Contingency Tables

Math 445, Spring 2017

A Permutation test for independence of two variables

Motivating example

The General Social Survey (GSS) is a sociological survey used to collect data on demographic characteristics and attitudes of residents of the United States. In 2014, the survey collected responses from 2538 respondents. We will consider two questions from this survey:

- Do you think the use of marijuana should be made legal, or not?
- What is the highest degree that you have earned?

Using these data, we will explore whether one's opinion on the legalization of marijuana is *associated* with their political views.

To begin, we'll consider a contingency table summarizing the breakdown of opinion by political views:

degree	legal	not legal	(all)	% in favor
bachelor	68	41	109	0.6238532
graduate	167	121	288	0.5798611
HIGH SCHOOL	84	129	213	0.3943662
JUNIOR COLLEGE	446	344	790	0.5645570
(all)	765	635	1400	0.5464286

Expected counts in R

To calculate the expected counts for use in a hypothesis test we first need to create a basic contingency table:

```
observed_tbl <- with(grass_df, table(degree, grass))
```

	legal	NOT LEGAL
bachelor	68	41
graduate	167	121
HIGH SCHOOL	84	129
JUNIOR COLLEGE	446	344

Then we use the function `outer` to take the outer product of the row and column totals vectors:

```
expected_tbl <- outer(rowSums(observed_tbl), colSums(observed_tbl)) / sum(observed_tbl)
```

	legal	NOT LEGAL
bachelor	59.56071	49.43929
graduate	157.37143	130.62857
HIGH SCHOOL	116.38929	96.61071
JUNIOR COLLEGE	431.67857	358.32143

Finally, we can calculate our test statistic:

```
sum((observed_tbl - expected_tbl)^2 / expected_tbl)
```

```
## [1] 24.85484
```

Algorithm

Store the data in a table with one row per observation and one column per variable.

Calculate a test statistic for the original data. Normally large values of the test statistic suggest dependence.

Repeat

Randomly permute the rows in one of the columns.

Calculate the test statistic for the permuted data.

Until we have enough samples

Calculate the p-value as the fraction of times the random statistics exceed the original statistic.

Optionally, plot a histogram of the resampled statistic values.

Implementation in R

We'll use the `chisq` function written by the authors. Notice that the input is a contingency table, which can be obtained using the `table` command.

```
chisq <- function(obs) {
  expected <- outer(rowSums(obs), colSums(obs)) / sum(obs)
  RES <- sum((obs - expected)^2 / expected)
  return(RES)
}
```

Extract the columns of the data frame that correspond to the categories of interest (this simplifies checking for errors). We don't need to do this in our example, since we only have two columns in the `grass_df` data frame.

We only want to permute the data corresponding to *complete cases*, so we need to exclude all of the missing values (coded as `NA` in R). This is easily done *after* you have selected the two columns of interest using the `na.omit` function:

```
# Notice the NAs
summary(grass_df)
```

```
##        grass                degree
##  legal    :870   bachelor      : 186
##  NOT LEGAL:704   graduate      : 472
##  NA's     :964   HIGH SCHOOL   : 330
##                  JUNIOR COLLEGE:1269
##                  NA's          : 281
```

```
# Exclude the missing values
grass_complete <- na.omit(grass_df)

# Check that it worked
summary(grass_complete)
```

```
##        grass                degree
##  legal    :765   bachelor      :109
##  NOT LEGAL:635   graduate      :288
##                  HIGH SCHOOL   :213
##                  JUNIOR COLLEGE:790
```

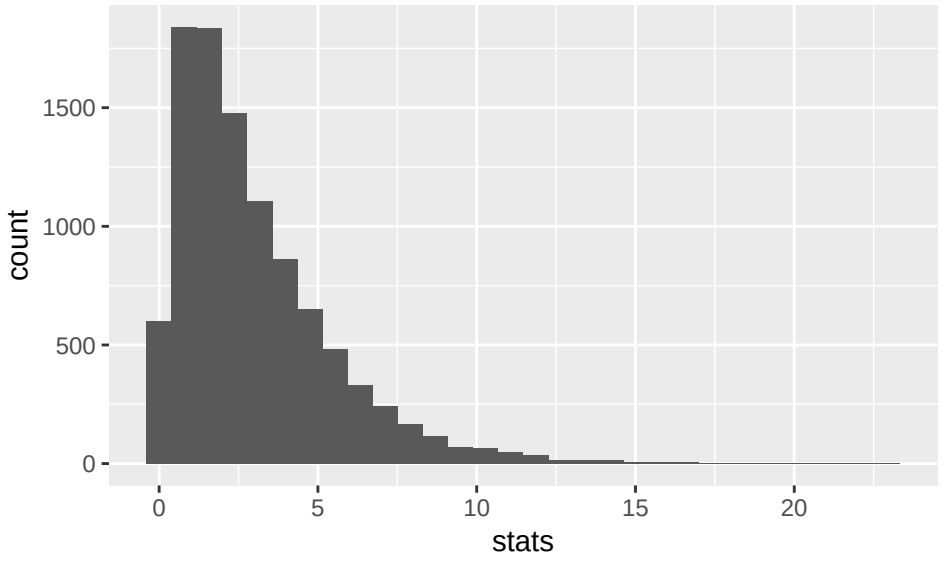
Now we're ready to permute the data. Remember that in each step we need to:

- Randomly permute the rows in one of the columns.
- Calculate the test statistic for the permuted data.

```
# make sure dplyr is loaded for the sample_n function
N <- 10^4 - 1
result <- numeric(N)
grass2 <- grass_complete$grass
degree2 <- grass_complete$degree
for(i in 1:N) {
  grass_permuted <- sample(grass2)
  gss_tbl <- table(degree2, grass_permuted)
  result[i] <- chisq(gss_tbl)
}
```

Now that we have the reference distribution we can inspect it using a histogram

```
library(ggplot2)
ggplot(data = data.frame(stats = result)) +
  geom_histogram(mapping = aes(stats))
```



and calculate a p-value

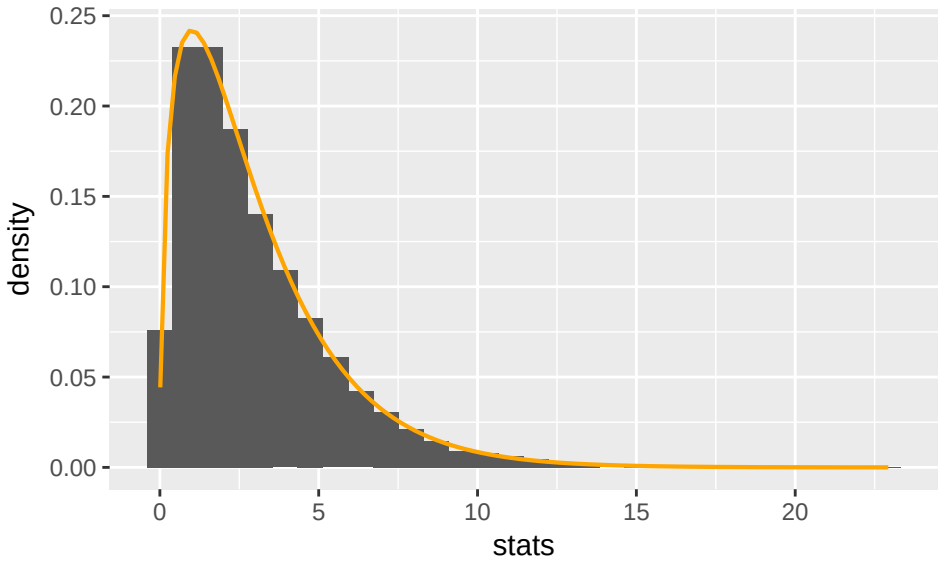
```
# Calculating the p-value
(sum(result >= chisq(observed_tbl)) + 1) / (N + 1)
```

```
## [1] 1e-04
```

What does the p-value indicate?

Using the χ^2 distribution

Below is the histogram of the reference distribution from our permutation test with a χ^2_3 density curve overlaid.



It is easy to calculate a p-value from a χ^2_m distribution using the `pchisq` function (i.e., the CDF):

```
1 - pchisq(24.85, df = 3)
```

```
## [1] 1.659666e-05
```

R can easily perform all of the calculations required for this hypothesis test in one simple function:

```
chisq.test(grass_df$degree, grass_df$grass)
```

```
##
##  Pearson's Chi-squared test
##
## data:  grass_df$degree and grass_df$grass
## X-squared = 24.855, df = 3, p-value = 1.656e-05
```