Sample

Joseph Lavicka

2021-04-26

```r
library(readr)
library(dplyr)
library(stringr)
library(tidyr)
library(purrr)
library(eiPack)
library(readxl)
library(writexl)
```

```r
df <- read_excel("sample_data.xlsx")

#### output name
x <- "sample_output.xlsx"

#### party column indices
a <- c(2:7)

#### main ethnic group indices
b <- c(8:14)

#### column indices to be combined into other group
c <- c(15)

#### column indices to be combined into unknown group
d <- c(16)
```

```r
colnames(df) -> df_colnames

data.frame("p" = rep("p", length(a)), "n" = 1:length(a)) %>%
  unite(p, n, col = "partyid", sep = "") %>%
  cbind(party = df_colnames[a]) -> partyid

if (is.null(d) == TRUE & is.null(c) == FALSE){
  df %>%
    mutate(pop = rowSums(df[c(b,c)]),
           valid = rowSums(df[a]),
           novote = pop - valid) %>%
    filter(novote < 0) -> df_na

  names(df)[a] <- partyid$partyid
```

```r
df %>%
  mutate(pop = rowSums(df[c(b,c)]),
         valid = rowSums(df[a]),
         novote = pop - valid,
         Other = rowSums(df[c])) %>%
  relocate(c(pop, valid, novote, Other), .after = last_col()) %>%
  filter(novote > 0) -> df

df[c(b,ncol(df))] -> agg

df[c(a,b,ncol(df)-c(0,1))]/df$pop -> df[c(a,b,ncol(df)-c(0,1))]

} else if (is.null(c) == TRUE & is.null(d) == FALSE) {
  df %>%
  mutate(pop = rowSums(df[c(b,d)]),
         valid = rowSums(df[a]),
         novote = pop - valid) %>%
  filter(novote < 0) -> df_na

names(df)[a] <- partyid$partyid

df %>%
  mutate(pop = rowSums(df[c(b,d)]),
         valid = rowSums(df[a]),
         novote = pop - valid,
         Unknown = rowSums(df[d])) %>%
  relocate(c(pop, valid, novote, Unknown), .after = last_col()) %>%
  filter(novote > 0) -> df

df[c(b,ncol(df))] -> agg

df[c(a,b,ncol(df)-c(0,1))]/df$pop -> df[c(a,b,ncol(df)-c(0,1))]

} else if (is.null(d) == TRUE & is.null(c) == TRUE){
df %>%
  mutate(pop = rowSums(df[c(b)]),
         valid = rowSums(df[a]),
         novote = pop - valid) %>%
  filter(novote < 0) -> df_na

names(df)[a] <- partyid$partyid

df %>%
  mutate(pop = rowSums(df[c(b)]),
         valid = rowSums(df[a]),
         novote = pop - valid) %>%
  relocate(c(pop, valid, novote), .after = last_col()) %>%
  filter(novote > 0) -> df

df[b] -> agg

df[c(a,b,ncol(df))]/df$pop -> df[c(a,b,ncol(df))]
```

```
  } else {
    df %>%
      mutate(pop = rowSums(df[c(b,c,d)]),
             valid = rowSums(df[a]),
             novote = pop - valid) %>%
      filter(novote < 0) -> df_na

    names(df)[a] <- partyid$partyid

    df %>%
      mutate(pop = rowSums(df[c(b,c,d)]),
             valid = rowSums(df[a]),
             novote = pop - valid,
             Unknown = rowSums(df[d]),
             Other = rowSums(df[c])) %>%
      relocate(c(pop, valid, novote, Unknown, Other), .after = last_col()) %>%
      filter(novote > 0) -> df

    df[c(b,ncol(df)-c(0,1))] -> agg

    df[c(a,b,ncol(df)-c(0:2))]/df$pop -> df[c(a,b,ncol(df)-c(0:2))]
}
```

Model

```
set.seed(42)

if (is.null(c) == TRUE & is.null(d) == FALSE | is.null(d) == TRUE & is.null(c) == FALSE){

  tune.out <- tuneMD(as.matrix(df[c(a,ncol(df)-1)]) ~ as.matrix(df[c(b,ncol(df))]), covariate = NULL, da

  ei.out <- ei.MD.bayes(as.matrix(df[c(a,ncol(df)-1)]) ~ as.matrix(df[c(b,ncol(df))]), total = "pop", da

      } else if (is.null(c) == TRUE & is.null(d) == TRUE){

        tune.out <- tuneMD(as.matrix(df[c(a,ncol(df))]) ~ as.matrix(df[b]), covariate = NULL, data = df


         ei.out <- ei.MD.bayes(as.matrix(df[c(a,ncol(df))]) ~ as.matrix(df[b]), total = "pop", data = d

         } else {

           tune.out <- tuneMD(as.matrix(df[c(a,ncol(df)-2)]) ~ as.matrix(df[c(b,ncol(df)-c(0,1))]), cova


           ei.out <- ei.MD.bayes(as.matrix(df[c(a,ncol(df)-2)]) ~ as.matrix(df[c(b,ncol(df)-c(0,1))]),

           }
```

National Estimates

```
## add mean of ei estimates
as.data.frame(ei.out$draws$Cell.counts)  %>%
  map(mean) %>%
  as.data.frame() %>%
  gather(key = "ethn_party", value = "mean") %>%
  mutate(mean = round(mean, 2),
         ethn_party = str_replace(ethn_party, "^ccount\\.", "")) %>%
  separate(ethn_party, sep = "\\.(?=p[[:digit:]])|\\.(?=novote)", into = c("ethn", "partyid")) %>%
  mutate(partyid = as.factor(partyid)) -> ei.est

## add standard deviation of ei estimates
as.data.frame(ei.out$draws$Cell.counts)  %>%
  map(sd) %>%
  as.data.frame() %>%
  gather(value = "sd") %>%
  mutate(sd = round(sd, 2)) %>%
  select(sd) -> ei.est[,4]

## add sum and percent of ethnic group totals from original dataset
agg %>%
  map(sum) %>%
  as.data.frame() %>%
  gather(key = "ethn", value = "total") %>%
  right_join(ei.est, by = "ethn") %>%
  mutate(percent = round((mean/total*100), 2)) %>%
  select(ethn, partyid, mean, sd, total, percent) -> ei.est

## add sum and percent of ethnic group totals from ei estimates
ei.est %>%
  group_by(ethn) %>%
  summarise(est_total = sum(mean)) %>%
  right_join(ei.est, by = "ethn") %>%
  mutate(est_percent = round((mean/est_total*100), 2)) %>%
  select(ethn, partyid, mean, sd, total, percent, est_total, est_percent) -> ei.est

## add sum and percent of voting population - totals minus novote party
ei.est %>%
  filter(partyid != "novote") %>%
  group_by(ethn) %>%
  summarise(est_vot_total = sum(mean)) %>%
  right_join(ei.est, by = "ethn") %>%
  mutate(est_vot_percent = round((mean/est_vot_total*100), 2)) %>%
  select(ethn, partyid, mean, sd, total, percent, est_total, est_percent, est_vot_total, est_vot_percen

## attach party names
full_join(ei.est, partyid, by = "partyid") -> ei.est
ei.est[,c(1,11,2:10)] -> ei.est
```

Table

```
ei.est %>%
  group_by(ethn, party) %>%
  summarise(est_vot_percent) %>%
  spread(ethn, est_vot_percent) -> tbl

tbl[1:nrow(tbl) - 1,] -> tbl
```

Codebook

```
read.me <- data.frame(name = c("table",
                               "output",
                               "mismatched data",
                               "",
                               "ethn",
                               "party",
                               "mean",
                               "sd",
                               "total",
                               "percent",
                               "est_total",
                               "est_percent",
                               "est_vot_total",
                               "est_vot_percent",
                               "",
                               "political parties",
                               "# of political parties",
                               "ethnic groups",
                               "groups merged into 'Other' category",
                               "groups merged into 'Unknown' category",
                               "# of ethnic groups (including 'Other' and 'Unknown')",
                               "",
                               "# of political units",
                               "# of mismatched observations"
                               ),
                      description = c("summary - percentage of votes cast by respective ethnic group fo
                                      "ei results of national estimates with major ethnic groups listed
                                      "observations where total valid votes are greater than total popu
                                      "",
                                      "ethnic group",
                                      "political party - coded relative to order of party in original da
                                      "average of ei estimates for number of votes cast by respective e
                                      "standard error of ei estimates for number of votes cast by respe
                                      "total ethnic group population from original dataset",
                                      "percentage of votes cast by respective ethnic group for respecti
                                      "total ethnic group population from ei results",
                                      "percentage of votes cast by respective ethnic group for respecti
                                      "total number of votes cast by respective ethnic group",
                                      "percentage of votes cast by respective ethnic group for respecti
                                      "",
                                      gsub("^c\\(|\\)$", "", paste(as.data.frame(partyid$party))),
                                      nrow(tbl),
                                      gsub("^c\\(|\\)$", "", paste(as.data.frame(df_colnames[b]))),
                                      if (is.null(c) == TRUE){
                                        paste("NA")
                                      } else {
                                      gsub("^c\\(|\\)$", "", paste(as.data.frame(df_colnames[c])))},
                                      if (is.null(d) == TRUE){
                                        paste("NA")
                                      } else {
                                      gsub("^c\\(|\\)$", "", paste(as.data.frame(df_colnames[d])))},
                                      ncol(tbl) - 1,
                                      "",
                                      nrow(df),
                                      nrow(df_na)
                                      ))
```

```
sheets <- list("read.me" = read.me, "table" = tbl, "output" = ei.est, "mismatched data" = df_na)
write_xlsx(sheets, path = x)
```