hw1

Liming Ning

2022/1/16

Case Study 1: Audience Size

Data Preparation

cleaning

```
# selection
talkdata = fread("data/Survey_results_final.csv", encoding = "UTF-8")
talkdata.selected =
  talkdata[,.(age = Answer.Age, # Note: some answers in Age are not numerics. Therefore, typeof Answer.
              gender = Answer.Gender,
              education = Answer.Education,
              income = Answer.HouseHoldIncome,
              sirius = `Answer.Sirius Radio`,
              wharton = `Answer.Wharton Radio`,
              worktime = WorkTimeInSeconds)]
talkdata$Reward[1:10] # question: 5 cents or 10 cents?
```

```
##  [1] "$0.05" "$0.05" "$0.05" "$0.05" "$0.05" "$0.05" "$0.05" "$0.05" "$0.05"
## [10] "$0.05"
```

```
# detect suspect observations
## age
talkdata.selected[!age %in% 10:100] # automatic type coersion when matching
```

```
##              age gender                                        education
## 1:                Male                                      select one
## 2:           223   Male            High school graduate (or equivalent)
## 3:        female Female Some college, no diploma; or Associate's degree
## 4: Eighteen (18)   Male            High school graduate (or equivalent)
## 5:             4   Male        Bachelor's degree or other 4-year degree
## 6:           27`   Male Some college, no diploma; or Associate's degree
##                income sirius wharton worktime
## 1:                                         5
## 2: $30,000 - $50,000     No      No       11
## 3:    Above $150,000    Yes      No       21
## 4: $30,000 - $50,000    Yes      No       29
## 5: $50,000 - $75,000    Yes      No       22
## 6: Less than $15,000     No      No       20
```

```
talkdata.selected[age == "Eighteen (18)", age := "18"] # imputation
talkdata.selected[age == "27`", age := "27"] # imputation
talkdata.selected = talkdata.selected[age %in% 10:100] # delete NAs
```

```
talkdata.selected[,age := as.numeric(age)]

## gender
unique(talkdata.selected$gender)
```

```
## [1] "Female" "Male"   ""
```

```
talkdata.selected[!gender %in% c("Male","Female")]
```

```
##    age gender                                          education            income
## 1:  47                        Graduate or professional degree $30,000 - $50,000
## 2:  47                        Graduate or professional degree $50,000 - $75,000
## 3:  29       Some college, no diploma; or Associate's degree $15,000 - $30,000
## 4:  31                        Graduate or professional degree $30,000 - $50,000
## 5:  25       Some college, no diploma; or Associate's degree Less than $15,000
## 6:  67       Some college, no diploma; or Associate's degree $50,000 - $75,000
##    sirius wharton worktime
## 1:    Yes      No       54
## 2:    Yes      No       15
## 3:    Yes      No       19
## 4:     No      No       15
## 5:    Yes      No       19
## 6:     No      No       32
```

```
talkdata.selected = talkdata.selected[gender != ""] # delete blanks

## education
unique(talkdata.selected$education)
```

```
## [1] "Some college, no diploma; or Associate's degree"
## [2] "Graduate or professional degree"
## [3] "Bachelor's degree or other 4-year degree"
## [4] "High school graduate (or equivalent)"
## [5] "Less than 12 years; no high school diploma"
## [6] "select one"
## [7] "Other"
```

```
talkdata.selected = talkdata.selected[!education %in% c("Other","select one")] # delete because they ar

## income
unique(talkdata.selected$income)
```

```
## [1] "$30,000 - $50,000"  "$15,000 - $30,000"  "$50,000 - $75,000"
## [4] "Above $150,000"     "Less than $15,000"  "$75,000 - $150,000"
## [7] ""
```

```
talkdata.selected = talkdata.selected[income != ""] # delete blanks

## sirius
unique(talkdata.selected$sirius)
```

```
## [1] "No"  "Yes" ""
```

```
talkdata.selected = talkdata.selected[sirius != ""] # delete blanks

## wharton
unique(talkdata.selected$wharton)
```

```
## [1] "No"  "Yes" ""
```

```
talkdata.selected = talkdata.selected[wharton != ""] # delete blanks

fwrite(talkdata.selected,"data/talkdata_cleaned.csv",row.names = F)
rm(list = ls())
## worktime: automatically recorded.
# possible improvements: use dplyr. get summary stats in one go and make imputation/dropping them.
# alternatives: do not delete some obs which seems to be valid expect for some missings/errors while we
```

summary stats

```
# age and worktime, integer
talkdata.selected = fread("data/talkdata_cleaned.csv",encoding = "UTF-8")
age.stat = talkdata.selected %>%
  summarise(mean = mean(age),min = min(age),median = median(age),max = max(age),"std. dev." = sd(age))
worktime.stat = talkdata.selected %>%
  summarise(mean = mean(worktime),min = min(age),median = median(worktime),max = max(age), "std. dev." =
cont.var.stat = rbind(age.stat,worktime.stat)
rownames(cont.var.stat) = c("age","worktime")
talkdata.size = nrow(talkdata.selected)
kbl(cont.var.stat, caption = "Summary Statistics for Non-categorical Variables", digits = 2, booktabs =
  kable_styling(latex_options = c("HOLD_position"))%>%
   footnote(general = paste("The table reports the summary statistics for non-categorical variables in
            threeparttable = T)
```

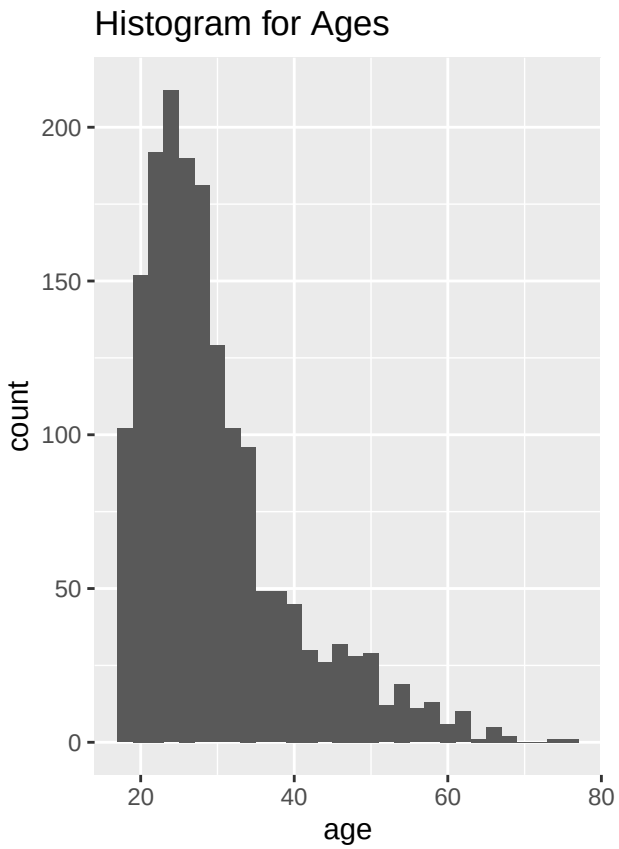
Table 1: Summary Statistics for Non-categorical Variables

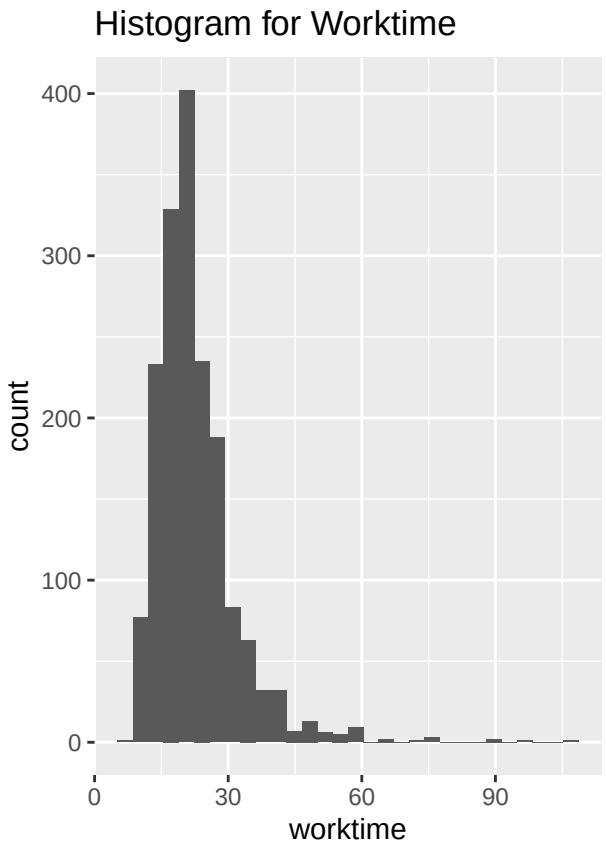
	mean	min	median	max	std. dev.
age	30.28	18	28	76	9.84
worktime	22.49	18	21	76	9.30

Note:
The table reports the summary statistics for non-categorical variables in the talkshow data. The valid sample size is 1725.

```
age.hist = ggplot(talkdata.selected,aes(x=age))+
  geom_histogram()+
  labs(title = "Histogram for Ages")
worktime.hist = ggplot(talkdata.selected,aes(x=worktime))+
  geom_histogram()+
  labs(title = "Histogram for Worktime")
plot_grid(age.hist,worktime.hist,nrow = 1) # right-skewed
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```





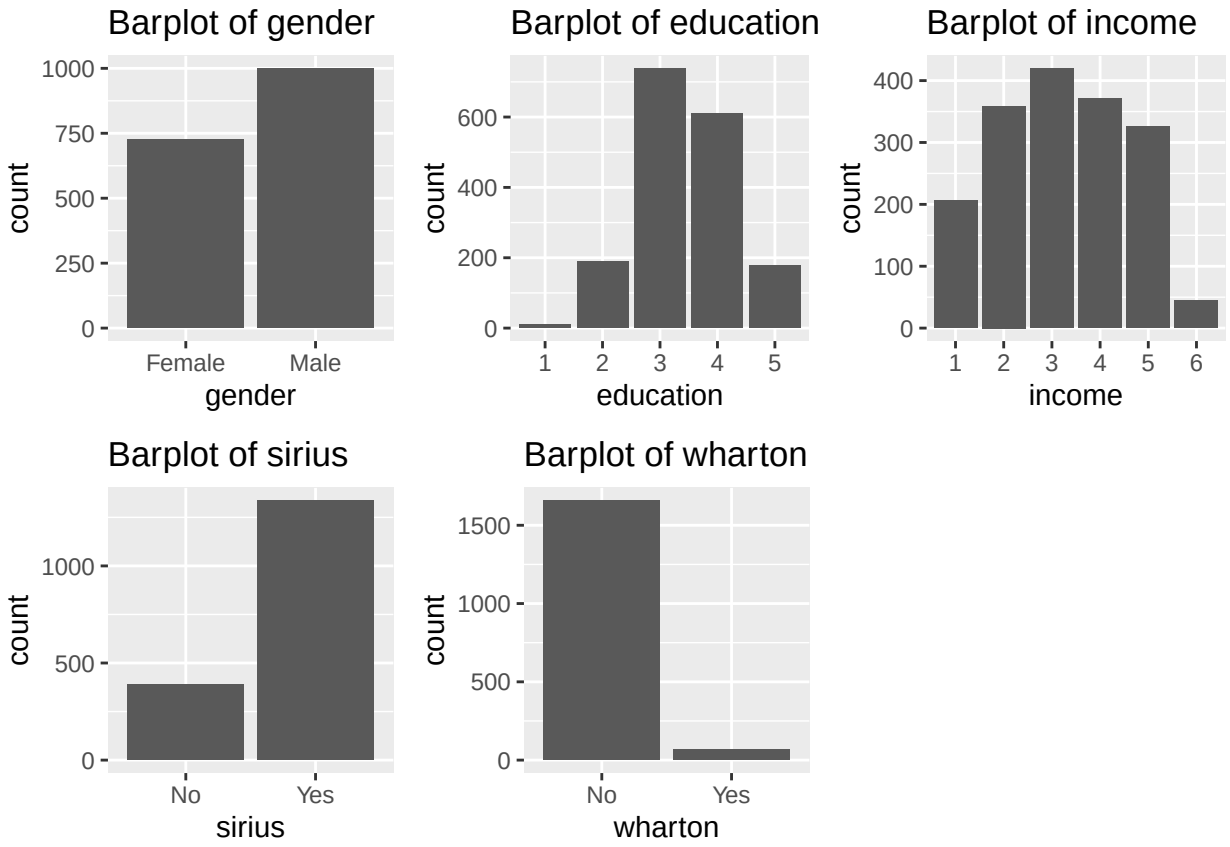

```
# categorical variables
## mapping
keywords.edu = data.table(education = unique(talkdata.selected$education))
keywords.edu[,order := c(3,5,4,2,1)]
keywords.edu = keywords.edu[order(order)]
for (i in 1:nrow(keywords.edu)) {
  talkdata.selected[education==keywords.edu$education[i],education:=keywords.edu$order[i]]
} # for education

keywords.income = data.table(income = unique(talkdata.selected$income))
keywords.income[,order := c(3,2,4,6,1,5)]
keywords.income = keywords.income[order(order)]
for (i in 1:nrow(keywords.income)) {
  talkdata.selected[income==keywords.income$income[i],income:=keywords.income$order[i]]
} # for income

## plots
get.bar = function(data,varname,x.label = varname,y.label = "count",...){

  ggplot(data,aes(x=eval(parse(text = varname))))+
    geom_bar(...)+
    xlab(x.label)+
    labs(title = paste("Barplot of ",varname,sep = ""))
}
bar.list = list()
key.catevar = c("gender","education","income","sirius","wharton")
```

```
for (i in 1:length(key.catevar)) {
  bar.list[[i]] = get.bar(talkdata.selected,key.catevar[i])
}
plot_grid(plotlist = bar.list,nrow = 2)
```



Notes: We map the education and income levels into integers for better exhibition. For education, 1 means *less than 12 years; no high school diploma*, 2 means *High school graduate (or equivalent)*, 3 means *Some college, no diploma; or Associate's degree*, 4 means *Bachelor's degree or other 4-year degree*, 5 means *Graduate or professional degree*. For income, 1 means *Less than $15,000*, 2 means *$15,000 - $30,000*, 3 means *$30,000 - $50,000*, 4 means *$50,000 - $75,000*, 5 means *$75,000 - $150,000*, 6 means *Above $150,000*.