Tutorial 1 DSA2101

Waseem

11/10/2021

1. There are missing values in the tables. Inspect the tables carefully and fill them in. The new arrests object should be free of NA values.

```
arrests <- readRDS("data/arrests.rds")
sapply(arrests,function(x) any(is.na(x[[1]])) || any(is.na(x[[2]])))
```

```
##  2011  2012  2013  2014  2015  2016  2017  2018  2019
##  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
```

```
arrests[['2011']]$age$Female[1] <- 2992 +1125 + 1008
arrests[['2019']]$age$Male[2] <- 9920 + 3208 + 10465
```

2. In the DataCamp R markdown course, you learnt about kable from the knitr package. Use it to display the tables for 2019:

```
library(knitr)
kable(arrests[["2019"]],
row.names= NA,
col.names= c('Citizenship status', 'Male', 'Female' ))
```

3. Retrieve the citizenship table from 2011 to 2015 inclusive and store it in a list object called qn3_list.

```
qn3_list<-lapply(arrests[1:5],function(x) x$citizenship)
```

4. Compute the total number of arrests in each year, and store them in a numeric vector qn4_vec.

```
qn4_vec<- sapply(arrests,function(x) sum( x[[1]][,-1]))
```

5. Use qn4_vec to retrieve and store only those years' data where the total number of arrests was less than 18000, but more than 17000. Store them in qn5_list.

```
qn5_list <- arrests[qn4_vec <18000 & 17000 > qn4_vec ]
```

6. Measure of association I: When we deal with 2x2 tables, one measure of association between the variables is the difference in proportions. Write a function prop_diff that takes in one of the data frames and returns the difference between proportion of Males for each row.

```
x<-arrests[[1]][[1]]
prop_diff<- function(x){
(x[[2]][[1]]/(x[[2]][[1]]+x[[3]][[1]]))-(x[[2]][[2]]/(x[[2]][[2]]+x[[3]][[2]]))
```

	Citizenship status	Male	Female
33	Singaporeans/ Permanent Residents/ Stateless	9920	2721
34	Foreigners	3208	1297

	Citizenship status	Male	Female
35	Above 21 Years Old	10465	3227
36	21 Years Old And Below	23593	791

```
}

qn6.1_vec <- sapply(arrests,function(y) prop_diff(y[[1]]))
qn6.2_vec <- sapply(arrests,function(y) prop_diff(y[[2]]))
qn6_df <- data.frame (qn6.1_vec,qn6.2_vec)
qn6_df
```

```
##       qn6.1_vec    qn6.2_vec
## 2011 0.08499876 -0.079035350
## 2012 0.08171101  0.008610853
## 2013 0.07155063  0.031289433
## 2014 0.06374857  0.023273612
## 2015 0.05766612 -0.016348598
## 2016 0.05546663  0.014762828
## 2017 0.06860536  0.007272868
## 2018 0.08541846 -0.010344112
## 2019 0.07265037 -0.203245767
```

7. Compute the log-odds ratio for each data.frame and store them in a data.frame called qn7_df, with one row for each year.

```
bob <- function (x){
log(x[[2]][[1]] * x[[3]][[2]]) / ( x[[2]][[2]] * x[[3]][[1]])

}
qn7.1_vec <- sapply(arrests,function(y) bob(y[[1]]))
qn7.2_vec <- sapply(arrests,function(y) bob(y[[2]]))
qn7_df <- data.frame (qn7.1_vec,qn7.2_vec)
qn7_df
```

```
##         qn7.1_vec    qn7.2_vec
## 2011 1.979125e-06 9.679670e-07
## 2012 2.045130e-06 1.650381e-06
## 2013 2.273519e-06 2.130018e-06
## 2014 2.005544e-06 1.938347e-06
## 2015 1.952595e-06 1.681588e-06
## 2016 1.847883e-06 1.901154e-06
## 2017 1.971599e-06 1.940951e-06
## 2018 1.922581e-06 1.815656e-06
## 2019 1.875378e-06 2.092227e-07
```