Goodness-of-Fit Tests

T.Scofield

You may click here to access the .qmd file.

In what follows, I will be working with data from the student survey data found at https://scofield.site/teaching/data/csv/ssurv.csv

```
ssurv = read.csv("https://scofield.site/teaching/data/csv/ssurv.csv")
names(ssurv)
```

```
 [1] "sex"           "class"         "gpa"           "height"
 [5] "pulse"         "childrank"     "numchildren"   "haircut"
 [9] "randomnum"     "speedtickets"  "cds"           "smoker"
[13] "hourssleep"    "selfhandedness" "momhandedness" "dadhandedness"
[17] "region"        "oncampus"      "cupscoffee"    "birthday"
[21] "overtwenty"
```

```
head(ssurv, n=3)
```

```
  sex class gpa height pulse childrank numchildren haircut randomnum
1   F    So 3.6     NA    NA         2           3       2         6
2   F    So 3.4     NA    NA         4           4      10         7
3   M    Fr 3.0     71    68         2           4       0        17
  speedtickets cds smoker hourssleep selfhandedness momhandedness dadhandedness
1            0  15    Non        8.0               R             L             R
2            0  10    Non        8.0               R             R             R
3            2  53    Non        5.5               R             R             R
    region oncampus cupscoffee birthday overtwenty
1 Suburban        Y          1       Th          Y
2    Rural        Y          0       Fr          N
3 Suburban        Y          0       Th          N
```

Obtaining the distribution for a categorical variable: A Quick Review

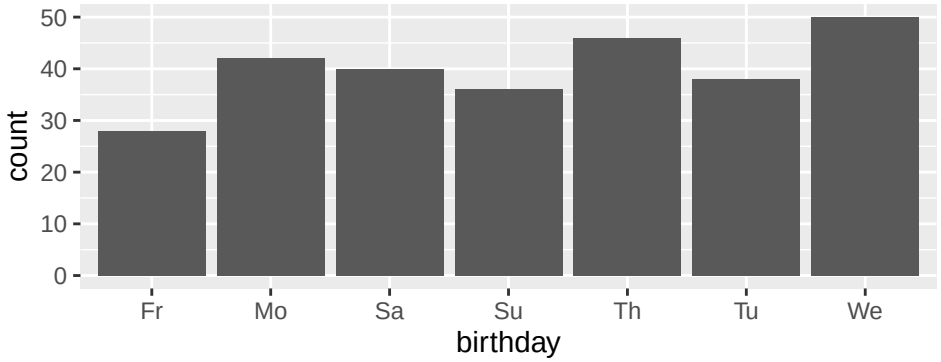
This can be done with a frequency table

```
tally(~birthday, data=ssurv)
```

```
birthday
Fr Mo Sa Su Th Tu We
28 42 40 36 46 38 50
```

or a bar chart

```
gf_bar(~birthday, data=ssurv)
```



We now turn to testing hypotheses for a single categorical variable. This is called a **goodness-of-fit test**

Step 1: Stating hypotheses for goodness-of-fit

If all days of the week were equally-likely to be the day of your birth, this would correspond to a null hypothesis:

$$\mathbf{H}_0 : p_{\text{Mo}} = p_{\text{Tu}} = p_{\text{We}} = p_{\text{Th}} = p_{\text{Fr}} = p_{\text{Sa}} = p_{\text{Su}} = \frac{1}{7}.$$

Here, p_{Mo} refers to the proportion of the population born on a Monday, p_{Tu} the proportion born on Tuesday, and so on. These are proportions, and since there is one for every possible day of the week, it is natural that, whatever their individual values, their sum is

$$p_{\text{Mo}} + p_{\text{Tu}} + p_{\text{We}} + p_{\text{Th}} + p_{\text{Fr}} + p_{\text{Sa}} + p_{\text{Su}} = 1.$$

It seemed natural for us to propose the days are equally-likely for a birth, so we set them all equal to 1/7 in our null hypothesis. The corresponding alternative simply negates the null; in this case, this may be stated as

$$\mathbf{H}_a : \text{ at least one of these proportions is not } 1/7.$$

Step 2: The test statistic (χ^2)

One needs sample data to test hypotheses, and from it we must calculate a *test statistic*. We have the student survey which has a `birthday` variable. It's frequency table is

```
tally(~birthday, data=ssurv) |> addmargins()
```

```
birthday
 Fr  Mo  Sa  Su  Th  Tu  We Sum
 28  42  40  36  46  38  50 280
```

I used `addmargins()` so we see that, in all, the data contains 280 respondents. The individual frequencies are called **observed counts**; we observe, for instance, that 46 of our respondents were born on a Thursday.

A natural test statistic takes into account the discrepancy (residual?) between what we have observed and what we would expect, if our null hypothesis were true. For each value of the variable (i.e., each day of the week), we note that

$$\begin{aligned}
\text{Monday}: \quad np_{\text{Mo}} &= 280 \cdot \frac{1}{7} = 40, \\
\text{Tuesday}: \quad np_{\text{Tu}} &= 280 \cdot \frac{1}{7} = 40, \\
\text{Wednesday}: \quad np_{\text{We}} &= 280 \cdot \frac{1}{7} = 40, \\
\text{Thursday}: \quad np_{\text{Th}} &= 280 \cdot \frac{1}{7} = 40, \\
\text{Friday}: \quad np_{\text{Fr}} &= 280 \cdot \frac{1}{7} = 40, \\
\text{Monday}: \quad np_{\text{Sa}} &= 280 \cdot \frac{1}{7} = 40, \\
\text{Sunday}: \quad np_{\text{Su}} &= 280 \cdot \frac{1}{7} = 40
\end{aligned}$$

That is, in a sample of $n = 280$ people, the equally-likelihood assumption would make us expect 40 as the count for each day of the week. These seven 40's are our **expected counts**.

The test statistic we compute from this data under our null hypothesis is called the **chi-square test statistic**:

$$\chi^2 = \sum_i \frac{(O_i - E_i)^2}{E_i},$$

where the O_i are the observed counts, and the E_i are the expected ones. Note that, were each of the observed counts to match the corresponding expected count, then our computed statistic would be $\chi^2 = 0$. On the other hand, each time $O_i \neq E_i$, it contributes a value that boosts χ^2. The larger the differences between observed and expected counts, the larger χ^2 will be.

We can compute this statistic with commands such as these:

```
observed = c(42, 38, 50, 46, 28, 40, 36)
expected = c(40, 40, 40, 40, 40, 40, 40)
sum((observed - expected)^2 / expected)
```

```
[1] 7.6
```

We can also get this χ^2 statistic using mosaic's `chisq()` command:

```
chisq( tally(~birthday, data=ssurv) )
```

```
X.squared
      7.6
```

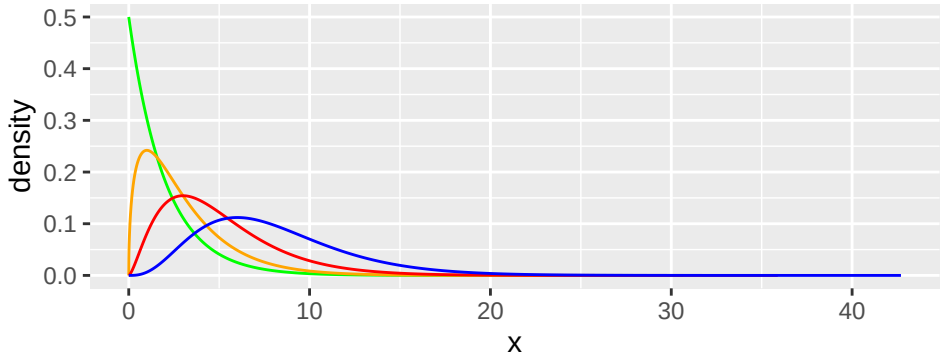
Step 3: Calculating the P-value

In other tests of hypotheses, we have had some **null distribution** we used as a reference to obtain a P-value. There are two ways to obtain one here, detailed below. One is easier than the other, but has limited use. I detail both here.

Use a chi-square distribution as the approximate null distribution

You may not have heard of chi-square distributions. We have not encountered them in this course until now. Compared with other families (normal, binomial, Student-T) we have encountered, they are most like Student-T in that they are distinguished by the number of degrees of freedom. They do **not** share the symmetric, bell-shaped appearance of Student-T distributions, however. Here are several displayed together.

```
gf_dist("chisq", params=c(df=2), color="green") |>
  gf_dist("chisq", params=c(df=3), color="orange") |>
  gf_dist("chisq", params=c(df=5), color="red") |>
  gf_dist("chisq", params=c(df=8), color="blue")
```



It is considered valid to use a chi-square distribution to obtain a P-value only when all expected counts are large enough. The Locks have settled on the rule that **each E_i should be at least 5**. In our running example, involving birthdays, each $E_i = 40$. Two further notes:

- To determine the correct number of degrees of freedom, count the number of values and subtract
    1. As 7 different days appear under birthday, we shall use `df=` $7 - 1 = 6$.
- The P-value corresponds to the **upper tail only**, the area beyond our test statistic.

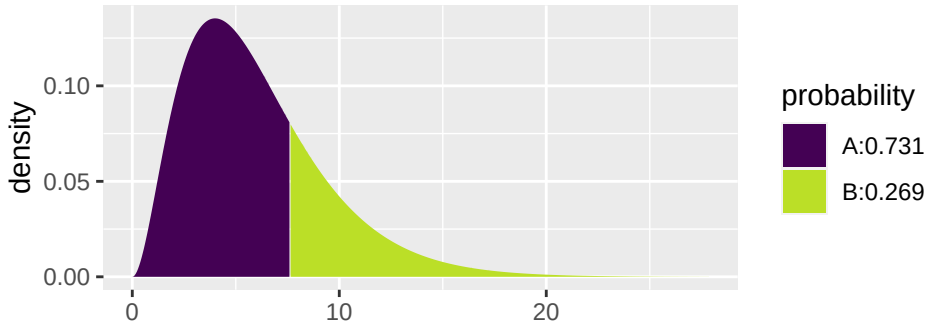
Our P-value in the running example comes from

```
1 - pchisq(7.6, df=6)
```

```
[1] 0.2688967
```

The `xpchisq()` command from the mosaic package helps to visualize this P-value as an area.

```
xpchisq(7.6, df=6)
```




```
[1] 0.7311033
```

Use simulation to build an approximate null distribution

You may be fortunate to have expected counts that all exceed 5. If not, the above method for computing a P-value is frowned upon. Another method is to build an approximate null distribution via simulation.

Under the null hypothesis, any sample would be like an iid drawn from a bag where the options are equally-likely. Such a process can be simulated and used to produce observed counts. Here, I have what amounts to a bag with one slip for each day of the week, so each day should have equal chance to be drawn. As our data had $n = 280$ observations, we will draw that many times.

```
bag = c("Su","Mo","Tu","We","Th","Fr","Sa")
simulatedFreqTable = tally(~resample(bag, size=280))
simulatedFreqTable
```

```
resample(bag, size = 280)
Fr Mo Sa Su Th Tu We
36 37 48 41 37 44 37
```

If we calculate a chi-square statistic using mosaic's `chisq()` command, it would represent one "typical" result in a world where the null hypothesis is true.

```
chisq( simulatedFreqTable )
```

```
X.squared
      3.1
```

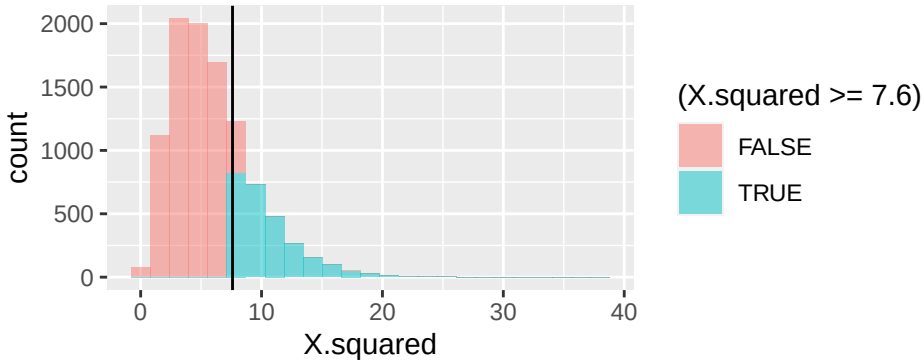
To simulate the appearance of the null distribution, we should produce many instances of typical χ^2-values arising in the null-hypothesis world.

```
manyX2s <- do(10000) * chisq( tally(~resample(bag, size=280)) )
head(manyX2s)
```

```
  X.squared
1      4.05
2      2.50
3      8.65
4     10.70
5      7.25
6      7.05
```

This simulated null distribution, with our test statistic of 7.6 thrown in appears like this:

```
gf_histogram(~X.squared, data=manyX2s, fill=~(X.squared>=7.6)) |> gf_vline(xintercept=7.6)
```



Computing a simulated P-value

Our goal, the approximate P-value, amounts to determining how much area is colored blue. We need the relative frequency associated with the event $\chi^2 \geq 7.6$ (the question being how often, under the null hypothesis, do we see values at least as extreme as our test statistic). Code to obtain this relative frequency can be as simple as:

```
prop(~(X.squared>=7.6), data=manyX2s)
```

```
prop_TRUE
   0.2662
```

This answer compares quite favorably with the one we obtained above using the chi-square distribution with 6 degrees of freedom.

Comparison of the Approach Described Above with the One-Proportion Test (from Section 6.3)

The approach we carried out above, named **Goodness-of-Fit**, which introduced for the first time a χ^2-statistic, was motivated by the desire to handle *non-binary* categorical variables. But there is no reason it cannot work on *binary* ones, as well. Suppose I wish to test whether a certain coin, whose outcomes offer just *two* options, is fair. My hypotheses are

$$\mathbf{H_0}: \ p_{\text{H}} = 0.5 \qquad \text{(with the implication that } p_{\text{T}} = 0.5\text{)}$$
$$\mathbf{H_a}: \ p_H \neq 0.5$$

To test these hypothesis requires data, and let us suppose I have flipped the coin 100 times resulting in 42 heads and 58 tails.

The 1-proportion approach of Section 6.3

I stated my null hypothesis as $p_H = 0.5$, so it is heads I am thinking of as successes. The rules of thumb are met for me to consider the sampling distribution of $\hat{p} \sim \text{Norm}(p, \sqrt{p(1-p)/n})$. So, under the null hypothesis, my null distribution is approximately $\text{Norm}(0.5, 0.05)$. All the same, I'll have more accurate results using continuity correction, which here means my standardized scores uses $42.5/100$ instead of $\hat{p} = 42/100$:

$$Z \ = \ \frac{42.5/100 - 0.5}{0.05} \ = \ -1.5.$$

My resulting P-value is

```
2*pnorm(-1.5)
```

```
[1] 0.1336144
```

The Goodness-of-Fit Approach, Via Simulation

Using $p_H = p_T = 0.5$ (the equally-likely outcomes hypothesis), we have

Values	Observed	Expected
Heads	42	50
Tails	58	50

The χ^2 statistic arising from our $n = 100$ sample coin flips is

$$\chi^2 \ = \ \frac{(42-50)^2}{50} + \frac{(58-50)^2}{50} \ = \ 2.56.$$

To simulate typical χ^2-values that would arise from a fair coin, flipped 100 times, we might run code like this:

```
bag = c("H", "T")
manyX2s <- do(10000) * chisq( tally(~resample(bag, size=100)))
head(manyX2s)
```

```
  X.squared
1      3.24
2      0.64
3      1.00
4      0.04
5      0.16
6      1.00
```

Again, I'll count how often it happens that $\chi^2 \geq 2.56$, and turn that into relative frequency by dividing by 10000:

```
prop( ~(X.squared >= 2.56), data=manyX2s )
```

```
prop_TRUE
   0.1307
```

This answer is similar to the P-value (`2*pnorm(-1.5)`) we obtained using the 1-proportion test, a fact which should be true, generally, given the rules of thumb are met. Just be aware you should expect this similarity of answers only when you compare right-tail area from the chi-square distribution with two-tail area using a 1-proportion test.