Descriptive Analysis of a Multilevel Data Set

Katherine Wolf

17 March, 2020

```
## cd
## 101 102 103 104 105 106 107 108 109 110 111 112 201 202 203 204 205 206 207 208
##  41  22  30  64  55  30  64  53  84  56  48  61  66  69  83  50  84  75  57  52
## 209 210 211 212 213 214 215 216 217 218 301 302 303 304 305 306 307 308 309 310
##  49  50  64  90  43  95  67  49  64  91  19  58  73  62  27  77 131 144  53  56
## 311 312 401 402 403 404 405 406 407 408 409 410 411 412 413 414 501 502 503
##  52  94 113  44  74  61  82  73 106  77  78  66  57 136 100  67  74  67  73
```

```
## cd
##     101     102     103     104     105     106     107     108     109     110
## 0.01025 0.00550 0.00750 0.01600 0.01375 0.00750 0.01600 0.01325 0.02100 0.01400
##     111     112     201     202     203     204     205     206     207     208
## 0.01200 0.01525 0.01650 0.01725 0.02075 0.01250 0.02100 0.01875 0.01425 0.01300
##     209     210     211     212     213     214     215     216     217     218
## 0.01225 0.01250 0.01600 0.02250 0.01075 0.02375 0.01675 0.01225 0.01600 0.02275
##     301     302     303     304     305     306     307     308     309     310
## 0.00475 0.01450 0.01825 0.01550 0.00675 0.01925 0.03275 0.03600 0.01325 0.01400
##     311     312     401     402     403     404     405     406     407     408
## 0.01300 0.02350 0.02825 0.01100 0.01850 0.01525 0.02050 0.01825 0.02650 0.01925
##     409     410     411     412     413     414     501     502     503
## 0.01950 0.01650 0.01425 0.03400 0.02500 0.01675 0.01850 0.01675 0.01825
```

```
## boro
##    1    2    3    4    5
##  608 1198  846 1134  214
```

```
## boro
##      1      2      3      4      5
## 0.1520 0.2995 0.2115 0.2835 0.0535
```

```
## agecat
##   1   2   3   4   5   6
## 350 685 815 808 612 690
```

```
## agecat
##       1       2       3       4       5       6    <NA>
## 0.08750 0.17125 0.20375 0.20200 0.15300 0.17250 0.01000
```

```
## racecat
##    1    2    3    4    5
## 1616 1055  164  958   95
```

```
## racecat
##       1       2       3       4       5    <NA>
## 0.40400 0.26375 0.04100 0.23950 0.02375 0.02800
```

```
## edcat
##   1   2   3   4   5
## 508 923 879 883 730
```

```
## edcat
##       1       2       3       4       5    <NA>
## 0.12700 0.23075 0.21975 0.22075 0.18250 0.01925
```

```
## inc3cat
##    1    2    3
## 1605 1093  722
```

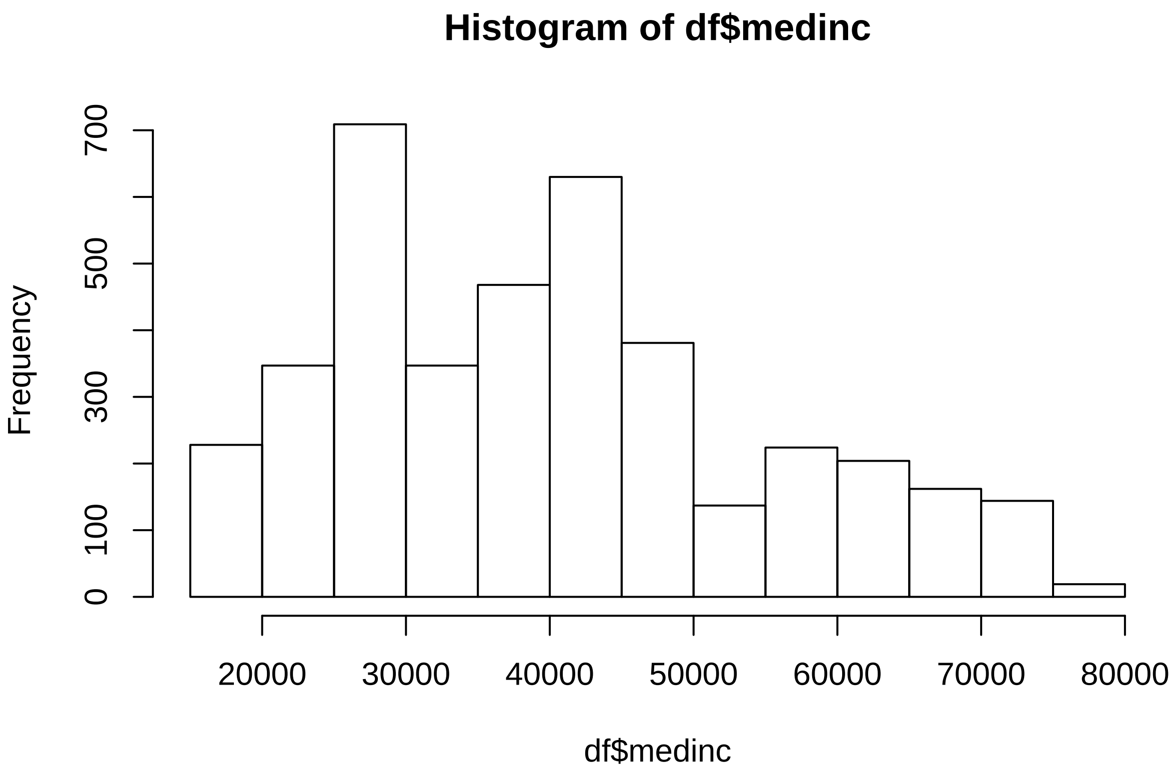
```
## inc3cat
##       1       2       3    <NA>
## 0.40125 0.27325 0.18050 0.14500
```

```
## binge
##    0    1
## 3562  438
```

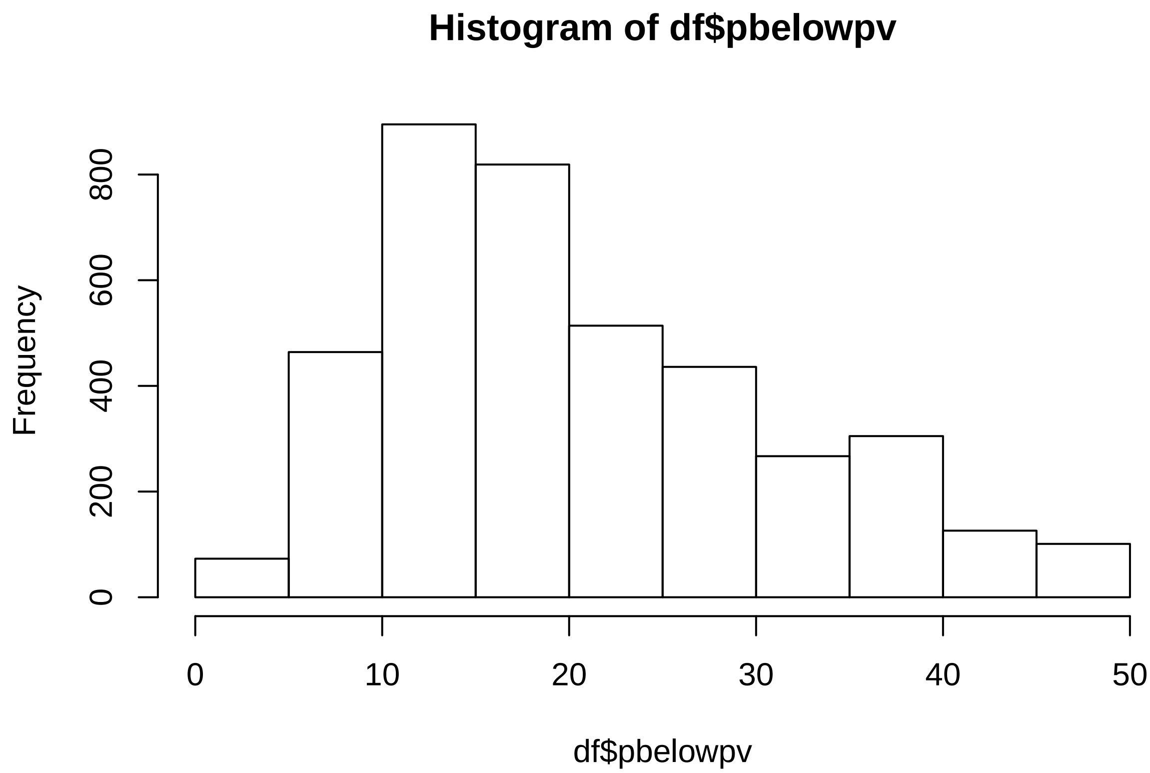
```
## binge
##      0      1
## 0.8905 0.1095
```

```
##     min median     mean   max       sd
## 1 16000  38965 40379.44 79475 15000.96
```

```
##        min  median     mean      max       sd
## 1 4.900159 19.1141 20.75689 45.66544 10.71602
```

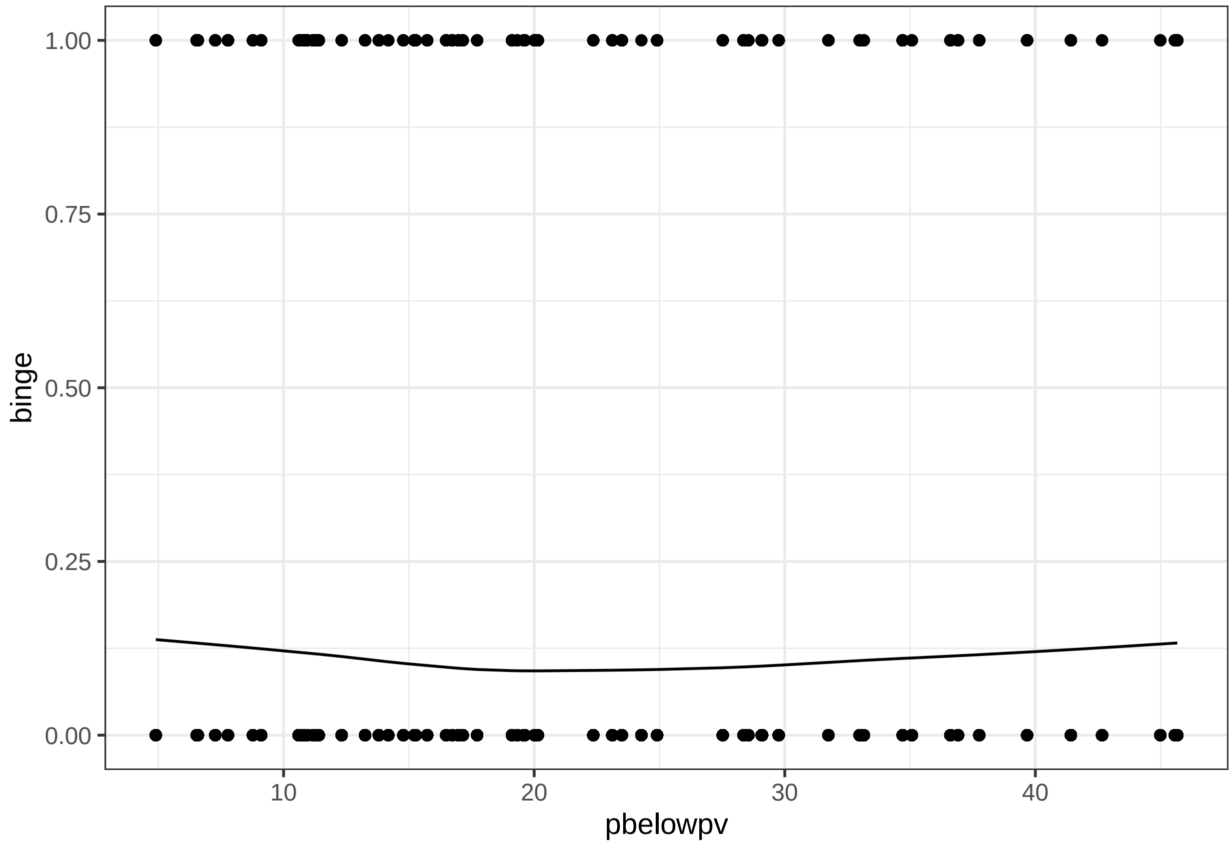


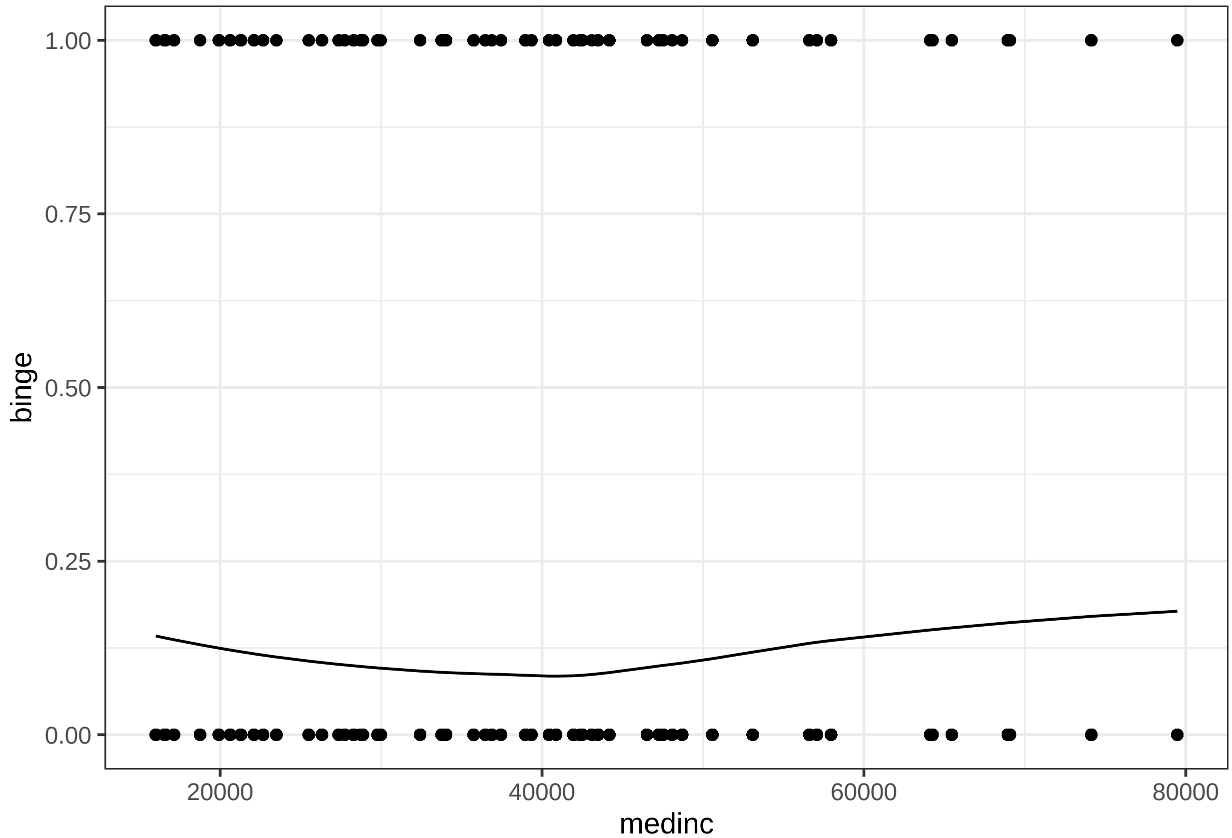
Histogram of df$medinc
Frequency
0
100
300
500
700
20000
30000
40000
50000
60000
70000
80000
df$medinc




```
##
##    1    2    3 <NA>
## 1605 1093  722  580
```

```
##
##       1       2       3    <NA>
## 0.40125 0.27325 0.18050 0.14500
```


1.00
0.75
0.50
0.25
0.00
binge
10
20
30
40
pbelowpv




```
##
##  Pearson's Chi-squared test
##
## data:  df$povq and df$binge
## X-squared = 7.6782, df = 3, p-value = 0.05315

##        df$binge
## df$povq   0   1
##       1 890 135
##       2 939 101
##       3 862  96
##       4 871 106

##        df$binge
## df$povq          0          1
##       1 0.86829268 0.13170732
##       2 0.90288462 0.09711538
##       3 0.89979123 0.10020877
##       4 0.89150461 0.10849539

##
##  Pearson's Chi-squared test
##
## data:  df$medincq and df$binge
## X-squared = 29.21, df = 3, p-value = 2.023e-06
```

```
##
##  Pearson's Chi-squared test
##
## data:  df$agecat and df$binge
## X-squared = 161.51, df = 5, p-value < 2.2e-16

##
##  Pearson's Chi-squared test
##
## data:  df$racecat and df$binge
## X-squared = 34.408, df = 4, p-value = 6.144e-07

##
##  Pearson's Chi-squared test
##
## data:  df$edcat and df$binge
## X-squared = 15.801, df = 4, p-value = 0.003299

##
##  Pearson's Chi-squared test
##
## data:  df$inc3catm and df$binge
## X-squared = 53.638, df = 3, p-value = 1.341e-11

##
##  Pearson's product-moment correlation
##
## data:  df$medinc and df$pbelowpv
## t = -127.38, df = 3998, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.9016850 -0.8894228
## sample estimates:
##        cor
## -0.8957241

##
##  Pearson's Chi-squared test
##
## data:  agecatm and inc3catm
## X-squared = 349.75, df = 18, p-value < 2.2e-16

##        inc3catm
## agecatm   0   1   2   3
##       0  31   8   0   1
##       1  70 170  77  33
##       2  57 282 212 134
##       3  75 298 257 185
##       4  85 291 245 187
##       5 104 215 173 120
##       6 158 341 129  62
```

```
##        inc3catm
## agecatm          0          1          2          3
##       0 0.77500000 0.20000000 0.00000000 0.02500000
##       1 0.20000000 0.48571429 0.22000000 0.09428571
##       2 0.08321168 0.41167883 0.30948905 0.19562044
##       3 0.09202454 0.36564417 0.31533742 0.22699387
##       4 0.10519802 0.36014851 0.30321782 0.23143564
##       5 0.16993464 0.35130719 0.28267974 0.19607843
##       6 0.22898551 0.49420290 0.18695652 0.08985507
```

```
## # weights:  45 (28 variable)
## initial  value 3695.731739
## iter  10 value 2931.759311
## iter  20 value 2898.521148
## iter  30 value 2888.656114
## final  value 2888.290158
## converged
```

```
## Call:
## multinom(formula = inc3cat ~ factor(agecat) + factor(racecat) +
##     factor(edcat), data = df)
##
## Coefficients:
##   (Intercept) factor(agecat)2 factor(agecat)3 factor(agecat)4 factor(agecat)5
## 2   -1.544601      0.07565201       0.3117244       0.3159686       0.1456632
## 3   -2.504560      0.06590732       0.5084491       0.5624438       0.1551581
##   factor(agecat)6 factor(racecat)2 factor(racecat)3 factor(racecat)4
## 2      -0.6877356       -0.5100436       -0.7364447       -0.9628249
## 3      -1.0940419       -1.2431933       -0.9840444       -1.5918134
##   factor(racecat)5 factor(edcat)2 factor(edcat)3 factor(edcat)4 factor(edcat)5
## 2        -0.445790      0.9705263       1.486249       2.285945       3.039878
## 3        -1.332637      0.8914704       1.941091       3.148123       4.274757
##
## Std. Errors:
##   (Intercept) factor(agecat)2 factor(agecat)3 factor(agecat)4 factor(agecat)5
## 2   0.2368367       0.1806596       0.1747156       0.1752042       0.1888811
## 3   0.3933508       0.2457323       0.2373412       0.2374721       0.2543142
##   factor(agecat)6 factor(racecat)2 factor(racecat)3 factor(racecat)4
## 2       0.1915145        0.1107800        0.2231841        0.1235579
## 3       0.2679901        0.1409942        0.2474547        0.1591730
##   factor(racecat)5 factor(edcat)2 factor(edcat)3 factor(edcat)4 factor(edcat)5
## 2        0.2832681      0.1885106      0.1875705      0.1928622      0.2171170
## 3        0.3813625      0.3578075      0.3433937      0.3410133      0.3533725
##
## Residual Deviance: 5776.58
## AIC: 5832.58
```

```r
knitr::opts_chunk$set(echo = FALSE,
                      warning = FALSE,
                      message = FALSE)


# need to call the libraries every time you begin a new R session (from one)
```

```
library(dplyr)
library(ggplot2)
library(nnet)
library(tidyverse)
library(tableone)
library(xtable)
library(knitr)
library(tableone)
library(kableExtra)
library(here)

# load packages necessary for this assignment
library(lme4)
library(sjstats)
library(gee)
library(car)


# # read in data - suppose the file dataset.csv contains continuous variables var1 and var2, and a bina
# df <- read.csv("working_data.csv")
#
# # create a categorical variable from a continuous variable
# df$catvar1 <- df$var1
# df$catvar1 <- ifelse(df$catvar1<=500,0,ifelse(df$catvar1>500,1,NA))
#
# # describe variables
# summary(df$var1)
# table(df$var1)
# hist(df$var1)
#
# with(df, table(var1,var2))
# with(df, table(var1,var2, exclude=NULL))
#
# df %>%  group_by(catvar1) %>% summarise(mean_outcome = mean(outcome))
#
#
# # bivariable relations
#
# # to do a Pearson's chi-squared test
# x2 <- chisq.test(df$var1, df$var2)
#
# # to see the results of the test
# x2
#
# # to see the table of observed
# x2$observed
#
# # to see the percents by row
# prop.table(x2$observed, 1)
#
# # to see the percents by column
# prop.table(x2$observed, 2)
#
```

```r
# # calculate correlation between variables
# cor.test(df$var1, df$var2)
#
# # summarize relationship between variables in a plot with a lowess line
# ggplot(df) + geom_point(aes(x=var1, y=outcome))  +
#   geom_line(aes(x=var1, y=predict(loess(outcome~var1)))) + theme_bw()



# GENERAL DESCRIPTION

# read in data
df <- read.csv("NYSES data for class.csv")

# frequencies and percentages for categorical variables
# before running these frequencies you might have applied formats if you find that helpful

with(df, table(cd), exclude=NULL)
with(df, prop.table(table(cd, exclude=NULL)))
with(df, table(boro), exclude=NULL)
with(df, prop.table(table(boro, exclude=NULL)))
with(df, table(agecat), exclude=NULL)
with(df, prop.table(table(agecat, exclude=NULL)))
with(df, table(racecat), exclude=NULL)
with(df, prop.table(table(racecat, exclude=NULL)))
with(df, table(edcat), exclude=NULL)
with(df, prop.table(table(edcat, exclude=NULL)))
with(df, table(inc3cat), exclude=NULL)
with(df, prop.table(table(inc3cat, exclude=NULL)))
with(df, table(binge), exclude=NULL)
with(df, prop.table(table(binge, exclude=NULL)))

# means etc and plots for continuous variables

df %>% summarise(min=min(medinc), median = median(medinc), mean=mean(medinc), max=max(medinc), sd=sd(me

df %>% summarise(min=min(pbelowpv), median = median(pbelowpv), mean=mean(pbelowpv), max=max(pbelowpv), 

hist(df$medinc)
hist(df$pbelowpv)



# MISSING DATA

table(df$inc3cat, exclude=NULL)
prop.table(table(df$inc3cat, exclude=NULL))

df$inc3catm <- ifelse(is.na(df$inc3cat),0, df$inc3cat)



# CATEGORIZING VARIABLES
```

```
# create quarters of neighborhood ses variables

df$povq <- ifelse(df$pbelowpv>=min(df$pbelowpv)& df$pbelowpv<=11.40486,1,
            ifelse(df$pbelowpv>11.40486 & df$pbelowpv<=19.1141,2,
              ifelse(df$pbelowpv>19.1141 & df$pbelowpv<=29.07797,3,
    ifelse(df$pbelowpv>29.07797 & df$pbelowpv<=max(df$pbelowpv),4, NA))))


df$medincq <- ifelse(df$medinc>=min(df$medinc) & df$medinc<=28780, 1,
                     ifelse(df$medinc>28780 & df$medinc<=38965, 2,
                ifelse(df$medinc>38965 & df$medinc<=48085,3,                ifelse(df$medinc>48085 &



# BIVARIABLE RELATIONS
ggplot(df) + geom_point(aes(x=pbelowpv, y=binge))  +
  geom_line(aes(x=pbelowpv, y=predict(loess(binge~pbelowpv)))) + theme_bw()

ggplot(df) + geom_point(aes(x=medinc, y=binge))  +
  geom_line(aes(x=medinc, y=predict(loess(binge~medinc)))) + theme_bw()

# bivariable relations with binge drinking
# can either save the test results as an object, then examine attributes of the object
# or just do the test and have the result printed
x2 <- chisq.test(df$povq, df$binge)
x2
x2$observed
prop.table(x2$observed, 1)

chisq.test(df$medincq, df$binge)
chisq.test(df$agecat, df$binge)
chisq.test(df$racecat, df$binge)
chisq.test(df$edcat, df$binge)
chisq.test(df$inc3catm, df$binge)

cor.test(df$medinc, df$pbelowpv)



# MORE ON MISSING INCOME

# need to create missing categories to include them in the chi-squared test

agecatm <- ifelse(is.na(df$agecat),0,df$agecat)
inc3catm <- ifelse(is.na(df$inc3cat),0,df$inc3cat)
racecatm <- ifelse(is.na(df$racecat),0,df$racecat)
edcatm <- ifelse(is.na(df$edcat),0,df$edcat)

x2 <- chisq.test(agecatm, inc3catm)
x2
x2$observed
prop.table(x2$observed, 1)
```

```r
x2 <- chisq.test(racecatm, inc3catm)
x2 <- chisq.test(edcatm, inc3catm)



mfit <- multinom(inc3cat ~ factor(agecat) + factor(racecat) + factor(edcat), data=df)

summary(mfit)

# because the regression drops those who are missing, need to identify those observations with missing

df$missing <- ifelse(is.na(df$agecat) | is.na(df$racecat) |is.na(df$edcat),1,0)

df$incpr1 <- ifelse(df$missing==1,NA,predict(mfit, type="probs")[,1])
df$incpr2 <- ifelse(df$missing==1,NA,predict(mfit, type="probs")[,2])
df$incpr3 <- ifelse(df$missing==1,NA,predict(mfit, type="probs")[,3])



#code from assignment 1 and the assignment 1 answer key that you need to read in data and create variab

# create quarters of neighborhood median income
# note that if you create this variable as an ordered factor type, it will not work correctly with the
df$medincq <- ifelse(df$medinc>=min(df$medinc) & df$medinc<=28780, 1,
                     ifelse(df$medinc>28780 & df$medinc<=38965, 2,
                            ifelse(df$medinc>38965 & df$medinc<=48085,3,
                                   ifelse(df$medinc>48085 & df$medinc<=max(df$medinc),4,NA))))

#for these model examples, binge is the outcome, medincq is a categorical variable of quarters of media

# # load packages used in this assignment
# # only need to install packages once
#
# install.packages("lme4")
# install.packages("sjstats")
# install.packages("gee")
# install.packages("car")


# load packages necessary for this assignment
library(lme4)
library(sjstats)
library(gee)
library(car)

# # RANDOM EFFECTS MODEL
# refit <- glmer(binge ~ factor(medincq) + factor(catvar1) + (1 | cd), family="binomial", data=df, nAGQ
# summary(refit)
# icc(refit)
#
# refit2 <- glm(binge ~ factor(medincq) + factor(catvar1), family="binomial", data=df, nAGQ=0)
# # the model with the random intercept must go first in the anova function
# anova(refit, refit2, test="Chisq")
```

```
#
# # POPULATION AVERAGE MODEL
# # need to sort the data set by cd because the gee function in R assumes ID values that are not physic
# df <- df[order(df$cd),]
#
# gfit <- gee(binge ~ factor(medincq) + factor(catvar1), id=cd, family="binomial", corstr="exchangeable
#
# gfit
# summary(gfit)
# # summary produces huge correlation matrix but also provides needed standard errors
#
# # R commands that could be helpful to know more about for this assignment - you can look them up usin
#
# linearHypothesis()
# anova()
# deltaMethod()
#
# # Note that when using linearHypothesis() and deltaMethod() with a GEE model, you have to supply the
```