Writing Customized R Code: Summary Statistics by Level/Group within Binary & Numeric Data

Bahirah Adewunmi

Contents


```
library(PASWR)
```

```
## Warning: package 'PASWR' was built under R version 3.3.3
```

```
## Loading required package: e1071
```

```
## Warning: package 'e1071' was built under R version 3.3.3
```

```
## Loading required package: MASS
```

```
## Warning: package 'MASS' was built under R version 3.3.3
```

```
## Loading required package: lattice
```

```
library(lattice)
library(e1071)
str(titanic3)
```

```
## 'data.frame':    1309 obs. of  14 variables:
##  $ pclass   : Factor w/ 3 levels "1st","2nd","3rd": 1 1 1 1 1 1 1 1 1 1 ...
##  $ survived : int  1 1 0 0 0 1 1 0 1 0 ...
##  $ name     : Factor w/ 1307 levels "Abbing, Mr. Anthony",..: 22 24 25 26 27 31 46 47 51 55 ...
##  $ sex      : Factor w/ 2 levels "female","male": 1 2 1 2 1 2 1 2 1 2 ...
##  $ age      : num  29 0.917 2 30 25 ...
##  $ sibsp    : int  0 1 1 1 1 0 1 0 2 0 ...
##  $ parch    : int  0 2 2 2 2 0 0 0 0 0 ...
##  $ ticket   : Factor w/ 929 levels "110152","110413",..: 188 50 50 50 50 125 93 16 77 826 ...
##  $ fare     : num  211 152 152 152 152 ...
##  $ cabin    : Factor w/ 187 levels "","A10","A11",..: 45 81 81 81 81 151 147 17 63 1 ...
##  $ embarked : Factor w/ 4 levels "","Cherbourg",..: 4 4 4 4 4 4 4 4 4 2 ...
##  $ boat     : Factor w/ 28 levels "","1","10","11",..: 13 4 1 1 1 14 3 1 28 1 ...
##  $ body     : int  NA NA NA 135 NA NA NA NA NA 22 ...
##  $ home.dest: Factor w/ 369 levels "","?Havana, Cuba",..: 309 231 231 231 231 237 163 25 23 229 ...
```

The functions I wrote use datasets like the `titanic3` dataset, which is available in the `PASWR` library. The `str` command on `titanic3` data produces the following table, which summarizes information about passengers on the Titanic, displaying summaries separately depending on the passenger's fare class (1st, 2nd or 3rd).

-All functions remove missing values from the summary statistics.

Variable	Missing	1st	2nd	3rd	P-value
Survival rate	0	61.9%	43%	25.5%	0
% Female	0	44.6%	38.3%	30.5%	0
Age	263	39.16 (14.55)	29.51 (13.64)	24.82 (11.96)	0
# siblings/spouses aboard	0	0.44 (0.61)	0.39 (0.59)	0.57 (1.3)	0.0279
# children/parents aboard	0	0.37 (0.72)	0.37 (0.69)	0.4 (0.98)	0.779
Fare ($)	1	87.51 (80.45)	21.18 (13.61)	13.3 (11.49)	0

lass.

Below is a function that summarizes numeric data by reporting the number of missing values, the means of each of the data's groups/levels, and the standard deviation of the groups, and the p-value, which tests whether the the means under each of the data's groups are statistically the same,

```
generateNumericSummary <- function(x, groups){
  doodles <- list(missing = NULL,means= NULL, sds= NULL, p.value= NULL, is.binary = NULL)
cnt.NA <- function(x) sum(is.na(x)==TRUE)
doodles$missing <- cnt.NA(x)
group.ttest <- aov(x ~ groups, getOption("na.omit"))
if(length(levels(groups))<=2){
  twolvlpval <-t.test(x~groups, getOption("na.omit"))
  doodles$means <-round(c(twolvlpval$estimate[[1]],twolvlpval$estimate[[2]]), digits = 2)}
else {doodles$means <- unique(round(group.ttest$fitted.values, digits=2))}
sd.NA <- function(x) sd(x,na.rm =TRUE)
doodles$sds1<- aggregate(x ~ groups, FUN= sd.NA)
doodles$sds <- round(doodles$sds1[,2], digits = 2)
if(length(levels(groups))<=2){
  twolvlpval <-t.test(x~groups, getOption("na.omit"))
  doodles$p.value <-twolvlpval$p.value
  }else{doodles$p.value <- summary(group.ttest)[[1]][["Pr(>F)"]][1]}
doodles$is.binary <- all(x[1:length(x)]==1 | x[1:length(x)]==0)
print(doodles[-6])
}
```

Below is a function that summarizes binary data (i.e. 1 or 0) by reporting the number of missing values, the proportions of each of the data's groups, and the p-value, which tests whether the proportions under each of the data's groups are statistically the same.

```
generateBinarySummary <- function(x, groups) {dandy <- list(missing = NULL,prop= NULL, p.value= NULL, is
cnt.NA <- function(x) sum(is.na(x)==TRUE)
dandy$missing <- cnt.NA(x)
dandy.aovtest <- aov(x ~ groups, getOption("na.omit"))
if(length(levels(groups))<=2){
```

```
  bin.twolvlpval <-t.test(x~groups, getOption("na.omit"))
  dandy$prop <-round(c(bin.twolvlpval$estimate[[1]],bin.twolvlpval$estimate[[2]]), digits = 3)
  }else{
    dandy$prop <- unique(round(dandy.aovtest$fitted.values, digits=4))
    }
dandy.ftest <- fisher.test(x, y=groups)
dandy$p.value <- dandy.ftest$p.value
dandy$is.binary <- all(x[1:length(x)]==1 | x[1:length(x)]==0)
dandy
}
```

Below is a function that incorporates the code chunks above to produce a vector that summarizes the data's groups, whether binary or numeric.

```
generateVariableSummary <- function(x, groups){
    if(all(x[1:length(x)]==1
           | x[1:length(x)]==0|
           is.na(x))){
    generateBinarySummary(x,groups)}
  else if (any(is.double(x)) &
           all(x[1:length(x)]!=1|x[1:length(x)]!=0 | is.na(x))){
    generateNumericSummary(x,groups)}
  else {
    return (NULL)}
}
```

Below is another function that lays the ground work to accept the output of the generateVariableSumaary function to produces a formatted summary vector of the binary and numeric data. The output of this function can be used in a series of row and column binds to produce a summary matrix.

```
formatVariableSummary <- function(var.summary) {
  if(length(var.summary)>4){
    a <- var.summary$missing
    d <- round(var.summary$p.value, digits=5)
    b <-c(round(var.summary$means, digits=4))
    c <- c(round(var.summary$sds, digits=2))
    f<- paste(b, " (", c , ")", sep="")
    c(a,f,d)
    }else{
      g <- round(var.summary$missing, digits = 0)
      h <- round(var.summary$p.value, digits = 3)
      i <-c(var.summary$prop, digits=3)
      k<- paste(i, "%", sep="")
      c(g,k,h)}
}
```