Cleaning/Preparing Human-Dimensions Data with Excel

(IN QUALTRICS)

* Close survey
* Export data from Qualtrics as csv (keep everything in CSV form)

(IN EXCEL)

* Remove invalid responses
  + Redundant responses (eye test first)
  + Individuals who do not fulfill survey criteria (i.e., underaged, wrong state)
* Recode short title w/o spaces for each question response (i.e., Age, Days.Hunting)
* Remove description row

Analyzing Human-Dimensions Data in Excel

(IN EXCEL)

* Select column of interest, insert pivot table
* Drag metric to both “Rows” and “Values”
* Select row labels column of pivot table, deselect: (blank), Select answer, Prefer not answer
  + The number in grand total is your total responses for this question: note this
* Right click “Count of X”, click value field setting, click show values as, click drop down
  + Click % column total, okay
  + This gives you the percent response for each group
  + If you need to combine groups (i.e., as non-white responses):
    - Copy over each percent and add up separate from table
  + Be sure to delete each sheet after you note your stats, as we are in CSV form
* For continuous responses (i.e., Age, Years of residence, Days hunting), be sure to visualize data or put in pivot table and check the range (either unrealistic high or low thrown out—i.e., Days hunting last year: 500 days; Age—150 years)
  + Remember this impacts the number of responses for each question

Analyzing Human-Dimensions Data in RStudio

(IN RStudio)

* Read in data and check format
* For each question, we want to convert our Y and all of our Xs into binary categories
* Unless we are interested in examining the role of unsurety vs surety in response, we can throw out NA or Not sure responses as NA
  + For continuous variables, we throw out all unrealistic responses (noted earlier)
    - Find the median
    - All values greater than median are “More” and get coded as 1
    - All values less than or equal to median are “Less” and get coded as 0

# Age

cwd$Age[cwd$Age > 1971] <- "Older"

cwd$Age[cwd$Age < 1972] <- "Younger"

* + For categorical variables:
    - 2 responses like Yes or No stay as Yes (1) or No (0)
    - 3+ responses like Race get categorized as White or Nonwhite

# Race

cwd$Race[cwd$Race == "White"] <- "White"

cwd$Race[cwd$Race == "Asian"] <- "Nonwhite"

cwd$Race[cwd$Race == "Black or African American"] <- "Nonwhite"

cwd$Race[cwd$Race == "American Indian or Alaskan Native"] <- "Nonwhite"

cwd$Race[cwd$Race == "Other"] <- "Nonwhite"

* For Likert scale responses (i.e., Not concerned 🡪 Very concerned)
  + Typically on 4- to 5-point scale
  + Can be done through logic or by where the median falls
  + If logic, typically for gauging knowledge or agreement
    - Ex. Likert on knowledge:

# QDM Knowledge

# More knowledgeable

cwd$QDM.Knowledge[cwd$QDM.Knowledge == "Good"] <- "More knowledgeable"

cwd$QDM.Knowledge[cwd$QDM.Knowledge == "Excellent"] <- "More knowledgeable"

# Less knowledgeable

cwd$QDM.Knowledge[cwd$QDM.Knowledge == "None"] <- "Less knowledgeable"

cwd$QDM.Knowledge[cwd$QDM.Knowledge == "Poor"] <- "Less knowledgeable"

* + If comparing the percent in one group that is more knowledgeable to the percent in another group that is more knowledgeable, we can use the median 🡪 logic is probably okay to use as well

Cleaning/Preparing Human-Dimensions Data with Rstudio

* Now that everything is in binary, we can run our stats
* Chi square are typical for comparing the percentage response for each X to the Y and seeing if they are statistically different
* Set up data table so we can run the Chi:

data.table <- table(cwd$Consume, cwd$Age)

* Visualize your table

data.table

* Run chi square

data.chi <- chisq.test(data.table)

* Can calculate counts for each column

# Calculate column-wise sums

col\_counts <- colSums(data.table)

# Add column counts as a new row to the contingency table

data.table\_with\_counts <- rbind(data.table, Col\_Counts = col\_counts)

# Print the table with column counts

print(data.table\_with\_counts)

* Report chi

print(data.chi)

* Run Cramer’s V: this tells us the relative strength of the relationship
  + weak (< .10), moderate (.10 - .15), or strong (≥ .15)
  + Package: library(rcompanion)

cramerV(data.table)

* Repeat this process for every X and Y (can be tedious, but if your code is set up then its just hammering the enter button and keeping track of your numbers)
* I would also suggest having your tables in Word ready to go so you can just plug and chug
* Data worth reporting: Sample size, percent for each response, p-val, chi square, degrees of freedom, cramer’s v
* In text, we only report the p-value if everything else is in the table
* In text, we want to report different data than in table, so “X group was N% more likely than Y group to M…” (basically dividing percentage of one group over the other group for response)
* If we are using a large sample size, Bonferroni correction factors are something to look into
  + This is pretty easy to incorporate, and it is applied to comparison-wise p-vals to standardize the experiment-wide error rate