hypothesis_testing_win19

January 11, 2019

1 1 Binomial Test

1.1 Example 1.1

Suppose we have the following data after tossing a coin several times:
[H, T, T, T, H, H, T, H, T, T, H, T, T, T, H, H, T, H, T, T, T, H, T, T, T, H, T, T, T, H, T, T]
Is this a fair coin?

```
In [1]: # create variable to store data
        coin_tosses <- c("H", "T", "T", "T", "H", "H", "T", "H", "T", "T", "H",
                         "T", "T", "T", "H", "H", "T", "H", "T", "T", "T", "H",
                         "T", "T", "T", "H", "T", "T", "T", "H", "T", "T")

        # get number of tosses
        n_tosses <- length(coin_tosses)

        # get number of heads
        n_heads <- sum(coin_tosses == "H")

        # print variables we created to check sanity
        print(n_tosses)
        print(n_heads)
```

```
[1] 32
[1] 11
```

```
In [3]: # run binomial test
        bin_test1 <- binom.test(n_heads, n_tosses)
        print(bin_test1)
```

```
        Exact binomial test

data:  n_heads and n_tosses
number of successes = 11, number of trials = 32, p-value = 0.5909
alternative hypothesis: true probability of success is not equal to 0.4
95 percent confidence interval:
```

```
 0.1857191 0.5319310
sample estimates:
probability of success
               0.34375
```

```
In [4]: # inspect the `test1` object more closely
        #attributes(bin_test1)
        str(bin_test1)
```

```
List of 9
 $ statistic  : Named num 11
  ..- attr(*, "names")= chr "number of successes"
 $ parameter  : Named num 32
  ..- attr(*, "names")= chr "number of trials"
 $ p.value    : num 0.591
 $ conf.int   : atomic [1:2] 0.186 0.532
  ..- attr(*, "conf.level")= num 0.95
 $ estimate   : Named num 0.344
  ..- attr(*, "names")= chr "probability of success"
 $ null.value : Named num 0.4
  ..- attr(*, "names")= chr "probability of success"
 $ alternative: chr "two.sided"
 $ method     : chr "Exact binomial test"
 $ data.name  : chr "n_heads and n_tosses"
 - attr(*, "class")= chr "htest"
```

1.2 Example 1.2

Suppose we are doing quality control for a medical device known to have a 0.001% failure rate. We are given a batch of 250000 to be tested. Of these, we find 17 defective devices. Does this batch have a significantly higher failure rate than our known failure rate?

```
In [5]: # specify our inputs
        n_defectives <- 17
        n_trials <- 250000
        p_failure <- 0.00001
```

```
In [6]: test2 <- binom.test(n_defectives, n_trials, p = p_failure, alternative = "greater")
        print(test2)
```

```

        Exact binomial test

data:  n_defectives and n_trials
number of successes = 17, number of trials = 250000, p-value =
1.557e-09
```

```
alternative hypothesis: true probability of success is greater than 1e-05
95 percent confidence interval:
 4.332901e-05 1.000000e+00
sample estimates:
probability of success
               6.8e-05
```

---

2 2 Pearson's χ^2 (goodness-of-fit) Test

2.1 Example 2.1

Suppose we want to determine whether or not a given die is loaded (i.e., not a fair die). Say we roll the die 100 times, and we obtain the following results:

Value	Count
1	13
2	21
3	15
4	17
5	20
6	14

Are we confident the die is fair?

```
In [12]: # create vector with our counts
         roll_cnts <- c(13, 21, 15, 17, 20, 14)

         # create vector with 6 elements, all 1/6
         probs <- c(1/2, rep(0.5/5, 5))

In [14]: # run test
         chsq_test1 <- chisq.test(roll_cnts, p = probs)

         # print the results
         print(chsq_test1)
```

```
        Chi-squared test for given probabilities

data:  roll_cnts
X-squared = 58.48, df = 5, p-value = 2.504e-11
```

```
In [ ]: str(chsq_test1)
```

3 3 Pearson's χ^2 (Independence) Test

3.1 Example 3.1

Suppose we would like to teach cats to dance. And we have two different training systems: using food as a reward, and using affection as a reward. Suppose that after a week of training the cats, we test their ability to dance. So, we have two categorical variables: *training* and *dance*. The results are below.

		Food as reward	Affection as reward
Cat Dances?	Yes	28	48
	No	10	114

From these data, are the *training* and *dance* varialbes independent?
*Source: Field *et al.* (2012)

```
In [15]: # construct tibble with our cat data
         cats <- data.frame(dance = c(rep(TRUE, 76), rep(FALSE, 124)),
                            training = c(rep("food", 28), rep("affection", 48),
                                         rep("food", 10), rep("affection", 114)))

In [16]: # sanity check to make sure data are correct
         xtab1 <- xtabs(~ dance + training, cats)
         print(xtab1)
```

```
       training
dance   affection food
  FALSE       114   10
  TRUE         48   28
```

```
In [17]: chsq1 <- chisq.test(cats$training, cats$dance)
         print(chsq1)
```

```

	Pearson's Chi-squared test with Yates' continuity correction

data:  cats$training and cats$dance
X-squared = 23.52, df = 1, p-value = 1.236e-06
```

4 4 Student's t-test

4.1 Example 4.1

Suppose you teach high school math and you would like to know whether your students perform at, above, or below average on the math portion of the SAT.

```
In [ ]: library(ggplot2)
```

```
In [ ]: # Define vector of student's SAT scores
        sat <- c(527, 554, 534, 541, 539, 542, 498, 512,
                 528, 531, 563, 566, 498, 503, 551, 582,
                 529, 549, 571, 523, 543, 588, 571)
```

```
In [ ]: ggplot() +
            geom_density(aes(x = sat), fill = "lightblue", colour = "skyblue", alpha = 0.5)
```

```
In [ ]: t.test(sat, mu = 527)
```

4.2 Example 4.2

```
In [ ]: spider <- read.csv("spiderlong.csv")
```

```
In [ ]: print(spider)
```

```
In [ ]: ggplot(spider, aes(x = anxiety, fill = group)) +
            geom_density(alpha = 0.5, colour = "grey")
```

```
In [ ]: ggplot(spider, aes(y = anxiety, x = group, fill = group)) +
            geom_boxplot(width = 0.2) +
            geom_jitter(width = 0.2)
```

```
In [ ]: t.test(anxiety ~ group, data = spider, var.equal = TRUE)
```

4.3 Example 4.3

Consider one of our first examples. Suppose we have developed some new medication to lower cholesterol. We randomly assign 50 patients each to a treatment and control group.

After 6 months, we measure their total cholesterol. We want to know if the treatment group's total cholesterol is different than the control group's.

```
In [ ]: library(tidyverse)

        # Read in data
        drug_trial <- read_csv("drug_trial_data.csv")
```

```
In [ ]: head(drug_trial)
```

```
In [ ]: library(reshape2)       # needed for melt()


        # Construct long-format data for ggplot
        drug_trial_long <- melt(drug_trial,
                                id.vars = c("id", "sex", "age", "group"),
                                measure.vars = c("time1", "time2"),
                                variable.name = "timepoint",
                                value.name = "cholesterol")

        ggplot(drug_trial_long, aes(y = cholesterol, x = timepoint)) +
            geom_jitter(width = 0.2, aes(colour = group))
```

```
In [ ]: t.test(drug_trial$time1, drug_trial$time2, paired = TRUE)
```

```
In [ ]:
```