Unit 5

R for Data Description

"It is difficult to understand why statisticians commonly limit their enquiries to Averages and do not revel in more comprehensive views. Their souls seem as dull to the charm of variety as that of the native of our flat English counties, whose retrospect of Switzerland was that, if its mountains could be thrown into its lakes, two nuisances could be got rid of at once."

- Sir Frances Galton (1822-1911)

Data description is done for at least two reasons – **data management (*e.g. - is the data clean and correct?*)** and **describing a sample (*who is actually represented, what do they "look like" with respect to the variables being studied?*).**

Data description for data management involves the production of descriptive statistics for ***every*** study variable: (1) to explore the distributions themselves (frequencies, shape, etc); and (2) to identify missing values, errors, and extremes.

Data description for reporting describes the analysis cohort itself. It also provides a sense of the extent to which the available sample is representative of the population of interest. It is used in intervention studies for the comparison of consenters and non-consenters and in retrospective studies for the comparison of cases and controls.

Table of Contents

Be sure to have downloaded from the course website:
relate100obs.Rdata
wws1000.Rdata
bplong.Rdata

Be sure to have installed (one-time) the following packages

Hmisc	**Rmisc**
summarytools	**psych**
Stargazer	**FSA**
tidyverse	**gmodels**

Learning Objectives

When you have finished this unit, you should be able to:

- List individual values of selected variables for selected individuals in a data set;
- Construct frequency, relative frequency, cumulative frequency, and cumulative relative frequency tables;
- Obtain standard summary statistics (eg – mode, median, mean, sd, se, percentiles) for selected variables for selected individuals in a data set;
- Construct cross-tabulations of discrete variable distributions, overall and for selections of individuals;
- Obtain correlations among multiple continuous variables, overall and for selections of individuals;

Suggestion – follow along!

These notes have been written so that you can follow along and practice the commands given.

Consider: (1) Downloading from the course website two data sets: (i) relate100obs.Rdata, (ii) bplong.Rdata, (iii) wws1000.Rdata.

(2) Printing out a hard copy of these notes to follow during an R Studio session; and

(3) Creating a saved R Markdown file as a "boiler" for future data description

Preliminaries – Prepare Data

green-comments **black-commands** **blue-results**

```
# Initialize session.
# Load R dataset relate100obs.Rdata
setwd("~/Desktop")
load(file="relate100obs.Rdata")

# Rename variables to be more meaningful
colnames(relate100obs)[colnames(relate100obs)=="R3483600"] <- "m_praise"
colnames(relate100obs)[colnames(relate100obs)=="R3485200"] <- "f_praise"
colnames(relate100obs)[colnames(relate100obs)=="R3828100"] <- "age"

# Unlike SAS, Stata etc., R has only one missing value designation and that is NA
# Convert source missing values to R missing value NA
relate100obs$m_praise[relate100obs$m_praise==-1] <- NA
relate100obs$m_praise[relate100obs$m_praise==-2] <- NA
relate100obs$m_praise[relate100obs$m_praise==-4] <- NA
relate100obs$m_praise[relate100obs$m_praise==-5] <- NA

relate100obs$f_praise[relate100obs$f_praise==-1] <- NA
relate100obs$f_praise[relate100obs$f_praise==-2] <- NA
relate100obs$f_praise[relate100obs$f_praise==-4] <- NA
relate100obs$f_praise[relate100obs$f_praise==-5] <- NA

relate100obs$age[relate100obs$age==-1] <- NA
relate100obs$age[relate100obs$age==-2] <- NA
relate100obs$age[relate100obs$age==-4] <- NA
relate100obs$age[relate100obs$age==-5] <- NA

# Label variables using package Hmisc and command label( )
library(Hmisc)
label(relate100obs$m_praise) <- "m_praise: Mother praises R for doing well"
label(relate100obs$f_praise) <- "f_praise: Father praises R for doing well"
label(relate100obs$age) <- "age: Age of R (years)"


# Label variable values using factor( ) with levels=c( ) and labels=c( )
relate100obs$m_praise <- factor(relate100obs$m_praise,levels = c(0,1,2,3,4,".",".s"),
                                labels = c("never", "rarely", "sometimes","usually","always",".",".s"))
relate100obs$f_praise <- factor(relate100obs$f_praise,levels = c(0,1,2,3,4,".",".s"),
                                labels = c("never", "rarely", "sometimes","usually","always",".",".s"))

# Command subset.  Keep only the variables of interest.
# Command save.  Save under new name.
relate100obs = subset(relate100obs, select = c("m_praise","f_praise","age") )
save(relate100obs,file="relatenew100.Rdata")
```

1. Data Set Description and Listing Observations

What do we mean by dataset description?

It can also mean producing distributions of study variables. But here, the focus is on the dataset structure, including such things as:

- Dataset name (and any notes)
- Number and names of variables
- Variable types and, importantly, storage
- Number of observations
- Variable value labels (if provided)

1.1. Illustration

green-comments **black-commands** **blue-results**

```
load(file="bplong.Rdata")

#  Obtain info re variable names, storage types, a look at some observations using str( )
 str(bplong)

'data.frame':240 obs. of  5 variables:
 $ patient: int  1 1 2 2 3 3 4 4 5 5 ...
 $ sex    : Factor w/ 2 levels "Male","Female": 1 1 1 1 1 1 1 1 1 1 ...
 $ agegrp : Factor w/ 3 levels "30-45","46-59",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ when   : Factor w/ 2 levels "Before","After": 1 2 1 2 1 2 1 2 1 2 ...
 $ bp     : int  143 153 163 170 153 168 153 142 146 141 ...
 - attr(*, "datalabel")= chr "fictional blood-pressure data"
 - attr(*, "time.stamp")= chr " 5 Oct 2018 13:46"
 - attr(*, "formats")= chr  "%8.0g" "%9.0g" "%9.0g" "%8.0g" ...
 - attr(*, "types")= int  65529 65530 65530 65530 65529
 - attr(*, "val.labels")= Named chr  "" "sex" "agegrp" "when" ...
  ..- attr(*, "names")= chr  "" "sex" "agegrp" "when" ...
 - attr(*, "var.labels")= chr  "Patient ID" "Sex" "Age Group" "Status" ...
 - attr(*, "version")= int 118
 - attr(*, "label.table")=List of 3
  ..$ when  : Named int  1 2
  .. ..- attr(*, "names")= chr  "Before" "After"
  ..$ sex   : Named int  0 1
  .. ..- attr(*, "names")= chr  "Male" "Female"
  ..$ agegrp: Named int  1 2 3
  .. ..- attr(*, "names")= chr  "30-45" "46-59" "60+"
 - attr(*, "expansion.fields")= list()
 - attr(*, "byteorder")= chr "LSF"


#  Obtain variable names using command names( )
names(bplong)

[1] "patient" "sex"     "agegrp"  "when"    "bp"


#  Obtain number of observations (sample size) using command nrow( )
nrow(bplong)

[1] 240
```

```
#  Obtain discrete variable value labels using command levels( )
levels(bplong$agegrp)
[1] "30-45" "46-59" "60+"

levels(bplong$sex)
[1] "Male"   "Female"

levels(bplong$when)
[1] "Before" "After"


#  Define new variable value labels using factor( ), labels=c( ) and levels=c( )
bplong$sex <- factor(bplong$sex,
                     labels = c(0,1),
                     levels = c("Male", "Female"))

bplong$agegrp <- factor(bplong$agegrp,
                      labels = c(1,2,3),
                      levels = c("30-45", "46-59","60+"))

bplong$when <- factor(bplong$when,
                        labels = c(1,2),
                        levels = c("Before", "After"))

#  Reorder the variables (columns) to be alphabetic using command sort( )
bplong<-bplong[sort(names(bplong))]


#  Obtain quick summary of variable distributions using command summary( )
summary(bplong)

   agegrp         bp           patient           sex          when
 30-45:80   Min.   :125.0   Min.   :  1.00   Male  :120   Before:120
 46-59:80   1st Qu.:144.0   1st Qu.: 30.75   Female:120   After :120
 60+  :80   Median :152.0   Median : 60.50
            Mean   :153.9   Mean   : 60.50
            3rd Qu.:162.2   3rd Qu.: 90.25
            Max.   :185.0   Max.   :120.00
```

1.2 Data Set Description Commands

{ Package } command	Example
Check structure of dataframe {base} str(dataframename)	str(bplong)
Obtain dimensions of dataframe {base} nrow(dataframename) # observations, n ncol(dataframename) # variables, p dim(dataframename) # rows and columns	nrow(bplong)
Obtain variable names {base} names(dataframename)	Names(bplong)
Obtain variable value labels of a variable {base} levels(dataframename$variablename)	levels(bplong$sex)
Reorder the variables (columns) to be alphabetical {base} dataframename) <- dataframename[sort(names(dataframename))]	bplong<-bplong[sort(names(bplong))]
Obtain summary statistics of every variable {base} summary(dataframename)	summary(Isoproterenol)
Nicer looking summary stats of every variable {stargazer} stargazer(dataframename,type="text", median=TRUE)	library(stargazer) stargazer(Isoproterenol,type="text", median=TRUE) What could go wrong: If the input is not a dataframe (for example, it is a tibble), you will only get a header. Solution: dataframename <- as.data.frame(dataframename)
Show n and variable type for every variable {psych} print(describeFast(dataframename),short=FALSE)	library(psych) print(describeFast(Isoproterenol,short=FALSE)
Consider this really nice dataset summary. {summarytools} print(dfSummary(dataframename))	library(summarytools) print(dfSummary(Isoproterenol))

1.3 Listing Individual Observations

{ Package } command	Example
List first 6 observations – EVERY variable {base} head(dataframename)	head(Isoproterenol)
List last 6 observations – EVERY variable {base} head(dataframename)	tail(Isoproterenol)
List selection of observations (rows) EVERY variable (column) {base} dataframename[ROW1:ROWLAST,]	ivf[1:6,] What could go wrong (1) must use square brackets now (2) must have comma and blank where columns would be
List selection of variables (columns) EVERY observation (row) {base} dataframename[, COLUMN1:COLUMNLAST]	ivf[,1:6] What could go wrong (1) must use square brackets now (2) must have blank and comma where rows would be
List first # observations of specific selection of variables {base} {base} dataframename[, c("var1", "var2")]	ivf[1:6,c("matage", "gestwks", "sex")]

2. One Variable Descriptions

2.1 Illustration

green-comments **black-commands** **blue-results**

```
#  If you have not already done so, load the data into your R session
load(file="bplong.Rdata")

#  Command factor. Recode labelled values to numeric values.
bplong$sex <- factor(bplong$sex,
                     labels = c(0,1),
                     levels = c("Male", "Female"))

bplong$agegrp <- factor(bplong$agegrp,
                        labels = c(1,2,3),
                        levels = c("30-45", "46-59","60+"))

bplong$when <- factor(bplong$when,
                          labels = c(1,2),
                          levels = c("Before", "After"))

#  (Handy for data management, actually):  Reorder columns by alphabetical order.
bplong<-bplong[c("agegrp", "bp","patient","sex","when")]

#  Obtain quick overview of distributions on EVERY variable using command summary( )
summary(bplong)

   agegrp          bp            patient           sex          when
 30-45:80   Min.   :125.0   Min.   :  1.00   Male  :120   Before:120
 46-59:80   1st Qu.:144.0   1st Qu.: 30.75   Female:120   After :120
 60+  :80   Median :152.0   Median : 60.50
            Mean   :153.9   Mean   : 60.50
            3rd Qu.:162.2   3rd Qu.: 90.25
            Max.   :185.0   Max.   :120.00

#  Single Variable Description - DISCRETE variable
#  Using package summarytools and command freq( )
library(summarytools)
freq(bplong$agegrp)

Frequencies
bplong$agegrp
Type: Factor (unordered)

             Freq   % Valid   % Valid Cum.   % Total   % Total Cum.
----------- ------ --------- -------------- --------- --------------
      30-45     80     33.33          33.33     33.33          33.33
      46-59     80     33.33          66.67     33.33          66.67
        60+     80     33.33         100.00     33.33         100.00
       <NA>      0                                0.00         100.00
      Total    240    100.00         100.00    100.00         100.00
```

```
#  Single Variable Description - CONTINUOUS variable
#  Using package summarytools and command descr( )
#  Use option transpose=T to obtain horizontal display
library(summarytools)
descr(bplong$bp,transpose=T)

Descriptive Statistics
bplong$bp
N: 240

            Mean   Std.Dev      Min       Q1   Median       Q3      Max      MAD      IQR      CV
-------- -------- --------- -------- -------- -------- -------- -------- ------- -------- ------
      bp   153.90     13.08   125.00   144.00   152.00   162.50   185.00   13.34    18.25   0.09

Table: Table continues below


           Skewness   SE.Skewness   Kurtosis   N.Valid   Pct.Valid
-------- ---------- ------------- ---------- --------- -----------
      bp       0.30          0.16      -0.45    240.00      100.00
```

2.2 One Discrete Variable

{ Package } command	Example
So you have it Quick descriptives on every variable (2 ways) {stargazer} stargazer(dataframename, type="text", median=TRUE) stargazer(dataframe="text",title="YOUR TITLE", out="TABLENAME.txt")	library(stargazer) stargazer(framingham, type="text", median=TRUE) stargazer(lung_demo,type="text",title="Table 1: Descriptives of Lung Study",out="table1.txt")
Frequency/Relative Frequency Table {summarytools} freq(dataframename$discretevariable)	library(summarytools) freq(wws1000$race)
Frequency/Relative Frequency Table Output in order of frequencies (largest first) {summarytools} freq(dataframename$discretevariable),order="freq")	library(summarytools) freq(wws1000$race, order=freq)

Brute force frequency/relative frequency table
 	
ntot <- length(discretevar) # sample size
var_freq <- table(discretevar) # frequencies
var_relfreq <- var_freq/ntot # rel. freqs
var_cum <- cumsum(var_freq) # cum. freqs
var_cumrel <- cumsum(var_relfreq) # cum rel freq
 	
# Create table using cbind() 	
tablename <- cbind(var_freq, var_relfreq, var_cum,
var_cumrel)
 	
# Label columns 	
colnames(tablename) <- c("Freq", "Rel Freq", "Cum
Freq", "Cum Rel Freq")
 	
# Display table 	
tablename
	ntot <- length(los)
	los_freq <- table(los)
	los_relfreq <- los_freq/ntot
	los_cum <- cumsum(los_freq)
	los_cumrel <- cumsum(los_relfreq)
	
	# Create q1table
	q1table <- cbind(los_freq, los_relfreq, los_cum,
	los_cumrel)
	
	# Label columns
	colnames(q1table) <- c("Freq", "Rel Freq", "Cum
	Freq", "Cum Rel Freq")
	
	# Display table
	q1table

2.3 One Continuous Variable

{ Package } command	Example
So you have it Quick descriptives on every variable (2 ways) {stargazer} stargazer(dataframename, type="text", median=TRUE) stargazer(dataframe="text",title="YOUR TITLE", out="TABLENAME.txt")	library(stargazer) stargazer(framingham, type="text", median=TRUE) stargazer(lung_demo,type="text",title="Table 1: Descriptives of Lung Study",out="table1.txt")
Detailed descriptives on SELECTED continuous variables using package stargazer and command stargazer( ) {stargazer} stargazer(dataframe[c("var",”var”)], type="text", summary.stat=c("n", "mean", "sd", "min", "p25", "median", "p75", "max")) Note – You pick and choose what statistics you want. I just listed some for you.	library(stargazer) stargazer(toyrdata[c("growth")],type="text",summary.stat=c("n", "mean", "sd", "min", "p25", "median", "p75", "max")) What could go wrong: You get header only. This may mean you have to declare your data to be a dataframe. Solution: Issue the following command first: toyrdata <- as.data.frame(toyrdata)
Detailed descriptives on SINGLE continuous variable using package summarytools and command descr( ) {summarytools} descr(dataframe$varname, stats = c("n.valid","mean", "sd", "min","q1", "med", "q3", "max","CV"), transpose = TRUE)	library(summarytools) descr(relate100obs$age, stats = c("n.valid","mean", "sd", "min","q1", "med", "q3", "max","CV"), transpose = TRUE)
Detailed descriptives on SINGLE continuous variable using package FSA and command Summarize( ) Summarize(varname ~1,data=dataframe,digits=2, na.rm=TRUE)	library(FSA) Summarize(yscore ~ 1,data=ptdata,digits=2, na.rm=TRUE)

3. Multiple Variable Descriptions

3.1 Illustration

green-comments **black-commands** **blue-results**

```
load(file="wws1000.Rdata")

# Convenient for data management.  Sort the variables in alphabetic order using sort( )
wws1000<-wws1000[sort(names(wws1000))]

# Quick summary statistics on every variable using command summary( ).
summary(wws1000)

      age             ccity          collgrad        currexp         everworked         fwt
 Min.   :21.00   Min.   :0.000   Min.   :0.000   Min.   : 0.000   Min.   :0.000   Min.   :0.000
 1st Qu.:34.00   1st Qu.:0.000   1st Qu.:0.000   1st Qu.: 1.000   1st Qu.:1.000   1st Qu.:2.000
 Median :37.00   Median :0.000   Median :0.000   Median : 3.000   Median :1.000   Median :4.000
 Mean   :36.28   Mean   :0.297   Mean   :0.241   Mean   : 5.115   Mean   :0.972   Mean   :4.356
 3rd Qu.:40.00   3rd Qu.:1.000   3rd Qu.:0.000   3rd Qu.: 8.000   3rd Qu.:1.000   3rd Qu.:7.000
 Max.   :83.00   Max.   :1.000   Max.   :1.000   Max.   :26.000   Max.   :1.000   Max.   :9.000
     grade           grade4          hours          idcode        industry         kidage1         kidage2
 Min.   : 4.00   Min.   :1.000   Min.   : 1.0   Min.   :   1   Min.   : 1.000   Min.   : 0.00   Min.   : 0.000
 1st Qu.:12.00   1st Qu.:2.000   1st Qu.:35.0   1st Qu.:1258   1st Qu.: 6.000   1st Qu.: 8.00   1st Qu.: 5.000
 Median :12.00   Median :2.000   Median :40.0   Median :2606   Median : 8.000   Median :10.00   Median : 7.000
 Mean   :13.12   Mean   :2.533   Mean   :37.4   Mean   :2591   Mean   : 8.089   Mean   :10.35   Mean   : 7.048
 3rd Qu.:15.00   3rd Qu.:3.000   3rd Qu.:40.0   3rd Qu.:3931   3rd Qu.:11.000   3rd Qu.:13.00   3rd Qu.: 9.000
 Max.   :18.00   Max.   :4.000   Max.   :80.0   Max.   :5159   Max.   :12.000   Max.   :21.00   Max.   :14.000
 NA's   :2       NA's   :2       NA's   :2                     NA's   :9        NA's   :235     NA's   :479
    kidage3        married       marriedyrs         metro          networth        nevermarried      numkids
 Min.   :0.00   Min.   :0.00   Min.   : 0.000   Min.   :0.000   Min.   :-7000.0   Min.   :0.000   Min.   :0.00
 1st Qu.:1.00   1st Qu.:0.00   1st Qu.: 0.000   1st Qu.:0.000   1st Qu.:-2774.1   1st Qu.:0.000   1st Qu.:1.00
 Median :3.00   Median :1.00   Median : 2.000   Median :1.000   Median : -651.4   Median :0.000   Median :2.00
 Mean   :3.43   Mean   :0.64   Mean   : 3.558   Mean   :0.704   Mean   :  817.9   Mean   :0.104   Mean   :1.53
 3rd Qu.:5.00   3rd Qu.:1.00   3rd Qu.: 7.000   3rd Qu.:1.000   3rd Qu.: 2585.3   3rd Qu.:0.000   3rd Qu.:2.00
 Max.   :7.00   Max.   :1.00   Max.   :11.000   Max.   :1.000   Max.   :33198.1   Max.   :1.000   Max.   :3.00
 NA's   :756
   occupation       prevexp           race           south          unempins          union
 Min.   : 1.000   Min.   : 0.000   Min.   :1.000   Min.   :0.000   Min.   :  0.00   Min.   :0.0000
 1st Qu.: 2.000   1st Qu.: 3.000   1st Qu.:1.000   1st Qu.:0.000   1st Qu.:  0.00   1st Qu.:0.0000
 Median : 3.000   Median : 5.000   Median :1.000   Median :0.000   Median :  0.00   Median :0.0000
 Mean   : 4.593   Mean   : 6.031   Mean   :1.275   Mean   :0.422   Mean   : 30.12   Mean   :0.2363
 3rd Qu.: 6.000   3rd Qu.: 9.000   3rd Qu.:2.000   3rd Qu.:1.000   3rd Qu.:  0.00   3rd Qu.:0.0000
 Max.   :13.000   Max.   :25.000   Max.   :3.000   Max.   :1.000   Max.   :298.00   Max.   :1.0000
 NA's   :6        NA's   :6                                                         NA's   :158
   uniondues          wage             wage2          yrschool
 Min.   : 0.000   Min.   :     0.0   Min.   : 0.000   Min.   : 8.00
 1st Qu.: 0.000   1st Qu.:     4.2   1st Qu.: 4.228   1st Qu.:12.00
 Median : 0.000   Median :     6.4   Median : 6.345   Median :12.00
 Mean   : 5.473   Mean   :   387.8   Mean   : 7.821   Mean   :13.16
 3rd Qu.:10.000   3rd Qu.:     9.6   3rd Qu.: 9.633   3rd Qu.:15.00
 Max.   :29.000   Max.   :380000.0   Max.   :40.200   Max.   :18.00
 NA's   :3                                          NA's   :2
```

```
# Multiple Variables Description:  TWO DISCRETE VARIABLES
# Crosstab using package summarytools and command ctable( )
library(summarytools)
ctable(wws1000$race,wws1000$numkids,prop="r", totals=TRUE)

Cross-Tabulation / Row Proportions
Variables: race * numkids
Data Frame: wws1000

------- --------- ------------- ------------- ------------- ------------- ----------------
          numkids             0             1             2             3            Total
   race
      1           183 (24.83%)  169 (22.93%)  204 (27.68%)  181 (24.56%)   737 (100.00%)
      2            49 (19.52%)   70 (27.89%)   71 (28.29%)   61 (24.30%)   251 (100.00%)
      3             3 (25.00%)    5 (41.67%)    2 (16.67%)    2 (16.67%)    12 (100.00%)
  Total           235 (23.50%)  244 (24.40%)  277 (27.70%)  244 (24.40%)  1000 (100.00%)
------- --------- ------------- ------------- ------------- ------------- ----------------



# Multiple Variables Description:  MULTIPLE CONTINUOUS VARIABLES
# Descriptives using package stargazer and option summar.stat=c( )
library(stargazer)
stargazer(wws1000[c("age","uniondues","wage")], type="text", summary.stat=c("n","mean","sd","min","max"))

==================================================
Statistic   N    Mean    St. Dev.   Min      Max
--------------------------------------------------
age       1,000 36.276    5.625     21       83
uniondues  997   5.473    8.953    0.000   29.000
wage      1,000 387.816 12,016.410 0.000 380,000.000
--------------------------------------------------


# Multiple Variable Description:  ONE CONTINUOUS by ONE DISCRETE
# Package dplyr and function filter( )to select variables and subset of interest (e.g. race==1)
library(tidyverse)
library(stargazer)

#------- Specify variables of interest and name the vector "myvars" --------*
myvars <- c('age', 'uniondues', 'wage')

#------- Command stargazer and filter. Obtain summary statistics for subset with race==1 ONLY
stargazer(filter(wws1000[, myvars],wws1000$race==1), type = "text",
  summary.stat = c("n","min","mean", "max", "sd"))

========================================
Statistic  N   Min   Mean   Max   St. Dev.
----------------------------------------
age       737  21   36.578   83    5.594
uniondues 736 0.000 5.355  29.000  8.959
wage      737 1.032 8.184  40.198  6.101
----------------------------------------
```

```
# Multiple Variable Descriptions – Pairwise Covariances and Correlations
#------- Specify variables of interest and name the vector “myvars” --------*
myvars <- c('age', 'uniondues', 'wage')

#  Covariances using command cov( ) with option use (to specify how to compute with missing data)
# and method
cov(wws1000[, myvars], y = NULL, use =  "pairwise.complete.obs",
    method = c("pearson"))

                  age     uniondues          wage
age         31.6354595    -0.7710138 -1.059739e+02
uniondues   -0.7710138    80.1591824 -2.082857e+03
wage      -105.9739220 -2082.8566178  1.443941e+08

#  Correlations using command cor( ) with option use (to specify how to compute with missing
data) # and method

cor(wws1000[, myvars], y = NULL, use =  "pairwise.complete.obs",
    method = c("pearson"))

                  age   uniondues         wage
age        1.000000000 -0.01529914 -0.001567968
uniondues -0.015299144  1.00000000 -0.019331057
wage      -0.001567968 -0.01933106  1.000000000
```

3.2 Two Discrete Variables

{ Package } command	Example
So you have it Quick descriptives on every variable (2 ways) {stargazer} stargazer(dataframename, type="text", median=TRUE) stargazer(dataframe="text",title="YOUR TITLE", out="TABLENAME.txt")	library(stargazer) stargazer(framingham, type="text", median=TRUE) stargazer(lung_demo,type="text",title="Table 1: Descriptives of Lung Study",out="table1.txt")
Two Way Crosstab - Method I {summarytools} Show counts only with(dataframe, ctable(rowvar,colvar,prop="n", totals=TRUE) Row Percents with(dataframe, ctable(rowvar,colvar,prop="r", totals=TRUE) Column Percents with(dataframe, ctable(rowvar,colvar,prop="c", totals=TRUE) 2x2 table COHORT: Row %, RR, and OR with(bplong, ctable(sex,when,prop="r",OR=TRUE,RR=TRUE)) 2x2 table CASE-CONTROL: Col % and OR with(bplong, ctable(sex,when,prop="c",OR=TRUE,RR=FALSE))	library(summarytools) with(wws1000,ctable(race,numkids,prop="r", totals=TRUE)
Two Way Crosstab – Method II {gmodels} CrossTable(dataframe$rvar,dataframe$cvar,digits=2, prop.r=TRUE,prop.c=FALSE,prop.t=FALSE, prop.chisq=FALSE, dnn=c("RowTitle","ColumnTitle"))	library(gmodels) CrossTable(wws1000$race,wws1000$numkids,digits=2, prop.r=TRUE,prop.c=FALSE,prop.t=FALSE, prop.chisq=FALSE, dnn=c("Race","Number of Children"))

3.3 One Discrete Variable and Multiple Continuous Variables

{ Package } command	Example
So you have it Quick descriptives on every variable (2 ways) {stargazer} stargazer(dataframename, type="text", median=TRUE) stargazer(dataframe="text",title="YOUR TITLE", out="TABLENAME.txt")	library(stargazer) stargazer(framingham, type="text", median=TRUE) stargazer(lung_demo,type="text",title="Table 1: Descriptives of Lung Study",out="table1.txt")
Using package summarytools {summarytools} with(dataframe,stby(data=continuousvar, INDICES=discretevar, FUN=descr,stats=c("statistic", "statistic")))	library(summarytools) with(wws1000, stby(data = age, INDICES =race, FUN = descr, stats = c("mean", "sd", "min", "med", "max")))
Using package FSA {FSA} Summarize(continuousvar~discretevar, data=dataframe,na.rm=TRUE)	library(FSA) Summarize(age~race,data=wws1000,na.rm=TRUE)
Using package Rmisc Note: ci half width of 95% CI {Rmisc} summarySE(data=dataframe,measurevar="continuousvar", groupvars=c("discretevar"),na.rm=TRUE)	library(Rmisc) summarySE(data=wws1000,measurevar="age", groupvars=c("race"),na.rm=TRUE)

3.4 Multiple Continuous Variables

{ Package } command	Example
So you have it Quick descriptives on every variable (2 ways) {stargazer} stargazer(dataframename, type="text", median=TRUE) stargazer(dataframe="text",title="YOUR TITLE", out="TABLENAME.txt")	library(stargazer) stargazer(framingham, type="text", median=TRUE) stargazer(lung_demo,type="text",title="Table 1: Descriptives of Lung Study",out="table1.txt")
Multiple Continuous Variables – Method I (2 ways) **{stargazer}** stargazer(dataframe[c("var1","var2","var3")], type="text", summary.stat=c("n","mean","sd","min","max")) myvars <- c('var1','var2', 'var3') stargazer(dataframe[myvars], type="text", summary.stat=c("n","mean","sd","min","max"))	library(stargazer) stargazer(wws1000[c("age","uniondues","wage")], type="text", summary.stat=c("n","mean","sd","min","max")) myvars <- c('age','uniondues', 'wage') stargazer(wws1000[myvars], type="text", summary.stat=c("n","mean","sd","min","max")) What could go wrong: You used double quotes in myvars, instead of single Solution: myvars <- c('var1','var2', 'var3')
Pairwise Covariances {base} myvars <- c('var1','var2', 'var3') cor(dataframe[, myvars], y = NULL, use = "pairwise.complete.obs", method = c("pearson"))	myvars <- c('age', 'uniondues', 'wage') cor(wws1000[, myvars], y = NULL, use = "pairwise.complete.obs", method = c("pearson"))