Paper 1 - Reproducibility Matrix

John Michael Schaffer II

## Prerequisites

The following code is required in order to carry out the computations of this paper.

# Load Package Dependencies  
library(PropCIs) # Clopper-Pearson Confidence Interval Method  
library(ggplot2) # Visualization Tool  
library(knitr) # Used to Create Tables

# Read mobility0.csv and store it as dataframe variable named mobility  
mobility = read.csv("http://dept.stat.lsa.umich.edu/~bbh/s485/data/mobility0.csv")  
  
# Set rownames  
row.names(mobility) <- mobility$zone

\*The data used in this analysis can be found at link provided above in the chunk of code.

## Overview

The computations in this document explore upward economic mobility of individuals within the four U.S. Census Bureau regions: Midwest, Northeast, South, and West. The following code consists of statistical analysis on each of the 40 individual zones and four U.S Census Bureau regions, which includes the calculation for Clopper-Pearson confidence intervals. In order to carry out this analysis, various data structures and functions were used to structure our data and compute effectively in addition to the generation of visuals and figures to represent data.

## R Code

### Confidence Intervals and Statistical Analysis

To begin the investivation of upward mobility in each zone, a new column vector named k is added to our mobility data frame. The *k* column variable, represents the # of individuals that achieved upward mobility in each zone and is calculated with the following equation:

This is shown in code below:

# Column vector k is created by multiplying each row by n.lowstart by p.upmover  
k = mobility$n.lowstart \* mobility$p.upmover  
  
# Next, we will add this vector to our mobility dataframe to associate each zone with the # of individuals that achieved upward mobility.  
mobility = data.frame(mobility, k)

By utilizing the modified mobility data frame, two column vectors that contain the lower and upper bounds of the confidence interval, denoted as *l.conf\_i* and *u.conf\_i* respectively, are constructed.

#### Clopper Pearson Confidence Interval - PropCIs Package Method

The code below uses the function exactci(…) from the PropCIs package to create the lower and upper bounds of the Clopper-Pearson confidence intervals for each zone:

conf.int = c()  
l.conf\_i\_PropCIs = c()  
u.conf\_i\_PropCIs = c()  
  
for (row in 1:nrow(mobility))  
{  
 conf.int = c(conf.int, exactci(mobility[row,"k"], mobility[row,"n.lowstart"], conf.level = 0.95))  
}  
  
for (i in 1:40){  
 l.conf\_i\_PropCIs = c(l.conf\_i\_PropCIs, conf.int[[i]][1])  
 u.conf\_i\_PropCIs = c(u.conf\_i\_PropCIs, conf.int[[i]][2])  
}  
  
# Now that we have two column vectors of the lower and upper bounds of the interval, we initialize a new data.frame instance and include these two column vectors  
mobility = data.frame(mobility, l.conf\_i\_PropCIs, u.conf\_i\_PropCIs)

#### Clopper Pearson Confidence Interval - Quantile of Beta Method

In order to test the validity of the packages calculations the values for the lower and upper bounds of each zones confidence interval are calculated through the Clopper-Pearson equation, which is:

In base R, the qbeta(…) function utilizes the above equation to calculate confidence intervals and is shown below:

# First we set out alpha value to indicate a 95% confidence interval  
alpha = rep(0.05, length(mobility$zone))  
  
# Next we calculate the lower and upper CP CI using the qbeta(...) function which utilizes the equation shown above.  
  
l.conf\_i\_calc = qbeta(alpha/2,mobility$k, mobility$n.lowstart - mobility$k + 1)  
  
u.conf\_i\_calc = qbeta(1-alpha/2, mobility$k + 1, mobility$n.lowstart - mobility$k)  
  
  
# Now that we have two column vectors of the lower and upper bounds of the interval, we initialize a new data.frame instance and include these two column vectors  
mobility = data.frame(mobility, l.conf\_i\_calc, u.conf\_i\_calc)

By viewing the mobility data matrix columns, we can see that both of the CIs are the same for the package calculation and qbeta(…) calculation:

head(mobility[,9:12])

## l.conf\_i\_PropCIs u.conf\_i\_PropCIs l.conf\_i\_calc  
## Albuquerque 0.0125485878 0.1654819 0.0125485878  
## Phoenix 0.0007443642 0.1532677 0.0007443642  
## Salt Lake City 0.0123485272 0.3169827 0.0123485272  
## San Jose 0.0473536266 0.3738417 0.0473536266  
## San Francisco 0.0011001686 0.2194866 0.0011001686  
## Las Vegas 0.0180376398 0.2305750 0.0180376398  
## u.conf\_i\_calc  
## Albuquerque 0.1654819  
## Phoenix 0.1532677  
## Salt Lake City 0.3169827  
## San Jose 0.3738417  
## San Francisco 0.2194866  
## Las Vegas 0.2305750

#### Formulation of Region Data

Furthering our analysis, we observe the differences of the four commuting zones.

# Create a data matrix for each commuting zone: west, midwest, northeast, south  
midwest = subset(mobility, region %in% "midwest")  
northeast = subset(mobility, region %in% "northeast")  
south = subset(mobility, region %in% "south")  
west = subset(mobility, region %in% "west")

Now that we have created subsets of zones with respect to their specific region, we can create variables for *n* (# of indviduals with a lowstart), *phat* (observed probability of upward mobility), and *k* (# of individuals that achieved upward mobility). After computing these values for each region, we will generate a mobility dataframe for regions.

# Initialize regionMobility dataframe variable  
regionsName = c("midwest", "northeast", "south", "west")  
regionMobility = data.frame(matrix(nrow = 4, ncol = 3))  
rownames(regionMobility) = regionsName  
colnames(regionMobility) = c("n", "phat", "k")  
  
# Next, we will create a list that contains our region dataframes and run a for loop to calculate n, phat, and k for each region.  
regions = list(midwest, northeast, south, west)  
i = 1  
for (region in regions){  
 regionMobility[i,] = list(sum(region$n.lowstart), sum(region$k)/sum(region$n.lowstart), sum(region$k))  
 i = i + 1  
}  
  
regionMobility = data.frame(regionMobility, regionsName)

Next, we generate Clopper-Pearson confidence intervals using the Beta method:

# First we set out alpha value to indicate a 95% confidence interval  
alpha = rep(0.05, 4)  
  
# Next we calculate the lower and upper CP CI using the qbeta(...) function which utilizes the equation shown above.  
  
l.conf\_i\_calc = qbeta(alpha/2,regionMobility$k, regionMobility$n - regionMobility$k + 1)  
  
u.conf\_i\_calc = qbeta(1-alpha/2, regionMobility$k + 1, regionMobility$n - regionMobility$k)  
  
  
# Now that we have two column vectors of the lower and upper bounds of the interval, we initialize a new data.frame instance and include these two column vectors  
regionMobility = data.frame(regionMobility, l.conf\_i\_calc, u.conf\_i\_calc)

### Figures and Visualizations

This code block contains a graph of all zones, their p.upmover value, and their confidence interval bounds at the hash marks region:

# Line plot with multiple groups  
p <- ggplot(data=mobility, aes(x=zone, y=p.upmover, group = 1)) + geom\_point() + theme(axis.title.x=element\_blank(),  
 axis.text.x=element\_blank(),  
 axis.ticks.x=element\_blank()) + labs(x = "Zones" , y = "Probability of Upward Mobility")  
  
p <- p + geom\_errorbar(aes(ymin = mobility$l.conf\_i\_calc, ymax = mobility$u.conf\_i\_calc), width = 0.2)  
  
p

This code contains the region table and graph:

regionTbl <- kable(regionMobility[,5:6], col.names = c("Lower Limit", "Upper Limit"))  
#regionGraph <- ggplot(data=regionMobility, aes(x=regionsName, y=phat, group = 1)) + geom\_point() + labs(x = "Region" , y = "Probability of Upward Mobility")  
#regionGraph <- regionGraph + geom\_errorbar(aes(ymin = regionMobility$l.conf\_i\_calc, ymax = regionMobility$u.conf\_i\_calc), width = 0.2)  
print(regionTbl)

##   
##   
## Lower Limit Upper Limit  
## ---------- ------------ ------------  
## midwest 0.0469972 0.1078831  
## northeast 0.0593248 0.1140218  
## south 0.0601388 0.1091505  
## west 0.0524429 0.1053641

#print(regionGraph)

\*Note that ggplot code is commented out becuase it does not run in RMarkdown