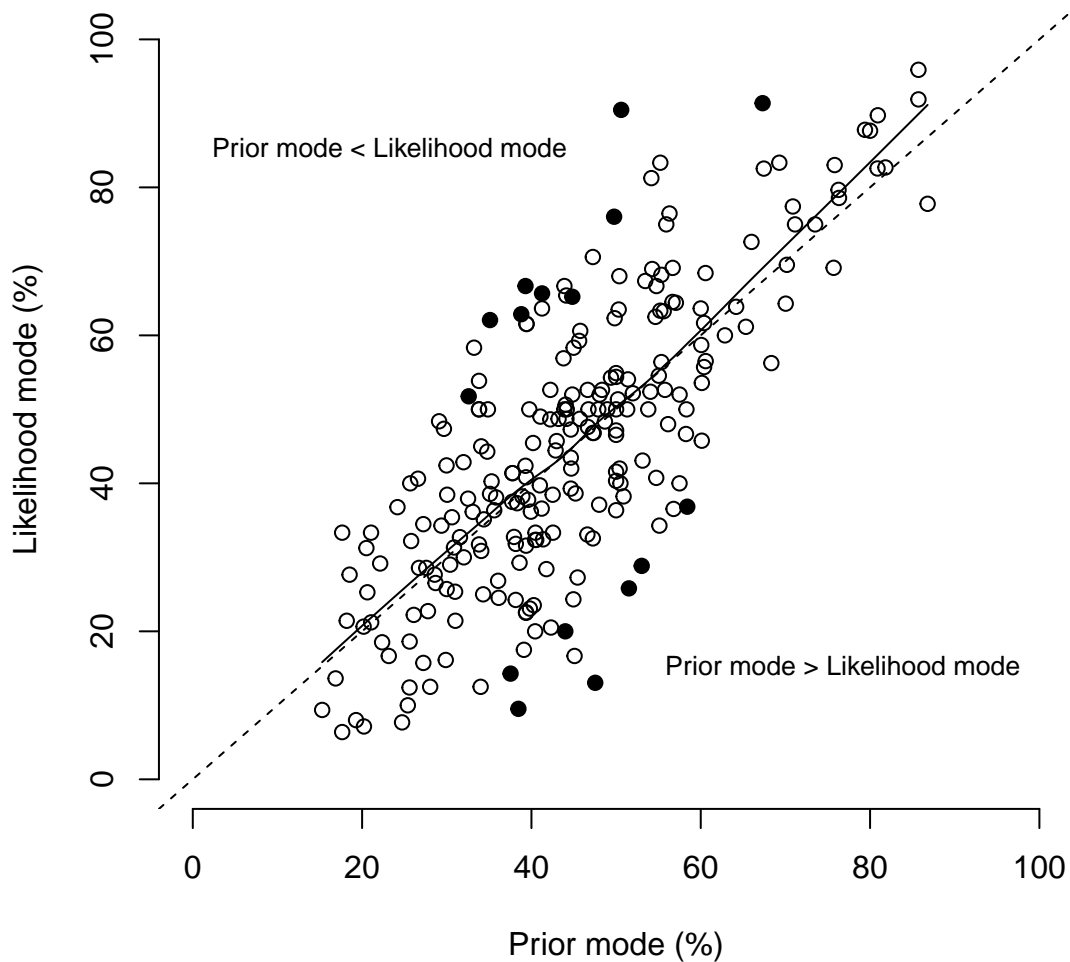


Figure 2: Prior-Likelihood Relationship

```
cor(results$priorMode, results$likeMode)
```

```
## [1] 0.772687
```

10 How precise is the likelihood estimate alone

For this, we assume total population of 100,000 with 17% aged 6-59 months and prevalence of SAM of 2%.

```
# Subset to results with PLC == TRUE
rejected <- results[results$PLC, ]
# calculate number of SAM
pop <- 100000 * 0.17 * 0.02
# calculate finite population correction factor
rejected$FPC <- sqrt((pop - rejected$likeD) / (pop - 1))
```

10.1 Relative precision

```
# Relative precision of surveys with PLC
rejected$likeRP <- (qnorm(0.975) *
  sqrt((rejected$likeMode *
    (1 - rejected$likeMode)) / rejected$likeD) *
  rejected$FPC) / rejected$likeMode

# Relative precision of an EPI coverage survey with p = likelihood mode,
# n = 120, and DEFF = 2.0?
rejected$epiRP <- (2.0 * qnorm(0.975) *
  sqrt((rejected$likeMode *
    (1 - rejected$likeMode)) / 210)) / rejected$likeMode

# How many have relative precision of better than or equal to the
# assumed EPI survey?

table(rejected$likeRP <= rejected$epiRP)

##
## FALSE  TRUE
##      6    10

prop.table(table(rejected$likeRP <= rejected$epiRP))

##
## FALSE  TRUE
## 0.375 0.625
```

```

# Proportion of SQUEAC assessments that fail by ...
#
#   prior likelihood conflict == TRUE AND precision worse than the
#   assumed EPI survey
#
failN <- sum(ifelse(rejected$likeRP > rejected$epiRP, 1, 0))
failP <- round(sum(ifelse(rejected$likeRP > rejected$epiRP, 1, 0)) /
               nrow(results) * 100, 2)

print(failN)

## [1] 6
print(failP)

## [1] 2.52

```


References

Michel Garenne, Douladel Willie, Bernard Maire, Olivier Fontaine, Roger Eeckels, André Briend, and Jan Van den Broeck. Incidence and duration of severe wasting in two African populations. *Public Health Nutrition*, 12(11):1974–1982, March 2009.