HW02-Stats in R

Cameron Gibson

March 2020

1 Reporting Binomial Test

Report

```
Exact binomial test

data:  s and f
number of successes = 1859, number of trials = 2360, p-value < 2.2e-16
alternative hypothesis: true probability of success is not equal to 0.5
95 percent confidence interval:
 0.7706496 0.8040569
sample estimates:
probability of success
             0.7877119
```

Two constructions of the dative alternation were tested for equiprobability in American Switchboard Corpus–1,859 instances of the *double object construction* and 501 instances of the *prepositional object construction*, generating 2360 total samples. A binomial test returned a p-value of 2.2e-16 and a 95% confidence interval of 0.7706496 and 0.8040569.

- function()
- as.numeric()
- data.frame()
- binom.test()
- dbinom()
- barplot()

Code

```
#binomtest.R

d.frame <- function(){

  success <- as.numeric(readline(prompt= "How many times did your experiment succeed? "))
  failure <- as.numeric(readline(prompt = "What are the total number of failures? " ))
  s.size <- success + failure
  d <- data.frame("Success" = success, "Total" = success + failure)
  return(d)
}

btest <- function(df){
  s <- df$Success
  f <- df$Total
  b <- binom.test(s, f, p = 0.5,
                  alternative = c("two.sided", "less", "greater"),
                  conf.level = 0.95)

  return(b)
}

density <-function(df, prob = 0.5){
  binom <- dbinom(df$Success, df$Total, prob)

  return(binom)
}

#this doesn't feel correct (don't like this at all)
plotter <- function(df, prob = 0.5){
  x <- df$Success
  y <- df$Total
  successes <- 1:x

  # do each calculation
  probs <- dbinom(successes, size = y, prob)

  # make a table from those two values
  probTable <- data.frame("Success" = successes, "Probs" = probs)
#  print(probTable)
  # display the table
  p <- barplot(height = probTable$Probs,
          names.arg = x,
          space = 0, las = 1,
          ylab = "Probability",
```

```
          xlab = "Syntactic Distribution")
#  p <-hist(probTable$Probs,
#           main = "Syntactic Distribution",
#           col = "Green",
#           border = "Blue",
#           xlab = "Number of DO selections",
#           ylab = "Probabilities")
#  return(p)
}
```

2 McNemar's Test

Report

```
     Exact binomial test

data:  x and n
number of successes = 943, number of trials = 1959, p-value = 0.1038
alternative hypothesis: true probability of success is not equal to 0.5
95 percent confidence interval:
 0.459029 0.503763
sample estimates:
probability of success
              0.481368
```

No clue what any of this means, but I'm NLP4 had more wins I think.

Code

```
#McNemarTest

#opens a tsv file
file_open <-function(filename){
  file <-as.data.frame(fread(filename)) #open file using fread(in data.table), converts
  return(file)
}
setwd('/Users/camerongibson/Dropbox/UNR/Spring 2020/R for Linguistics/hw02-cgibson6279')
tfile <- file_open("PTB.tsv") # creates global variable for opened file

#takes two inputs and sums the c
tag_checker <-function(file, check1, check2){
  s1.correct <- file$gold.tag == check1 #checks correct tags for system 1

  s2.correct <-file$gold.tag == check2 # check correct tags for system 2
  #print(s1.correct)
  #print(s2.correct)
```

```
  comp_check <- sum(s1.correct & !s2.correct) #check how many times system one is correc
  return(comp_check)
}

#creates global variables for each system correct count
Stanford <- tag_checker(tfile, check1 = tfile$Stanford.tag, check2 = tfile$NLP4J.tag)
NLP4J <- tag_checker(tfile, check1 = tfile$NLP4J.tag, check2 = tfile$Stanford.tag)

#run mcnemars test on each system
mcn_test <- function(s1, s2){
  x = min(s1,s2)
  n = s1 + s2
  m = binom.test(x, n, p = 0.5,
             alternative = c("two.sided", "less", "greater"),
             conf.level = 0.95)

  return(m)
}

results <- mcn_test(Stanford, NLP4J)
print(results)
```