chi_square_goodness_of_fit

October 29, 2020

1 Using Pearson's χ^2 Test for Categorical Data

2 1. Pearson's χ^2 Test

What if our categorical variables has more than 2 categories?

There are a few options when you have a variable with more than 2 categories - Exact Multinomial Test (EMT package in *R*) - *G*-Test for Goodness-of-Fit (also called likelihood ratio test) - Pearson's χ^2 (Goodness-of-Fit) Test

3 1.1 Pearson's χ^2 (Goodness-of-Fit) Test

Pearson's χ^2 goodness-of-fit test can be used when we have some categorical variable, X, where each X_i is a value from one of K categories, and where $K \geq 2$ and we have an expected probability, P_k, for each category.

3.1 1.2 Pearson's χ^2 Goodness-of-Fit Test Example

Suppose we want to determine whether or not a die is loaded (i.e., not a fair die). Say we roll the die 100 times, and we obtain the following results:

Face	Count
1	13
2	21
3	15
4	17
5	20
6	14

Are we confident that this is a fair die?

3.1.1 1.2.1 Pearson's χ^2 Test Example (cont.)

The test statistic is χ^2 and is computed using:

$$\chi^2 = \sum_{k=1}^{K} \frac{(O_k - E_k)^2}{E_k},$$

where K is the number of categories, O_k is the observed count for category k, and E_k is the expected count for category k under the null hypothesis. The degrees of freedom are: $df = K - 1$.

3.1.2 1.2.2 Pearson's χ^2 Test Example (cont.)

The χ^2 test statistic follows the χ^2 distribution, a continuous distribution with a single parameter—the degrees of freedom (i.e., df).

[1.] Image source: https://stats.libretext.org

3.1.3 1.2.3 Pearson's χ^2 Test Example (cont.)

With this χ^2 and df, we evaluate probability of observed data if the null hypothesis is true. - Note that Pearson's χ^2 goodness-of-fit test assumes observations are independent from one another

[1.] Image source: https://actuarialmodelingtopics.wordpress.com

3.1.4 1.2.4 Using the `chisq.test()` Function

```
In [31]: roll_cnts <- c(13, 21, 15, 17, 20, 14)          # create vector with our counts

         probs <- rep(1/6, 6)                            # create vector with 6 elements, all

In [32]: test1 <- chisq.test(roll_cnts, p = probs)       # run test

         print(test1)


        Chi-squared test for given probabilities

data:  roll_cnts
X-squared = 3.2, df = 5, p-value = 0.6692
```

3.1.5 1.2.5 Using `str()` on Output of `chisq.test()`

```
In [17]: str(test1)              # examine components of test object

List of 9
 $ statistic: Named num 3.2
  ..- attr(*, "names")= chr "X-squared"
 $ parameter: Named num 5
  ..- attr(*, "names")= chr "df"
 $ p.value  : num 0.669
 $ method   : chr "Chi-squared test for given probabilities"
 $ data.name: chr "roll_cnts"
 $ observed : num [1:6] 13 21 15 17 20 14
 $ expected : num [1:6] 16.7 16.7 16.7 16.7 16.7 ...
 $ residuals: num [1:6] -0.8981 1.0614 -0.4082 0.0816 0.8165 ...
 $ stdres   : num [1:6] -0.9839 1.1628 -0.4472 0.0894 0.8944 ...
```

```
 - attr(*, "class")= chr "htest"
```

```
In [35]: test1$residuals
```

1. -0.898146239020498 2. 1.06144555520604 3. -0.408248290463863 4. 0.0816496580927732 5. 0.816496580927727 6. -0.65319726474218

```
In [ ]:
```