

# Tutorial 10 Solutions

Notation used in class	Notation used in the textbook
$n_{i\bullet} = \sum_{j=1}^c n_{ij}$ $i = 1, \dots, r$	$r_i$ $i = 1, \dots, r$
$n_{\bullet j} = \sum_{i=1}^r n_{ij}$ $j = 1, \dots, c$	$c_j$ $j = 1, \dots, c$

## Question 1

Consider the problem of testing the *hypothesis of independence*:

$H_0$  : There exist probabilities  $p_{i\bullet}$  and  $p_{\bullet j}$ ,  $i = 1, \dots, r, j = 1, \dots, c$ , such that

$$p_{ij} = p_{i\bullet} p_{\bullet j} \quad \text{for all } i = 1, \dots, r, \quad j = 1, \dots, c.$$

$H_1$  : Not  $H_0$ .

Here,  $p_{ij}$  represents the probability that an object or individual selected randomly from the population under study will belong to category  $i$  of argument 1 and category  $j$  of argument 2.

The data is represented by  $n_{ij}$ ,  $i = 1, \dots, r, j = 1, \dots, c$ , which counts the number of observations that fall in category  $i$  of argument 1 and category  $j$  of argument 2.

Note that the probabilities  $p_{i\bullet}$  and  $p_{\bullet j}$  must satisfy  $\sum_{i=1}^r p_{i\bullet} = 1$  and  $\sum_{j=1}^c p_{\bullet j} = 1$ .

Consider estimating  $p_{i\bullet}$  and  $p_{\bullet j}$  under  $H_0$ . Show that the MLEs of  $p_{i\bullet}$  and  $p_{\bullet j}$  are given by:

$$\hat{p}_{i\bullet} = \frac{n_{i\bullet}}{n}, \quad i = 1, \dots, r$$

$$\hat{p}_{\bullet j} = \frac{n_{\bullet j}}{n}, \quad j = 1, \dots, c.$$

Let

$$\boldsymbol{\theta} = \{p_{i\bullet} p_{\bullet j}, \quad i = 1, \dots, r, \quad j = 1, \dots, c\}$$

be the  $r \times c$  dimensional vector of unknown parameters under  $H_0$ . Since we have a multinomial experiment, if  $H_0$  is true, then the likelihood function takes the following form:

$$\mathcal{L}(\boldsymbol{\theta} | \text{all } n_{ij}) = \frac{n!}{\prod_{i=1}^r \prod_{j=1}^c n_{ij}!} \prod_{i=1}^r \prod_{j=1}^c p_{ij}^{n_{ij}}$$

$$\begin{aligned}
&= \frac{n!}{\prod_{i=1}^r \prod_{j=1}^c n_{ij}!} \prod_{i=1}^r \prod_{j=1}^c (p_{i\bullet} p_{\bullet j})^{n_{ij}} \quad (\text{Under } H_0) \\
&= \frac{n!}{\prod_{i=1}^r \prod_{j=1}^c n_{ij}!} \left( \prod_{i=1}^r p_{i\bullet}^{\sum_j n_{ij}} \right) \left( \prod_{j=1}^c p_{\bullet j}^{\sum_i n_{ij}} \right) \\
&= \frac{n!}{\prod_{i=1}^r \prod_{j=1}^c n_{ij}!} \left( \prod_{i=1}^r p_{i\bullet}^{n_{i\bullet}} \right) \left( \prod_{j=1}^c p_{\bullet j}^{n_{\bullet j}} \right) \quad (1)
\end{aligned}$$

Recall that the  $\hat{\theta}$  that maximizes the likelihood function, will also maximize the log-likelihood function. Taking the logarithm of (1) and ignoring any terms that do not depend on the unknown parameters, our quantity of interest becomes:

$$\sum_{i=1}^r n_{i\bullet} \log(p_{i\bullet}) + \sum_{j=1}^c n_{\bullet j} \log(p_{\bullet j}).$$

Hence, in order to find the MLE,  $\hat{\theta}$  (given all  $n_{ij}$ ) of  $\theta$ , we need to solve the following system of equations:

$$\begin{cases} \frac{\partial}{\partial p_{i\bullet}} \left( \sum_{i=1}^r n_{i\bullet} \log(p_{i\bullet}) + \sum_{j=1}^c n_{\bullet j} \log(p_{\bullet j}) \right) = 0, & i = 1, \dots, r \\ \frac{\partial}{\partial p_{\bullet j}} \left( \sum_{i=1}^r n_{i\bullet} \log(p_{i\bullet}) + \sum_{j=1}^c n_{\bullet j} \log(p_{\bullet j}) \right) = 0, & j = 1, \dots, c \end{cases}$$

This system simplifies to:

$$\begin{cases} \frac{\partial}{\partial p_{i\bullet}} \sum_{i=1}^r n_{i\bullet} \log(p_{i\bullet}) = 0, & i = 1, \dots, r \\ \frac{\partial}{\partial p_{\bullet j}} \sum_{j=1}^c n_{\bullet j} \log(p_{\bullet j}) = 0, & j = 1, \dots, c \end{cases} \quad (2)$$

where

$$\sum_{i=1}^r p_{i\bullet} = 1 \quad \text{and} \quad \sum_{j=1}^c p_{\bullet j} = 1,$$

and hence,

$$p_{r\bullet} = 1 - \sum_{i=1}^{r-1} p_{i\bullet} \quad \text{and} \quad p_{\bullet c} = 1 - \sum_{j=1}^{c-1} p_{\bullet j}. \quad (3)$$

Substituting (3) into (2) results in the new system:

$$\begin{cases} \frac{n_{i\bullet}}{p_{i\bullet}} - \frac{n_{r\bullet}}{p_{r\bullet}} = 0 & i = 1, \dots, r-1 \\ \frac{n_{\bullet j}}{p_{\bullet j}} - \frac{n_{\bullet c}}{p_{\bullet c}} = 0 & j = 1, \dots, c-1 \end{cases} \quad (4)$$

Denote

$$A = \frac{n_{r\bullet}}{p_{r\bullet}} \quad \text{and} \quad B = \frac{n_{\bullet c}}{p_{\bullet c}}.$$

It then follows from (4) that

$$n_{i\bullet} = A p_{i\bullet}, \quad i = 1, \dots, r \quad (4a)$$

$$n_{\bullet j} = B p_{\bullet j}, \quad j = 1, \dots, c \quad (4b)$$

Since  $\sum_{i=1}^r n_{i\bullet} = n$  and  $\sum_{i=1}^r p_{i\bullet} = 1$ , it follows from (4a) that  $A = n$ . Similarly, since  $\sum_{j=1}^c n_{\bullet j} = n$  and  $\sum_{j=1}^c p_{\bullet j} = 1$ , it follows from (4b) that  $B = n$ . Hence, the solutions  $\hat{p}_{i\bullet}$  and  $\hat{p}_{\bullet j}$  to the system in (4) are:

$$\hat{p}_{i\bullet} = \frac{n_{i\bullet}}{n}, \quad i = 1, \dots, r$$


$$\hat{p}_{\bullet j} = \frac{n_{\bullet j}}{n}, \quad j = 1, \dots, c$$

Note: It can be shown (though time consuming) that the matrix of second derivatives of the log-likelihood at point

$$\theta = (\hat{p}_{i\bullet}, \hat{p}_{\bullet j}, \quad i = 1, \dots, r, \quad j = 1, \dots, c)$$

is *negative definite*, i.e. the estimators  $\hat{p}_{i\bullet}$  and  $\hat{p}_{\bullet j}$  are indeed the MLEs of  $p_{i\bullet}$  and  $p_{\bullet j}$ , respectively, for  $i = 1, \dots, r$  and  $j = 1, \dots, c$ .

## Question 2

(14.18)  A study of the amount of violence viewed on television as it relates to the age of the viewer yielded the results shown in the accompanying table for 81 people. (Each person in the study was classified, according to the person's TV viewing habits, as a low-violence or high-violence viewer.) Do the data indicate that viewing of violence is not independent of age of viewer, at the 5% significance level?

Viewing	Age		
	16-34	35-54	55 and over
Low violence	8	12	21
High violence	18	15	7

The hypotheses we are interested in testing are:

$H_0$  : The viewing of violence is independent of age

vs

$H_1$  : The viewing of violence is not independent of age

We begin by entering the data into **R**.

```
television <- matrix(
  c(8, 12, 21, 18, 15, 7), byrow=TRUE, nrow=2, ncol=3,
  dimnames = list(c("Low violence", "High violence"), c("16-34", "35-54", "55+"))
)
```

```
television
```

```
##           16-34 35-54 55+
## Low violence      8    12  21
## High violence    18    15   7
```

We can perform the required chi-square test using the `chisq.test()` function introduced in last week's tutorial.

```
chisq.test(television)
```


```
##
## Pearson's Chi-squared test
##
## data:  television
## X-squared = 11.169, df = 2, p-value = 0.003756
```

From the output above, we can see that the  $p$ -value is 0.003756. Since the  $p$ -value is less than 0.05, we reject the null hypothesis in favour of the alternative. There is sufficient evidence to support the claim that the viewing of violence is not independent of age.

### Question 3

(14.26) A manufacturer of buttons wished to determine whether the fraction of defective buttons produced by three machines varied from machine to machine. Samples of 400 buttons were selected from each of the three machines, and the number of defectives were counted for each sample. The results are shown in the table below. Do these data present sufficient evidence to indicate that the fraction of defective buttons varied from machine to machine?

Machine number	Number of defectives
1	16
2	24
3	9

(a)  Test, using  $\alpha = 0.05$ , with a  $\chi^2$  test.

Let  $p_i$  be the true fraction of defective buttons produced by one of the three machines,  $i = 1, \dots, 3$ . Then the hypotheses we are interested in testing are:

$$H_0 : p_1 = p_2 = p_3 = p \quad \text{vs} \quad H_1 : \text{At least one } p_i \neq p$$

Next, we expand our table with a column for counts of non-defectives. If we were doing this question by hand, we could also include the row totals, column totals, and grand total.

Machine number	Number of defectives	Number of non-defectives	Total
1	16	384	400
2	24	376	400
3	9	391	400
Total	49	1151	1200

Note that if we were approaching this question by hand, the procedure is identical to that of working with contingency tables, as we have done in the previous tutorial. The calculation of the test statistic is the same and its degrees of freedom follows the same formula. The main difference lies in the statement of the hypotheses of interest.

To perform this test in **R**, we begin by entering the data and then passing it to the `chisq.test()` function.

```
buttons <- matrix(
  c(16, 384, 24, 376, 9, 391), byrow=TRUE, nrow=3, ncol=2,
  dimnames = list(paste("Machine", 1:3), c("Defectives", "Non-defectives"))
)
```


```
buttons
```

```
##           Defectives Non-defectives
## Machine 1         16          384
## Machine 2         24          376
## Machine 3          9          391
```

```
chisq.test(buttons)
```

```
##
## Pearson's Chi-squared test
##
## data:  buttons
## X-squared = 7.1916, df = 2, p-value = 0.02744
```

The  $p$ -value of this test is 0.02744. Since the  $p$ -value is less than 0.05, we reject the null hypothesis in favour of the alternative. There is sufficient evidence to support the claim that the fraction of defective buttons varies from machine to machine.

- (b)  Test, using  $\alpha = 0.05$ , with a likelihood ratio test. (Refer to exercise 10.106, covered in Tutorial 5 Question 2.)

Imitating the setup of exercise 10.106 (Tutorial 5 Question 2), the likelihood ratio is calculated as:

$$\lambda = \frac{\left(\frac{\sum n_i}{1200}\right)^{\sum n_i} \left(1 - \frac{\sum n_i}{1200}\right)^{1200 - \sum n_i}}{\prod_{i=1}^3 \left(\frac{n_i}{400}\right)^{n_i} \left(1 - \frac{n_i}{400}\right)^{400 - n_i}},$$

where  $n_i$  counts the number of defective buttons produced by machine  $i$ . Prior to computing the value of  $-2 \log(\lambda)$ , we should use log properties to simplify the expression, as demonstrated in Tutorial 5 Question 2.

In **R**, we calculate the value of our test statistic by reusing the code from Tutorial 5 Question 2 and making some minor modifications:

```

n <- c(16, 24, 9)
sum_n <- sum(n)

statistic <- -2 * (sum_n * log(sum_n / 1200) + (1200 - sum_n) * log(1 - sum_n / 1200)
                 - sum(n * log(n / 400)) - sum((400 - n) * log(1 - n / 400)))

statistic

## [1] 7.378884

```

The degrees of freedom is calculated as:

$$k = \dim(\Omega) - \dim(\Omega_0) = 3 - 1 = 2,$$

for we have three free parameters in the unrestricted space, and one free parameter in the space governed by the null hypothesis.

The (approximate)  $p$ -value of the test is:

```

pchisq(statistic, df=3-1, lower.tail=FALSE)

## [1] 0.02498594

```

Since the  $p$ -value is less than 0.05, we reject the null hypothesis in favour of the alternative once again and make the same concluding remarks as in (a).

## Question 4

(14.38) Counts on the number of items per cluster (or colony or group) must necessarily be greater than or equal to one. Thus, the Poisson distribution generally does not fit these kinds of counts. For modelling counts on phenomena such as number of bacteria per colony, number of people per household, and number of animals per litter, the *logarithmic series* distribution often proves useful. This discrete distribution has probability function given by

$$p(y | \theta) = -\frac{1}{\ln(1 - \theta)} \cdot \frac{\theta^y}{y}, \quad y = 1, 2, 3, \dots, \quad 0 < \theta < 1,$$

where  $\theta$  is an unknown parameter.

(a) Show that the MLE  $\hat{\theta}$  of  $\theta$  satisfies the equation

$$\bar{Y} = \frac{\hat{\theta}}{-(1 - \hat{\theta}) \ln(1 - \hat{\theta})}, \quad \text{where } \bar{Y} = \frac{1}{n} \sum_{i=1}^n Y_i.$$

The likelihood function is given by:

$$\begin{aligned}
\mathcal{L}(\theta | \mathbf{y}) &= \prod_{i=1}^n p(y_i | \theta) \\
&= \left( -\frac{1}{\ln(1 - \theta)} \right)^n \frac{\theta^{\sum_{i=1}^n y_i}}{\prod_{i=1}^n y_i}.
\end{aligned}$$

The log-likelihood function is given by:

$$\ell(\theta | \mathbf{y}) = n \ln \left( -\frac{1}{\ln(1-\theta)} \right) + \sum_{i=1}^n y_i \ln(\theta) - \sum_{i=1}^n \ln(y_i).$$


The first derivative of the log-likelihood with respect to  $\theta$  is:

$$\frac{d\ell}{d\theta} = \frac{n}{\ln(1-\theta)(1-\theta)} + \frac{1}{\theta} \sum_{i=1}^n y_i$$

We then set the first derivative equal to zero by evaluating at  $\theta = \hat{\theta}$ .

$$\begin{aligned} \frac{d\ell}{d\theta} \Big|_{\theta=\hat{\theta}} &= 0 \\ \frac{n}{\ln(1-\hat{\theta})(1-\hat{\theta})} + \frac{1}{\hat{\theta}} \sum_{i=1}^n y_i &= 0 \\ \frac{1}{\hat{\theta}} \sum_{i=1}^n y_i &= \frac{-n}{\ln(1-\hat{\theta})(1-\hat{\theta})} \\ \frac{1}{n} \sum_{i=1}^n y_i &= \frac{-\hat{\theta}}{\ln(1-\hat{\theta})(1-\hat{\theta})} \\ \bar{y} &= \frac{-\hat{\theta}}{\ln(1-\hat{\theta})(1-\hat{\theta})}, \end{aligned}$$

as desired.

- (b)  The data in the following table give frequencies of observation for counts on the number of bacteria per colony, for a certain type of soil bacteria.

Bacteria per colony	1	2	3	4	5	6	7+
Number of colonies observed	359	146	57	41	26	17	29

Test the hypothesis that these data fit a logarithmic series distribution. Use  $\alpha = 0.05$ . (Notice that the value  $\bar{y}$  must be approximated because we do not have exact information on counts greater than six.)

We first read the data into **R**.

```
bacteria <- data.frame(
  number = 1:7,
  observed_count = c(359, 146, 57, 41, 26, 17, 29)
)
```

```
bacteria
```

```
## number observed_count
## 1      1           359
## 2      2           146
```

```
## 3      3      57
## 4      4      41
## 5      5      26
## 6      6      17
## 7      7      29
```

Since we do not have a closed form solution for  $\hat{\theta}$ , we must use numerical methods to solve for  $\hat{\theta}$  (via intermediate value theorem). To do this, we first take the result from (a) and rearrange it to obtain a function that equals zero.

$$\bar{y} = \frac{-\hat{\theta}}{\ln(1 - \hat{\theta})(1 - \hat{\theta})}$$

$$\bar{y} \cdot \ln(1 - \hat{\theta})(1 - \hat{\theta}) = -\hat{\theta}$$

$$\hat{\theta} + \bar{y} \cdot \ln(1 - \hat{\theta})(1 - \hat{\theta}) = 0$$

Since this function depends on the mean count, we should calculate this now. Recall that since this is tabulated count data, the mean is *not* calculated by simply averaging the observed count values.

```
mean_count <- with(bacteria, sum(number * observed_count) / sum(observed_count))

mean_count
```

```
## [1] 2.105185
```

We then define the required function of  $\hat{\theta}$  as a function in **R**.

```
g <- function(theta) {
  theta + mean_count * log(1 - theta) * (1 - theta)
}
```

To find a root of this function, we use the `uniroot()` function that is built into **R**. Since we have the restriction that  $\theta \in (0, 1)$ , we should search within this interval.

```
uniroot(g, interval=c(0.0001, 0.9999))
```

```
## $root
## [1] 0.738445
##
## $f.root
## [1] 3.302951e-07
##
## $iter
## [1] 7
##
## $init.it
## [1] NA
##
## $estim.prec
## [1] 6.103516e-05
```

The output of the `uniroot()` function gives more information than what we require. The value of interest is contained under `$root`. Let us store this in a variable called `theta_mle`.



```
theta_mle <- uniroot(g, interval=c(0.0001, 0.9999))$root
```

```
theta_mle
```

```
## [1] 0.738445
```

Now that we have an estimate for  $\theta$  via  $\hat{\theta}$ , we need the expected counts, which are obtained by multiplying the overall sample size with the expected probabilities. These probabilities will be obtained from the probability mass function of the logarithmic series distribution. Since this distribution is not built into **R**, we will need to create our own. I will do so while also following the conventions of existing distribution functions in **R**, such as starting the function name with a **d** and making sure that it is vectorized over its main argument (i.e. if I supply a vector of length five, I should obtain a vector of length five back as the output. However, unlike the distribution functions that already exist in **R**, I will not vectorize over the `theta` argument, since for the purposes of this question, we will only be using a single value for `theta`.

```
dlogseries <- function(x, theta) {  
  if (any(x %% 1 != 0)) {  
    warning("`x` contains values that are not whole numbers.")  
  }  
  
  if (any(x < 1)) {  
    warning("`x` contains values that are not greater than or equal to 1.")  
  }  
  
  if (theta < 0 | theta > 1) {  
    stop("`theta` must be between 0 and 1.")  
  }  
  
  ifelse((x %% 1 != 0) | (x < 1), 0, - (theta^x) / (x * log(1 - theta)))  
}
```

In the above function, if any values of the `x` violate the flags (not being a whole number or not being greater than or equal to one), then the function should return a value of zero in those positions, while also displaying a warning to the user. This is to be consistent with the other distribution functions in **R**. For valid values of `x`, we simply return the probability from the probability mass function. If the value of `theta` supplied is outside of the valid bounds, then we return an error instead of a warning.

We can perform a quick test to check that our function works properly.

```
dlogseries(-1, theta=theta_mle)
```

```
## Warning in dlogseries(-1, theta = theta_mle): `x` contains values that are not  
## greater than or equal to 1.
```

```
## [1] 0
```

```
dlogseries(2.3, theta=theta_mle)
```

```
## Warning in dlogseries(2.3, theta = theta_mle): `x` contains values that are not  
## whole numbers.
```

```
## [1] 0
```

```
dlogseries(c(5, -2, 1.2, 2), theta=theta_mle)
```

```
## Warning in dlogseries(c(5, -2, 1.2, 2), theta = theta_mle): `x` contains values
## that are not whole numbers.

## Warning in dlogseries(c(5, -2, 1.2, 2), theta = theta_mle): `x` contains values
## that are not greater than or equal to 1.

## [1] 0.03274584 0.00000000 0.00000000 0.20330202
```

On the topic of function writing, contrary to what may have been taught in other courses, the last line of your function should never be a variable assignment. For debugging purposes, we always want our function to return *visible* output in the console when called on its own. In addition, we should not use explicit `return()` statements unless we break/exit out of our function earlier than expected, e.g. if we violated one of the flags in the `if` statements and wanted an early exit out of the function.

In calculating our expected probabilities, note that we only pass the values from 1 through 6 into our `dlogseries()` function. This is because the seventh group is a collapsed group and we want our expected probabilities to sum to 1.

```
bacteria <- bacteria |>
  transform(expected_prob = c(dlogseries(1:6, theta=theta_mle),
                             1 - sum(dlogseries(1:6, theta=theta_mle))))

bacteria
```

```
##   number observed_count expected_prob
## 1      1           359      0.55062196
## 2      2           146      0.20330202
## 3      3            57      0.10008490
## 4      4            41      0.05543040
## 5      5            26      0.03274584
## 6      6            17      0.02015083
## 7      7            29      0.03766404
```

Now that we have the expected probabilities, we can obtain the expected counts by multiplying the expected probabilities by the overall sample size.

```
bacteria <- bacteria |>
  transform(expected_count = sum(observed_count) * expected_prob)

bacteria
```

```
##   number observed_count expected_prob expected_count
## 1      1           359      0.55062196      371.66983
## 2      2           146      0.20330202      137.22886
## 3      3            57      0.10008490       67.55731
## 4      4            41      0.05543040       37.41552
## 5      5            26      0.03274584       22.10344
## 6      6            17      0.02015083       13.60181
## 7      7            29      0.03766404       25.42323
```

As mentioned in Tutorial 9 Question 3, we calculated expected counts rather than passing the observed counts and expected probabilities into the `chisq.test()` function. This is because **R** does not know that we made an additional estimate by estimating the MLE of  $\theta$ , thereby reducing our degrees of freedom by one. As such, the degrees of freedom and  $p$ -value returned by the `chisq.test()` function will be incorrect.

However, as we are using software, the test statistic of the chi-square test is not difficult to compute.

```
statistic <- with(bacteria, sum((observed_count - expected_count)^2 / expected_count))  
  
statistic
```

```
## [1] 5.024837
```

The degrees of freedom of our test statistic is  $7 - 1 - 1 = 5$  since we have seven categories, lose one degree of freedom for estimating  $\theta$ , and lose one degree of freedom for the constraint that our probabilities must sum to one. The  $p$ -value for this (upper-tailed) test is:

```
pchisq(statistic, df=7-1-1, lower.tail=FALSE)
```

```
## [1] 0.4128566
```

Since the  $p$ -value is not less than 0.05, we fail to reject the null hypothesis. As such, there is insufficient evidence to support the claim that the data do not come from a logarithmic series distribution.