Analysis of Wald confidence interval

Vivek Agarwal

This is a demonstration of the fact that the Wald confidence interval does not always have the stated confidence level, $1-\alpha$, where α, is the probability of rejecting the null hypothesis when it is true.

```
require(knitr)
```

```
## Loading required package: knitr
```

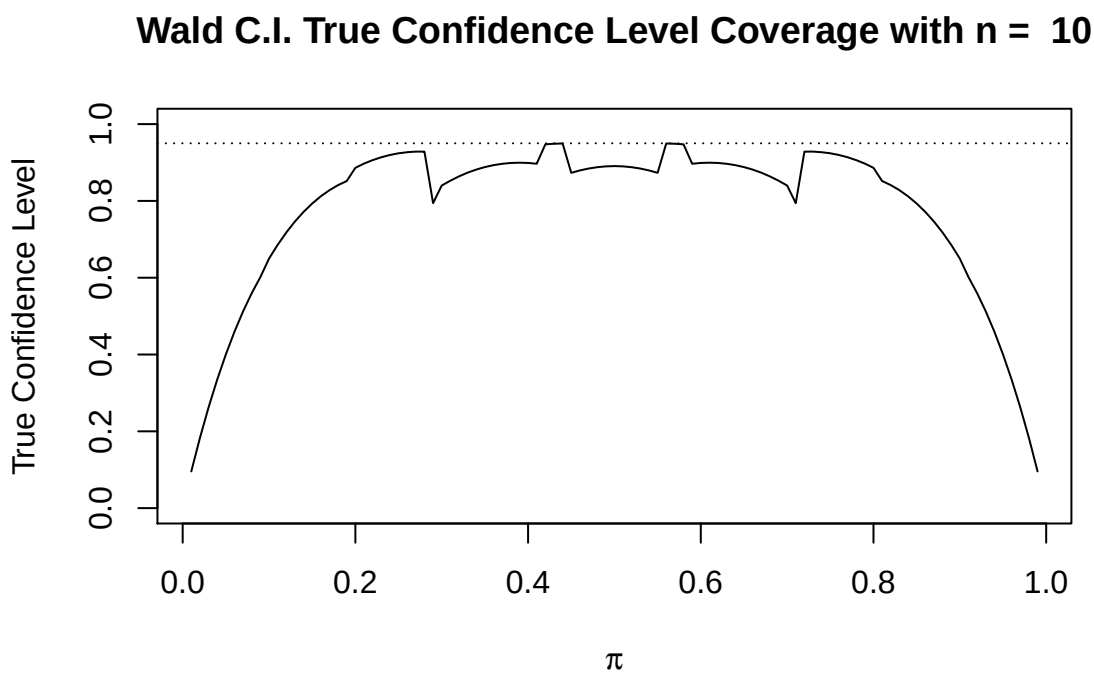
```
## Warning: package 'knitr' was built under R version 3.5.2
```

```
opts_chunk$set(tidy.opts=list(width.cutoff=80),tidy=TRUE)


pi = 0.6 # true parameter value of the probability of success
alpha = 0.05 # significane level
n = 10 # number of trials

wald.CI.true.coverage = function(pi, alpha=0.05, n) {

  # Objective:
  #    Calculate the true confidence level of a Wald Confidence (given pi, alpha, and n)

  # Input:
  #    pi: the true parameter value
  #    alpha: significance level
  #    n: the number of trials

  # Return:
  #    wald.df: a data.frame containing
  #    (1) observed number of success, w
  #    (2) MLE of pi, pi.hat
  #    (3) Binomial probability of obtaining the number of successes from n trials, pmf
  #    (4) lower bound of the Wald confidence interval, wald.CI_lower.bound
  #    (5) upper bound of the Wald confidence interval, wald.CI_upper.bound
  #    (6) whether or not an interval contains the true parameter, covered.pi

  w = 0:n

  pi.hat = w/n

  pmf = dbinom(x=w, size=n, prob=pi)

  var.wald = pi.hat*(1-pi.hat)/n # variance in wald model
```

```
  wald.CI_lower.bound = pi.hat - qnorm(p = 1-alpha/2)*sqrt(var.wald)
  wald.CI_upper.bound = pi.hat + qnorm(p = 1-alpha/2)*sqrt(var.wald)

  covered.pi = ifelse(test = pi>wald.CI_lower.bound,
                      yes = ifelse(test = pi<wald.CI_upper.bound,
                                   yes=1, no=0), no=0)

  wald.CI.true.coverage = sum(covered.pi*pmf)

  wald.df = data.frame(w, pi.hat,
                       round(data.frame(pmf,
                                        wald.CI_lower.bound,
                                        wald.CI_upper.bound),
                             4), covered.pi)

  return(wald.df)
}

# Call the function with user-provided arguments (pi, alpha, n) to
# generate the data.frame that contains
# (1) the observed number of success, w
# (2) MLE of pi, pi.hat
# (3) Binomial probability of obtaining the number of successes from n trials, pmf
# (4) the lower bound of the Wald confidence interval, wald.CI_lower.bound
# (5) the upper bound of the Wald confidence interval, wald.CI_upper.bound
# (6) whether or not an interval contains the true parameter, covered.pi

wald.df = wald.CI.true.coverage(pi=0.6, alpha=0.05, n=10)

# Obtain the true confidence level from the Wald Confidence,
# given pi, alpha, and n
wald.CI.true.coverage.level = sum(wald.df$covered.pi*wald.df$pmf)

# Generalize the above computation to a sequence of pi's

# Generate an example sequence of pi (feel free to make the increment smaller)

draw.true.wald.conf<-function(alpha, n){
  pi.seq = seq(0.01, 0.99, by=0.01)

  # Create a matrix to store (1) pi and (2) the true confidence level of
  # the Wald Confidence Interval corresponding to the specific pi
  wald.CI.true.matrix = matrix(data=NA,nrow=length(pi.seq),ncol=2)

  # Loop through the sequence of pi's to obtain the true confidence level of
  # the Wald Confidence Interval corresponding to the specific pi
  counter=1
```

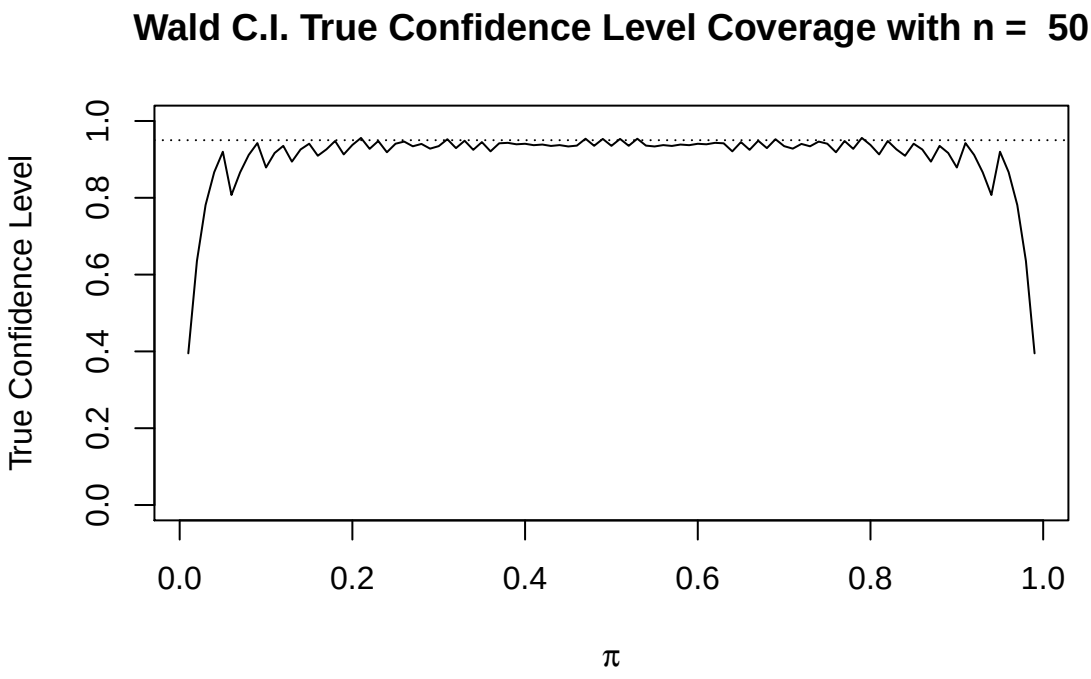
```
  for (pi in pi.seq) {
      wald.df2 = wald.CI.true.coverage(pi=pi, alpha=alpha, n=n)
      #print(paste('True Coverage is', sum(wald.df2$covered.pi*wald.df2$pmf)))
      wald.CI.true.matrix[counter,] = c(pi,sum(wald.df2$covered.pi*wald.df2$pmf))
      counter = counter+1
  }
  #str(wald.CI.true.matrix)
  #wald.CI.true.matrix[1:5,]

  # Plot the true coverage level (for given n and alpha)
  plot(x=wald.CI.true.matrix[,1],
       y=wald.CI.true.matrix[,2],
       ylim=c(0,1),
       main = paste("Wald C.I. True Confidence Level Coverage with n = ", n),
       xlab=expression(pi),
       ylab="True Confidence Level",
       type="l")
  abline(h=1-alpha, lty="dotted")
}
```

```
draw.true.wald.conf(.05, 10)
```

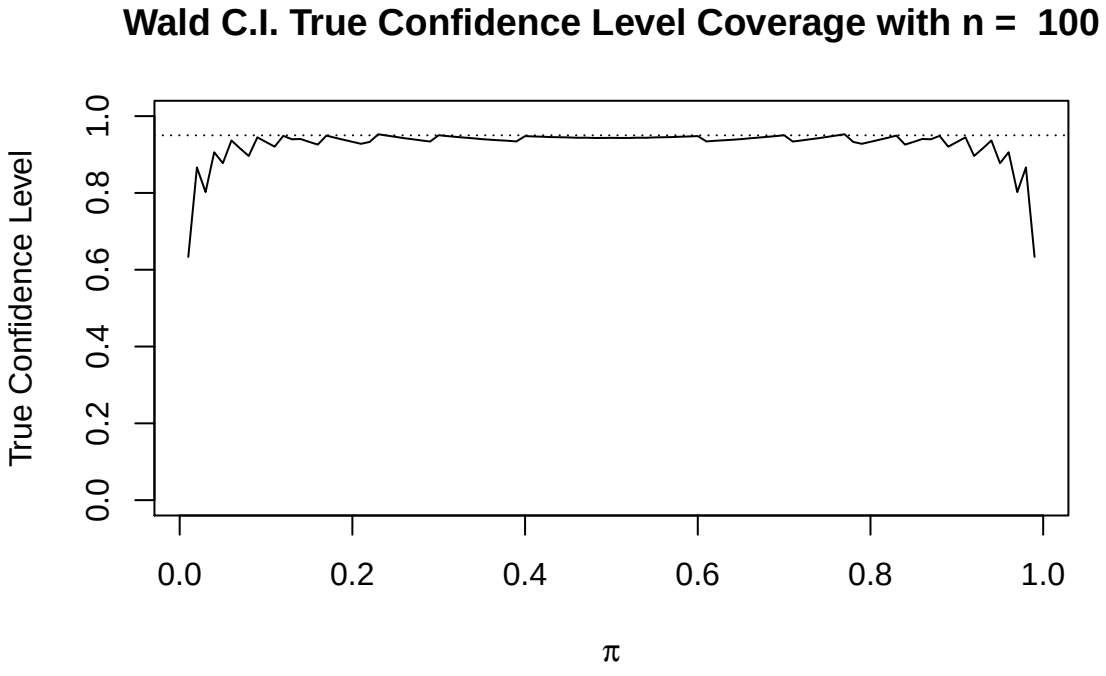


Using the code above to draw graphs for $n = 50, n = 100, n = 500$

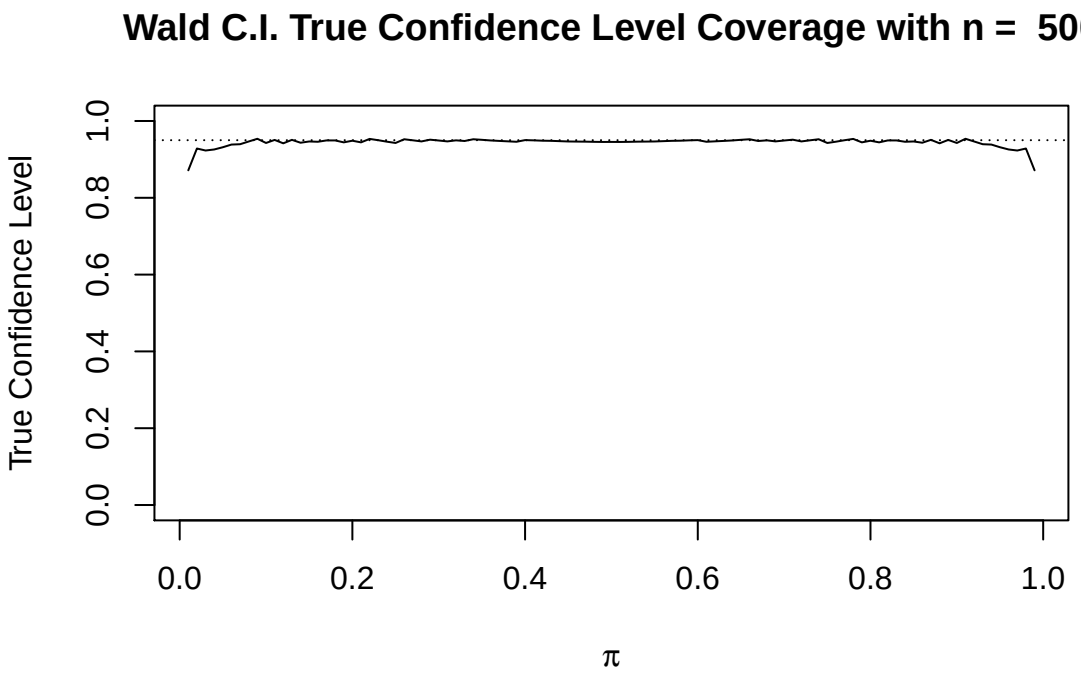
```
draw.true.wald.conf(.05, 50)
```




```
draw.true.wald.conf(.05, 100)
```



```
draw.true.wald.conf(.05, 500)
```



Takeaways:

1. True confidence interval becomes closer to 1 - alpha as n increases.
2. When pi is close to 0 or 1, the true confidence interval is lower as compared to when pi is closer to 0.5. This difference reduces as n increases.
3. True confidence interval stays at or lower than 1-alpha. In other words Wald interval is always greater than or equal to true confidence.