DATA2002

Chi-squared goodness of fit tests

Garth Tarr

THE UNIVERSITY OF SYDNEY

Goodness of fit tests for discrete distributions

Radiation exposure

The goal in biological dosimetry is to estimate the dose of **ionizing radiation**, absorbed by an exposed individual by using chromosome damage in peripheral lymphocytes.

When radiation exposure occurs, the damage in DNA is randomly distributed between cells producing chromosome aberrations. The outcome of interest is the number of aberrations observed. The number of aberrations typically follows a Poisson distribution, the rate of which depends on the dose.

The table below shows the number of chromosome aberrations from a patient exposed to radiation after the nuclear accident of Stamboliyski (Bulgaria) in 2011 (Puig and Weiß, 2020).

Number of aberrations	0	1	2	3	4	5	6	7
Frequency	117	94	51	15	0	0	0	1

We want to test whether the random variable generating this data follows a **Poisson** distribution.

Poisson distribution

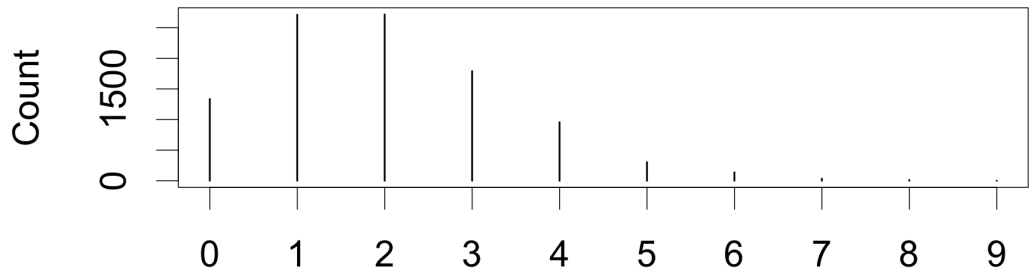
A **Poisson** random variable represents the probability of a given number of events occurring in a fixed interval (e.g. number of events in a fixed period of time) if these event occur independently with some known average rate λ per unit time.

If X is a Poisson random variable with rate parameter λ, the probability mass function is:

$$P(X = k) = e^{-\lambda}\frac{\lambda^k}{k!}, \quad k = 0, 1, 2, \dots$$

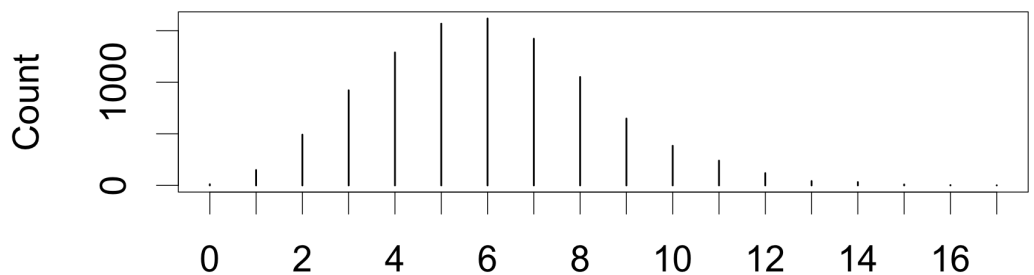
$\lambda = 2$

```
plot(table(rpois(n=10000, lambda=2)), ylab = "Count")
```



$\lambda = 6$

```
library(dplyr) # for %>%
rpois(n=10000, lambda=6) %>% table() %>%
  plot(ylab = "Count")
```



Chi-squared tests for discrete distributions

Suppose we have a sample $x_1, x_2, \ldots, x_n$.

We want to test whether the sample is taken from a population with a given distribution function $F_0(x|\theta_1, \theta_2, \ldots, \theta_h)$ where θ_l are parameters of the distribution.

We may count the frequencies y_i for each value of x_j and compare them to the expected frequencies, e_i, calculated using the expected probabilities, p_i, from the hypothesised distribution, $F_0(x|\theta_1, \theta_2, \ldots, \theta_h)$.

This is a *general* chi-squared goodness-of-fit test with test statistic,

$$T = \sum_{i=1}^{k} \frac{(Y_i - e_i)^2}{e_i} = \sum_{i=1}^{k} \frac{(Y_i - np_i)^2}{np_i}.$$

Chi-squared tests for discrete distributions

However the model parameters $\theta_1, \theta_2, \ldots, \theta_h$ are usually **unknown** and have to be estimated from the sample.

In this case, the expected probabilities p_i are replaced by their estimates $\hat{p}_i$.

Then the observed test statistic is

$$t_0 = \sum_{i=1}^{k} \frac{(y_i - n\hat{p}_i)^2}{n\hat{p}_i},$$

and the approximate p-value is

$$P(\chi^2_{k-1-q} \geq t_0).$$

Note the degrees of freedom are $k - 1 - q$ where q is the number of parameters we need to estimate.

Radiation exposure

- Let X be a random variable such that $X \sim \mathrm{Poisson}(\lambda)$.
- Let y_i be the observed frequency of $X = i$.
- The expected probabilities p_i are given by the probability mass function,

$$P(X = i) = p_i = e^{-\lambda} \frac{\lambda^i}{i!} \quad \text{for } i = 0, 1, 2, \ldots,$$

  where p_i denote the probability of the number of chromosome aberrations in the i-th category.
- Note that for a Poisson distribution both $\mathrm{E}(X)$ and $\mathrm{Var}(X)$ are equal to the parameter λ.

Since λ is unknown, we estimate λ by the sample mean $\hat{\lambda} = \bar{x} = 248/278 = 0.89$.

i	y_i	$\hat{p}_i = e^{-\hat{\lambda}}\hat{\lambda}^i/i!$	$\hat{e}_i = n\hat{p}_i$	$\frac{(y_i-\hat{e}_i)^2}{\hat{e}_i}$
0	117	$\frac{0.89^0 e^{-0.89}}{0!} = 0.4098$	$278 \times 0.4098 = 113.92$	$\frac{(117-113.92)^2}{113.92} = 0.08$
1	94	$\frac{0.89^1 e^{-0.89}}{1!} = 0.3656$	$278 \times 0.3656 = 101.63$	$\frac{(94-101.63)^2}{101.63} = 0.57$
2	51	$\frac{0.89^2 e^{-0.89}}{2!} = 0.1631$	$278 \times 0.1631 = 45.33$	$\frac{(51-45.33)^2}{45.33} = 0.71$
3	15	$\frac{0.89^3 e^{-0.89}}{3!} = 0.0485$	$278 \times 0.0485 = 13.48$	$\frac{(15-13.48)^2}{13.48} = 0.17$
4	0	$\frac{0.89^4 e^{-0.89}}{4!} = 0.0108$	$278 \times 0.0108 = 3.01$	$\frac{(0-3.01)^2}{3.01} = 3.01$
5	0	$\frac{0.89^5 e^{-0.89}}{5!} = 0.0019$	$278 \times 0.0019 = 0.54$	$\frac{(0-0.54)^2}{0.54} = 0.54$
6	0	$\frac{0.89^6 e^{-0.89}}{6!} = 0.0003$	$278 \times 0.0003 = 0.08$	$\frac{(0-0.08)^2}{0.08} = 0.08$
≥ 7	1	$1 - 0.4098 - 0.3656 - \ldots - 0.0003 = 0.00004$	$278 \times 0.00004 = 0.01$	$\frac{(1-0.01)^2}{0.01} = 96.40$
Total	278	1	278	101.56

Note that the numbers have been rounded. Calculations were actually done using R, so there's likely rounding errors, e.g. `(1-0.01)^2/0.01 = 98.01` but I've reported R's more accurate number (`96.40`) in the table which didn't suffer rounding errors.

- But wait! There are a number of cells where the expected number of counts is < 5 which violates an assumption.
- We combine the last five classes so that the final group is ≥ 3 with observed frequency $y_{\geq 3} = 15 + 1 = 16$, the expected count is $\hat{e}_{\geq 3} = 13.48 + 3.01 + 0.54 + 0.08 + 0.01 = 17.12$, and the contribution to the chi-square test statistic is $\frac{(16-17.12)^2}{17.12} = 0.07$.

i	y_i	$\hat{p}_i = e^{-\hat{\lambda}}\hat{\lambda}^i/i!$	$\hat{e}_i = n\hat{p}_i$	$\frac{(y_i-\hat{e}_i)^2}{\hat{e}_i}$
0	117	$\frac{0.89^0 e^{-0.89}}{0!} = 0.4098$	$278 \times 0.4098 = 113.92$	$\frac{(117-113.92)^2}{113.92} = 0.08$
1	94	$\frac{0.89^1 e^{-0.89}}{1!} = 0.3656$	$278 \times 0.3656 = 101.63$	$\frac{(94-101.63)^2}{101.63} = 0.57$
2	51	$\frac{0.89^2 e^{-0.89}}{2!} = 0.1631$	$278 \times 0.1631 = 45.33$	$\frac{(51-45.33)^2}{45.33} = 0.71$
≥ 3	16	$1 - 0.4098 - 0.3656 - 0.1631 = 0.0615$	$278 \times 0.0615 = 17.12$	$\frac{(16-17.12)^2}{17.12} = 0.07$
Total	278	1	278	1.43

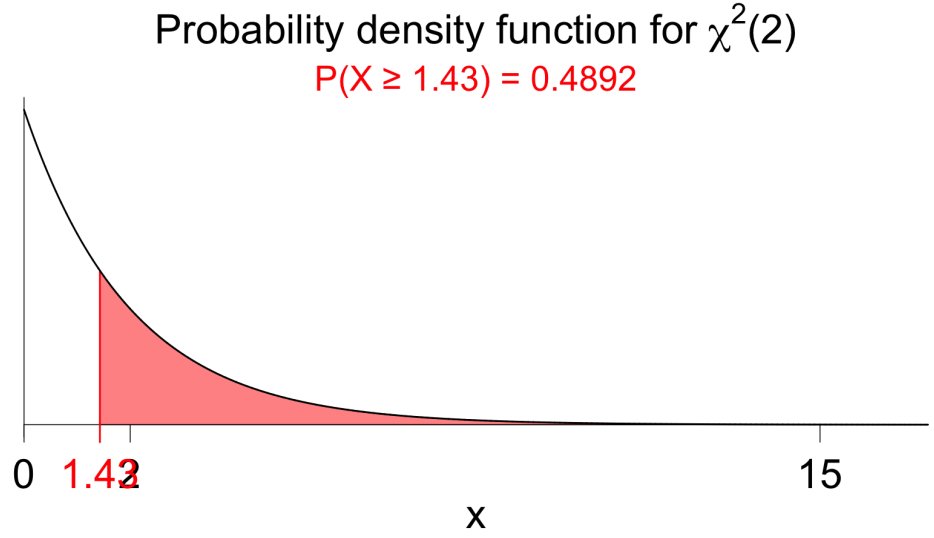
- The final observed test statistic is, $t_0 = 0.08 + 0.57 + 0.71 + 0.07 = 1.43$.

There's a little rounding error here, the actual test statistic, without rounding error is 1.4372. Don't stress about rounding errors, in practice you will have R do everything for you and so it's not important whether you round to 2 or 3 or 4 decimal places.

Hypothesis test

- **Hypothesis:** H_0: the data come from a Poisson distribution vs H_1: the data do not come from a Poisson distribution.
- **Assumptions:** The expected frequencies, $e_i = np_i \geq 5$. Observations are independent.
- **Test statistic:** $T = \sum_{i=1}^{k} \frac{(Y_i - np_i)^2}{np_i}$.
  Under H_0, $T \sim \chi^2_2$ approximately.
- **Observed test statistic:** $t_0 = 1.43$
- **P-value:**
  $P(T \geq t_0) = P(\chi^2_2 \geq 1.43) = 0.49$

- **Decision:** Since the p-value is greater than 0.05, we do not reject the null hypothesis. The data are consistent with a Poisson distribution.



In R

```
y = c(117, 94, 51, 15, 0, 0, 0, 1) # input the observed counts
x = 0:7 # define the corresponding groups
n = sum(y) # total number of samples (sample size)
k = length(y) # number of groups
(lam = sum(y * x)/n) # estimate the lambda parameter
```

```
## [1] 0.8920863
```

```
p = dpois(x, lambda = lam) # obtain the p_i from the Poisson pmf
p
```

```
## [1] 4.097999e-01 3.655769e-01 1.630631e-01 4.848878e-02
## [5] 1.081404e-02 1.929412e-03 2.868670e-04 3.655859e-05
```

```
p[8] = 1 - sum(p[1:7]) # redefine the 8th element P(>=7) NOT P(7)
round(p, 5)
```

```
## [1] 0.40980 0.36558 0.16306 0.04849 0.01081 0.00193 0.00029
## [8] 0.00004
```

Note: `p` is a vector of length 8, because `x` is a vector of length 8. We can extract the first three elements in the vector using `p[1:7]` where `1:7` means all the integers between 1 and 3 inclusive, this is the same as `p[c(1,2,3,4,5,6,7)]`. We can access and overwrite the 8th element in a vector in a similar way. `p[8]` would print out the 8th element, but `p[8] = 1 - sum(p[1:7])` overwrites the 8th element.

```
(ey = n * p)  # calculate the expected frequencies
```

```
## [1] 113.92436722 101.63037076  45.33153228  13.47988010
## [5]   3.00630420   0.53637658   0.07974904   0.01141984
```

```
ey >= 5  #check assumption e_i >= 5 not all satisfied
```

```
## [1]  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE
```

```
# combine adjacent classes to satisfy assumptions
(yr = c(y[1:3], sum(y[4:8])))
```

```
## [1] 117  94  51  16
```

```
(eyr = c(ey[1:3], sum(ey[4:8])))
```

```
## [1] 113.92437 101.63037  45.33153  17.11373
```

```
(pr = c(p[1:3], sum(p[4:8])))
```

```
## [1] 0.40979988 0.36557687 0.16306307 0.06156018
```

```
kr = length(yr)  # number of combined classes
(t0 = sum((yr - eyr)^2/eyr))  # test statistic
```

```
## [1] 1.43721
```

```
(pval = 1 - pchisq(t0, df = kr - 1 - 1))  # p-value
```

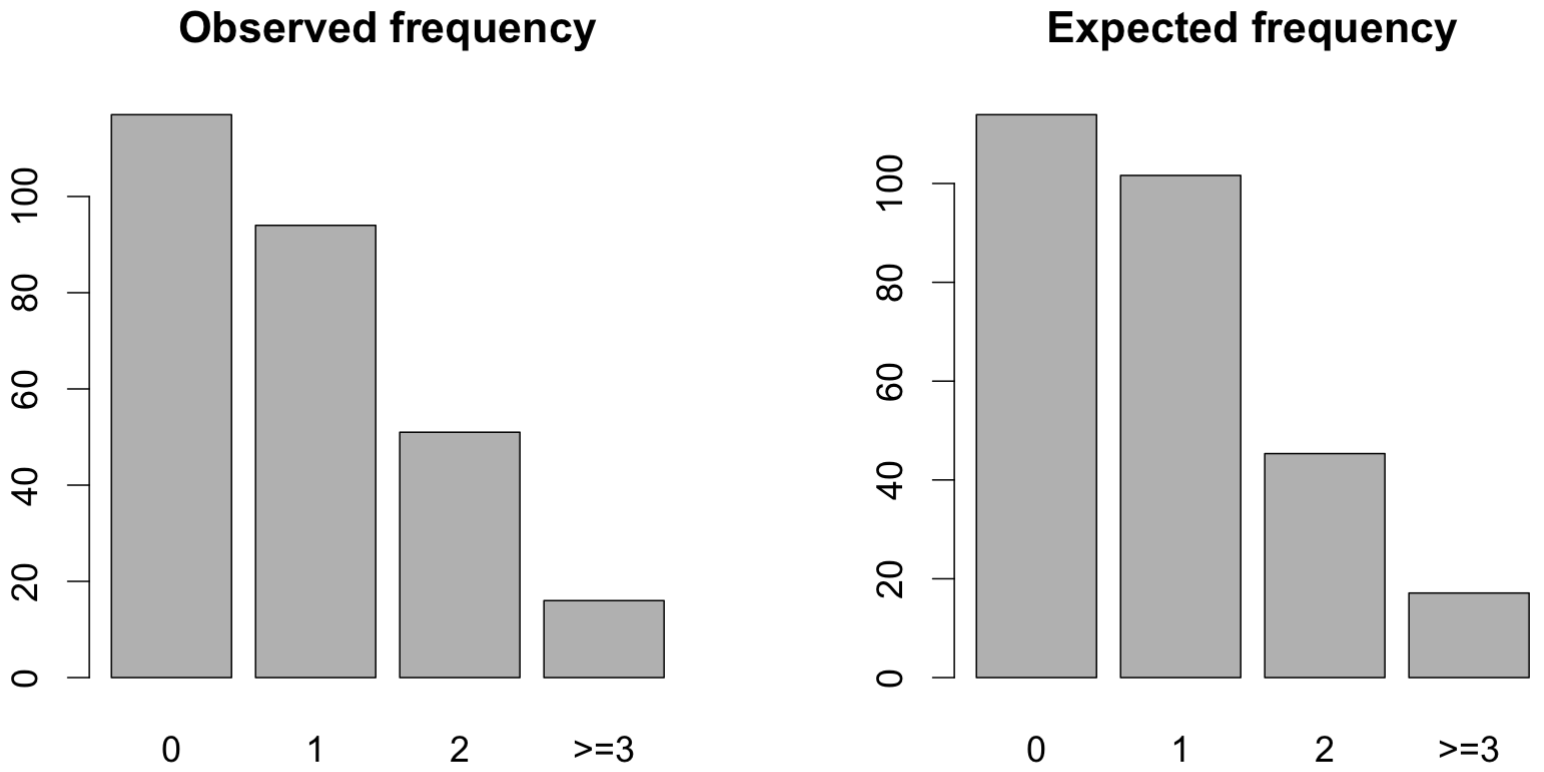
```
## [1] 0.4874317
```

```
chisq.test(yr, p = pr)  # note the (incorrect) degrees of freedom
```

```
##
##     Chi-squared test for given probabilities
##
## data:  yr
## X-squared = 1.4372, df = 3, p-value = 0.6968
```

Why are the degrees of freedom in `chisq.test()` wrong?

```r
xr = c("0", "1", "2", ">=3")  # group labels
par(mfrow = c(1, 2), cex = 1.5)  # plot options
barplot(yr, names.arg = xr, main = "Observed frequency")
barplot(eyr, names.arg = xr, main = "Expected frequency")
```



R packages and functions

- `rpois()` generate pseudo-random data from a Poisson distribution
- `dpois()` probabilities from a Poisson distribution
- `table()` tabulate discrete data
- `nrow()`, `mean()`, `sd()` and `var()`
- `plot()` a generic function that generates different plots depending on what you feed it. E.g. when you feed it a `table` object, it plots a frequency distribution.
- `1:n` returns a vector of integers between `1` and `n` inclusive
- `x[1:3]` returns the first 3 values in the vector `x`
- `x[4] = 41` sets the 4th element in the object `x` to be `41`

References

For further details see Larsen and Marx (2012), sections 10.3 and 10.4.

Larsen, R. J. and M. L. Marx (2012). *An Introduction to Mathematical Statistics and its Applications*. 5th ed. Boston, MA: Prentice Hall. ISBN: 978-0-321-69394-5.

Puig, P. and C. H. Weiß (2020). "Some goodness-of-fit tests for the Poisson distribution with applications in Biodosimetry". In: *Computational Statistics & Data Analysis* 144, p. 106878. ISSN: 0167-9473. DOI: 10.1016/j.csda.2019.106878.