Goodness of Fit and Independence

STAT-S520

Arturo Valdivia

04-11-23

General Setting

- Partition the sample space of interest, S, into k events or cells
  - $E_1 \cup E_2 \cup \cdots \cup E_k = S$
  - $E_1, \ldots, E_k$ are pairwise disjoint
- Test various hypotheses about the probabilities of those events.
- Given $E_1, \ldots, E_k$, let $p_j = P(E_j)$ and the vector of cell probabilities $\vec{p} = (p_1, \ldots, p_k)$
- Let Π be the set of all possible probability vectors $\vec{\pi} = (\pi_1, \ldots, \pi_k)$ as long as
  - $\pi_1, \ldots, \pi_k \geq 0$ and
  - $\pi_1 + \cdots + \pi_k = 1$

Hypotheses

- We test

$$H_0 : \overrightarrow{p} \in \Pi_0 \qquad \text{versus} \qquad H_1 : \overrightarrow{p} \in \Pi_1$$

where Π_0 and Π_1 are disjoint sets of probability vectors whose union is Π.

Example 1

Construct $S, E_1, \ldots, E_k$, and $\overrightarrow{p} = (p_1, \ldots, p_k)$ under the null hypothesis that a 6-sided die is fair.

Observed and Expected Cell Counts

- The sample: repeat the experiment n times and let o_j be the number of times that E_j appears, we call this the observed cell count of cell j.
- Goodness-of-fit tests compare observed cell counts to expected cell counts.
  - Expected cell count for cell j, e_j, is obtained assuming the null hypothesis is true.
  - If p_j is the probability of observing E_j under H_0 and the total number of observed values is n, cell j's expected count is $e_j = p_j * n$.

Test Statistics

- Pearson's chi-squared statistic:

$$X^2 = \sum_{j=1}^{k} \frac{(o_j - e_j)^2}{e_j}$$

- The Likelihood ratio chi-squared statistic is

$$G^2 = 2 \sum_{j=1}^{k} o_j \log \left(\frac{o_j}{e_j} \right)$$

- Both X^2 and G^2 statistics can be approximated by a chi-squared distribution.

Example 1: Fair Die (continued)

Let's assume we observed the following data (counts)

```
obs = c(3407, 3631, 3176, 2916, 3448, 3422)
n = sum(obs)
p = rep(1/6,6) #probabilities under the null
exp = n*p
exp
```

```
## [1] 3333.333 3333.333 3333.333 3333.333 3333.333 3333.33
```

```
X2 = sum((obs - exp)^2/exp)
X2
```

```
## [1] 94.189
```

```
G2 = sum(2*obs*log(obs/exp))
G2
```

```
## [1] 95.80227
```

Degrees of Freedom

- The correct degrees of freedom is the difference between the dimensions of the unrestricted and the restricted sets of possible $p_1, \ldots, p_k$
- The unrestricted set has $k - 1$ dimensions (k probabilities, but they must sum to 1)
- The restricted set has less than $k - 1$ dimensions. It is determined by how many probabilities are free to vary.

Example 1 (continued)

Determine whether a 6-sided die is fair. Then

$$H_0 : p_1 = p_2 = \cdots = p_6 = \frac{1}{6}$$

- The unrestricted set has $6 - 1 = 5$ probabilities that are free to vary.
- The null hypothesis specifies a single point, e.g., $p_1 = \cdots = p_6 = 1/6$,
  - No probabilities are free to vary
  - The restricted set has dimension 0.
- The degrees of freedom needed are $df = (6 - 1) - 0 = 5$

Example 1 (continued)

```
df = (6 - 1) - 0
1 - pchisq(X2, df)
```

```
## [1] 0
```

```
1 - pchisq(G2, df)
```

```
## [1] 0
```

Simulation-Based Approach

```
die= as.character(1:6)
die.vec = rep(die,obs)
df1 = data.frame(die.vec)
null_dist <- df1 %>%
  specify(response = die.vec) %>%
  hypothesize(null = "point",
              p = c("1" = 1/6, "2" = 1/6, "3" = 1/6, "4" =
  generate(reps = 1000, type = "draw") %>%
  calculate(stat = "Chisq")
null_dist %>%
  get_p_value(obs_stat = X2, direction = "greater")
```

```
## Warning: Please be cautious in reporting a p-value of 0.
## approximation based on the number of `reps` chosen in th
## `?get_p_value()` for more information.
```

```
## # A tibble: 1 x 1
##   p_value
```

Exercise 2 (ISI 13.4 Exercise 3)

According to Mendelian genetics, a recessive trait will appear in an offspring if and only if both parents contribute a recessive gene. If each parent has a dominant and a recessive gene, then the probability that their offspring will display the recessive trait is $1/4$.

A certain strain of tomato is either tall (dominant trait) or dwarf (recessive trait). The same strain has either cut leaves (dominant trait) or potato leaves (recessive trait). Let E_1 denote tall cut-leaf offspring, let E_2 denote tall potato-leaf offspring, let E_3 denote dwarf cut-leaf offspring, and let E_4 denote dwarf potato-leaf offspring.

Exercise 2 (ISI 13.4 Exercise 3 continued)

In 1931, J. W. MacArthur reported experimental results for $n = 1611$ offspring. MacArthur observed $o_1 = 926$, $o_2 = 288$, $o_3 = 293$, and $o_4 = 104$. Using this information, find:

a. $\vec{p}$, the probability of each E_j (under H_0)
b. The expected counts (under H_0)
c. The test statistic
d. The degrees of freedom
e. The conclusion to the test

(work in R)

Exercise 3: (ISI 13.4 Exercise 6: Using the Poisson Distribution)

Let $X(S) = \{0, 1, 2, \dots\}$. The random variable X is said to have a Poisson distribution with intensity parameter $\mu \in (0, \infty)$, if X has a probability mass function

$$f(x) = P(X = x) = \frac{\mu^x e^{-\mu}}{x!}$$

We write $X \sim Poisson(\mu)$ and it can be shown that $EX = VarX = \mu$. The Poisson distribution frequently arises when counting arrivals in a fixed time interval.

Example 3 (ISI 13.4 Exercise 6 continued)

In 1910, E. Rutherford and M. Geiger counted the numbers of alpha-particle scintillations observed in each of $n = 2608$ 72-intervals. Now we partition $X(S)$ by setting $E_j = \{j - 1\}$ for $j = 1, \ldots, 10$ and $E_{11} = \{10, 11, 12, \ldots\}$. The null hypothesis states that counts of alpha-particle scintillations follow a Poisson distribution. Obtain the vector of $\vec{p}$ that represents the null hypothesis, using the proposed partition. Estimate μ, using the following counts:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	57	203	383	525	532	408	273	139	45	27	10	4	0	1	1

Example 3 (ISI 13.4 Exercise 6 continued)

Using $\hat{\mu}$, find:

a. The expected counts (under H_0)
b. The test statistic
c. The degrees of freedom
d. The conclusion to the test

(work in R)

Independence: Setting

- Let S be the sample space of our experiment and
  - $A_1, \ldots, A_r$ partition S into r cells
  - $B_1, \ldots, B_c$ also partition S into c cells
- Think of As and Bs as two variables with different categories (partition)

Independence: Setting

- We care about the relationship between As and Bs
- We define a third partition by

$$E_{ij} = A_i \cap B_j$$

- Partitions $A_1, \ldots, A_r$ and $B_1, \ldots, B_c$ are mutually independent if and only if

$$P(E_{ij}) = P(A_i) \cdot P(B_j)$$

  for each ij pair.
- We use the chi-squared methods developed above to check if independence holds

Example 2

Two partitions of criminals, one by type of crime (arson, rape, violence, stealing, coining, fraud) and the other by alcohol consumption (drinker, abstainer). Here is the sample (counts) observed:

```
##           drink abstain
## arson        50      43
## rape         88      62
## violence    155     110
## stealing    379     300
## coining      18      14
## fraud        63     144
```

Example 2 (continued)

The expected counts using the outer product (%o%)

```
exp = rowSums(obs)%o%colSums(obs)/sum(obs)
exp
```

```
##                drink   abstain
## arson      49.10870  43.89130
## rape       79.20757  70.79243
## violence  139.93338 125.06662
## stealing  358.54628 320.45372
## coining    16.89762  15.10238
## fraud     109.30645  97.69355
```

Test

- The test statistic is

$$G^2 = 2 \sum_{i=1}^{r} \sum_{j=1}^{c} o_{ij} \log \left(\frac{o_{ij}}{e_{ij}} \right)$$

Degrees of Freedom

- Unrestricted set: $rc - 1$
- Restricted set: $(r - 1) + (c - 1)$
- Degrees of freedom:
  $(rc - 1) - [(r - 1) + (c - 1)] = (r - 1)(c - 1)$

Example 2 (continued)

```
G2 = sum(2*obs*log(obs/exp))
G2
```

```
## [1] 50.51729
```

```
df = (6 - 1)*(2 - 1)
1 - pchisq(G2, df)
```

```
## [1] 1.085962e-09
```

Example 2 Simulation-Based

```
library(tidyverse)
df.obs = as.data.frame(obs)
data2 <- df.obs %>%
  rownames_to_column("crime") %>%
  gather("alcohol",value,-crime) %>%
  rowwise() %>%
  mutate(value = list(1:value)) %>%
  unnest(value) %>%
  select(-value)
```

Example 2 Simulation-Based (continued)

```
null_dist <- data2 %>%
  specify(alcohol ~ crime) %>%
  hypothesize(null = "independence") %>%
  generate(reps = 1000, type = "permute") %>%
  calculate(stat = "Chisq")

null_dist %>%
  get_p_value(obs_stat = X2, direction = "greater")
```

```
## Warning: Please be cautious in reporting a p-value of 0.
## approximation based on the number of `reps` chosen in th
## `?get_p_value()` for more information.

## # A tibble: 1 x 1
##   p_value
##     <dbl>
## 1       0
```