STAT 457 Delinquency Problem

Martha Eichlersmith

Is delinquency related to birth order?

```
alpha <- c(127, 123,  93, 17)/360 #most delinquent
gamma <- c(345, 209, 158, 65)/777 #least delinquent
parameters <- rbind(alpha, gamma)
total <- c(sum(alpha), sum(gamma))
diff <- .5*c(alpha-gamma, sum(alpha-gamma))
table <- rbind( cbind(parameters, total), diff)
rownames(table) <- c("Most Delinquent", "Least Delinquent", "(1/2)Difference (Most - Least)")
colnames(table) <- c("Oldest", "In-between", "Youngest", "Only Child", "Total")
#knitr::kable(table, booktabs=TRUE, 'latex', digits=dec, caption="Row Proportion") %>%
#  kableExtra::kable_styling(latex_options="hold_position" )
table
```

```
##                                    Oldest In-between   Youngest
## Most Delinquent                 0.35277778  0.3416667 0.25833333
## Least Delinquent                0.44401544  0.2689833 0.20334620
## (1/2)Difference (Most - Least) -0.04561883  0.0363417 0.02749356
##                                 Only Child        Total
## Most Delinquent                 0.04722222 1.000000e+00
## Least Delinquent                0.08365508 1.000000e+00
## (1/2)Difference (Most - Least) -0.01821643 3.469447e-17
```

```
func_diff <- function(matrix){
diff.matrix <- cbind( #most - least
   matrix[,1] - matrix[,5] #diff for Oldest
  ,matrix[,2] - matrix[,6] #diff for in-between
  ,matrix[,3] - matrix[,7] #diff for youngest
  ,matrix[,4] - matrix[,8] #diff for only-child
   )
diff.matrix
}

func_values <- function(vec){
  it <- length(vec)
  CI.val <- decimal(quantile(vec, c(0.025, 0.975)), dec)
  CI.95 <- paste("(", paste(CI.val, collapse=", "), ")")
  mean <- decimal(mean(vec), dec)
  pval <- length(vec[vec > 0])/it
  vec <- c(it, mean, CI.95, pval)
  vec
}

func_delinq <- function(it){
set.seed(050104)
prop <- rdirichlet(it, t(parameters))
diff <- cbind( #most - least
   prop[,1] - prop[,5] #diff for Oldest
  ,prop[,2] - prop[,6] #diff for in-between
  ,prop[,3] - prop[,7] #diff for youngest
  ,prop[,4] - prop[,8] #diff for only-child
   )
apply(diff, 2, func_values)
}
```

```r
it.vec <- c(1e04, 1e05, 1e06)
it.1 <- func_delinq(it.vec[1])
it.2 <- func_delinq(it.vec[2])
it.3 <- func_delinq(it.vec[3])
```

```r
func_table <- function(i, name){
table <- cbind(it.1[,i], it.2[,i] , it.3[,i])
rownames(table) <- c("Iterations", "Mean", "95% CI", "P(diff>0)")
title <- paste("Difference in Proportion for", name, ": Most Delinquent - Least Delinquent")
table
#knitr::kable(table, booktabs=T, 'latex', caption=title) %>%
#  kableExtra::kable_styling(latex_options="hold_position" ) #hold table in place
}

func_table(1, "Oldest")
```

```
##            [,1]                    [,2]
## Iterations "10000"                 "100000"
## Mean       "-0.04753"              "-0.04558"
## 95% CI     "( -0.79796, 0.71730 )" "( -0.80043, 0.73221 )"
## P(diff>0)  "0.4313"                "0.43721"
##            [,3]
## Iterations "1000000"
## Mean       "-0.04565"
## 95% CI     "( -0.79722, 0.73221 )"
## P(diff>0)  "0.436071"
```

```r
func_table(2, "In-Between")
```

```
##            [,1]                    [,2]
## Iterations "10000"                 "100000"
## Mean       "0.04349"               "0.03782"
## 95% CI     "( -0.66624, 0.74943 )" "( -0.67684, 0.73751 )"
## P(diff>0)  "0.5723"                "0.56754"
##            [,3]
## Iterations "1000000"
## Mean       "0.03614"
## 95% CI     "( -0.67381, 0.74165 )"
## P(diff>0)  "0.564099"
```

```r
func_table(3, "Youngest")
```

```
##            [,1]                    [,2]
## Iterations "10000"                 "100000"
## Mean       "0.02591"               "0.02642"
## 95% CI     "( -0.64223, 0.68651 )" "( -0.61413, 0.67724 )"
## P(diff>0)  "0.558"                 "0.56111"
##            [,3]
## Iterations "1000000"
## Mean       "0.02761"
## 95% CI     "( -0.61398, 0.68130 )"
## P(diff>0)  "0.563085"
```

```r
func_table(4, "Only")
```

```
##            [,1]                    [,2]
## Iterations "10000"                 "100000"
## Mean       "-0.01886"              "-0.01806"
## 95% CI     "( -0.44356, 0.29082 )" "( -0.43012, 0.28790 )"
## P(diff>0)  "0.3603"                "0.36121"
```

```
##            [,3]
## Iterations "1000000"
## Mean       "-0.01811"
## 95% CI     "( -0.42690, 0.28280 )"
## P(diff>0)  "0.359778"
```

```
func_pval.chisq <- function(pvals){
  x <- -2*sum(log(as.numeric(pvals)))
  combined.pval <-  pnorm(x, 2*length(pvals))
}

pvals.vec <- cbind(it.1[4,], pval.2 <- it.2[4,], pval.3 <- it.3[4,])

combined <- apply(pvals.vec, 2, func_pval.chisq)
combined.pvals <- rbind(it.vec, decimal(combined, dec))
rownames(combined.pvals) <- c("Iterations", "Combined p-value")

title <- paste("Combined p-values:","$X = -2\\sum_{i=1}^4(p_i) \\sim \\chi^2_{df=8}$")
combined.pvals
```

```
##                  [,1]      [,2]      [,3]
## Iterations       "10000"   "1e+05"   "1e+06"
## Combined p-value "0.02310" "0.02168" "0.02265"
```

```
#knitr::kable(combined.pvals, booktabs=T, 'latex', caption=title) %>%
#  kableExtra::kable_styling(latex_options="hold_position" )
```

There is evidence that birth order has an effect on delinquency rates. There may be an argument that younger brothers are more delinquent, but my younger brother is not a delinquent (anecdotal evidence!).