ZULF'S FORMAL REPLACEMENT FOR PEARSON CHI-SQUARE INDEPENDENCE TEST

ZULFIKAR MOINUDDIN AHMED

1. Pearson Chi-Square Independence Does not Tell us Proportions being same or not

Karl Pearson's Chi-square independence test does not do what I would like from a frequency table.

Suppose I have two categorical variables X and Y. Then I make a frequency table. Suppose $X \in \{x_1, \ldots, x_m\}$ and $Y \in \{y_1, \ldots, y_n\}$. I have now a table with say $x_1, x_2, \ldots x_m$ as row labels and $y_1, y_2, \ldots, y_n$ as column labels. Then I fill up the table by counts for (X, Y) measurements of the same subjects.

I now have a rectangular array of non-negative integers.

What I would like to know if whether the first row proportions, say $q_1^1, \ldots, q_n^1$ are statistically identical to all the other rows or not.

2. What is Chi-Square and Why One Needs Care to use Chi-Square

This is extremely elementary but professional scientists do not need to unpack it so often. The Chi-square distributions are sums of squares of normal distributions. For example, χ_5^2 the Chi-square distribution with 5 degrees of freedom is just the distribution of $Z_1^2 + \cdots + Z_5^2$ where $Z_1, \ldots, Z_5 \sim N(0, 1)$ are all standard normal. This is really beautiful except there is trouble in paradise when proportions arise.

Suppose you take proportions of the first row normalized to 1. So $(q_1^1, \ldots, q_n^1)$; they will sum to 1 and satisfy

$$0 \leq q_s^1 \leq 1$$

for $s = 1, \ldots, n$. Where are normal distributions going to arise here at all? You can't just randomly say q_s^1 is remotely normal. And then you say, well, fine, it's *log*-normal. It's not.

$$\log(q_s^1) \in (-\infty, 0]$$

So the natural thing that could be normal (regardless of standard deviation) is actually:

$$g(q_s^1) =: g_{\text{propnorm}}(q_s^1) = \log(-\log(q_s^1))$$

This proportion-normalizer is totally outside the range of vision of statistical analysts of categorical data. Now we actually get full coverage of $\mathbf{R}$ by $g(q)$ where q is a proportion.

And this is my suggestion for one part of the problem, find some normal variable candidates. The second part of the problem is what to do about the standard deviation.

Date: April 28, 2021.

For this problem, I say just use the standard deviation of the set

$$\{g(q_1^1), \ldots, g(q_n^1)\}$$

if $n >> 2$. Otherwise just use something like average absolute differences, i.e. average over $|g(q_a^1) - g(q_b^1)|$ as $a, b \in \{1, \ldots, n\}$. Let's call this σ. Then let

$$\mu = \frac{1}{n} \sum_s g(q_s^1)$$

Finally produce the test statistic

$$t = \sum_{1 \leq r \leq m-1, 1 \leq s \leq n-1} (g(q_s^r) - g(q_s^1))^2 / \sigma^2$$

The last column is missing because probabilities sum to 1. Finally, this statistic can be checked against a Chi-square distribution with some justification. Let's see, the degrees of freedom will be

$$df = (n-1) * (m-1)$$

So this is my proposal. Use this statistic and check against chi-square with *df* degrees of freedom.

3. Full R Implementation

```
g<-function(x){
  log(-log(x))
}

zulf.sigma<-function(z){
  n<-length(z)
  D<-matrix(0, nrow=n, ncol=n)
  s<-0
  for (j in 2:n){
    for (k in 1:(n-1)){
      gg = abs( z[j] - z[k])
      D[j,k] <- gg
      D[k,j] <- gg
      s<-s + gg
    }
  }
  out<- 2*s/(n^2-n)
  out
}

zulf.chisq<-function( data ){
  m<-dim(data)[1]
  n<-dim(data)[2]
  t<-0
  v0<-g(data[1,1:(n-1)]/sum(data[1,1:(n-1)]))
  sigma0 <- zulf.sigma( v0 )
  mu0 <- mean(v0)
  for (j in 2:m){
```

```
    v<-g(data[j,1:(n-1)]/sum(data[j,1:(n-1)]))
    sigma <- zulf.sigma( v )
    mu <- mean(v)
    t<- t + sum( ((v-mu)/sigma-(v0-mu0)/sigma0)^2)
  }
  df<-(n-1)*(m-1)
  t0<-qchisq(0.95, df=df)
  pval <- 1 - pchisq( t, df=df)
  list(tstat=t,pval=pval,crit=t0)
}
```