DATA 606 HW 1

2023-01-10

Contents

Question 1

```
# Set seed
set.seed(2023)

# Load packages
library(survey)
library(sampling)
library(dplyr)
# Load data
data <- read.csv("ks-projects-201801.csv")
```

Checking the variable names and dimensions of the data

```
dim(data)
```

```
## [1] 378661     15
```

```
colnames(data)
```

```
##  [1] "ID"               "name"             "category"         "main_category"
##  [5] "currency"         "deadline"         "goal"             "launched"
##  [9] "pledged"          "state"            "backers"          "country"
## [13] "usd.pledged"      "usd_pledged_real" "usd_goal_real"
```

The data frame has 15 columns and 378661 rows.

Question 2

```r
set.seed(2023)
# Sample 5000 indexes
N = nrow(data) # Number of rows
n = 5000 # Sample size
idx = sample(1:N, size = n, replace= FALSE)
samp = data[idx,]
data_sample <- data.frame(samp, pw=rep(N/n,n), fpc=rep(N,n))
svy <- svydesign(id=~0, strata= NULL, weights=~pw,data=data_sample, fpc=~fpc)
mean_sd <- svymean(~usd_pledged_real, svy)
mean_sd
```

```
##                     mean    SE
## usd_pledged_real 8227.9 800.5
```

Using a simple random sample, we get an estimated population mean for usd_pledged_real of 8227.9 and standard deviation of 800.5.

Question 3

```r
set.seed(2023)
unique(data$main_category)
```

```
##  [1] "Publishing"   "Film & Video" "Music"        "Food"         "Design"
##  [6] "Crafts"       "Games"        "Comics"       "Fashion"      "Theater"
## [11] "Art"          "Photography"  "Technology"   "Dance"        "Journalism"
```

```r
table(data$main_category)
```

```
##
##          Art       Comics       Crafts        Dance       Design      Fashion
##        28153        10819         8809         3768        30070        22816
## Film & Video         Food        Games   Journalism        Music  Photography
##        63585        24602        35231         4755        51918        10779
##   Publishing   Technology      Theater
##        39874        32569        10913
```

```r
Ny = c(39874, 63585, 51918, 24602, 30070, 8809, 35231, 10819, 22816, 10913, 28153, 10779, 32569, 3768, 4
fpcs = round(5000*Ny/N)
idx2 = sampling::strata(data, stratanames = c("main_category"), size = fpcs, method = "srswor")
samp2 = getdata(data, idx2)

newdata2 <- data.frame(samp2, pw=1/samp2$Prob, fpc=c(rep(Ny[1], fpcs[1]), rep(Ny[2], fpcs[2]), rep(Ny[3
svy2 <- svydesign(id=~1, strata = ~main_category, weights=~pw,data = newdata2, fpc=~fpc)
mean_sd2 = svymean(~usd_pledged_real, svy2)
mean_sd2
```

```
##                     mean     SE
## usd_pledged_real 9891.6 1542.4
```

Using a stratified sample, we get an estimated population mean for usd_pledged_real of 8567.8 and a standard deviation 844.22.

Question 4

```
set.seed(2023)
idx3<-sampling:::cluster(data, clustername = "country", size = 2, method = "srswor")
clus<-getdata(data, idx3)
length(unique(data$country))
```

```
## [1] 23
```

```
clus$pw=rep(23/2,dim(clus)[1])
unique(clus$country)
```

```
## [1] "MX" "SE"
```

```
nrow(clus[clus$country == "MX",]) + nrow(clus[clus$country == "SE",])
```

```
## [1] 3509
```

```
clus$fpc = c(rep(23, 3509))


scluster<-svydesign(id=~country, weights=~pw, data = clus, fpc=~fpc)
mean_sd3<-svymean(~usd_pledged_real, scluster)
mean_sd3
```

```
##                       mean     SE
## usd_pledged_real 4515.8 2985.4
```

Using a clustered sample, we get an estimated population mean for usd_pledged_real of 11605 and a standard deviation of 159.73.

Question 5

```
ratio<-svyratio(~usd_pledged_real, ~usd_goal_real, svy)
ratio
```

```
## Ratio estimator: svyratio.survey.design2(~usd_pledged_real, ~usd_goal_real, svy)
## Ratios=
##                   usd_goal_real
## usd_pledged_real     0.1717997
## SEs=
##                   usd_goal_real
## usd_pledged_real    0.07391139
```

```r
# Predicted 500k
pred1<-predict(ratio, 500000)
# Predicted 1000k
pred2<-predict(ratio, 1000000)
# Predicted 2000k
pred3<-predict(ratio, 2000000)

pred1
```

```
## $total
##                   usd_goal_real
## usd_pledged_real      85899.85
##
## $se
##                   usd_goal_real
## usd_pledged_real      36955.69
```

```r
pred2
```

```
## $total
##                   usd_goal_real
## usd_pledged_real      171799.7
##
## $se
##                   usd_goal_real
## usd_pledged_real      73911.39
```

```r
pred3
```

```
## $total
##                   usd_goal_real
## usd_pledged_real      343599.4
##
## $se
##                   usd_goal_real
## usd_pledged_real      147822.8
```

We get an estimated ratio of 0.1718 with a standard deviation of 0.07391.

For Goal amount of 500k: Predicted USD Pledged: 85899.85

Standard Deviation: 36955.69

For Goal amount of 1000k: Predicted USD Pledged: 171799.7

Standard Deviation: 73911.39

For Goal amount of 2000k: Predicted USD Pledged: 343599.4

Standard Deviation: 147822.8

Session Info

```
sessionInfo()
```

```
## R version 4.1.1 (2021-08-10)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19044)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_Canada.1252  LC_CTYPE=English_Canada.1252
## [3] LC_MONETARY=English_Canada.1252 LC_NUMERIC=C
## [5] LC_TIME=English_Canada.1252
##
## attached base packages:
## [1] grid      stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
## [1] dplyr_1.0.7     sampling_2.9    survey_4.1-1    survival_3.2-11
## [5] Matrix_1.3-4
##
## loaded via a namespace (and not attached):
##  [1] compiler_4.1.1   pillar_1.6.3     tools_4.1.1      digest_0.6.28
##  [5] evaluate_0.14    lifecycle_1.0.1  tibble_3.1.2     lattice_0.20-44
##  [9] pkgconfig_2.0.3  rlang_0.4.11     DBI_1.1.1        yaml_2.2.1
## [13] xfun_0.26        fastmap_1.1.0    stringr_1.4.0    knitr_1.36
## [17] generics_0.1.0   vctrs_0.3.8      mitools_2.4      tidyselect_1.1.1
## [21] glue_1.4.2       R6_2.5.1         fansi_0.5.0      rmarkdown_2.11
## [25] purrr_0.3.4      magrittr_2.0.1   htmltools_0.5.2  ellipsis_0.3.2
## [29] MASS_7.3-54      splines_4.1.1    assertthat_0.2.1 lpSolve_5.6.17
## [33] utf8_1.2.2       stringi_1.7.5    crayon_1.4.1
```