Writing a simple R package in S3.

Ronald (Ryy) Glenn Thomas

Table of contents

Figure 1: S3 OOP in R

1 Introduction

Ok! It time to step up my R programming to S3. But how to start?

I'll begin by reading the relevant chapters in Advanced R (add reference).

Other references:

Introduction to Scientific Programming and Simulation using R. Jomes. Maillardet, Robinson.

A Simple Guide to S3 Methods | R-bloggers

Why your S3 method isn't working | R-bloggers

Dealing with S3 methods in R with a simple example | R-bloggers

Video on S3 Classes in R by Dr Andrew Robinson | R-bloggers

Unexported S3 Methods and R Packages | R-bloggers

Simple Guide to S3 Methods | R-bloggers

The S3 OOP system | R-bloggers

2 Appendix zz.table1.c

```
zz.table1.c = function(df, form, pv=TRUE, totl=TRUE, grps=TRUE) {
if (!require("pacman")) install.packages("pacman", repo="cran.rstudio.com")
pacman::p_load(janitor )
prep =  function(df, form) {
dfr = df %>%
    ungroup %>%
    sel(all.vars(form[[3]]))
df_list = dfr %>%  split( df_grp) %>%
list_merge(., "Total" = dfr) %>%
purrr::transpose()
}
process1 = function(x){
pv_chr = data.frame(x[["Total"]],df_grp) %>%
pvalue_chr
ll = x[[length(x)]]%>%  as.factor %>% levels
ll_indent = paste("\\hspace{5mm}   ",ll)
sum_chr = x %>%
    lapply(function(x) factor(x, levels=ll)) %>%
    map(categ)   %>% as_tibble  %>%
    cbind(variable=ll_indent, ., 'p-value'=NA) %>%
    mut(variable=as.character(variable)) %>%
    rbind(NA, .)
sum_chr[1,ncol(sum_chr)]=pv_chr
# browser()
return(sum_chr)
}
process2 = function(x){
pv_num = data.frame(x[["Total"]],df_grp) %>%
   pvalue_num
sum_num = x %>%
   map_chr(contin)  %>%
   bind_rows  %>%
   cbind(variable=NA,., 'p-value'=pv_num)    %>%
```

```
   mut(variable=as.character(variable))
return(sum_num)
}
contin= function(x) {
s1 = zz.sum.min(x)
paste0(s1['Mean'],"$\\pm$", s1['SD'], " ({\\scriptsize $",s1['N'],"$})") }
categ = function(x) {
prps = table(x) %>% prop.table  %>%  round(2)*100
cnts_prps =  table(x)%>%
paste0(.," ({\\scriptsize $",prps,"$})")
}
pvalue_num = function(df) {
tidy(anova(lm(as.formula(paste(names(df),  collapse="~")), data = df)))$p.value[1]
}
pvalue_chr = function(df) {
tab =  table(df[,1], df[,2])
ifelse((nrow(tab) >=2 & ncol(tab) >=2),
       stats::fisher.test(tab,simulate.p.value=T)$p.value, NA)
}
fieldclass =sapply(df, class)%>%  enframe %>%
slice(match(all.vars(form[[3]]),name))
groupclass =sapply(df, class)%>%  enframe %>%
slice(match(all.vars(form[[2]]),name))
df_grp<- df %>%  pull(groupclass$name)
df2 = prep(df, form)
out = df2 %>%
  map_if(fieldclass$value=="numeric" | fieldclass$value=="integer", function(x){process2(x)}
       map_if(fieldclass$value=="character", function(x){process1(x)})   %>%
imap(function(x,y) {
         y2 = ifelse(fieldclass$value[fieldclass$name == y]=="character",
             paste(y, "-- {\\scriptsize no. (\\%)}"), y)
     x[1,1]=y2
#  browser()
     x
       }) %>%
   bind_rows()
on= names(out)
nn = tabyl(df_grp)%>%
    adorn_totals() %>%
```

```
    pull(n)
names(out) = paste(rep("{\\bf",length(on)),on, c("",paste0("\\scriptsize(n=",nn,")"),""),rep
if (!grps) out = out %>%  sel(contains("variable"),contains("Total"),contains("p-value"))
if (!pv) {
    out = out %>%  sel(-contains("p-value"))}
if (!totl) out = out %>%  sel(-contains("Total"))
return(out)
}
```