# Assignment 1 - Foundations of Probability

**Name:** Soufiane Fadel
**Date:** September 26, 2018

## Exercise 1

**a)** ♣ Under the binomial distribution assumption, let's note:

$$\begin{aligned} w &: \text{ the number of students who answered "yes" (number of successes)} \\ \pi &: \text{ the probability of success} \\ n &: \text{ the number of students} \end{aligned}$$

Since $w = 0$ and $n = 25$, the "ML" estimate $\hat{\pi} = \frac{w}{n} = \frac{0}{25} = 0$. by using the Wald method, the 95% confidence interval for $\pi$ is :

$$\begin{aligned} \hat{\pi} & \pm Z_{1-\alpha/2}\sqrt{\frac{\hat{\pi}(1-\hat{\pi})}{n}} \quad \text{with} \quad \alpha = 5\% \\ &= 0 \pm 1.96\sqrt{\frac{0(1-0)}{25}} \\ &= (0, 0) \end{aligned}$$

♣ NO! in this case the Wald interval doesn't work, we have just one value for $\pi = 0$. (When the observation falls at the boundary of the sample space, often Wald methods do not provide sensible answers.)

**b)** ♣ **Wilson confidence interval**
the 95% ($\alpha = 5\%$) Wilson interval is:

$$\begin{aligned} \tilde{\pi} & \pm \frac{Z_{1-\alpha/2}\sqrt{n}}{n + Z^2_{1-\alpha/2}}\sqrt{\hat{\pi}(1-\hat{\pi}) + \frac{Z^2_{1-\alpha/2}}{4n}} \quad \textit{where } \tilde{\pi} = \frac{w + Z^2_{1-\alpha/2}/2}{n + Z^2_{1-\alpha/2}} \\ &= \frac{0 + 1.96^2/2}{25 + 1.96^2} \pm \frac{1.96\sqrt{25}}{25 + 1.96^2}\sqrt{0(1-0) + \frac{1.96^2}{4 * 25}} \\ &= (0, 0.133) \end{aligned}$$

♣ **Agresti-Coull confidence interval**
the 95% ($\alpha = 5\%$) Agresti-Coull interval is:

$$\begin{aligned} \tilde{\pi} & \pm Z_{1-\alpha/2}\sqrt{\frac{\tilde{\pi}(1-\tilde{\pi})}{n + Z^2_{1-\alpha/2}}} \quad \textit{where } \tilde{\pi} = \frac{w + Z^2_{1-\alpha/2}/2}{n + Z^2_{1-\alpha/2}} \\ &= \frac{0 + 1.96^2/2}{25 + 1.96^2} \pm \sqrt{\frac{\frac{0+1.96^2/2}{25+1.96^2}\left(1 - \frac{0+1.96^2/2}{25+1.96^2}\right)}{25 + 1.96^2}} \\ &= (-0.0244, 0.1576) \end{aligned}$$

## Exercise 2

```
#####################################################
# Basic computations
#####################################################
> w<-0 # Sum(y_i)
> n<-25
> alpha<-0.05
> pi.hat<-w/n
```

```
> p.tilde<-(w + qnorm(p = 1-alpha/2)^2 /2) / (n + qnorm(p = 1-alpha/2)^2)


####################################################
# Wald C.I
####################################################
> var.wald<-pi.hat*(1-pi.hat)/n
> round(pi.hat + qnorm(p = c(alpha/2, 1-alpha/2)) * sqrt(var.wald),4)

#output
[1] 0 0


####################################################
# Wilson C.I
####################################################
> round(p.tilde + qnorm(p = c(alpha/2, 1-alpha/2)) * sqrt(n) / (n+qnorm(p
     = 1-alpha/2)^2) * sqrt(pi.hat*(1-pi.hat) + qnorm(p = 1-alpha/2)^2/(4*
    n)),4)

#output
[1] 0.0000 0.1332


####################################################
# Agresti-Coull C.I
####################################################
> var.ac<-p.tilde*(1-p.tilde) / (n+qnorm(p = 1-alpha/2)^2)
> round(p.tilde + qnorm(p = c(alpha/2, 1-alpha/2)) *sqrt(var.ac),4)

#output
[1] -0.0244 0.1576
```

**Exercise 3**

**a)** The binomial log likelihood function is :

$$l = \log(\mathcal{L}) = w\log(\pi) + (n-w)\log(1-\pi)$$

Differentiating with respect to $\pi$

$$\frac{\partial l}{\partial \pi} = \frac{w}{\pi} - \frac{n-w}{1-\pi} = \frac{w-n\pi}{\pi(1-\pi)}$$

Equating this to 0 gives the likelihood equation, which has solution $\hat{\pi} = \frac{w}{n}$

**b)**

case 1:

$$\begin{aligned} w = 0 \quad &\Longrightarrow \quad l = n\log(1-\pi) \\ &\Longrightarrow \quad \max_{\pi\in(0,1)}(l) = n\log(1-0) = 0 \end{aligned}$$

case 2:

$$\begin{aligned} w = n \quad &\Longrightarrow \quad l = n\log(1-\pi) \\ &\Longrightarrow \quad \max_{\pi\in(0,1)}(l) = n\log(1) = 0 \end{aligned}$$

case 3:

$$\begin{aligned} 0 < w < n \quad &\Longrightarrow \quad \frac{\partial^2 l}{\partial \pi^2} = -\frac{w}{\pi^2} - \frac{n-w}{(1-\pi)^2} < 0 \\ &\Longrightarrow \quad \text{l is a concave function} \\ &\Longrightarrow \quad \text{l have a unique global maximum which is } \hat{\pi} = \frac{w}{n} \end{aligned}$$

As a conclusion $\hat{\pi} = \frac{w}{n} = \underset{(0,1)}{\text{argmax}}\ l$ is indeed the value that maximizes the log likelihood function ' over the interval $0 \leq \pi \leq 1$.

**Exercise 4** (Exercise 13 in Section 1.3 of the textbook)

**a)**

```
> w <- 4
> n <- 10
> alpha <- 0.05
> pi.hat <- w/n
> library(binom)
> binom.confint(x = w, n = n, conf.level = 1-alpha, methods = "lrt")

#output
  method x n mean lower upper
1 lrt 4 10 0.4 0.1456425 0.7000216
```

**b)**

```
#################################################################
# Basic computations
#################################################################
> alpha<-0.05
> n<-40
> w<-0:n
> pi.hat<- w/n
> pi.seq<-seq(from = 0.001, to = 0.999, by = 0.0005)
> save.true.conf <-0:(length(pi.seq)-1)


###################################################################
# Loop over each pi that the true confidence level is calculated on
###################################################################
> counter<-1
> for(pi in pi.seq) {
        pmf<-dbinom(x = w, size = n, prob = pi)
        LRT.int<-binom.confint(x = w, n = n, conf.level = 1-alpha,
            methods ="lrt")
        save.LRT<-ifelse(test=pi>LRT.int$lower, yes=ifelse( test=pi<
            LRT.int$upper, yes = 1, no = 0), no = 0)
        LRT<-sum(save.LRT*pmf)
        save.true.conf[counter]<- LRT
        counter<-counter+1
  }
```

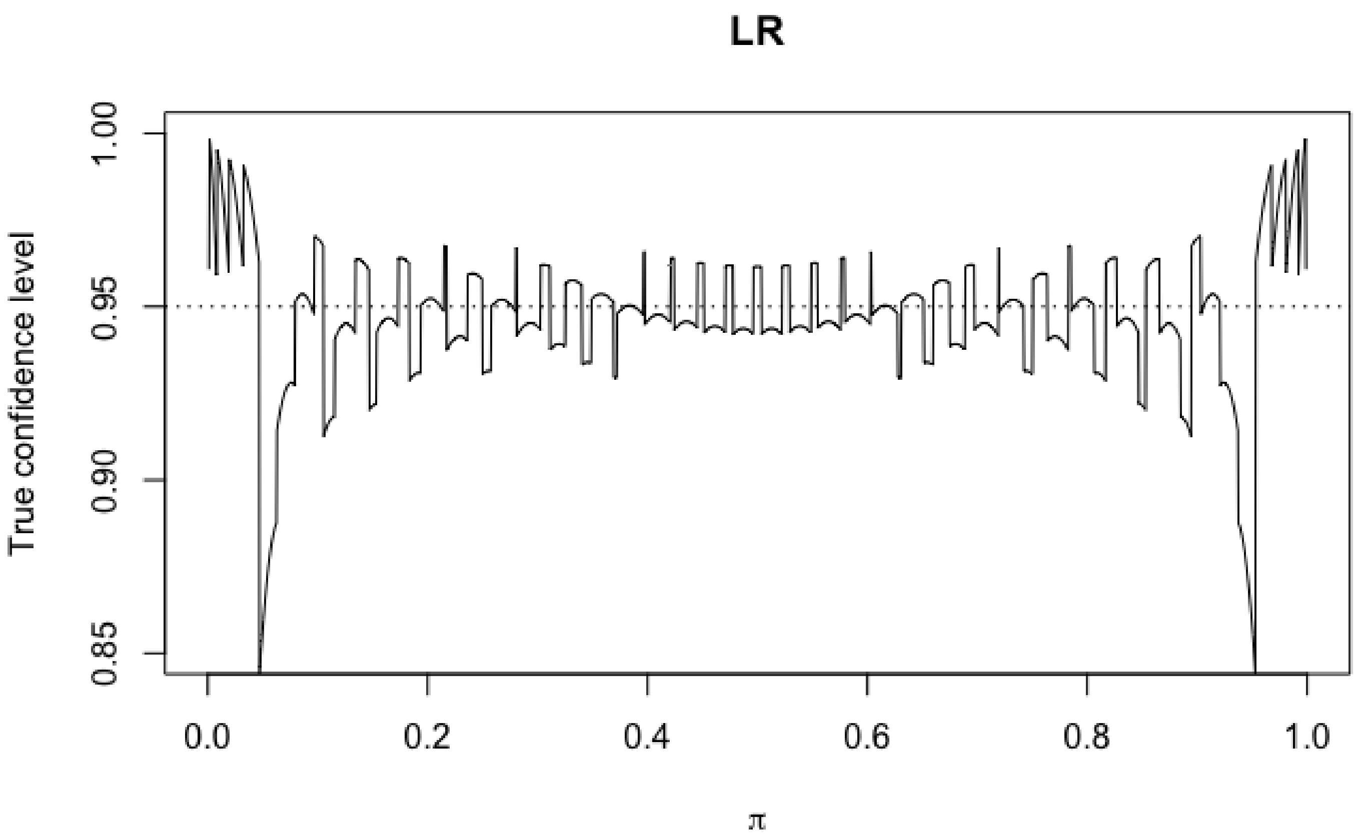


Figure 1: Plot of the true confidence levels

```
#####################################################################
# Plots
#####################################################################
    > plot(x =pi.seq , y = save.true.conf, main = "LR", xlab =
      expression(pi), ylab = "True confidence level", type = "l",
          ylim = c(0.85,1))
    > abline(h = 1-alpha, lty = "dotted")
```

**c)**

Figure bellow is taken from the textbook (figure 1.3), we can affirm that :

- the true confidence level for Wald interval is too low to be on the plot at extreme values of $\pi$, ans also it's below the the state value 0.95, so the Wald interval is exclude.
- The true confidence level interval's for LR is similar to the Wilson interval from values of approximately 0.2 to 0.8.
- We remark that close to 0 and 1 the true confidence LR level decrease, like the Wilson interval.
- the Wilson interval generally is a little better next to 0 and 1.
- The LR interval can be quite conservative, like the Agresti-Coull interval between $0.2ans0.8$, and $\pi$ is close to 0 and 1 the Agresti-Coull is better .
- LR and Clopper-Pearson are quite similar on the extreme value 0 and 1.
- LR confidence intervals often are used in some more complicated contexts where better intervals are not available

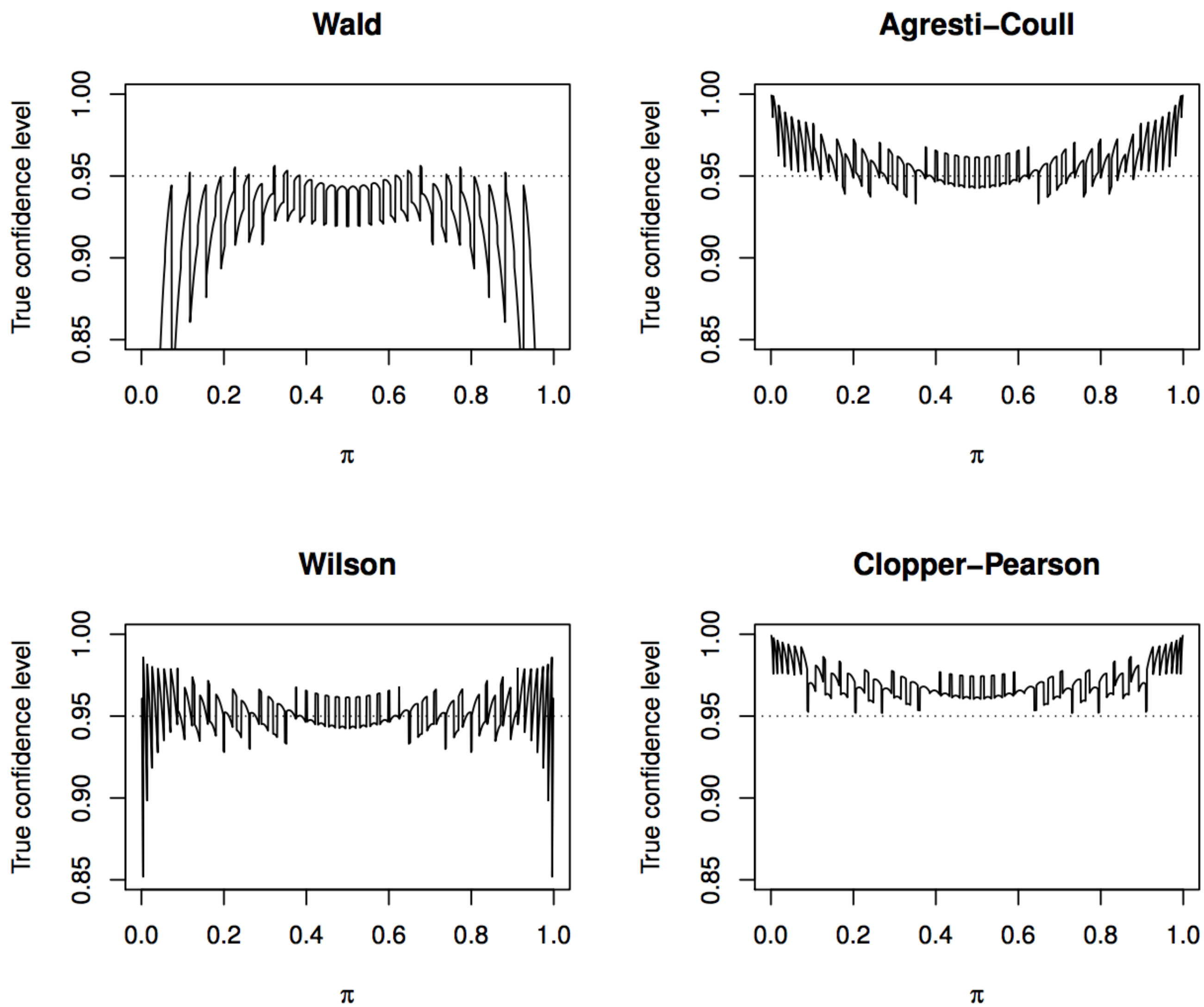


**Figure 1.3**: True confidence levels with $n = 40$ and $\alpha = 0.05$.

$\Longrightarrow$ we can choose the Clopper-Pearson interval as the a good interval because it's the more conservative.

**Exercise 5** (Exercise 15 in Section 1.3 of the textbook)

**a)** We want confidence intervals that place the parameter within as narrow a range as possible, while maintaining at least the stated confidence level. If we wanted intervals that had greater coverage, we would have stated a higher confidence level, but on the other hand we lose the brevity of the interval, it's a (Inverse relation)!

**b)**

```
> alpha<-0.05
> pi<-0.16
> n<-40
> w<-0:n
> pi.hat<-w/n
```

```
> pmf<-dbinom(x = w, size = n, prob = pi)
> var.wald<-pi.hat*(1-pi.hat)/n
> lower<-pi.hat - qnorm(p = 1-alpha/2) * sqrt(var.wald)
> upper<-pi.hat + qnorm(p = 1-alpha/2) * sqrt(var.wald)
> Lenth.wald<-upper-lower
> sum(Lenth.wald*pmf)
[1] 0.2215248
```

**c)**

```
##########################################################
# Basic calculations
##########################################################
alpha <- 0.05
n <- 40
w <- 0:n
pi.hat <- w / n
p.tilde <- (w + qnorm(p = 1 - alpha/2) ^ 2 / 2) / (n + qnorm(1 - alpha/2)^2)
pi.seq <- seq(from = 0.001, to = 0.999, by = 0.0005)

##########################################################
# Save true confidence levels in a matrix
##########################################################
save.exp.lenght <- matrix(data = NA, nrow = length(pi.seq), ncol = 5)

#################################################################
#Loop over each pi that the true confidence level is calculated on
##################################################################
counter <- 1
for (pi in pi.seq) {
  pmf <- dbinom(x = w, size = n, prob = pi)

  #Wald
  lower.wald <- pi.hat - qnorm(p = 1 - alpha/2) * sqrt(pi.hat * (1 - pi.hat)/
      n)
  upper.wald <- pi.hat + qnorm(p = 1 - alpha/2) * sqrt(pi.hat * (1 - pi.hat)/
      n)
  save.wald <- upper.wald - lower.wald
  wald <- sum(save.wald * pmf)

  #Agresti-Coull
  lower.AC <- p.tilde - qnorm(p = 1 - alpha/2) * sqrt(p.tilde * (1 - p.tilde)
      /(n + qnorm(1 - alpha/2)^2))
  upper.AC <- p.tilde + qnorm(p = 1 - alpha/2) * sqrt(p.tilde * (1 - p.tilde)
       / (n + qnorm(1 - alpha/2)^2))
  save.AC <- upper.AC - lower.AC
  AC <- sum(save.AC * pmf)

  #Wilson
  lower.wilson <- p.tilde - qnorm(p = 1 - alpha/2) * sqrt(n) / (n + qnorm(1 -
       alpha/2)^2) * sqrt(pi.hat * (1 - pi.hat) + qnorm(1 -alpha/2)^2 / (4 *
      n))
  upper.wilson <- p.tilde + qnorm(p = 1 - alpha / 2) * sqrt(n) / (n + qnorm(1
       - alpha/2)^2) * sqrt(pi.hat * (1 - pi.hat) + qnorm(1 - alpha/2)^2/ (4
```

```r
    * n))
  save.wilson <- upper.wilson - lower.wilson
  wilson <- sum(save.wilson * pmf)


  #Clopper-Pearson / we use binom.confint()
  library(package = binom)
  lower.CP <- binom.confint(x = w, n = n, conf.level = 1 - alpha, methods = "
    exact" )$lower

  upper.CP <- binom.confint(x = w, n = n, conf.level = 1 - alpha, methods = "
    exact" )$upper

  #Find true confidence level
  save.CP <- upper.CP - lower.CP
  CP <- sum(save.CP * pmf)
  save.exp.lenght[counter, ] <- c(pi, wald, AC, wilson, CP)
  counter <- counter + 1
}
#################################################################
#Plots
#################################################################
plot( x = save.exp.lenght[, 1], y = save.exp.lenght[, 2], main = "Expected
   Length for different methods", xlab = expression(pi), ylab = "Expected
   Length", type = "l", ylim = c(0, 0.35), lty = 5 )

lines(x = save.exp.lenght[, 1], y = save.exp.lenght[, 3], lty = 1, col = "
   blue")

lines(x = save.exp.lenght[, 1], y = save.exp.lenght[, 4], lty = 2, col = "
   red")

lines(x = save.exp.lenght[, 1], y = save.exp.lenght[, 5], lty = 3, col = "
   forestgreen")

legend(x = 0, y = 0.07, legend = c("Wald", "Agresti-Caffo", "Wilson", "
   Clopper-Pearson"), lty = c(5, 1, 2, 3), bty = "n", col = c("black", "blue
   ", "red", "forestgreen"))
```

**d)**

- ♠ it is clear that the "**Agresti-Caffo and Wilson**" method is the most efficient in terms of ( expected length and the true confidence level ).
- ♠ for a value of pi between 0.2 and 0.8 the interval of wald and Clopper-Pearson is a little bigger compared to the others ("Agresti-Caffo" and "Wilson"), except that we see that for the wald true confidence level is quite weak, so there is still a bad choice.

$\Longrightarrow$ As conclusion the best one is the wilson interval because it place the parameter within as narrow a range as possible, while maintaining at least the stated confidence level.

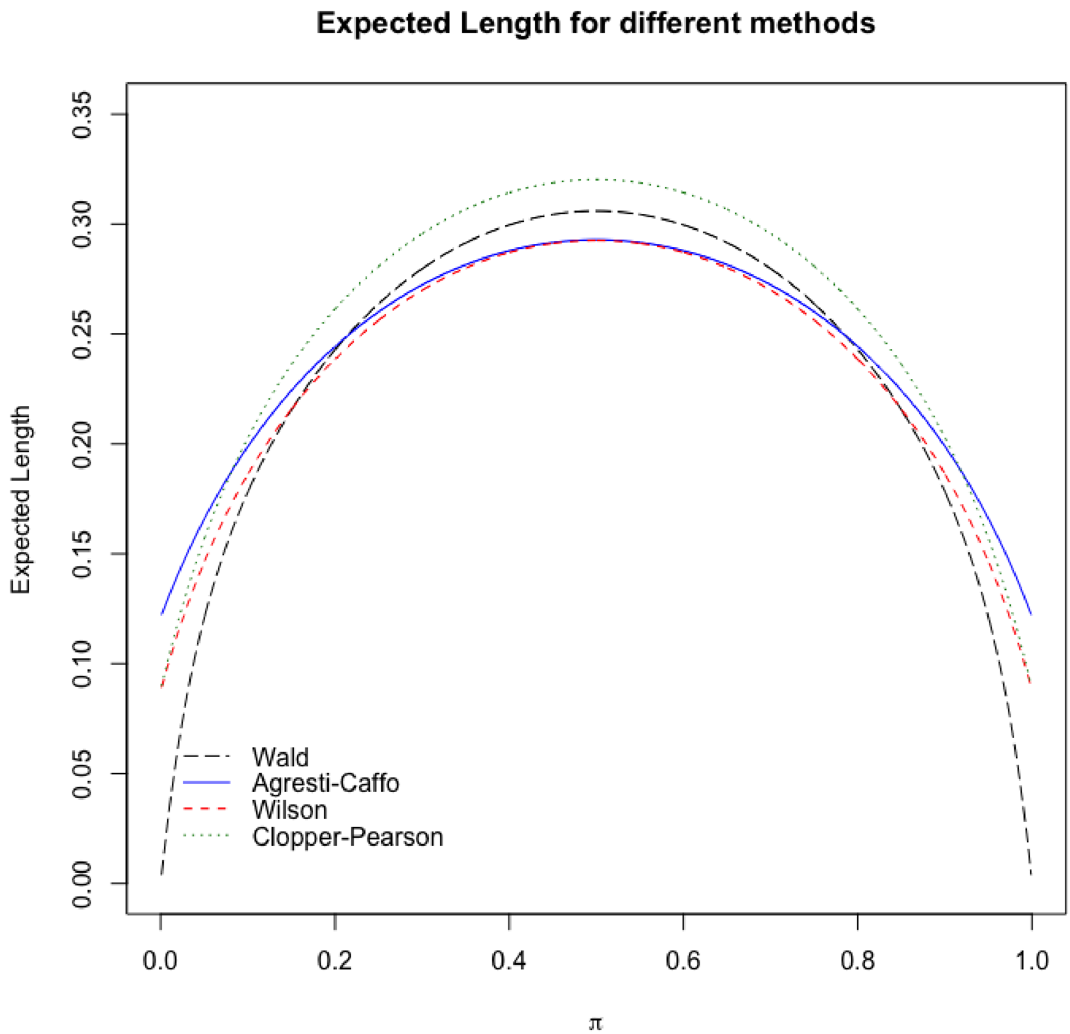


Figure 2: expected length for n = 40, $\pi = 0.16$

**e)**

**Outline the steps that will be needed to find the length using Monte Carlo**

- ♠ Simulate 1,000 samples using the rbinom() function with $n = 40$ and $\pi = 0.16$,
- ♠ Calculate the 95% Wald confidence interval for each sample.
- ♠ Calculate length of intervals
- ♠ Take the mean of those length

**Below is the R code:**

```
##############################################################
# the estimated expected length using Monte Carlo simulation
##############################################################

pi<- 0.16
alpha <- 0.05
```

```
n <- 40

#Number of binomial samples of size n
numb.bin.samples<-1000

set.seed(4716)
w<-rbinom(n = numb.bin.samples, size = n, prob = pi)
counts<-table(x = w)

pi.hat<-w/n
var.wald<-pi.hat*(1-pi.hat)/n
lower<-pi.hat - qnorm(p = 1-alpha/2) * sqrt(var.wald)
upper<-pi.hat + qnorm(p = 1-alpha/2) * sqrt(var.wald)

save<- upper - lower
exp.lenght <- mean(save)

> exp.lenght
[1] 0.2364643
```

**Exercise 6** (Exercise 17 in Section 1.3 of the textbook)

**a)**

♣ **Wald confidence interval for the difference in probabilities of success $\pi_1 - \pi_2$**
the 95% ($\alpha = 5\%$) Wald interval is:

$$\begin{aligned} \hat{\pi}_1 - \hat{\pi}_2 \quad & \pm Z_{1-\alpha/2}\sqrt{\frac{\hat{\pi}_1(1-\hat{\pi}_1)}{n} + \frac{\hat{\pi}_2(1-\hat{\pi}_2)}{n}} \quad \text{with} \quad \alpha = 5\% \\ = \quad & (-0.05362769, 0.49593538) \end{aligned}$$

♣ **R code**

```
c.table<-array(data = c(22, 10, 4, 6), dim = c(2,2), dimnames
    = list(First =c("No time-out", "Time-out"), Second = c("
    Success", "Failure")))

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

alpha<-0.05
pi.hat1<-pi.hat.table[1,1]
pi.hat2<-pi.hat.table[2,1]

#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] -0.05362769 0.49593538
```

♣ **Agresti-Caffo confidence interval for the difference in probabilities of success $\pi_1 - \pi_2$**
the 95% ($\alpha = 5\%$) Wald interval is:

$$\begin{aligned} \tilde{\pi}_1 - \tilde{\pi}_2 & \pm Z_{1-\alpha/2}\sqrt{\frac{\tilde{\pi}_1(1-\tilde{\pi}_1)}{n+2} + \frac{\tilde{\pi}_2(1-\tilde{\pi}_2)}{n+2}} \quad where\ \tilde{\pi}_i = \frac{w_i+1}{n_i+2} \quad i = 1,2 \\ = & \quad (-0.05584623, 0.47648115) \end{aligned}$$

♣ **R code**

```
c.table<-array(data = c(22, 10, 4, 6), dim = c(2,2),dimnames =
    list(First = c("No time-out", "Time-out"), Second = c("
    Success", "Failure")))

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

alpha<-0.05
pi.hat1<-pi.hat.table[1,1]
pi.hat2<-pi.hat.table[2,1]

#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] -0.05584623 0.47648115
```

- **Interpretation**
  the difference in time-out method success probabilities given no time-out method success is between " -0.05584623 & 0.47648115 " for Agresti-Caffo, and "-0.05362769 & 0.49593538 " for Wald.
  Because this interval contains 0, we cannot detect if time-out method is successful or not. This means that either there is no difference, or there is a difference, but it was not detected in this sample.

**b)**

in this our case we have the data as binary observations, so the score statistic is the same as the chi-squared statistic in the Pearson's chi-squared test. Hence **we can calculate Pearson's chi-squared test only**.

♣ **the Pearson's chi-squared test**
The test statistic is formed by computing (observed count - estimated expected count)$^2$/(estimated expected count) over all observed counts, the Pearson chi-square test statistic is :

$$X^2 = \sum_{i=1}^{2}\left(\frac{(w_i - n_i\bar{\pi})^2}{n_i\bar{\pi}} + \frac{(n_i - w_i - n_i(1-\bar{\pi}))^2}{n_i n_i(1-\bar{\pi})}\right)$$

♣ **R code**

```
> prop.test(x = c.table, conf.level = 0.95, correct =
    FALSE)


  #OUTPUT
```

```
#2-sample test for equality of proportions without
    continuity correction

#data: c.table
#X-squared = 2.6704, df = 1, p-value = 0.1022
#alternative hypothesis: two.sided
#95 percent confidence interval:
# -0.05362769 0.49593538
#sample estimates:
# prop 1 prop 2
# 0.8461538 0.6250000
```

♣ **likelihood ratio test**

The LRT test statistic can be shown to be

$$\begin{aligned} -2\log(\lambda) &= -2\log\left[w_1 \log\left(\frac{\tilde{\pi}}{\hat{\pi}_1}\right) + (n_1 - w_1)\log\left(\frac{1-\tilde{\pi}}{1-\hat{\pi}_1}\right)\right. \\ &+ \left. w_2 \log\left(\frac{\tilde{\pi}}{\hat{\pi}_2}\right) + (n_2 - w_2)\log\left(\frac{1-\tilde{\pi}}{1-\hat{\pi}_2}\right)\right] \end{aligned} \quad (2)$$

♣ **R code**

```
 pi.bar <- colSums ( c.table ) [1]/sum( c.table )
log.Lambda <- c.table [1 ,1]*log ( pi.bar/pi.hat.table [1
    ,1]) +
 c.table [1 ,2]*log ((1 - pi.bar ) / (1- pi.hat.table [1
     ,1]) ) + c.table [2 ,1] * log ( pi.bar/pi.hat.table [2
      ,1]) + c.table [2 ,2]*log((1 - pi.bar ) /(1 - pi.hat.
     table [2 ,1]) )
test.stat <- -2* log.Lambda
test.stat
#Output
2.610628
```

**c)**

♣ **R code for the estimated relative risk : $\widehat{RR} : \widehat{\pi}_1/\widehat{\pi}_2$**

```
> cat("The sample relative risk is", round(pi.hat1/pi.hat2, 4)
    , "\n \n")
#OUTPUT
The sample relative risk is 1.3538
```

- **Interpretation**
  A Success is 1.3538 times as likely for those kickers with No time-out method than those with time-out method.

♣ **R code for relative risk confidence interval**

```
 alpha<-0.05
 n1<-sum(c.table[1,])
 n2<-sum(c.table[2,])

ci<-exp(log(pi.hat1/pi.hat2) + qnorm(p = c(alpha/2, 1-alpha/2)
    )* sqrt((1-pi.hat1)/(n1*pi.hat1) + (1-pi.hat2)/(n2*pi.hat2
    )))

> round(ci, 4)
[1] 0.8954 2.0470
```

- **Interpretation**
  The 95% confidence interval is $0.8954 < RR < 2.0470$. Therefore, with 95% confidence, the probability of having success with No time-out methods is between 0.8954 and 2.0470 times as large for those kickers who use time-out.

**d)**

♣ **R code for the estimated odds risk :** $\widehat{OR} : \frac{\widehat{\pi}_1/(1-\widehat{\pi}_1)}{\widehat{\pi}_2(1-\widehat{\pi}_2)}$

```
> OR.hat<- c.table[1,1] * c.table[2,2] / (c.table[2,1] * c.
    table[1,2])

> cat("The sample Odd risk is", round(OR.hat,4), "\n \n")
#OUTPUT
The sample Odd risk is 3.3
```

- **Interpretation**
  The estimated odds of a success are 3.3 times as large as in group with No time-out method than in group using time-out method.

♣ **R code for Odd risk confidence interval**

```
> alpha<-0.05
> var.log.or<-1/c.table[1,1] + 1/c.table[1,2] + 1/c.table[2,1]
     + 1/c.table[2,2]
> OR.CI<-exp(log(OR.hat) + qnorm(p = c(alpha/2, 1-alpha/2))*
    sqrt(var.log.or))
> round(OR.CI, 2)
[1] 0.76 14.34
```

- **Interpretation**
  With 95% confidence, the odds of having success with No time-out methods are between 0.76 and 14.34 times as large when the kickers use time-out methods. throw is made.

**e)** Because 1 is within the interval, there is insufficient evidence to indicate if th icing the kicker is a good strategy to follow