**Chapter 6 – Additional topics**

**Section 6.2 – Exact inference**

Most of the inference procedures examined so far rely on a statement similar to the following:

As the sample size goes to infinity, the statistic's distribution approaches a chi-square (or normal) distribution.

Infinity is the one sample size that we can never have! Thus, we have not used the EXACT distribution for our statistics.

Fortunately, there are many situations where distributions, like a chi-square, serve as a good approximation to the actual distribution for the statistic. However, especially when the sample size is small, these distributions often serve as poor approximations. Here’s a quote R. A. Fisher’s *Statistical Methods for Research Workers* 1st edition (1926) book on this topic:

… the traditional machinery of statistical processes is wholly unsuited to the needs of practical research. Not only does it take a cannon to shoot a sparrow, but it misses the sparrow! The elaborate mechanism built on the theory of infinitely large samples is not accurate enough for simple laboratory data. Only by systematically tackling small sample problems on their merits does it seem possible to apply accurate tests to practical data.

The purpose of this section is to develop inference procedures that do not rely on large-sample approximations. These inference procedures are often referred to as EXACT in the sense that they use the *actual* distribution for a statistic.

**Section 6.2.1 – Fisher's exact test for independence**

Hypergeometric distribution

Here’s the classic set up for a random variable with a hypergeometric probability distribution:

Suppose an urn has a red balls and b blue balls with n = a + b. Suppose k ≤ n balls are randomly drawn from the urn without replacement. Let M be the number of red balls drawn out.

The random variable M has a hypergeometric distribution with density function of

 for m = 0, 1,…, k

subject to m ≤ a and k – m ≤ b. Note that a, n, b, and k are FIXED values. The only random variable is M!

An important part of this set up is that each ball has an equal probability of being drawn out of the urn, regardless of color.

Example: Balls in an urn (Tea.R)

Suppose there are n = 8 balls in an urn with a = 4 of them red and b = 4 of them blue. Suppose k = 4 balls are drawn from the urn. What is the probability of observing M = 3 red balls?



The probability mass function is

|  |  |
| --- | --- |
| m | P(M = m) |
| 0 | 0.0143 |
| 1 | 0.2286 |
| 2 | 0.5143 |
| 3 | 0.2286 |
| 4 | 0.0143 |

Is it reasonable to observe M ≥ 3?

R code and output:

> # All possible probabilities

> M <- 0:4

> # Syntax for dhyper(m, a, b, k)

> data.frame(M, prob = round(dhyper(M, 4, 4, 4), 4))

 M prob

1 0 0.0143

2 1 0.2286

3 2 0.5143

4 3 0.2286

5 4 0.0143

Fisher’s exact test

The hypergeometric distribution can be used to find the probability of observing a particular contingency table under independence! Below are tables demonstrating how:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Response |  |
|  |  | 1 | 2 |  |
| Group | 1 | w1 | n1 – w1 | n1 |
| 2 | w2 | n2 – w2 | n2 |
|  |  | w+ | n – w+ | n |
|  |  |  |  |  |
|  |  | Urn |  |
|  |  | Drawn out | Remaining |  |
| Color | Red | m | a – m | a |
| Blue | k – m | b – k + m | b |
|  |  | k | n – k | n |

With the binomial distribution model for these contingency tables that were first seen in Section 1.2, we only had n1 and n2 (row totals) as fixed prior to taking a sample. In order to use the hypergeometric, we would also need to assume W+ and n – W+ (column totals) as fixed. This can be true for some situations, but it will not be for most. Fortunately, one can condition on W+ in the binomial model to obtain the hypergeometric model. The mathematical details are discuss in Exercise #1 of the book.

Question: If n1, n2, and w+ are known, how many cell values in the 2×2 portion of the table need to be known before all four values are known?

For the first table under independence, each element of groups 1 and 2 has the same probability of obtaining the 1 response. This is the same as each ball in the second table having the same probability of being drawn out of the urn, regardless of color.

Example: Lady tasting tea (Tea.R)

Sir Ronald Fisher exploited the use of the hypergeometric distribution one day in Cambridge, England, in the late 1920s. During an afternoon tea with a group of colleagues, a “lady” in this group (Muriel Bristol) claimed that she could determine whether the milk or the tea was poured first into a cup. Fisher devised the follow experiment to test this lady’s claim:

1. Eight cups were set aside for the experiment.
2. Tea was poured first into 4 cups and milk was poured first into the remaining 4 cups.
3. The lady was blinded to which had tea or milk poured first, but she knew there were four of each.
4. The lady tasted the cups of tea and provided a tea or milk poured first response.

Salsburg (2001) gives a more thorough discussion in his book [“The Lady Tasting Tea: How Statistics Revolutionized Science in the Twentieth Century”](http://www.amazon.com/Lady-Tasting-Tea-Statistics-Revolutionized/dp/0716741067).

Below is a **hypothetical** outcome of the experiment:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Lady’s response |  |
|  |  | Milk | Tea |  |
| Actual | Milk | 3 | 1 | 4 |
| Tea | 1 | 3 | 4 |
|  |  | 4 | 4 | 8 |

If the lady could not actually differentiate which was added to a cup first, then her selection of milk responses would in reality be four random cups, and the odds of a milk response would be the same regardless of whether milk or tea was truly added first (odds for row 1 equal to odds for row 2). Thus, we have independence.

Under independence, the probabilities for each possible contingency table are the same as we found earlier with the hypergeometric:

|  |  |  |
| --- | --- | --- |
| m | P(M = m) |  |
| 0 | 0.0143 | 0 |
| 1 | 0.2286 | 1/9 |
| 2 | 0.5143 | 1 |
| 3 | 0.2286 | 9 |
| 4 | 0.0143 | >9 |

This table could have been written in terms of w1 as well. I have added odds ratios to the table to help you see the amount of observed dependence.

P-values are probabilities an event is at least as extreme as what was observed. Under independence, the above table then provides information needed to compute a p-value for a test of independence! Thus, the probability of randomly guessing three or more of the milk cups correctly is P(M ≥ 3) = 0.2286 + 0.0143 = 0.2429.

Questions:

* Would you be surprised if the lady obtained 3 or more milk cups correct if she did not truly know the difference?
* Would you be surprised if the lady obtained 4 milk cups correct if she did not truly know the difference?
* What does OR > 1 mean here?
* What does OR < 1 mean here?
* What are H0 and Ha here for a test of independence?

The lady did actually get all 4 correct! Here’s how Fisher could have used R to perform the test:

> c.table <- array(data = c(4, 0, 0, 4), dim = c(2,2),

 dimnames = list(Actual = c("Milk", "Tea"), Response =

 c("Milk", "Tea")))

> c.table

 Response

Actual Milk Tea

 Milk 4 0

 Tea 0 4

> fisher.test(x = c.table, alternative = "greater")

 Fisher's Exact Test for Count Data

data: c.table

p-value = 0.01429

alternative hypothesis: true odds ratio is greater than 1

95 percent confidence interval:

 2.003768 Inf

sample estimates:

odds ratio

 Inf

> fisher.test(x = c.table)

 Fisher's Exact Test for Count Data

data: c.table

p-value = 0.02857

alternative hypothesis: true odds ratio is not equal to 1

95 percent confidence interval:

 1.339059 Inf

sample estimates:

odds ratio

 Inf

The default for fisher.test() is to perform a two-sided test (alternative = "two.sided"). The p-value for a two-sided test is simply found by adding all probabilities that are less than or equal to the probability corresponding to what was observed. For this case, the probability was simply 0.0143 + 0.0143 = 0.0286.

Note: All exact inference procedures tend to be conservative for hypothesis testing.

Larger than 2×2 tables

Fisher’s exact test can be extended to tables larger than 2×2 by using the multiple hypergeometric distribution. This distribution is shown in my book, and the application of it is very similar to the 2×2 case. We will see an example shortly.

**Section 6.2.2 – Permutation test for independence**

Permutation tests are a general way to perform hypothesis tests. The test is most often performed via Monte Carlo simulation to obtain a very good estimate of the exact distribution of a statistic of interest. In simple cases, Monte Carlo distribution is not needed as demonstrated next.

The actual distribution for X2 is not , but rather closely related to the hypergeometric distribution. Below is the table from the lady tasting tea example but now with X2 included:

|  |  |  |  |
| --- | --- | --- | --- |
| m | P(M = m) |  | X2 |
| 0 | 0.0143 | 0 | 8 |
| 1 | 0.2286 | 1/9 | 2 |
| 2 | 0.5143 | 1 | 0 |
| 3 | 0.2286 | 9 | 2 |
| 4 | 0.0143 | >9 | 8 |

Therefore, the exact distribution for X2 (think of X2 as a random variable) is

P(X2 = 0) = 0.5143,

P(X2 = 2) = 0.2286 + 0.2286 = 0.4572, and

P(X2 = 8) = 0.0143 + 0.0143 = 0.0286,

where rounding error leads to these probabilities here not summing to 1. Below is a plot of the CDF for this distribution and a  (code in Tea.R):



There are some differences between the two CDFs!

A permutation test for independence can use this exact distribution then to calculate p-values. Thus, if X2 was observed to be 2, then the p-value is P(X2 ≥ 2) = 0.4572 + 0.0286 = 0.4858.

When the sample and/or contingency table size is large, it will not be as easy to use the hypergeometric or multiple hypergeometric distributions. Instead, a form of Monte Carlo simulation can be used to obtain a very good approximation to the exact distribution of X2. Similar to what was done in Section 3.2, a large number, say B, of contingency tables are simulated under independence. The test statistic is calculated for each of these tables, say , to build an approximation to the exact distribution. The p-value is then simply .

The difference between Section 3.2 and here is HOW the contingency tables are simulated. We are now going to keep the row AND column totals fixed! Consider again the hypothetical table for the lady tasting tea example:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Lady’s response |  |
|  |  | Milk | Tea |  |
| Actual | Milk | 3 | 1 | 4 |
| Tea | 1 | 3 | 4 |
|  |  | 4 | 4 | 8 |

We can re-write this table in a “raw” data representation as shown on the left below.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Row | Column |  | Row | Column |  | Row | Column |
| 1 | z1 = 1 |  | 1 | z2 = 1 |  | 1 | z1 = 1 |
| 1 | z2 = 1 |  | 1 | z1 = 1 |  | 1 | z2 = 1 |
| 1 | z3 = 1 |  | 1 | z3 = 1 |  | 1 | z7 = 1 |
| 1 | z4 = 2 |  | 1 | z4 = 1 |  | 1 | z4 = 1 |
| 2 | z5 = 1 |  | 2 | z5 = 1 |  | 2 | z5 = 1 |
| 2 | z6 = 2 |  | 2 | z6 = 1 |  | 2 | z8 = 1 |
| 2 | z7 = 2 |  | 2 | z7 = 1 |  | 2 | z3 = 1 |
| 2 | z8 = 2 |  | 2 | z8 = 1 |  | 2 | z6 = 1 |

Notice there are four 1’s and four 2’s in the “row” portion and four 1’s and four 2’s in the “column” portion. Also, notice there are three times where row = 1 and column = 1, just like in the 2×2 contingency table representation. The zj, for j = 1, …, 8, labels are included to help us track the 1’s and 2’s when we re-order them shortly.

Under independence, the row numbers and the column numbers are equally likely to be paired up! Thus, we could “mix up” or “permute” the row numbers and also do the same for the column numbers to obtain one possible contingency table. Because the total number of 1’s and 2’s is fixed for both the row and column variable, we can actually just permute the column (or row) values only. Thus, there are 8! = 40,320 different permutations with each having an equal probability to occur due to independence.

The middle and right side raw data representations show two possible permutations (X2 = 2 in the middle and X2 = 0 on the right). In total, there are 20,736 different permutations that lead to X2 = 0. Notice that 20,736/40,320 = 0.5143, which is the same probability as obtained earlier with the hypergeometric.

The algorithm that can be used to perform the permutation test by Monte Carlo simulation is as follows:

* + 1. Randomly permute the column numbers while keeping the row numbers unchanged.
		2. Calculate X2 for each simulated data set; call these values ‘s to differentiate them from the original data’s X2 value
		3. Plot a histogram 
		4. Calculate ; this is the p-value for a hypothesis test

Note that step 1 is just sampling the column numbers without replacement.

Example: Fiber enriched crackers (FiberExact.R)

> diet <- read.csv(file = "C:\\data\\Fiber.csv")

> head(diet)

 fiber bloat count

1 bran high 0

2 gum high 5

3 both high 2

4 none high 0

5 bran medium 1

6 gum medium 3

> # Match order given at DASL

> diet$fiber<-factor(x = diet$fiber, levels = c("none",

 "bran", "gum", "both"))

> diet$bloat<-factor(x = diet$bloat, levels = c("none",

 "low", "medium", "high"))

> diet.table<-xtabs(formula = count ~ fiber + bloat, data =

 diet)

> diet.table

 bloat

fiber none low medium high

 none 6 4 2 0

 bran 7 4 1 0

 gum 2 2 3 5

 both 2 5 3 2

> fisher.test(x = diet.table)

 Fisher's Exact Test for Count Data

data: diet.table

p-value = 0.06636

alternative hypothesis: two.sided

Fisher’s exact test produces a p-value of 0.0664.

One way to perform the permutation test is through using the simulate.p.value = TRUE argument in chisq.test():

> set.seed(8912)

> chisq.test(x = diet.table, correct = FALSE,

 simulate.p.value = TRUE, B = 1000)

 Pearson's Chi-squared test with simulated p-value (based

 on 1000 replicates)

data: diet.table

X-squared = 16.943, df = NA, p-value = 0.03896

There were B = 1,000 different permutations used. Is this enough? Because the p-value here is simply a “proportion of successes”, we could use a confidence interval from Section 1.1 to obtain a range which we expect the exact p-value to fall within:

> set.seed(8912)

> save.p <- chisq.test(x = diet.table, correct = FALSE,

 simulate.p.value = TRUE, B = 1000)

> library(package = binom)

> binom.confint(x = round(save.p$p.value\*1000,0), n = 1000,

 conf.level = 1-0.05, methods = "wilson")

 method x n mean lower upper

1 wilson 39 1000 0.039 0.02865896 0.05286931

One could also take a larger number of permutations:

> set.seed(8912)

> save.p2 <- chisq.test(x = diet.table, correct = FALSE,

 simulate.p.value = TRUE, B = 100000)

> binom.confint(x = round(save.p2$p.value\*100000,0), n =

 100000, conf.level = 1-0.05, methods = "wilson")

 method x n mean lower upper

1 wilson 4552 1e+05 0.04552 0.04424545 0.04682946

Similar to Section 3.2, I would like to actually see the histogram for all of the  values. By plotting a  on this histogram too, one can determine how well the  distribution approximates the exact distribution. Unfortunately, chisq.test() does not provide the histogram so we will need to create the plot ourselves. Below is the process that I used to perform the test and create the plot:

> # Put the data into its raw form

> set1 <- as.data.frame(as.table(diet.table))

> tail(set1) # Notice 2 obs. for fiber = both and bloat =

 high

 fiber bloat Freq

11 gum medium 3

12 both medium 3

13 none high 0

14 bran high 0

15 gum high 5

16 both high 2

> set2 <- set1[rep(1:nrow(set1), times = set1$Freq), -3]

> tail(set2) # Notice 2 obs. for fiber = both and bloat =

 high

 fiber bloat

15.1 gum high

15.2 gum high

15.3 gum high

15.4 gum high

16 both high

16.1 both high

> # Check

> xtabs(formula = ~ set2[,1] + set2[,2])

 set2[, 2]

set2[, 1] none low medium high

 none 6 4 2 0

 bran 7 4 1 0

 gum 2 2 3 5

 both 2 5 3 2

> X.sq <- chisq.test(set2[,1], set2[,2], correct = FALSE)

> X.sq$statistic

X-squared

 16.94267

Warning message:

In chisq.test(set2[, 1], set2[, 2], correct = FALSE) :

 Chi-squared approximation may be incorrect

> # Do one permutation to illustrate

> set.seed(4088)

> set2.star <- data.frame(row = set2[,1], column =

 sample(set2[,2], replace = FALSE))

> xtabs(formula = ~ set2.star[,1] + set2.star[,2])

 set2.star[, 2]

set2.star[, 1] none low medium high

 none 4 3 2 3

 bran 6 4 1 1

 gum 3 3 5 1

 both 4 5 1 2

> X.sq.star <- chisq.test(set2.star[,1], set2.star[,2],

 correct = FALSE)

> X.sq.star$statistic

X-squared

 8.200187

Warning message:

In chisq.test(set2.star[, 1], set2.star[, 2], correct = FALSE) :

 Chi-squared approximation may be incorrect

> # Permutation test

> B <- 1000

> X.sq.star.save <- matrix(data = NA, nrow = B, ncol = 1)

> set.seed(1938)

> # options(warn = -1)

> for(i in 1:B) {

 set2.star <- data.frame(row = set2[,1], column =

 sample(set2[,2], replace = FALSE))

 X.sq.star <- chisq.test(set2.star[,1], set2.star[,2],

 correct = FALSE)

 X.sq.star.save[i,1] <- X.sq.star$statistic

 }

There were 50 or more warnings (use warnings() to see the first 50)

> mean(X.sq.star.save >= X.sq$statistic)

[1] 0.05

> # options(warn = 0)

> summarize <- function(result.set, statistic, df, B,

 color.line = "red") {

 par(mfrow = c(1,3), mar = c(5,4,4,0.5))

 # Histogram

 hist(x = result.set, main = "Histogram", freq = FALSE,

 xlab = expression(X^{"2\*"}))

 curve(expr = dchisq(x = x, df = df), col = color.line,

 add = TRUE, lwd = 2)

 segments(x0 = statistic, y0 = -10, x1 = statistic, y1 =

 10)

 # Compare CDFs

 plot.ecdf(x = result.set, verticals = TRUE, do.p =

 FALSE, main = "CDFs", lwd = 2, col = "black", xlab =

 expression(X^"2\*"), ylab = "CDF")

 curve(expr = pchisq(q = x, df = df), col = color.line,

 add = TRUE, lwd = 2, lty = "dotted")

 legend(x = df, y = 0.4, legend = c(expression(Perm.),

 substitute(chi[df1]^2, list(df1 = df))), lwd =

 c(2,2), col = c("black", color.line), lty =

 c("solid", "dotted"), bty = "n")

 # QQ-Plot

 chi.quant <- qchisq(p = seq(from = 1/(B+1), to = 1-

 1/(B+1), by = 1/(B+1)), df = df)

 plot(x = sort(result.set), y = chi.quant, main = "QQ-

 plot", xlab = expression(X^{"2\*"}), ylab = "Chi-

 square quantiles")

 abline(a = 0, b = 1)

 par(mfrow = c(1,1))

 # p-value

 mean(result.set >= statistic)

 }

> summarize(result.set = X.sq.star.save, statistic =

 X.sq$statistic, df = (nrow(diet.table)-1) \*

 (ncol(diet.table)-1), B = B)

[1] 0.05



The p-value here is very similar to what we had before. Also, we see that a  distribution does a very good job with approximating the distribution of X2.

Question: How could we find estimates of P(X2 = x2) in order to obtain the estimated PMF? Note that the lowercase x2 is used to emphasize a particular observed value.

Final comments:

* STAT 950 discusses permutation tests for other statistical problems. A general book on permutation tests written at a MS-stat level is *Introduction to Modern Nonparametric Statistics* by James J. Higgins (2003).
* The homework gives a contingency table where a  distribution approximation works poorly.

**Section 6.2.3 – Exact logistic regression**

This section extends the ideas of permutation tests to a logistic regression setting. The exact distribution for the sufficient statistics of the regression parameters are found through permutation test methods.