**Chapter 1 – Analyzing a binary response, part 1**

What is a categorical (qualitative) variable?

* Field goal result – success or failure
* Patient survival – yes or no
* Criminal offense convictions – murder, robbery, assault, …
* Highest attained education level – HS, BS, MS, PhD (ordinal properties)
* Food for breakfast – cereal, bagel, eggs,…
* Annual income - <15,000, 15,000-<25,000,
25,000-<40,000, ≥40,000 (ordinal properties)
* Religious affiliation

We live in a categorical world!

Chapters 1 and 2 focus on binary response categorical variables.

**Section 1.1 – One binary variable**

We will examine a binary response variable observed from a homogeneous population.

Goal: Estimate the overall probability of observing one of two possible outcomes for this random variable.

* This is often equated with the “probability of success” for an individual item in the population.
* Equivalently, this is the overall prevalence of successes in the population because each item has the same probability of success.

In the future: Section 1.2 will extend the methods from Section 1.1 to a heterogeneous setting where individual items come from one of two categories. Chapter 2 completes the extension to a heterogeneous population where we use a regression model to estimate the binary response probability.

**Section 1.1.1 – Bernoulli and binomial probability distributions**

Suppose Y = 1 is a success where the probability of a success is P(Y = 1) = π. Also, suppose Y = 0 is a failure. Goal is to estimate π.

Bernoulli probability mass function (PMF)

P(Y = y) =  for y = 0 or 1

Notice that P(Y = 1) = π and P(Y = 0) = 1 – π

Often, you observe multiple success/failure observations. Let Y1, …, Yn denote random variables for these observations. If the random variables are independent and have the same probability of success π, then we can use a binomial PMF for .

Binomial PMF

P(W = w) = 
for w = 0, 1, …, n

Notes:

*  = n choose w
* W is a random variable denoting the number of “successes” out of n trials
* W has a fixed number of possibilities – 0, 1, …, n
* n is a fixed constant
* π is a parameter denoting the probability of a “success” with values between 0 and 1.

Question: Why examine probability distributions?

Example: Field goal kicking

Suppose a field goal kicker attempts 5 field goals during a game and each field goal has the same probability of being successful (the kick is made). Also, assume each field goal is attempted under similar conditions; i.e., distance, weather, surface,….

Below are the characteristics that must be satisfied in order for the binomial distribution to be used.

1. There are n identical trials.

n = 5 field goals attempted under the exact same conditions

1. Two possible outcomes of a trial. These are typically referred to as a success or failure.

Each field goal can be made (success) or missed (failure)

1. The trials are independent of each other.

The result of one field goal does not affect the result of another field goal.

1. The probability of success, denoted by π, remains constant for each trial. The probability of a failure is 1-π.

Suppose the probability a field goal is good is 0.6; i.e., P(success) = π = 0.6.

1. The random variable, W, represents the number of successes.

Let W = number of field goals that are good. Thus, W can be 0, 1, 2, 3, 4, or 5.

Because these 5 items are satisfied, the binomial probability mass function can be used and W is called a binomial random variable.

Mean and variance for Binomial random variable

E(W) = nπ

Var(W) = nπ(1-π)

Proofs would be covered in a mathematical statistics course like UNL’s STAT 462, 880, and 882.

Example: Field goal kicking (Binomial.R)

Suppose π = 0.6, n = 5. What are the probabilities for each possible value of w?

P(W=0) = 

=  ≈ 0.0102

For W=0,…,5:

| **W** | **P(W = w)** |
| --- | --- |
| 0 | 0.0102 |
| 1 | 0.0768 |
| 2 | 0.2304 |
| 3 | 0.3456 |
| 4 | 0.2592 |
| 5 | 0.0778 |

E(W) = nπ = 5\*0.6 = 3 and

Var(W) = nπ(1-π) = 5\*0.6\*(1-0.6) = 1.2

R code and output:

> dbinom(x = 1, size = 5, prob = 0.6)

[1] 0.0768

> dbinom(x = 0:5, size = 5, prob = 0.6)

[1] 0.01024 0.07680 0.23040 0.34560 0.25920 0.07776

> pmf<-dbinom(x = 0:5, size = 5, prob = 0.6)

> save<-data.frame(w = 0:5, prob = round(x = pmf, digits =

 4))

> save

 w prob

1 0 0.0102

2 1 0.0768

3 2 0.2304

4 3 0.3456

5 4 0.2592

6 5 0.0778

> #While not necessary, a new plotting window can be opened

 with

> x11(width = 6, height = 6, pointsize = 12)

> #win.graph() works too for Windows computers

> plot(x = save$w, y = save$prob, type = "h", xlab = "w",

 ylab = "P(W=w)", main = "Plot of a binomial PMF for n=5,

 pi=0.6", panel.first = grid(col="gray", lty="dotted"),

 lwd = 2)

> abline(h = 0) 

Example: Simulating observations from a binomial PMF (binomial.R)

The purpose of this example is to show how one can “simulate” observing a random sample of observations from a population characterized by a binomial distribution.

Why would someone want to do this?

Use the rbinom() function in R.

> #Generate observations from a Binomial distribution

> set.seed(4848)

> bin5<-rbinom(n = 1000, size = 5, prob = 0.6)

> bin5[1:20]

 [1] 3 2 4 1 3 1 3 3 3 4 3 3 3 2 3 1 2 2 5 2

> mean(bin5)

[1] 2.991

> var(bin5)

[1] 1.236155

> table(x = bin5)

x

0 1 2 3 4 5

12 84 215 362 244 83

> hist(x = bin5, main = "Binomial with n=5, pi=0.6, 1000

 bin. observations", probability = TRUE, breaks = -

 0.5:5.5, ylab = "Relative frequency")

> -0.5:5.5

[1] -0.5 0.5 1.5 2.5 3.5 4.5 5.5



Notes:

* The shape of the histogram looks similar to the shape of the actual binomial distribution.
* The mean and variance are close to what we expect them to be!

**Section 1.1.2 – Inference for the probability of success**

Introduction to maximum likelihood estimation

Please note that some of this material is taken from Appendix B.

Suppose the success or failure of a field goal in football can be modeled with a Bernoulli(π) distribution. Let Y = 0 if the field goal is a failure and Y = 1 if the field goal is a success. Then the probability distribution for Y is:

P(Y = y) = 

where π denotes the probability of success.

Suppose we would like to estimate π for a 40 yard field goal. Let y1,…,yn denote a random sample of observed field goal results at 40 yards. Thus, these yi’s are either 0’s or 1’s. Given the resulting data (y1,…,yn), the “likelihood function” measures the plausibility of different values of π:





Suppose w = 4 and n = 10. Given this observed information, we would like to find the corresponding parameter value for π that produces the largest probability of obtaining this particular sample. The following table can be formed to help find this parameter value:

|  |  |
| --- | --- |
| π |  |
| 0.2 | 0.000419 |
| 0.3 | 0.000953 |
| 0.35 | 0.001132 |
| 0.39 | 0.001192 |
| 0.4 | 0.001194 |
| 0.41 | 0.001192 |
| 0.5 | 0.000977 |

Calculations in R (LikelihoodFunction.R in Appendix B):

> sum.y<-4

> n<-10

> pi<-c(0.2, 0.3, 0.35, 0.39, 0.4, 0.41, 0.5)

> Lik<-pi^sum.y\*(1-pi)^(n-sum.y)

> data.frame(pi, Lik)

 pi Lik

1 0.20 0.0004194304

2 0.30 0.0009529569

3 0.35 0.0011317547

4 0.39 0.0011918935

5 0.40 0.0011943936

6 0.41 0.0011919211

7 0.50 0.0009765625

> #Likelihood function plot

> curve(expr = x^sum.y\*(1-x)^(n-sum.y), xlim = c(0,1),

 xlab = expression(pi), ylab = "Likelihood

 function")



Note that π = 0.4 is the “most plausible” value of π for the observed data because this maximizes the likelihood function. Therefore, 0.4 is the maximum likelihood estimate (MLE).

In general, the MLE can be found as follows:

1. Find the natural log of the likelihood function, 
2. Take the derivative of  with respect to π.
3. Set the derivative equal to 0 and solve for π to find the maximum likelihood estimate. Note that the solution is the maximum of  provided certain “regularity” conditions hold (see Mood, Graybill, Boes, 1974).

For the field goal example:



where log means natural log.



 

Therefore, the maximum likelihood estimator of π is the proportion of field goals made. To avoid confusion between a parameter and a statistic, we will denote the estimator as  = /n.

Maximum likelihood estimation will be extremely important in this class!!!

For additional examples with maximum likelihood estimation, please see Section 9.15 of my STAT 380 lecture notes at <http://www.chrisbilder.com/stat380/schedule.htm>.

Properties of maximum likelihood estimators

 will vary from sample to sample. We can mathematically quantify this variation for maximum likelihood estimators in general as follows:

* For a large sample, maximum likelihood estimators can be treated as normal random variables.
* For a large sample, the variance of the maximum likelihood estimator can be computed from the second derivative of the log likelihood function.

Thus, in general for a maximum likelihood estimator  for θ, we can say that



for a large sample Y1, …, Yn, where



Why do you think this is important to know?

The use of “for a large sample” can also be replaced with the word “asymptotically”. You will often hear these results talked about using the phrase “asymptotic normality of maximum likelihood estimators”.

You are not responsible to do derivations as shown in the next example (on exams). This example is helpful to see in order to understand how R will be doing these and more complex calculations.

Example: Field goal kicking



The large sample variance of  is

.

To find this, note that,



 because only the Yi’s

 are random variables





Since π is a parameter, we replace it with its corresponding estimator to obtain .

Thus, = .

This same result is derived on p. 474 of Casella and Berger (2002).

See Chapter 18 of Ferguson (1996) for more on the “asymptotic normality” of maximum likelihood estimators.

Wald confidence interval

Because , we can rewrite this as a standardized statistic:



Also, because we have a probability distribution here, we can quantify with a level of certainty that observed values of the statistic are within a particular range:



where  is the 1 – α/2 quantile from a standard normal. For example, if α = 0.05, we have Z0.975 = 1.96:

> qnorm(p = 1-0.05/2, mean = 0, sd = 1)

[1] 1.959964

Note that I specially chose  and  for symmetry. Of course, .

If we rearrange items within the P(⋅), we obtain



Thus, if α is chosen to be small, we are fairly certain the expression within P(⋅) will hold true. When we substitute the observed values of  and  into the expression, we obtain the (1 – α)100% “Wald” confidence interval for θ as



Notice this interval follows the typical form of a confidence interval for a parameter:

Estimator ± (distributional value)\*(standard deviation of estimator)

Confidence interval for π

Because  is a maximum likelihood estimator, we can use a Wald confidence interval for π:



When  is close to 0 or 1, two problems may occur:

1. Calculated limits may be less than 0 or greater than 1, which is outside the boundaries for a probability.
2. When  = 0 or 1,  = 0 for n > 0. This leads to the lower and upper limits to be exactly the same (0 for  = 0 or 1 for  = 1).

Example: Field goal kicking (CIpi.R)

Suppose  = w = 4 and n = 10. The 95% confidence interval is



0.0964 < π < 0.7036

R code and output:

> w<-4

> n<-10

> alpha<-0.05

> 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)

> round(data.frame(lower, upper), 4)

 lower upper

1 0.0964 0.7036

> #Quicker

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

 sqrt(var.wald), 4)

[1] 0.0964 0.7036

Interpretations:

* With 95% confidence, true probability of success is between 0.0964 and 0.7036.
* The 95% confidence interval is 0.0964 < π < 0.7036. We would expect that 95% of all similarly constructed intervals to contain π.

Incorrect interpretations:

* The probability that π is between 0.0964 and 0.7036 is 95%.
* 95% of the time π is between 0.0964 and 0.7036.
* The 95% confidence interval is 0.0964 < π < 0.7036. We would expect that 95% of all π values to be within these limits.

What’s the difference between “With 95% confidence” and “The probability … is 95%”?

Problems!!!

1. Remember the interval “works” if the sample size is large. The field goal kicking example has n = 10 only!
2. The discreteness of the binomial distribution often makes the normal approximation work poorly even with large samples.

The result is a confidence interval that is often too “liberal”. This means when 95% is stated as the confidence level, the true confidence level is often lower.

On the other hand, there are “conservative” intervals. These intervals have a true confidence level larger than the stated level.

The problems with this particular confidence interval have been discussed for a long time in the statistical literature. There have been many, many alternative confidence intervals for π proposed. Some of the most important papers on this came out around the year 2000:

* Agresti, A. and Coull, B. (1998). Approximate is better than “exact” for interval estimation of binomial proportions. *The American Statistician* 52(2), 119-126.
* Agresti, A. and Caffo B. (2000). Simple and effective confidence intervals for proportions and differences of proportions result from adding two successes and two failures. *The American Statistician* 54(4), 280-288.
* Agresti, A. and Min, Y. (2001). On small-sample confidence intervals for parameters in discrete distributions. *Biometrics* 57, 963-971.
* Blaker, H. (2000). Confidence curves and improved exact confidence intervals for discrete distributions. *The Canadian Journal of Statistics* 28, 783-798.
* Brown, L., Cai, T., and DasGupta, A. (2001). Interval estimation for a binomial proportion. *Statistical Science* 16(2), 101-133.
* Brown, L., Cai, T., and DasGupta, A. (2002). Confidence intervals for a binomial proportion and asymptotic expansions. *Annals of Statistics* 30(1), 160-201.

Brown et al. (2001) serves as a summary of all the proposed methods and gives the following recommendations:

* For n < 40, use the Wilson or Jeffrey’s prior interval.

Wilson interval:



with .

Where does interval come from?

Consider the hypothesis test for H0:π = π0 vs. Ha:π ≠ π0 using the test statistic of



The limits of the Wilson confidence interval come from “inverting” the test. This means finding the set of π0 such that



is satisfied. Through solving a quadratic equation, the interval limits are derived. Going through the derivation is part of the homework.

Jeffrey’s prior interval:

The Wald and Wilson intervals are frequentist inference procedures. When α = 0.05, they can be interpreted as:

We would expect 95% of all similarly constructed intervals to contain the parameter π.

Of course, this does not mean that the one interval obtained from one sample has a 95% probability of containing the parameter.

A Bayesian credible interval gives you the interpretation that ONE interval has a 95% probability of containing π when α = 0.05. A Jeffrey’s prior interval is one of these intervals. Please see Section 6.6.1 for a discussion.

* For n ≥ 40, use the Agresti-Coull (Agresti and Coull, 1998) interval.

The (1-α)100% confidence interval is



This is essentially a Wald interval where we add  successes and  failures to the observed data. In fact, when α = 0.05, Z1-α/2 = 1.96 ≈ 2. Then

.

Thus, two successes and two failures are added. Also, notice how



can be thought of as an adjusted estimate of π. For values of w close to 0, . For values of w close to n, . Think about how this affects the interval!

When n < 40, the Agresti-Coull interval is generally still better than the Wald interval. The Wilson interval can be used when n ≥ 40 as well, and it is generally better than the Agresti-Coull. The two advantages that the Agresi-Coull interval can have over the Wilson are:

1. A little easier to calculate by hand. Of course, why would someone do the calculations by hand in the first place?!
2. For VERY extreme values of π, the Wilson interval can be liberal and the Agresti-Coull interval can be conservative. However, there are small fixes that can be performed to improve their performance. Please see Section 1.1.3 in the book for a discussion.

Example: Field goal kicking (CIpi.R)

Below is the code used to calculate each confidence interval. Note that I should have used pi.tilde rather than p.tilde as the name for !

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

 = 1-alpha/2)^2)

> p.tilde

[1] 0.4277533

> #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(1-alpha/2)^2/(4\*n)), 4)

[1] 0.1682 0.6873

> #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)

[1] 0.1671 0.6884

How useful would these confidence intervals be?

The binom package in R provides a simple function to do these calculations as well. Here is an example of how I used its function:

> library(package = binom)

> binom.confint(x = w, n = n, conf.level = 1 –

 alpha, methods = "all")

 method x n mean lower upper

1 agresti-coull 4 10 0.4000000 0.16711063 0.6883959

2 asymptotic 4 10 0.4000000 0.09636369 0.7036363

3 bayes 4 10 0.4090909 0.14256735 0.6838697

4 cloglog 4 10 0.4000000 0.12269317 0.6702046

5 exact 4 10 0.4000000 0.12155226 0.7376219

6 logit 4 10 0.4000000 0.15834201 0.7025951

7 probit 4 10 0.4000000 0.14933907 0.7028372

8 profile 4 10 0.4000000 0.14570633 0.6999845

9 lrt 4 10 0.4000000 0.14564246 0.7000216

10 prop.test 4 10 0.4000000 0.13693056 0.7263303

11 wilson 4 10 0.4000000 0.16818033 0.6873262

A modification of the Wilson interval formula is used by this function when calculating the interval’s limits at particular values of w. When w = 1, the lower limit of the interval is replaced by –log(1 – α)/n. When w = n – 1, the upper limit of the interval is replaced by 1+ log(1 – α)/n. Please see my book for details for why this modification is used.

We will learn about another confidence interval for π, called the “Clopper-Pearson” interval, a little later in Section 1.1.

**Section 1.1.3 – True confidence levels for confidence intervals**

Below is a comparison of the performance of the four confidence intervals (Brown et al. 2001). The values on the y-axis represent the true confidence level (coverage) of the confidence intervals. Each of the confidence intervals are supposed to be 95%!



What does the “true confidence” or “coverage” level mean?

* Suppose a random sample of size n = 50 is taken from a population and a 95% Wald confidence interval is calculated.
* Suppose another random sample of size n = 50 is taken from the same population and a 95% Wald confidence interval is calculated.
* Suppose this process is repeated 10,000 times.
* We would expect 9,500 out of 10,000 (95%) confidence intervals to contain π.
* Unfortunately, this does not often happen. It only is guaranteed to happen when n = ∞ for the Wald interval.
* The true confidence or coverage level is the percent of times the confidence intervals contain or “cover” π.

How can this “true confidence” or “coverage” level be calculated?

Simulate the process of taking 10,000 samples and calculating the confidence interval each time using a computer. This is called a Monte Carlo simulation.

The plots of the previous page essentially do this for many possible values of π (0.0005 to 0.9995 by 0.0005). For example, the true confidence level using the Wald interval is approximately 0.90 for π = 0.184.

Note: The authors of the paper actually use a different method which does not require computer simulation (to be discussed shortly). However, computer simulation will provide just about the same answer if the number of times the process (take a sample, calculate the interval) is repeated a large number of times. Plus, it can be used in many more situations where computer simulation is the only option.

Brown et al. (2001) also discuss other confidence intervals. After the paper, there are a set of responses by other statisticians. Each agrees the Wald interval should NOT be used; however, they are not all in agreement with which other interval to use!

Example: Calculate **estimated** true confidence level for Wald (ConfLevel.R)

To match what was given in the plots, I changed n to 50 in the corresponding program for this example. Also, I changed π to be 0.184. Only 1,000 samples are taken (rather than 10,000 mentioned previously) to speed up the in-class demonstration.

> pi<-0.184

> alpha<-0.05

> n<-50

> w<-0:n

> pi.hat<-w/n

> numb.bin.samples<-1000 #Number of binomial samples of

 size n

> set.seed(4516)

> w<-rbinom(n = numb.bin.samples, size = n, prob = pi)

> counts<-table(x = w)

> counts

x

 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17

 3 6 28 51 70 122 118 144 146 113 85 47 34 19 10 4

> pi.hat<-w/n

> pi.hat[1:10]

 [1] 0.18 0.18 0.20 0.22 0.24 0.18 0.22 0.10 0.16 0.12

> 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)

> data.frame(w, pi.hat, lower, upper)[1:10,]

 w pi.hat lower upper

1 9 0.18 0.07351063 0.2864894

2 9 0.18 0.07351063 0.2864894

3 10 0.20 0.08912769 0.3108723

4 11 0.22 0.10517889 0.3348211

5 12 0.24 0.12162077 0.3583792

6 9 0.18 0.07351063 0.2864894

7 11 0.22 0.10517889 0.3348211

8 5 0.10 0.01684577 0.1831542

9 8 0.16 0.05838385 0.2616161

10 6 0.12 0.02992691 0.2100731

> save<-ifelse(test = pi>lower, yes = ifelse(test =

 pi<upper, yes = 1, no = 0), no = 0)

> save[1:10]

 [1] 1 1 1 1 1 1 1 0 1 1

> mean(save)

[1] 0.898

> true.conf<-mean(save)

> cat("An estimate of the true confidence level is:",

 round(true.conf,4), "\n")

An estimate of the true confidence level is: 0.898

Important: Notice how the calculations are performed using vectors.

What we are doing here is actually estimating the probabilities for  = 0, 1, …, n through the use of Monte Carlo simulation. We sum up the probabilities for wwhen its corresponding confidence interval contains π. Through calculating each possible confidence interval for π when n = 50, one will notice that π = 0.184 is within the interval for w = 6 to 15. Thus,

> table(w) # Frequency for each w observed

w

 2 3 4 5 6 7 8 9 10 11 12 13

 3 6 28 51 70 122 118 144 146 113 85 47

 14 15 16 17

 34 19 10 4

> prop.w<-table(w)/numb.bin.samples #Proportion for each w

> obs.w<-as.integer(names(table(w))) # Obtain w number

> binom.prob<-round(dbinom(x = obs.w, size = n, prob =

 pi),4)

> data.frame(w = obs.w, obs.prop = prop.w, binom.prob = binom.prob)

 w obs.prop.w obs.prop.Freq binom.prob

1 2 2 0.003 0.0024

2 3 3 0.006 0.0086

3 4 4 0.028 0.0229

4 5 5 0.051 0.0474

5 6 6 0.070 0.0802

6 7 7 0.122 0.1137

7 8 8 0.118 0.1378

8 9 9 0.144 0.1451

9 10 10 0.146 0.1341

10 11 11 0.113 0.1100

11 12 12 0.085 0.0806

12 13 13 0.047 0.0531

13 14 14 0.034 0.0317

14 15 15 0.019 0.0171

15 16 16 0.010 0.0084

16 17 17 0.004 0.0038

> sum(prop.w)

[1] 1

> sum(binom.prob) # Note: not equal to 1 because some possible values of w were not observed

[1] 0.9969

One could find the EXACT true confidence level without Monte Carlo simulation! Below are the steps:

1. Find all possible intervals that one could have with w = 0, 1, …, n.
2. Form I(w) = 1 if the interval for a w contains π and 0 otherwise.
3. Calculate the true confidence level as



This is what Brown et al. (2001) did for their paper. The key to using a non-simulation based approach is there are a finite number of possible values for the random variable of interest. In other settings beyond confidence intervals for π, this will usually not occur and simulation will be the only approach for a finite sample size n.

Example: Calculate ~~estimated~~ true confidence level for Wald (ConfLevel.R)

> alpha<-0.05

> pi<-0.184

> n<-50

> 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)

> save<-ifelse(test = pi>lower, yes = ifelse(test =

 pi<upper, yes = 1, no = 0), no = 0)

> sum(save\*pmf)

[1] 0.9034437

> data.frame(w, pi.hat, round(data.frame(pmf, lower,

 upper),4), save)

 w pi.hat pmf lower upper save

1 0 0.00 0.0000 0.0000 0.0000 0

2 1 0.02 0.0004 -0.0188 0.0588 0

3 2 0.04 0.0024 -0.0143 0.0943 0

4 3 0.06 0.0086 -0.0058 0.1258 0

5 4 0.08 0.0229 0.0048 0.1552 0

6 5 0.10 0.0474 0.0168 0.1832 0

7 6 0.12 0.0802 0.0299 0.2101 1

8 7 0.14 0.1137 0.0438 0.2362 1

9 8 0.16 0.1378 0.0584 0.2616 1

10 9 0.18 0.1451 0.0735 0.2865 1

11 10 0.20 0.1341 0.0891 0.3109 1

12 11 0.22 0.1100 0.1052 0.3348 1

13 12 0.24 0.0806 0.1216 0.3584 1

14 13 0.26 0.0531 0.1384 0.3816 1

15 14 0.28 0.0317 0.1555 0.4045 1

16 15 0.30 0.0171 0.1730 0.4270 1

17 16 0.32 0.0084 0.1907 0.4493 0

18 17 0.34 0.0038 0.2087 0.4713 0

<OUTPUT EDITED>

> sum(dbinom(x = 6:15, size = n, prob = pi))

[1] 0.9034437

How can you perform this simulation for more than one π and then produce a coverage plot similar to p. 1.31? See ConfLevelsWaldOnly.R and ConfLevel4Intervals.R discussed in my book.

Why does the true confidence level (coverage) oscillate? The reason involves the discreteness of a binomial random variable. Please see my book for an example with n = 40 and π = 0.156 or 0.157.

Clopper-Pearson interval

Perhaps you always want to a have true confidence level at or greater than the stated level. If so, you can use the Clopper-Pearson interval!

First, we need to discuss what a beta probability distribution represents. Let V be a random variable from a beta distribution. Then



for 0 < v < 1, a > 0, and b > 0. Note that Γ( ) is the “gamma” function where Γ(c) = (c-1)! for an integer c. The gamma function is more generally defined as

 for c>0.

The α quantile of a beta distribution, denoted by vα or beta(α; a, b), is

.

Below are plots of various Beta distributions from Welsh (1996). The left extreme of each plot denotes 0 and the right extreme of each plot denotes 1 because 0 < v < 1 by definition for the distribution.



The (1-α)100% Clopper-Pearson confidence interval is

beta(α/2; w, n-w+1) < π < beta(1-α/2; w+1, n-w)

This is derived using the relationship between the binomial and beta distributions (see problem #2.40 on p. 82 of Casella and Berger, 2002). If w = 0, the lower limit is taken to be 0. If w = n, the upper limit is taken to be 1. Because a beta distribution is used and because how lower and upper limits are found when w = 0 or n, the interval’s limits are always between 0 and 1!

Often, the interval is stated using an F-distribution instead. This comes about the relationship between beta and F-distributions. The two intervals are the same in the end. I use the beta distribution version becomes it is more compact and easy to calculate via a computer.

This Clopper-Pearson (CP) interval is GUARANTEED to have a true confidence (coverage) level ≥ 1-α! While this is good, the confidence interval is usually wider than other intervals. Why would a wide interval be bad?

Brown et al. (2001, p. 113) say this interval is “wastefully conservative and is not a good choice for practical use”. Examine the coverage plot for it (n=50, p. 113 of Brown et al. 2001):



Maybe this is too harsh?

Intervals like the Clopper-Pearson are called “exact” confidence intervals because they use the exact distribution for W (binomial). This does not mean their true confidence level is exactly (1-α)100% as you can see above.

The Blaker (2000, 2001) interval provides a little less conservative version of an exact interval. If you are interested in details about it, please see the paper and a short discussion in my book.

Example: Field goal kicking (CIpi.R)

Below is the code used to calculate the Clopper-Pearson confidence interval.

> #Clopper-Pearson

> round(qbeta(p = c(alpha/2, 1-alpha/2), shape1 =

 c(w, w+1), shape2 = c(n-w+1, n-w)),4)

[1] 0.1216 0.7376

The binom package calls the Clopper-Pearson interval the “exact” interval.

> library(binom)

> binom.confint(x = w, n = n, conf.level = 1 –

 alpha, methods = "exact")

 method x n mean lower upper

1 exact 4 10 0.4 0.1215523 0.7376219

A summary of all intervals:

|  |  |  |
| --- | --- | --- |
| **Name**  | **Lower** | **Upper** |
| Wald | 0.0964 | 0.7036 |
| Agresti-Coull | 0.1671 | 0.6884 |
| Wilson | 0.1682 | 0.6873 |
| Clopper-Pearson | 0.1216 | 0.7376 |
|  |  |  |
| Jeffrey’s | 0.1531 | 0.6963 |
| Blaker | 0.1500 | 0.7171 |

Hypothesis tests for π

When only one simple parameter is of interest, such as π, confidence intervals are generally preferred over hypothesis tests. This is because a confidence interval gives a range of possible parameter values, which a hypothesis test cannot.

Still, if you want to perform a hypothesis test of H0:π = π0 vs. Ha:π ≠ π0, one way is to use the test statistic of



as mentioned earlier when discussing the Wilson confidence interval. You can reject H0 if |Z0| > Z1-α/2. This type of test is usually called a “score” test. Score tests are general likelihood-based inference procedures – please see p. 494 of Casella and Berger (2002) for details and Appendix B. Because the Wilson interval results from a score test, the interval is often referred to as a “score interval” as well.

Another way to perform a hypothesis test of H0:π = π0 vs. Ha:π ≠ π0 is a likelihood ratio test (LRT). Because this test is VERY frequently used in the analysis of categorical data (beyond testing for π), we will spend some time discussing it here.

The LRT statistic, Λ, is the ratio of two likelihood functions. The numerator is the likelihood function maximized over the parameter space restricted under the null hypothesis. The denominator is the likelihood function maximized over the unrestricted parameter space. The test statistic is written as:



Wilks (1935, 1938) shows that –2log(Λ) can be approximated by a  for a large sample and under Ho where u is the difference in dimension between the alternative and null hypothesis parameter spaces. See Casella and Berger (2002, p. 374) and Appendix B for more background on the LRT.

Questions:

* Suppose Λ is close to 0, what does this say about H0? Explain.
* Suppose Λ is close to 1, what does this say about H0? Explain.
* When using –2log(Λ), will large or small value values indicate H0 is false?

Example: Field goal kicking (no program)

Continuing the field goal example, suppose the hypothesis test H0:π = 0.5 vs. Ha:π ≠ 0.5 is of interest. Remember that w = 4 and n = 10.

The numerator of Λ is the maximum possible value of the likelihood function under the null hypothesis. Because π = 0.5 is the null hypothesis, the maximum can be found by just substituting π = 0.5 in the likelihood function:



Then



The denominator of Λ is the maximum possible value of the likelihood function under the null OR alternative hypotheses. Because this includes all possible values of π here, the maximum is achieved when the MLE is substituted for π in the likelihood function! As shown previously, the maximum value is 0.001194.

Therefore,



Then –2log(Λ) = -2log(0.8179) = 0.4020 is the test statistic value. The critical value is  = 3.84 using α = 0.05:

> qchisq(p = 0.95, df = 1)

[1] 3.841459

There is not sufficient evidence to reject the hypothesis that π = 0.5.