R CHEATSHEET • MTH107

Class R FAQ

by Derek H. Ogle, revised Dec-17

Data

Get Data

ENTER RAW DATA:

1. Enter data in Excel (variables in columns, individuals in rows, first row has variable names, no spaces or special characters).
2. Save as "Comma Separated Values (*.CSV)" file in your local directory/folder.

DATA PROVIDED BY PROFESSOR:

1. Goto Data Specific to MTH107 on Resources page.
2. Right-click on "data" link and save to your local directory/folder.

Load CSV

1. Start script and save it in the same folder that contains the CSV file.
2. Select Session, Set Working Directory, To Source File Location menus.
3. Copy resulting setwd() code to script.
4. Use read.csv() to load data into dfobj.

```
dfobj <- read.csv("filename.csv")
```

5. Observe structure of data.frame.

```
str(dfobj)
```

Filter Individuals

Individuals that meet a certain condition (or conditions) are filtered from the dfobj data.frame with filterD().

```
newdf <- filterD(dfobj,cond)
```

where cond may be as follows with var replaced with a variable name and value replaced with a number or category level (*if* value *is text then it must be in quotes*):

```
var == value              # equal to
var != value              # not equal to
var > value               # greater than
var >= value              # greater than or equal
var < value               # less than
var <= value              # less than or equal
var %in% c("val","val","val")  # in the list
cond | cond               # either condition met
cond, cond                # both conditions met
```

Individual in row rownum is selected with:

```
dfobj[rownum,]
```

Individual in row rownum is excluded with:

```
dfobj[-rownum,]
```

Exploratory Data Analysis

Univariate

QUANTITATIVE – Summary statistics (mean, median, SD, IQR, etc.) and a histogram for the qvar variable.

```
hist(~qvar,data=dfobj,xlab="qvar label")
Summarize(~qvar,data=dfobj,digits=#)
```

QUANTITATIVE BY GROUP – Summary statistics and histograms for the qvar variable separated by groups in the gvar variable.

```
hist(qvar~gvar,data=dfobj,xlab="qvar label")
Summarize(qvar~gvar,data=dfobj,digits=#)
```

CATEGORICAL – Frequency and percentage tables and bar chart for the cvar variable.

```
( freq1 <- xtabs(~cvar,data=dfobj) )
percTable(freq1,digits=#)
barplot(freq1,xlab="cvar label",
              ylab="Frequency")
```

Bivariate

QUANTITATIVE – Correlation (r) and scatterplot for the qvarY and qvarX variables.

```
plot(qvarY~qvarX,data=dfobj,
     ylab="yvar label",xlab="xvar label")
corr(~qvarY+qvarX,data=dfobj)
```

CATEGORICAL – Frequency and percentage tables for the cvarRow and cvarCol variables.

```
( freq2 <- xtabs(~cvarRow+cvarCol,
                 data=dfobj) )
percTable(freq2)              # total/table %
percTable(freq2,margin=1)     # row %
percTable(freq2,margin=2)     # column %
```

Models

Normal Distributions

```
distrib(val,mean=meanval,sd=sdval,
        type="q",lower.tail=FALSE)
```

where

- val is a value of the quantitative variable or area (i.e., percentage as a proportion).
- meanval is population mean (μ)
- sdval is standard deviation (σ) or error (SE)
- type="q" is included for reverse calculations
- lower.tail=FALSE is included for "right-of" calculations

For SE use (where nval=sample size):

```
sd=sdval/sqrt(nval)
```

Linear Regression

The best-fit line between the rspvar response and expvar explanatory variables.

```
( bfl <- lm(rspvar~expvar,data=dfobj) )
```

A visual of the best-fit line.

```
fitPlot(bfl,ylab="rspvar lbl",xlab="expvar lbl")
```

The r^2 value.

```
rSquared(bfl)
```

Predict a value of rspvar given a specific expval value of the expvar.

```
predict(bfl,data.frame(expvar=expval))
```

Hypothesis Testing

Quantitative

ONE SAMPLE:

```
z.test(dfobj$qvar,mu=mu0,alt=HA,
       conf.level=confval,sd=sdval)
t.test(dfobj$qvar,mu=mu0,alt=HA,
       conf.level=confval)
```

TWO SAMPLE:

```
levenesTest(qvar~gvar,data=dfobj)
t.test(qvar~gvar,data=dfobj,alt=HA ,
       conf.level=confval,var.equal=TRUE)
```

- qvar is the quant. response variable in dfobj
- mu0 is the population mean in H_0
- HA is "two.sided" for a not equals, "less" for a less than, or "greater" for a greater than H_A
- confval is the confidence level (e.g., 0.95)
- sdval is the popn. standard deviation (σ)
- gvar is a categorical variable in dfobj that identifies the groups
- var.equal=TRUE if the population variances are thought to be equal

Categorical

ONE SAMPLE:
Goodness-of-fit test for observed frequencies in the freq1 table and expected values (or proportions) in exp.p.

```
( gof <- chisq.test(freq1,p=exp.p,
        rescale.p=TRUE,correct=FALSE) )
```

TWO SAMPLE:
Chi-square for freq2 two-way observed frequency table (with response variable in columns and groups in rows).

```
( chi <- chisq.test(freq2,correct=FALSE) )
```

With the following follow-up analyses:

- Extract the expected values.

```
gof$expected   or   chi$expected
```

- Extract the residuals.

```
gof$residuals   or   chi$expected
```

- Confidence intervals *for goodness-of-fit*.

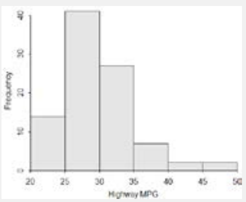
```
gofCI(gof,digits=3)
```

Data

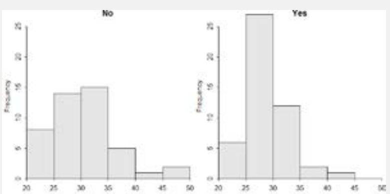
```
> library(NCStats)
> setwd("C:/aaaWork/Web/GitHub/NCMTH107")
> dfobj <- read.csv("93cars.csv")
> str(dfobj)
'data.frame':    93 obs. of  26 variables:
  $ Type     : Factor w/ 6 levels "Compact","Large",..: 4 ...
  $ HMPG     : int  31 25 26 26 30 31 28 25 27 25 ...
  $ Manual   : Factor w/ 2 levels "No","Yes": 2 2 2 2 2 1 ...
  $ Weight   : int  2705 3560 3375 3405 3640 2880 3470 ...
  $ Domestic: Factor w/ 2 levels "No","Yes": 1 1 1 1 1 2 ...

> justSporty <- filterD(dfobj,Type=="Sporty")
> justHMPGgt30 <- filterD(dfobj,HMPG>30)
> noDomestics <- filterD(dfobj,Domestic!="Yes")
> Sprty_Small <- filterD(dfobj,Type %in% c("Sporty","Small"))
```

Univariate EDA

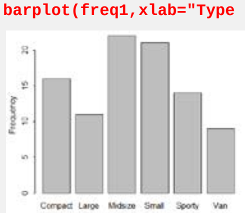
```
> Summarize(~HMPG,data=dfobj,digits=1)
      n   mean      sd     min      Q1 median      Q3     max
   93.0   29.1     5.3    20.0    26.0   28.0    31.0    50.0
> hist(~HMPG,data=dfobj,xlab="Highway MPG")
```




```
> Summarize(HMPG~Domestic,data=dfobj,digits=1)
  Domestic  n mean  sd min Q1 median Q3 max
 1       No 45 30.1 6.2  21 25     30 33  50
 2      Yes 48 28.1 4.2  20 26     28 30  41
> hist(HMPG~Domestic,data=dfobj,xlab="Highway MPG")
```

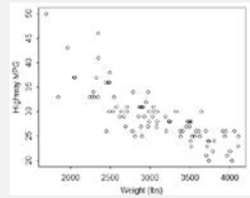



```
> ( freq1 <- xtabs(~Type,data=dfobj) )
 Type
 Compact   Large Midsize   Small  Sporty     Van
      16      11      22      21      14       9
> percTable(freq1,digits=1)
 Type
 Compact   Large Midsize   Small  Sporty     Van     Sum
    17.2    11.8    23.7    22.6    15.1     9.7   100.1
> barplot(freq1,xlab="Type of Car",ylab="Frequency")
```



Bivariate EDA

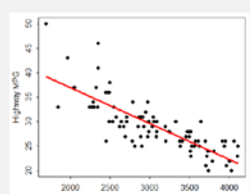
```
> plot(HMPG~Weight,data=dfobj,ylab="Highway MPG")
       xlab="Weight (lbs)")
```



```
> corr(HMPG~Weight,data=dfobj)
  [1] -0.8106581

> ( freq2 <- xtabs(~Domestic+Manual,data=dfobj) )
          Manual
  Domestic No Yes
       No   6  39
       Yes 26  22
> percTable(freq2,digits=1)
          Manual
  Domestic    No   Yes   Sum
       No    6.5  41.9  48.4
       Yes  28.0  23.7  51.7
       Sum  34.5  65.6 100.1
> percTable(freq2,margin=1,digits=1)
          Manual
  Domestic    No   Yes   Sum
       No   13.3  86.7 100.0
       Yes  54.2  45.8 100.0
> percTable(freq2,margin=2,digits=1)
          Manual
  Domestic    No   Yes
       No   18.8  63.9
       Yes  81.2  36.1
       Sum 100.0 100.0
```

Linear Regression

```
> ( bfl <- lm(HMPG~Weight,data=dfobj) )
  Coefficients:
  (Intercept)       Weight
    51.601365    -0.007327

> fitPlot(bfl,xlab="Weight (lbs)",ylab="Highway MPG")
```



```
> rSquared(bfl)
  [1] 0.6571665
> predict(bfl,data.frame(Weight=3000))
  29.62019
```

Hypothesis Tests

```
> z.test(dfobj$HMPG,mu=26,alt="greater",conf.level=0.95,sd=6)
  z= 4.9601, n= 93, Std. Dev= 6.000, Std. Dev of the sample
  mean = 0.622, p-value = 3.523e-07
  alternative hypothesis: true mean is greater than 26
  95 percent confidence interval:
   28.06264       Inf
  sample estimates:
  mean of dfobj$HMPG
            29.08602

> t.test(dfobj$HMPG,mu=26,alt="two.sided",conf.level=0.99)
  t = 5.5818, df = 92, p-value = 2.387e-07
  alternative hypothesis: true mean is not equal to 26
  99 percent confidence interval:
   27.63178 30.54026
  sample estimates:
  mean of x
   29.08602

> levenesTest(HMPG~Domestic,data=dfobj)
        Df F value  Pr(>F)
  group  1  5.3595 0.02286 *
        91

> t.test(HMPG~Manual,data=dfobj,alt="less",conf.level=0.99,
      var.equal=TRUE)
  t = -4.2183, df = 91, p-value = 2.904e-05
  alt. hypothesis: true difference in means is less than 0
  99 percent confidence interval:
        -Inf -1.980103
  sample estimates:
   mean in group No mean in group Yes
           26.12500          30.63934

> exp <- c(1,1,1,1,1,1)/6
> (gof<-chisq.test(freq1,p=exp,rescale.p=TRUE,correct=FALSE))
  X-squared = 8.871, df = 5, p-value = 0.1143

> gof$expected
  Compact   Large Midsize   Small  Sporty     Van
     15.5    15.5    15.5    15.5    15.5    15.5

> gof$residuals
  Compact    Large  Midsize    Small   Sporty      Van
  0.12700 -1.14300  1.65100  1.39700 -0.38100 -1.65100

> gofCI(gof,digits=3)
          p.obs p.LCI p.UCI p.exp
  Compact 0.172 0.109 0.261 0.167
  Large   0.118 0.067 0.199 0.167
  Midsize 0.237 0.162 0.332 0.167
  Small   0.226 0.153 0.321 0.167
  Sporty  0.151 0.092 0.237 0.167
  Van     0.097 0.052 0.174 0.167

> ( chi <- chisq.test(freq2,correct=FALSE) )
  Pearson's Chi-squared test with freq2
  X-squared = 17.1588, df = 1, p-value = 3.438e-05

> chi$expected
          Manual
  Domestic       No      Yes
       No  15.48387 29.51613
       Yes 16.51613 31.48387

> chi$residuals
          Manual
  Domestic         No        Yes
       No  -2.410160   1.745645
       Yes  2.333627  -1.690214
```