

STATS 205: Homework Assignment 5

Brian Liu

6/10/2019

Solution to Problem 1

We say that two observations X_1 and X_2 are *independent* of one another with respect to a collection of events \mathcal{A} if

$$Pr \{X_1 \in A \text{ and } X_2 \in B\} = Pr \{X_1 \in A\} Pr \{X_2 \in B\}$$

where A and B are any two not necessarily distinct sets of outcomes belonging to \mathcal{A}^3 .

– 2.2.1 Independent Observations; Permutation, Parametric, and Bootstrap Tests of Hypotheses; Good, Phillip I

In deciding whether your own observations are exchangeable and a permutation test applicable, the key question is the one we posed in the very first chapter: Under the null hypothesis of no differences among the various experimental or survey groups, can we exchange the labels on the observations without significantly affecting the results?

– 2.2.2 Exchangeable Observations; Permutation, Parametric, and Bootstrap Tests of Hypotheses; Good, Phillip I

Solution to Problem 2

```
cysticerici <- c(28.9, 32.8, 12.0, 9.9, 15.0, 38.0, 12.5, 36.5, 8.6, 26.8);cysticerici
```

```
## [1] 28.9 32.8 12.0 9.9 15.0 38.0 12.5 36.5 8.6 26.8
```

```
worms_reco <- c(1.0, 7.7, 7.3, 7.9, 1.1, 3.5, 18.9, 33.9, 28.6, 25.0); worms_reco
```

```
## [1] 1.0 7.7 7.3 7.9 1.1 3.5 18.9 33.9 28.6 25.0
```

The null hypothesis is that the mean weight of introduced cysticerici *has no correlation with* the mean weight of worms recovered. That is,

$$H_0 : \tau = 0$$

The alternative hypothesis is that the mean weight of introduced cysticerici is *positively correlated with* the mean weight of worms recovered. That is,

$$H_A : \tau > 0$$

To test the null hypothesis against the alternative hypothesis, we will use the Kendall test, a distribution-free test for independence based on signs.

```
cor.test(x = cysticerici, y = worms_reco, method = "kendall", alt = "greater")
```

```
##
```

```
## Kendall's rank correlation tau
```

```
##
```

```
## data: cysticerci and worms_reco
## T = 19, p-value = 0.7578
## alternative hypothesis: true tau is greater than 0
## sample estimates:
##      tau
## -0.1555556
```

The p -value is 0.7578, which is *not* significant at the $\alpha = 0.05$ level. There is *not enough* evidence that the mean weight of introduced cysticerci is *positively correlated with* the mean weight of worms recovered.

Solution to Problem 3

```
cysticerci <- c(28.9, 32.8, 12.0, 9.9, 15.0, 38.0, 12.5, 36.5, 8.6, 26.8)
worms_reco <- c(1.0, 7.7, 7.3, 7.9, 1.1, 3.5, 18.9, 33.9, 28.6, 25.0)
cor.test(x = cysticerci, y = worms_reco, method = "kendall", alt = "greater")
```

```
##
## Kendall's rank correlation tau
##
## data: cysticerci and worms_reco
## T = 19, p-value = 0.7578
## alternative hypothesis: true tau is greater than 0
## sample estimates:
##      tau
## -0.1555556
```

The estimate for $\tau = -0.1555556$.

Solution to Problem 4

```
brain_weight = c(515, 286, 469, 410, 461, 436, 479, 198, 389, 262, 536); length(brain_weight)

## [1] 11

fiber_count = c(32500, 26800, 11410, 14850, 23640, 23820, 29840, 21830, 24650, 22500, 26000); length(fiber_count)

## [1] 11

library(bootstrap)
theta.hat = cor(brain_weight, fiber_count); theta.hat

## [1] 0.1604644

# library(partitions)
# n = 1000
# allCompositions = compositions(n, n); allCompositions[,1:5]
# allCompositions.sub = allCompositions[, sample(1:dim(allCompositions)[2], size=1000, replace=FALSE)]

# draw.bootstrap.samples = function(df){
#   n = dim(df)[1]
#   ind = sample(n, replace = TRUE)
#   cor.bootstrap.replicate = cor(df[ind, "LSAT"], df[ind, "GPA"])
#   return(cor.bootstrap.replicate)
# }
# R = 1000
```

```
# theta.hat.star = replicate(R, draw.bootstrap.samples(law))
# # make a ggplot
# library(ggplot2)
# theta.hat.star.df = data.frame(theta.hat.star = theta.hat.star)
# ggplot(theta.hat.star.df) +
#   geom_density(aes(x = theta.hat.star, y = ..scaled..),
#   fill = "lightblue") +
#   geom_hline(yintercept=0, colour="white", size=1) +
#   theme_bw() +
#   ylab("density") +
#   xlab(bquote(hat(theta))) +
#   geom_vline(xintercept = theta.hat, col = "red")+
#   scale_y_continuous(expand = c(0,0))
```

Solution to Problem 5

```
cysticerci <- c(28.9, 32.8, 12.0, 9.9, 15.0, 38.0, 12.5, 36.5, 8.6, 26.8)
worms_reco <- c(1.0, 7.7, 7.3, 7.9, 1.1, 3.5, 18.9, 33.9, 28.6, 25.0)
```

The null hypothesis is that the mean weight of introduced cysticerci *has no correlation with* the mean weight of worms recovered. That is,

$$H_0 : r_s < r_{s,\alpha}$$

The alternative hypothesis is that the mean weight of introduced cysticerci is *positively correlated with* the mean weight of worms recovered. That is,

$$H_A : r_s \geq r_{s,\alpha}$$

Otherwise, do not reject.

To test the null hypothesis against the alternative hypothesis, we will use the Spearman test, a distribution-free test for independence based on ranks.

```
# this method of performing the test was given in the textbook
library(SuppDists)
qSpearman(p = 0.05, r = 10)
```

```
## [1] -0.5393939
```

Since $r_{s,\alpha} = -0.5393939$, we will reject the null hypothesis only if $r_s \geq -0.5393939$.

Calculating r_s ,

```
cor(x = cysticerci, y = worms_reco, method = "spearman")
```

```
## [1] -0.2
```

Since $r_s = -0.2$ and $r_{s,\alpha} = -0.5393939$, the statement $r_s \geq r_{s,\alpha}$ is *true*. Thus, we *reject* the null hypothesis. There is *sufficient* evidence that the mean weight of introduced cysticerci is *positively correlated with* the mean weight of worms recovered.

NOTE: At this point, I tried to use `cor.test()` with `method = "spearman"` but I got a different result than I expected, and I'm not sure why. Maybe I'm interpreting the output incorrectly?

```
cor.test(x = cysticerici, y = worms_reco, method = "spearman", alternative = "greater")
```

```
##
## Spearman's rank correlation rho
##
## data: cysticerici and worms_reco
## S = 198, p-value = 0.72
## alternative hypothesis: true rho is greater than 0
## sample estimates:
## rho
## -0.2
```

The p -value is 0.72, which is *not* significant at the $\alpha = 0.05$ level. There is *not enough* evidence that the mean weight of introduced cysticerici is *positively correlated with* the mean weight of worms recovered.

Solution to Problem 6

```
x = c(0, 5000, 10000, 15000, 20000, 25000, 30000, 100000)
y = c(0.924, 0.988, 0.992, 1.118, 1.133, 1.145, 1.157, 1.357)
```

The null hypothesis is that the mean weight of introduced cysticerici *has no correlation with* the mean weight of worms recovered. That is,

$$H_0 : \beta = \beta_0$$

$$H_0 : \beta = 0$$

The alternative hypothesis is that the mean weight of introduced cysticerici is *positively correlated with* the mean weight of worms recovered. That is,

$$H_A : \beta > \beta_0$$

$$H_A : \beta > 0$$

To test the null hypothesis against the alternative hypothesis, we will use the Theil test, a distribution-free test for the slope of the regression line.

```
library(NSM3)
```

```
## Loading required package: combinat
##
## Attaching package: 'combinat'
## The following object is masked from 'package:utils':
##
##      combn
## Loading required package: MASS
## Loading required package: partitions
## Loading required package: survival
## fANCOVA 0.5-1 loaded
```

```
## Registered S3 methods overwritten by 'ggplot2':
##   method      from
## [.quosures    rlang
## c.quosures    rlang
## print.quosures rlang
```

```
theil(x, y, alpha=0.05, beta.0=0, type = "u")
```



```
## Alternative: beta greater than 0
## C = 28, C.bar = 1, P = 0
## beta.hat = 0
## alpha.hat = 0.975
##
## 1 - alpha = 0.95 upper bound for beta:
## -Inf, 0
```

```
theil.fit = theil (x,
  y,
  beta.0 = 0 ,
  slopes=TRUE,
  type = "u",
  doplot = FALSE)
theil.fit
```

```
## Alternative: beta greater than 0
## C = 28, C.bar = 1, P = 0
## beta.hat = 0
## alpha.hat = 0.975
##
## All slopes:
## i j      S.ij
## 1 2 1.280000e-05
## 1 3 6.800000e-06
## 1 4 1.293333e-05
## 1 5 1.045000e-05
## 1 6 8.840000e-06
## 1 7 7.766667e-06
```

```
## 1 8 4.330000e-06
## 2 3 8.000000e-07
## 2 4 1.300000e-05
## 2 5 9.666667e-06
## 2 6 7.850000e-06
## 2 7 6.760000e-06
## 2 8 3.884211e-06
## 3 4 2.520000e-05
## 3 5 1.410000e-05
## 3 6 1.020000e-05
## 3 7 8.250000e-06
## 3 8 4.055556e-06
## 4 5 3.000000e-06
## 4 6 2.700000e-06
## 4 7 2.600000e-06
## 4 8 2.811765e-06
## 5 6 2.400000e-06
## 5 7 2.400000e-06
## 5 8 2.800000e-06
## 6 7 2.400000e-06
## 6 8 2.826667e-06
## 7 8 2.857143e-06
##
##
## 1 - alpha = 0.95 upper bound for beta:
## -Inf, 0
```

```
theil.output = theil(x,
  y,
  beta.0 = 0,
  slopes=TRUE,
  type = "u", doplot = FALSE, alpha = .05)
c(theil.output$L, theil.output$U)
```

```
## [1] -Inf    0
```

TODO: Interpret these results correctly.

Solution to Problem 7

```
height = c(42.8, 63.5, 37.5, 39.5, 45.5, 38.5, 43.0, 22.5, 37.0, 23.5, 33.0, 58.0); length(height)

## [1] 12

weight = c(40.0, 93.5, 35.5, 30.0, 52.0, 17.0, 38.5, 8.5, 33.0, 9.5, 21.0, 79.0); length(weight)

## [1] 12

heart_catheter_length = c(37.0, 49.5, 34.5, 36.0, 43.0, 28.0, 37.0, 20.0, 33.5, 30.5, 38.5, 47.0); length(heart_catheter_length)

## [1] 12

model1 <- lm(heart_catheter_length ~ height + weight)
summary(model1)

##
```

```

## Call:
## lm(formula = heart_catheter_length ~ height + weight)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -7.048 -1.258 -0.259  1.899  7.004
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  21.0084     8.7512   2.401  0.0399 *
## height        0.1964     0.3606   0.545  0.5993
## weight        0.1908     0.1652   1.155  0.2777
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.943 on 9 degrees of freedom
## Multiple R-squared:  0.8053, Adjusted R-squared:  0.7621
## F-statistic: 18.62 on 2 and 9 DF,  p-value: 0.0006336
cor.test(x = height, y = weight, method = "pearson")

##
## Pearson's product-moment correlation
##
## data:  height and weight
## t = 11.003, df = 10, p-value = 6.576e-07
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.8634453 0.9893154
## sample estimates:
##          cor
## 0.9610936
confint(model1, conf.level = 0.95)

##              2.5 %      97.5 %
## (Intercept)  1.2119071 40.8048885
## height      -0.6193422  1.0120555
## weight      -0.1827991  0.5644547
library(Rfit)
rfit(heart_catheter_length ~ height + weight)

## Call:
## rfit.default(formula = heart_catheter_length ~ height + weight)
##
## Coefficients:
## (Intercept)      height      weight
## 25.9212658    0.0291139    0.2455696

```