140.616.HW6.Delahanty.Jeremy

Jeremy Delahanty

2024-04-03

1

1a) After treatment, 2 out of 10 cells respond to a certain antigen. Calculate a 95% confidence interval for the probability p of response.

We can calculate a confidence interval very simply using a binomial test function built into R! This is because we have two outcomes of interest that are binary (does or does not respond).

```
# Define total number of samples
n <- 10

# Define number of successes
num_responders <- 2

# Calculate the confidence interval for this distribution's parameters.
result <- binom.test(num_responders, n)

# Get the values of the confidence intervals using the $ operator in R
interval_95 <- result$conf.int

# Display the confidence interval
cat("95% Confidence Interval is:", interval_95)
```

```
## 95% Confidence Interval is: 0.02521073 0.5560955
```

1b) In a second experiment, 0 out of 10 cells respond to another certain antigen. Test the null hypothesis that the response rate is 10% versus the alternative that the response rate is less than 10%.

To test for whether the response rate is less than 10%, we can use the `binom.test` function once again! Our null hypothesis is th at our response rate, or probability of success, is 10% and the alternative is that it is less than 10%.

```
# Define number of samples
n <- 10

# Define number of successes
num_responders <- 0
```

```r
# Define probability we're investigating at null hypothesis
pnull <- 0.10

# Use binom.test again, but define the alternative test as "less"
# so it is specific to the test investigating whether the probability
# is indeed less than 10%. No cat is necessary since this will print all the
# results for us.
binom.test(x = num_responders, n = 10, p = pnull, alternative = "less")
```

```
##
##  Exact binomial test
##
## data:  num_responders and 10
## number of successes = 0, number of trials = 10, p-value = 0.3487
## alternative hypothesis: true probability of success is less than 0.1
## 95 percent confidence interval:
##  0.0000000 0.2588656
## sample estimates:
## probability of success
##                      0
```

We see that because the P-Value is greater than our significance level of 0.05 and our confidence interval includes a probability greater than 0.10, we fail to reject our null hypothesis that the proportion of responders being less than 0.10.

2 Consider the table found in the homework. Do outcomes 1 through 5 look equally likely? Use a chi-square and likelihood ratio test to answer this question.

In this case, our null hypothesis is that all outcomes are equally likely. Since there are 5 possibilities, our probability for equal likelihood of these outcomes is 20%. Using this, we can use the Chi Square test and then follow up with the likelihood ratio test.

```r
# Define the counts of each outcome
outcomes <- c(17, 19, 22, 25, 17)

# Define expected frequencies if null is true
null_probabilities <- c(0.2, 0.2, 0.2, 0.2, 0.2)

# Define expected counts if null is true
expected_counts <- sum(outcomes) * null_probabilities

# Use the chisq.test function
chisq.test(x = outcomes, p = null_probabilities)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  outcomes
## X-squared = 2.4, df = 4, p-value = 0.6626
```

```r
# Calculate the likelihood ratio test with:
# LRT = 2 * sum(counts * log(counts/expected))
lrt <- 2 * sum(outcomes*log(outcomes/expected_counts))

# Display the LRT statistic
cat("Likelihood ratio test statistic:", lrt, "\n")
```

```
## Likelihood ratio test statistic: 2.350393
```

```r
# Calculate the LRT p-value by using  the degrees of freedom (categories - 1)
# with the pchisq test. State that lower.tail = FALSE because we are only looking
# at the right side of the distribution.
lrt_pvalue <- pchisq(q = lrt, df = 4, lower.tail = FALSE)

# Display the p-value
cat("The p-value from the likelihood ratio test: ", lrt_pvalue)
```

```
## The p-value from the likelihood ratio test:  0.67161
```

Our results from both the chi-squared test as well as likelihood ratio test lead us to reject the null hypothesis. Each have p-values and test statistics that are similar in magnitude and are substantially larger than the significance, alpha, of 0.05. We can therefore conclude that not all outcomes are equally likely in this dataset.

3

Among 1,000 subjects in a genotyping stsudy, we see 649 AA, 300 AB, and 51 BB at a single locus. Use the chi-square and likelihood ratio test to find out whether or not the locus is in Hardy-Weinberg equilibrium.

This test will assume that the null hypothesis is that the locus is indeed in HW equilibrium, with the alternative bering that it is not in HW equilibrium.

We define `p` as the frequency of the allele A and `q` as the frequency of allele B. These two probabilities must equal 1:

$p + q = 1$

We can then define each combination of alleles as the following:

AA: p^2 BB: q^2 AB: $p \times q$

And then define the final formula for equal to 1:

$1 = p^2 + 2pq + q^2$

With these definitions, we can now use R to calculate each of these values and determine whether the population is in HW equilibrium! We can use the method described in class for this calculation.

```r
# Define observed allele counts, AA, AB, BB and the total observations
observed_alleles <- c(649, 300, 51)
total_observations <- sum(observed_alleles)

# Estimate allele frequencies with equation from class
```

```r
# (AA + AB/2) / total
a_est <- (observed_alleles[1] + observed_alleles[2]/2)/total_observations
# Since p (a_est) + q (b_est) = 1, we can subtract a_est from 1!
b_est <- 1 - a_est
# AA probability is a_est^2
aa_expected <- a_est ^ 2
# AB probability is 2 * a_est * b_est
ab_expected <- 2 * a_est * b_est
# BB probability is b_est^2
bb_expected <- b_est ^2

# Turn values into a vector for computation
expected_counts <- c(aa_expected * total_observations,
                     ab_expected * total_observations,
                     bb_expected * total_observations
                     )

# Define degrees of freedom for chi-square test:
# Calculated from k - s - 1 where k is number of outcome classes, s is the
# number of parameters estimated. Then, calculate the test statistic
dof <- 1
chisq <- sum((observed_alleles - expected_counts)^2/expected_counts)

# Using this test statistic, get the p-value:
chisq_pvalue <- pchisq(q = chisq, df = dof, lower.tail = FALSE)

# Display the p-value
cat("P-value from chi-square test:", chisq_pvalue, "\n")
```

```
## P-value from chi-square test: 0.03688831
```

```r
# Now get the likelihood ratio test statistic with:
# LRT = 2 * sum(observed_genotype * log(actual_genotype / expected_genotype))
lrt <- 2 * sum(observed_alleles * log(observed_alleles/expected_counts))

# Calculate the p-value
lrt_pvalue <- pchisq(q = lrt, df = dof, lower.tail = FALSE)

# Display the p-value
cat("P-value of the likelihood ratio test:", lrt_pvalue)
```

```
## P-value of the likelihood ratio test: 0.04112428
```

We see that our observed p-values for the chi-square test and likelihood ratio test are similar and below our pre-set alpha value of 0.05. We can thus reject the null hypothesis and conclude that the locus is not in fact in HW equilibrium!

Here is an alternative method for calculating these statistics a classmate demonstrated:

```r
# Define observed allele counts, AA, AB, BB and the total observations
observed_alleles <- c(649, 300, 51)
total_observations <- sum(observed_alleles)
```

```r
# Calculate the total of each allele
# This is 2*AA + AB
a_count <- 2*observed_alleles[1] + observed_alleles[2]
# For B, it is 2*BB + AB
b_count <- 2*observed_alleles[3] + observed_alleles[2]
total_alleles <- a_count + b_count

# Calculate allele frequencies
# This is value of p
a_freq <- a_count / total_alleles
# This is value of q
b_freq <- b_count / total_alleles

# Now calculate expected genotypes under HW
aa_expected <- a_freq^2 * total_observations
ab_expected <- (2*a_freq*b_freq) * total_observations
bb_expected <- b_freq^2 * total_observations
expected_genotypes <- c(aa_expected, ab_expected, bb_expected)

# Next, use the Chi-square test with degrees of freedom = k - s - 1
# where k is number of outcome classes, s is the number of paramters estimated.
dof <- length(observed_alleles) - 2

# Calculate the chi-square statistic
chisq <- sum((observed_alleles - expected_genotypes)^2/expected_genotypes)

# Display the test statistic
cat("Chi-square test stastistic:", chisq, "\n")
```

```
## Chi-square test stastistic: 4.355561
```

```r
# Now, calculate the p-value of this statistic with pchisq
chisq_pvalue <- pchisq(q = chisq, df = dof, lower.tail = FALSE)

# Display this p-value
cat("P-value of chi-square statistic:", chisq_pvalue, "\n")
```

```
## P-value of chi-square statistic: 0.03688831
```

```r
# Next, perform the likelihood ratio test with the following method:
# LRT = 2 * sum(observed_genotype * log(actual_genotype / expected_genotype))
lrt <- 2*sum(observed_alleles * log(observed_alleles/expected_genotypes))

# Display this p-value
cat("Likelihood ratio test statistic:", lrt, "\n")
```

```
## Likelihood ratio test statistic: 4.170886
```

```r
# Finally, find the p-value from this test statistic. We can use the same
# degrees of freedom as before
lrt_pvalue <- pchisq(q = lrt, df = dof, lower.tail = FALSE)
```

```r
# Display the p-value
cat("P-value of likelihood ratio test:", lrt_pvalue)
```

```
## P-value of likelihood ratio test: 0.04112428
```