Week 8 Exercises: Chi-Square Test of Independence and Correlation

Chi-Square Exercise: There is a dataset built into R called HairEyeColor that contains a contingency table for n=592 statistics students. Four variants of hair color are crossed with four variants of eye color with cell counts of the number of people fitting each description. Your goal is to conduct a thorough chi-square analysis using both conventional and Bayesian techniques.

1. First, just for simplicity, combine the males and females into one 4x4 contingency table:

   HEcombined <- HairEyeColor[, ,1] + HairEyeColor[, ,2]

   Note the double commas in the selector: HairEyeColor is a 3D table! Review the result by typing HEcombined at the console. Check the total number of observations in the new dataset with sum(HEcombined). Add comments to describe what you see.

2. Review the proportion of total observations in each cell with:

   HEcombined/sum(HEcombined)

   Review the results with your group, going row by row or column by column. Add comments to describe what you see. Later, we are going to focus on a subset of four cells where the real action is located. Can you pick out which ones they might be?

3. Calculate the chi-square value and test the overall significance of the test of independence. You can use the chisq.test() procedure to accomplish this:

   chiOut <- chisq.test(HEcombined)
   chiOut

   Paste the console output into a comment. Describe the results of the significance test in a comment. Also answer these questions: Why are there nine degrees of freedom? What is the null hypothesis? Is the null hypothesis rejected?

4. Examine the "residuals" from the chi-square test with chiOut$residuals. Residuals represent how far an observed value was from the expected value. A large positive residual means that the observation for a cell was much higher than expected. A large negative residual means that the observation for a cell was much lower than expected. Large residuals (negative or positive) indicate the cells that made the most powerful contribution to the value of chi-square. Therefore, cells with large residuals show where the "action" is with respect to non-independence.

5. Based on the residuals, select two rows and two columns from HEcombined that have the most interesting stuff happening in them. Using the square brackets notation, you can access smaller subsets of a matrix or data frame. For example, this accesses the first two rows and the third and fourth columns of mtcars. Try it!

   mtcars[c(1,2), c(3,4)]

   Assign the results of your subsetting to a new matrix called *HEsmall*.

6. Make sure to library(BayesFactor) before running the next line. Conduct a Bayesian contingency table analysis with the following line of code:

   ctOut <- contingencyTableBF(HEsmall, sampleType="poisson", posterior=TRUE, iterations=10000)

   Review the results with summary(ctOut). Add comments to describe what you see.

7. Examine histograms of the posterior distributions **of the proportions of different colored eyes**. Here's an example of code that will plot the ratios formed by the entries in the first row:

   firstRowRatio <- ctOut[,"lambda[1,1]"] / ctOut[,"lambda[1,2]"]
   hist(firstRowRatio)

   Copy and modify that code so that you can also calculate the proportions for the second row. Store the result in a new variable. Describe what you are seeing and make sense out of it in a comment. You might want to refer back to HEsmall to remember what the original count was in each cell.

8. After completing step 7, you now have two different posterior distributions of ratio values, one for each row. The difference between these two distributions makes a third posterior distribution that captures the extent to which the proportions from the first two rows differ. Explain the histogram in a comment. Just based on a visual inspection, does the HDI overlap with zero? Is there a credible difference between the two rows with respect to the proportions of different colored eyes?

Correlation Exercise: In the handouts area there is a CSV file with the name altCorrSample.csv. This contains a derivation of the Science data set from the ltm package. You do not need to load the ltm package to use it. Several columns have been added in order that the new data set contains some rank order and metric variables in addition to the original ordinal data. In particular, note these column definitions:

- science – a metric variable which is the sum of participant ratings of five items pertaining to attitudes about science
- tech – a metric variable which is the sum of participant ratings of two items pertaining to attitudes about technology
- sciRank – a rank order variable that ranks participants in order of their attitudes about science
- techRank – a rank order variable that ranks participants in order of their attitudes about technology
- optimist – A Boolean variable indicating whether a participant was generally optimistic about science and technology (not used in this exercise)

9. Read in the CSV file. Run str() to confirm that you have 175 observations of 12 variables.

10. Display histograms of science, tech, sciRank, and techRank. Add comments indicating what you see.

11. Correlate the two rank order variables (sciRank and techRank) using three different correlation coefficients and compare the results. Make sure to test significance using cor.test(). The cor() and cor.test() functions both take an extra argument called method= which can be set to one of three correlation techniques:
    - method= "pearson" – This produces the Pearson product moment correlation, suitable for normally distributed metric variables (also the default if method is not specified)
    - method= "kendall" – This produces Kendall's Tau, suitable for arbitrary distributions with or without outliers
    - method= "spearman" – This produces Spearman's Rho, suitable for rank order variables (which are generally uniformly distributed)

12. Correlate the two metric variables (science and tech) using three different correlation coefficients and compare the results. Make sure to test significance using cor.test(). Write a comment describing the results.