**Section 1.2.2 – Confidence intervals for the difference of two probabilities**

From the previous set of notes, we had the following:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | Response | |  |
|  |  | Success | Failure |  |
| Group | 1 | π1 | 1 – π1 | 1 |
| 2 | π2 | 1 – π2 | 1 |

We would to compare π1 and π2. There are a few different ways to perform this comparison. This section of the notes focuses on π1 – π2.

We saw before that the estimated probability of success for one binary variable  can be treated as an approximate normal random variable with mean π and variance  for a large sample. Using the notation in this chapter, this means that

*  has an approximate normal distribution mean 1 and variance 1(1 – 1)/n1 for large n1
*  has an approximate normal distribution mean 2 and variance 2(1 – 2)/n2 for large n2

Note that  and  are treated as random variables here. One could more compactly write this as  for large nj.

The statistic that estimates  is . One can show that



for large n1 and n2. Where does the variance come from?

 because  and  are independent random variables. Some of you may have seen the following: Let X and Y be independent random variables and let a and b be constants. Then Var(aX + bY) = a2Var(X) + b2Var(Y).

The estimate of the variance is then



The Wald confidence interval for 1 – 2 is then

 ± Z1-α/2

Do you remember the problems with the Wald interval for π? Similar problems occur here ☹.

Agresti and Caffo (2000) recommend adding two successes and two failures to the data for an interval of ANY level of confidence. Let

 and 

be the adjusted estimated probability of successes for the two groups. The Agresti-Caffo confidence interval is



Example: Larry Bird (Bird.R)

Below is some code and output from earlier:

> c.table <- array(data = c(251, 48, 34, 5), dim = c(2,2), dimnames = list(First = c("made", "missed"), Second =

c("made", "missed")))

> c.table

Second

First made missed

made 251 34

missed 48 5

> pi.hat.table <- c.table/rowSums(c.table)

> pi.hat.table

Second

First made missed

made 0.8807018 0.11929825

missed 0.9056604 0.09433962

> pi.hat1 <- pi.hat.table[1,1]

> pi.hat2 <- pi.hat.table[2,1]

New code and output:

> alpha <- 0.05

> # Wald

> var.wald <- pi.hat1\*(1-pi.hat1) / sum(c.table[1,]) +

pi.hat2\*(1-pi.hat2) / sum(c.table[2,])

> pi.hat1 - pi.hat2 + qnorm(p = c(alpha/2, 1-alpha/2)) \*

sqrt(var.wald)

[1] -0.11218742 0.06227017

> # Agresti-Caffo

> pi.tilde1 <- (c.table[1,1] + 1) / (sum(c.table[1,]) + 2)

> pi.tilde2 <- (c.table[2,1] + 1) / (sum(c.table[2,]) + 2)

> var.AC <- pi.tilde1\*(1-pi.tilde1) / (sum(c.table[1,]) + 2) + pi.tilde2\*(1-pi.tilde2) / (sum(c.table[2,]) + 2)

> pi.tilde1 - pi.tilde2 + qnorm(p = c(alpha/2, 1-alpha/2))

\* sqrt(var.AC)

[1] -0.10353254 0.07781192

Therefore, the 95% Wald confidence interval is

-0.1122 < π1 - π2 < 0.0623

and the 95% Agresti-Caffo confidence interval is

-0.1035 < π1 - π2 < 0.0778

There is not sufficient evidence to indicate a difference in the probabilities of success. What does this mean in terms of the Bird’s free throw shooting?

Other ways to perform these calculations:

> # Calculations using the PropCIs package

> library(package = PropCIs)

> # Wald

> wald2ci(x1 = c.table[1,1], n1 = sum(c.table[1,]), x2 =

c.table[2,1], n2 = sum(c.table[2,]), conf.level = 0.95,

adjust = "Wald")

data:

95 percent confidence interval:

-0.11218742 0.06227017

sample estimates:

[1] -0.02495862

> # Agresti-Caffo

> wald2ci(x1 = c.table[1,1], n1 = sum(c.table[1,]), x2

= c.table[2,1], n2 = sum(c.table[2,]), conf.level =

0.95, adjust = "AC")

data:

95 percent confidence interval:

-0.10353254 0.07781192

sample estimates:

[1] -0.01286031

> # Wald

> prop.test(x = c.table[,1], n = rowSums(c.table), conf.level = 0.95, correct = FALSE)

2-sample test for equality of proportions without

continuity correction

data: c.table[, 1] out of rowSums(c.table)

X-squared = 0.27274, df = 1, p-value = 0.6015

alternative hypothesis: two.sided

95 percent confidence interval:

-0.11218742 0.06227017

sample estimates:

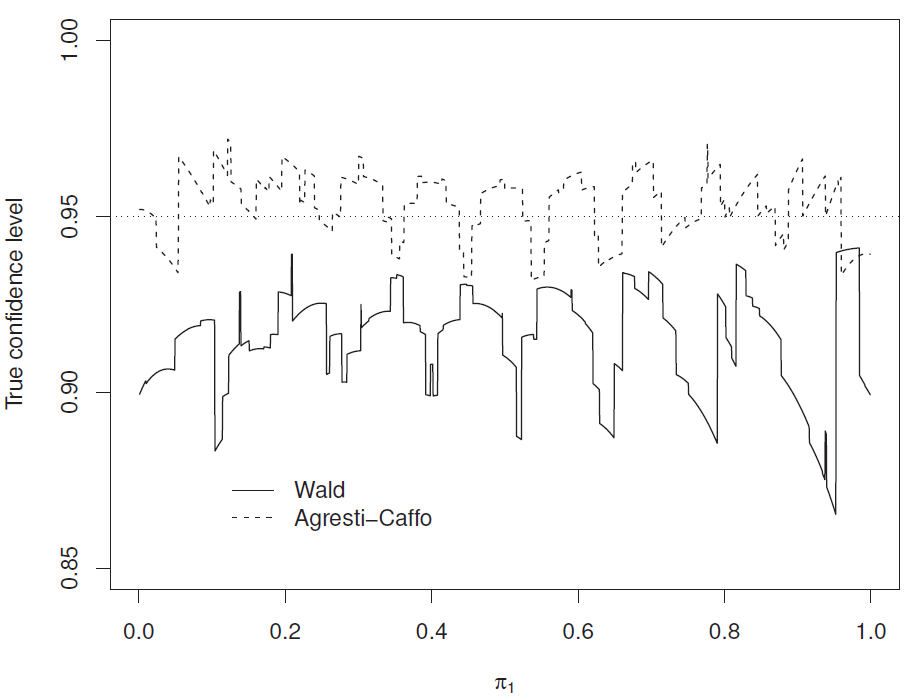
prop 1 prop 2

0.8807018 0.9056604

One could also use x = c.table[,1]+1 and n = rowSums(c.table)+2 in prop.test() to get the Agresti-Caffo interval.

Example: Actual true confidence level for 1 – 2 intervals (ConfLevelTwoProb.R)

Below is a plot for n1 = n2 = 10, 2 = 0.4, and α = 0.05.



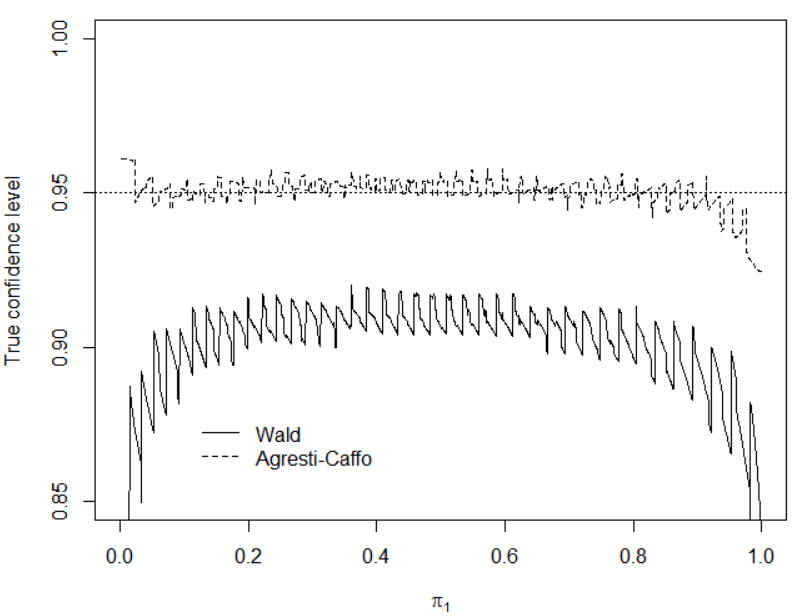
What does this plot say about the actual true confidence levels for the intervals?

How are the actual true confidence levels calculated here?

Examine the steps used for intervals involving  alone and extend them to intervals for 1 – 2. Suppose there is only one value of 1.

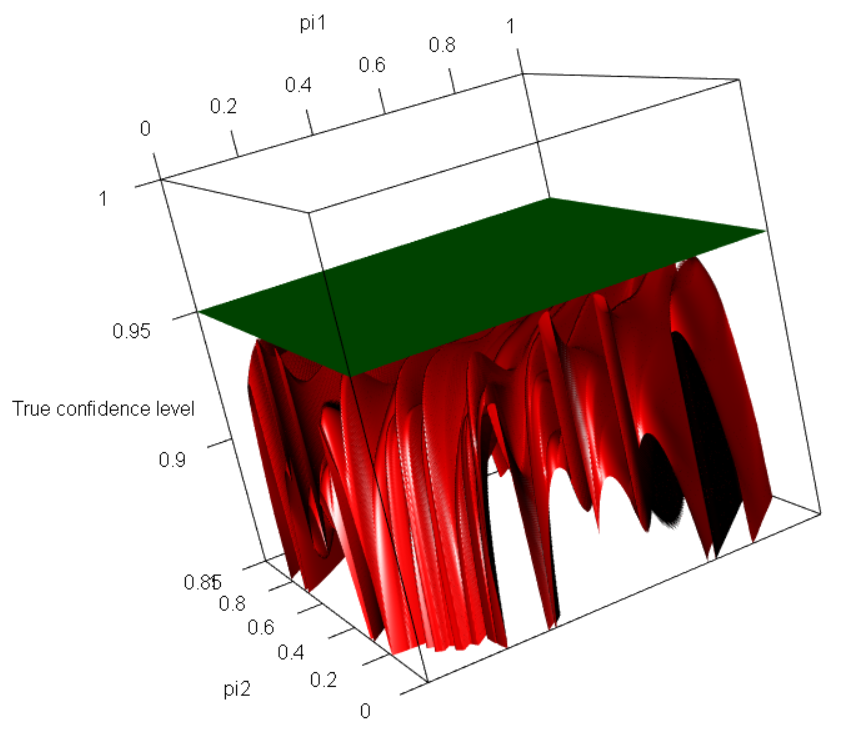
1. Find all possible intervals that one could have with \_\_\_\_\_\_\_\_\_.
2. Form I(\_\_) = 1 if \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ and 0 otherwise.
3. Calculate the true confidence level as \_\_\_\_\_\_\_\_\_\_.

What about other values for n1, n2, and 2? One can use my program to investigate it! For example, here’s what happens when n1 = 40, n2 = 10, 2 = 0.3:

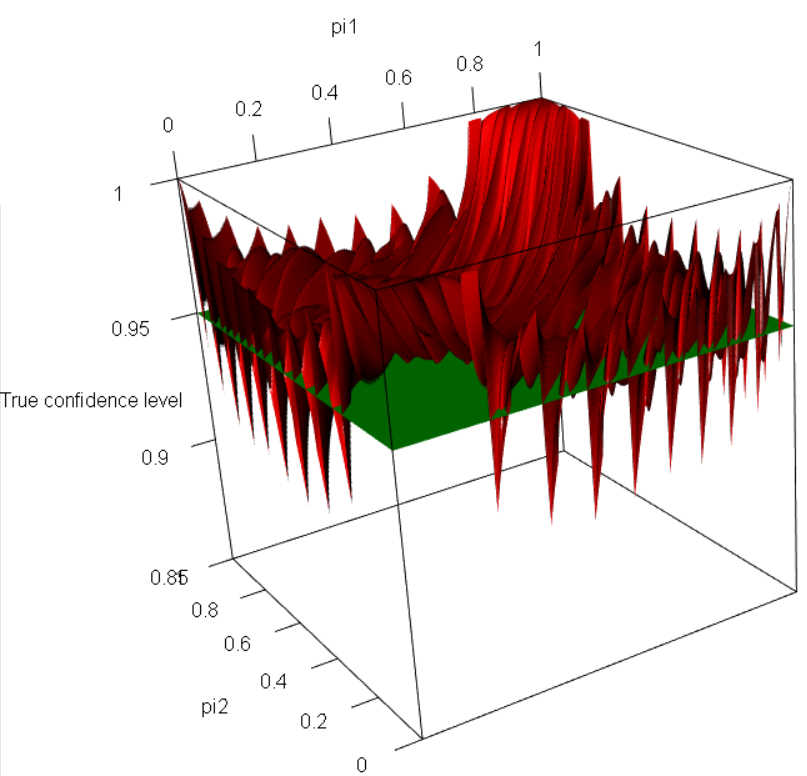


One can allow 1 to vary as well leading to a 3D plot. These plots can be created with the rgl package. The next set of plots use n1 = n2 = 10, and α = 0.05.

Wald interval:



Agresti-Caffo interval:



Overall, we can see the Agresti and Caffo interval tends to be much better than the Wald interval.

Other confidence intervals can be calculated. One interval worth noting is the score interval. This interval takes a score statistic (to be discussed shortly) for the hypothesis test of H0:π1 – π2 = d vs. Ha: π1 – π2 ≠ d:

,

where  and  denote the MLEs of 1 and 2 under the constraint that π1 – π2 = d, and inverts it to find the lower and upper limits for the interval. In other words, find the set of d values such that



is satisfied. There is no closed-form expression that can be written, unlike the Wilson interval for . Therefore, iterative numerical procedures need to be used to find the lower and upper bounds. The diffscoreci() function of the PropCIs package calculates the interval.