Learning loop functions using Flags data set: lapply, sapply, tapply, vapply

Arin Parsa

6/6/2020

Contents

Flags data set by UCI ML Rep

http://archive.ics.uci.edu/ml/datasets/Flags

Flags Data Set Information

This data file contains details of various nations and their flags. In this file the fields are separated by spaces (not commas). With this data you can try things like predicting the religion of a country from its size and the colors in its flag.

10 attributes are numeric-valued. The remainder are either Boolean- or nominal-valued.

Attribute Information:

1. name: Name of the country concerned
2. landmass: 1=N.America, 2=S.America, 3=Europe, 4=Africa, 4=Asia, 6=Oceania
3. zone: Geographic quadrant, based on Greenwich and the Equator; 1=NE, 2=SE, 3=SW, 4=NW
4. area: in thousands of square km
5. population: in round millions
6. language: 1=English, 2=Spanish, 3=French, 4=German, 5=Slavic, 6=Other Indo-European, 7=Chinese, 8=Arabic, 9=Japanese/Turkish/Finnish/Magyar, 10=Others
7. religion: 0=Catholic, 1=Other Christian, 2=Muslim, 3=Buddhist, 4=Hindu, 5=Ethnic, 6=Marxist, 7=Others
8. bars: Number of vertical bars in the flag
9. stripes: Number of horizontal stripes in the flag
10. colours: Number of different colours in the flag
11. red: 0 if red absent, 1 if red present in the flag
12. green: same for green
13. blue: same for blue
14. gold: same for gold (also yellow)
15. white: same for white
16. black: same for black
17. orange: same for orange (also brown)

18. mainhue: predominant colour in the flag (tie-breaks decided by taking the topmost hue, if that fails then the most central hue, and if that fails the leftmost hue)
19. circles: Number of circles in the flag
20. crosses: Number of (upright) crosses
21. saltires: Number of diagonal crosses
22. quarters: Number of quartered sections
23. sunstars: Number of sun or star symbols
24. crescent: 1 if a crescent moon symbol present, else 0
25. triangle: 1 if any triangles present, 0 otherwise
26. icon: 1 if an inanimate image present (e.g., a boat), otherwise 0
27. animate: 1 if an animate image (e.g., an eagle, a tree, a human hand) present, 0 otherwise
28. text: 1 if any letters or writing on the flag (e.g., a motto or slogan), 0 otherwise
29. topleft: colour in the top-left corner (moving right to decide tie-breaks)
30. botright: Colour in the bottom-left corner (moving left to decide tie-breaks)

Create a vector of column names, and read file as table into a data frame

```
colnames <- c("country", "landmass", "zone", "area", "population", "language", "religion", "bars", "str

# get flag data set from UCI. If this fails, get the dataset from the repo of this markdown file
dataset <-"https://archive.ics.uci.edu/ml/machine-learning-databases/flags/flag.data"

# read data into a table.
flags <- read.table(dataset, sep=",", col.names = colnames, as.is = TRUE)
```

Verify data is read properly and all countries are loaded

```
flags[,1]
```

```
##   [1] "Afghanistan"              "Albania"
##   [3] "Algeria"                  "American-Samoa"
##   [5] "Andorra"                  "Angola"
##   [7] "Anguilla"                 "Antigua-Barbuda"
##   [9] "Argentina"                "Argentine"
##  [11] "Australia"                "Austria"
##  [13] "Bahamas"                  "Bahrain"
##  [15] "Bangladesh"               "Barbados"
##  [17] "Belgium"                  "Belize"
##  [19] "Benin"                    "Bermuda"
##  [21] "Bhutan"                   "Bolivia"
##  [23] "Botswana"                 "Brazil"
##  [25] "British-Virgin-Isles"     "Brunei"
##  [27] "Bulgaria"                 "Burkina"
##  [29] "Burma"                    "Burundi"
##  [31] "Cameroon"                 "Canada"
##  [33] "Cape-Verde-Islands"       "Cayman-Islands"
##  [35] "Central-African-Republic" "Chad"
##  [37] "Chile"                    "China"
##  [39] "Colombia"                 "Comorro-Islands"
##  [41] "Congo"                    "Cook-Islands"
##  [43] "Costa-Rica"               "Cuba"
##  [45] "Cyprus"                   "Czechoslovakia"
##  [47] "Denmark"                  "Djibouti"
```

```
##  [49] "Dominica"                "Dominican-Republic"
##  [51] "Ecuador"                 "Egypt"
##  [53] "El-Salvador"             "Equatorial-Guinea"
##  [55] "Ethiopia"                "Faeroes"
##  [57] "Falklands-Malvinas"      "Fiji"
##  [59] "Finland"                 "France"
##  [61] "French-Guiana"           "French-Polynesia"
##  [63] "Gabon"                   "Gambia"
##  [65] "Germany-DDR"             "Germany-FRG"
##  [67] "Ghana"                   "Gibraltar"
##  [69] "Greece"                  "Greenland"
##  [71] "Grenada"                 "Guam"
##  [73] "Guatemala"               "Guinea"
##  [75] "Guinea-Bissau"           "Guyana"
##  [77] "Haiti"                   "Honduras"
##  [79] "Hong-Kong"               "Hungary"
##  [81] "Iceland"                 "India"
##  [83] "Indonesia"               "Iran"
##  [85] "Iraq"                    "Ireland"
##  [87] "Israel"                  "Italy"
##  [89] "Ivory-Coast"             "Jamaica"
##  [91] "Japan"                   "Jordan"
##  [93] "Kampuchea"               "Kenya"
##  [95] "Kiribati"                "Kuwait"
##  [97] "Laos"                    "Lebanon"
##  [99] "Lesotho"                 "Liberia"
## [101] "Libya"                   "Liechtenstein"
## [103] "Luxembourg"              "Malagasy"
## [105] "Malawi"                  "Malaysia"
## [107] "Maldive-Islands"         "Mali"
## [109] "Malta"                   "Marianas"
## [111] "Mauritania"              "Mauritius"
## [113] "Mexico"                  "Micronesia"
## [115] "Monaco"                  "Mongolia"
## [117] "Montserrat"              "Morocco"
## [119] "Mozambique"              "Nauru"
## [121] "Nepal"                   "Netherlands"
## [123] "Netherlands-Antilles"    "New-Zealand"
## [125] "Nicaragua"               "Niger"
## [127] "Nigeria"                 "Niue"
## [129] "North-Korea"             "North-Yemen"
## [131] "Norway"                  "Oman"
## [133] "Pakistan"                "Panama"
## [135] "Papua-New-Guinea"        "Parguay"
## [137] "Peru"                    "Philippines"
## [139] "Poland"                  "Portugal"
## [141] "Puerto-Rico"             "Qatar"
## [143] "Romania"                 "Rwanda"
## [145] "San-Marino"              "Sao-Tome"
## [147] "Saudi-Arabia"            "Senegal"
## [149] "Seychelles"              "Sierra-Leone"
## [151] "Singapore"               "Soloman-Islands"
## [153] "Somalia"                 "South-Africa"
## [155] "South-Korea"             "South-Yemen"
```

```
## [157] "Spain"                    "Sri-Lanka"
## [159] "St-Helena"                "St-Kitts-Nevis"
## [161] "St-Lucia"                 "St-Vincent"
## [163] "Sudan"                    "Surinam"
## [165] "Swaziland"                "Sweden"
## [167] "Switzerland"              "Syria"
## [169] "Taiwan"                   "Tanzania"
## [171] "Thailand"                 "Togo"
## [173] "Tonga"                    "Trinidad-Tobago"
## [175] "Tunisia"                  "Turkey"
## [177] "Turks-Cocos-Islands"      "Tuvalu"
## [179] "UAE"                      "Uganda"
## [181] "UK"                       "Uruguay"
## [183] "US-Virgin-Isles"          "USA"
## [185] "USSR"                     "Vanuatu"
## [187] "Vatican-City"             "Venezuela"
## [189] "Vietnam"                  "Western-Samoa"
## [191] "Yugoslavia"               "Zaire"
## [193] "Zambia"                   "Zimbabwe"
```

Various operations to learn lapply and sapply

```
# create a subset for data for flag colors
flag_colors <- flags[, 11:17]
head(flag_colors)
```

```
##   red green blue gold white black orange
## 1   1     1    0    1     1     1      0
## 2   1     0    0    1     0     1      0
## 3   1     1    0    0     1     0      0
## 4   1     0    1    1     1     0      1
## 5   1     0    1    1     0     0      0
## 6   1     0    0    1     0     1      0
```

```
lapply(flag_colors, sum) #returns a list of number of flags based on color
```

```
## $red
## [1] 153
##
## $green
## [1] 91
##
## $blue
## [1] 99
##
## $gold
## [1] 91
##
## $white
## [1] 146
##
## $black
## [1] 52
##
## $orange
```

```
## [1] 26
```

```
sapply(flag_colors, sum) # sapply returns a vector since list value has only one element which the numb
```

```
##    red  green   blue   gold  white  black orange
##    153     91     99     91    146     52     26
```

```
sapply(flag_colors, mean) # find the mean of flag colors
```

```
##       red     green      blue      gold     white     black    orange
## 0.7886598 0.4690722 0.5103093 0.4690722 0.7525773 0.2680412 0.1340206
```

```
flag_shapes <- flags[,19:23] #returns a data frame of flag shapes
lapply(flag_shapes, range)  #returns a list containing range of flag shapes (how many flags per shape)
```

```
## $circles
## [1] 0 4
##
## $crosses
## [1] 0 2
##
## $saltires
## [1] 0 1
##
## $quarters
## [1] 0 4
##
## $sunstars
## [1]  0 50
```

```
shape_mat <-sapply(flag_shapes, range) # sapply on range will return a matrix this time since list elem
shape_mat
```

```
##      circles crosses saltires quarters sunstars
## [1,]       0       0        0        0        0
## [2,]       4       2        1        4       50
```

```
unique(c(3, 4, 5, 5, 5, 6, 6)) #an example to show that unique function will return only 3, 4, 5, 6
```

```
## [1] 3 4 5 6
```

```
unique_vals_list <- lapply(flags, unique) # returns a list of unique values per column in flags data fr
unique_vals_list
```

```
## $country
##   [1] "Afghanistan"              "Albania"
##   [3] "Algeria"                  "American-Samoa"
##   [5] "Andorra"                  "Angola"
##   [7] "Anguilla"                 "Antigua-Barbuda"
##   [9] "Argentina"                "Argentine"
##  [11] "Australia"                "Austria"
##  [13] "Bahamas"                  "Bahrain"
##  [15] "Bangladesh"               "Barbados"
##  [17] "Belgium"                  "Belize"
##  [19] "Benin"                    "Bermuda"
##  [21] "Bhutan"                   "Bolivia"
##  [23] "Botswana"                 "Brazil"
##  [25] "British-Virgin-Isles"     "Brunei"
```

```
##  [27] "Bulgaria"                 "Burkina"
##  [29] "Burma"                    "Burundi"
##  [31] "Cameroon"                 "Canada"
##  [33] "Cape-Verde-Islands"       "Cayman-Islands"
##  [35] "Central-African-Republic" "Chad"
##  [37] "Chile"                    "China"
##  [39] "Colombia"                 "Comorro-Islands"
##  [41] "Congo"                    "Cook-Islands"
##  [43] "Costa-Rica"               "Cuba"
##  [45] "Cyprus"                   "Czechoslovakia"
##  [47] "Denmark"                  "Djibouti"
##  [49] "Dominica"                 "Dominican-Republic"
##  [51] "Ecuador"                  "Egypt"
##  [53] "El-Salvador"              "Equatorial-Guinea"
##  [55] "Ethiopia"                 "Faeroes"
##  [57] "Falklands-Malvinas"       "Fiji"
##  [59] "Finland"                  "France"
##  [61] "French-Guiana"            "French-Polynesia"
##  [63] "Gabon"                    "Gambia"
##  [65] "Germany-DDR"              "Germany-FRG"
##  [67] "Ghana"                    "Gibraltar"
##  [69] "Greece"                   "Greenland"
##  [71] "Grenada"                  "Guam"
##  [73] "Guatemala"                "Guinea"
##  [75] "Guinea-Bissau"            "Guyana"
##  [77] "Haiti"                    "Honduras"
##  [79] "Hong-Kong"                "Hungary"
##  [81] "Iceland"                  "India"
##  [83] "Indonesia"                "Iran"
##  [85] "Iraq"                     "Ireland"
##  [87] "Israel"                   "Italy"
##  [89] "Ivory-Coast"              "Jamaica"
##  [91] "Japan"                    "Jordan"
##  [93] "Kampuchea"                "Kenya"
##  [95] "Kiribati"                 "Kuwait"
##  [97] "Laos"                     "Lebanon"
##  [99] "Lesotho"                  "Liberia"
## [101] "Libya"                    "Liechtenstein"
## [103] "Luxembourg"               "Malagasy"
## [105] "Malawi"                   "Malaysia"
## [107] "Maldive-Islands"          "Mali"
## [109] "Malta"                    "Marianas"
## [111] "Mauritania"               "Mauritius"
## [113] "Mexico"                   "Micronesia"
## [115] "Monaco"                   "Mongolia"
## [117] "Montserrat"               "Morocco"
## [119] "Mozambique"               "Nauru"
## [121] "Nepal"                    "Netherlands"
## [123] "Netherlands-Antilles"     "New-Zealand"
## [125] "Nicaragua"                "Niger"
## [127] "Nigeria"                  "Niue"
## [129] "North-Korea"              "North-Yemen"
## [131] "Norway"                   "Oman"
## [133] "Pakistan"                 "Panama"
```

```
## [135] "Papua-New-Guinea"         "Parguay"
## [137] "Peru"                     "Philippines"
## [139] "Poland"                   "Portugal"
## [141] "Puerto-Rico"              "Qatar"
## [143] "Romania"                  "Rwanda"
## [145] "San-Marino"               "Sao-Tome"
## [147] "Saudi-Arabia"             "Senegal"
## [149] "Seychelles"               "Sierra-Leone"
## [151] "Singapore"                "Soloman-Islands"
## [153] "Somalia"                  "South-Africa"
## [155] "South-Korea"              "South-Yemen"
## [157] "Spain"                    "Sri-Lanka"
## [159] "St-Helena"                "St-Kitts-Nevis"
## [161] "St-Lucia"                 "St-Vincent"
## [163] "Sudan"                    "Surinam"
## [165] "Swaziland"                "Sweden"
## [167] "Switzerland"              "Syria"
## [169] "Taiwan"                   "Tanzania"
## [171] "Thailand"                 "Togo"
## [173] "Tonga"                    "Trinidad-Tobago"
## [175] "Tunisia"                  "Turkey"
## [177] "Turks-Cocos-Islands"      "Tuvalu"
## [179] "UAE"                      "Uganda"
## [181] "UK"                       "Uruguay"
## [183] "US-Virgin-Isles"          "USA"
## [185] "USSR"                     "Vanuatu"
## [187] "Vatican-City"             "Venezuela"
## [189] "Vietnam"                  "Western-Samoa"
## [191] "Yugoslavia"               "Zaire"
## [193] "Zambia"                   "Zimbabwe"
##
## $landmass
## [1] 5 3 4 6 1 2
##
## $zone
## [1] 1 3 2 4
##
## $area
##   [1]   648    29  2388     0  1247  2777  7690    84    19     1   143    31
##  [13]    23   113    47  1099   600  8512     6   111   274   678    28   474
##  [25]  9976     4   623  1284   757  9561  1139     2   342    51   115     9
##  [37]   128    43    22    49   284  1001    21  1222    12    18   337   547
##  [49]    91   268    10   108   249   239   132  2176   109   246    36   215
##  [61]   112    93   103  3268  1904  1648   435    70   301   323    11   372
##  [73]    98   181   583   236    30  1760     3   587   118   333  1240  1031
##  [85]  1973  1566   447   783   140    41  1267   925   121   195   324   212
##  [97]   804    76   463   407  1285   300   313    92   237    26  2150   196
## [109]    72   637  1221    99   288   505    66  2506    63    17   450   185
## [121]   945   514    57     5   164   781   245   178  9363 22402    15   912
## [133]   256   905   753   391
##
## $population
##  [1]   16    3   20    0    7   28   15    8   90   10    1    6  119    9   35
## [16]    4   24    2   11 1008    5   47   31   54   17   61   14  684  157   39
```

```
## [31]   57  118   13   77   12   56   18   84   48   36   22   29   38   49   45
## [46]  231  274   60
##
## $language
##  [1] 10  6  8  1  2  4  3  5  7  9
##
## $religion
## [1] 2 6 1 0 5 3 4 7
##
## $bars
## [1] 0 2 3 1 5
##
## $stripes
##  [1]  3  0  2  1  5  9 11 14  4  6 13  7
##
## $colors
## [1] 5 3 2 8 6 4 7 1
##
## $red
## [1] 1 0
##
## $green
## [1] 1 0
##
## $blue
## [1] 0 1
##
## $gold
## [1] 1 0
##
## $white
## [1] 1 0
##
## $black
## [1] 1 0
##
## $orange
## [1] 0 1
##
## $mainhue
## [1] "green"  "red"    "blue"   "gold"   "white"  "orange" "black"  "brown"
##
## $circles
## [1] 0 1 4 2
##
## $crosses
## [1] 0 1 2
##
## $saltires
## [1] 0 1
##
## $quarters
## [1] 0 1 4
##
```

```
## $sunstars
##  [1]  1  0  6 22 14  3  4  5 15 10  7  2  9 50
##
## $crescents
## [1] 0 1
##
## $triangle
## [1] 0 1
##
## $icon
## [1] 1 0
##
## $animate
## [1] 0 1
##
## $text
## [1] 0 1
##
## $topleft
## [1] "black"  "red"    "green"  "blue"   "white"  "orange" "gold"
##
## $botright
## [1] "green"  "red"    "white"  "black"  "blue"   "gold"   "orange" "brown"
```

```
v <- sapply(unique_vals_list, length) #sapply returns a vector of number of unique values per column
v
```

```
##    country   landmass       zone       area population   language   religion
##        194          6          4        136         48         10          8
##       bars    stripes     colors        red      green       blue       gold
##          5         12          8          2          2          2          2
##      white      black     orange    mainhue    circles    crosses   saltires
##          2          2          2          8          4          3          2
##   quarters   sunstars  crescents   triangle       icon    animate       text
##          3         14          2          2          2          2          2
##    topleft   botright
##          7          8
```

```
l <- lapply(unique_vals_list, length) #lapply returns a list containing the number of unique values per
l
```

```
## $country
## [1] 194
##
## $landmass
## [1] 6
##
## $zone
## [1] 4
##
## $area
## [1] 136
##
## $population
## [1] 48
##
```

```
## $language
## [1] 10
##
## $religion
## [1] 8
##
## $bars
## [1] 5
##
## $stripes
## [1] 12
##
## $colors
## [1] 8
##
## $red
## [1] 2
##
## $green
## [1] 2
##
## $blue
## [1] 2
##
## $gold
## [1] 2
##
## $white
## [1] 2
##
## $black
## [1] 2
##
## $orange
## [1] 2
##
## $mainhue
## [1] 8
##
## $circles
## [1] 4
##
## $crosses
## [1] 3
##
## $saltires
## [1] 2
##
## $quarters
## [1] 3
##
## $sunstars
## [1] 14
##
```

```
## $crescents
## [1] 2
##
## $triangle
## [1] 2
##
## $icon
## [1] 2
##
## $animate
## [1] 2
##
## $text
## [1] 2
##
## $topleft
## [1] 7
##
## $botright
## [1] 8
```

```
v["landmass"] # returns number of unique landmasses. Note the use of [ since sapply(unique_vals_list, l
```

```
## landmass
##        6
```

```
l[["landmass"]] # returns number of unique landmasses. Note the use of [[ since lapply(unique_vals_list
```

```
## [1] 6
```

```
sapply(flags,unique) #returns unique values in the flag data frame
```

```
## $country
##   [1] "Afghanistan"              "Albania"
##   [3] "Algeria"                  "American-Samoa"
##   [5] "Andorra"                  "Angola"
##   [7] "Anguilla"                 "Antigua-Barbuda"
##   [9] "Argentina"                "Argentine"
##  [11] "Australia"                "Austria"
##  [13] "Bahamas"                  "Bahrain"
##  [15] "Bangladesh"               "Barbados"
##  [17] "Belgium"                  "Belize"
##  [19] "Benin"                    "Bermuda"
##  [21] "Bhutan"                   "Bolivia"
##  [23] "Botswana"                 "Brazil"
##  [25] "British-Virgin-Isles"     "Brunei"
##  [27] "Bulgaria"                 "Burkina"
##  [29] "Burma"                    "Burundi"
##  [31] "Cameroon"                 "Canada"
##  [33] "Cape-Verde-Islands"       "Cayman-Islands"
##  [35] "Central-African-Republic" "Chad"
##  [37] "Chile"                    "China"
##  [39] "Colombia"                 "Comorro-Islands"
##  [41] "Congo"                    "Cook-Islands"
##  [43] "Costa-Rica"               "Cuba"
##  [45] "Cyprus"                   "Czechoslovakia"
```

```
##  [47] "Denmark"                 "Djibouti"
##  [49] "Dominica"                "Dominican-Republic"
##  [51] "Ecuador"                 "Egypt"
##  [53] "El-Salvador"             "Equatorial-Guinea"
##  [55] "Ethiopia"                "Faeroes"
##  [57] "Falklands-Malvinas"      "Fiji"
##  [59] "Finland"                 "France"
##  [61] "French-Guiana"           "French-Polynesia"
##  [63] "Gabon"                   "Gambia"
##  [65] "Germany-DDR"             "Germany-FRG"
##  [67] "Ghana"                   "Gibraltar"
##  [69] "Greece"                  "Greenland"
##  [71] "Grenada"                 "Guam"
##  [73] "Guatemala"               "Guinea"
##  [75] "Guinea-Bissau"           "Guyana"
##  [77] "Haiti"                   "Honduras"
##  [79] "Hong-Kong"               "Hungary"
##  [81] "Iceland"                 "India"
##  [83] "Indonesia"               "Iran"
##  [85] "Iraq"                    "Ireland"
##  [87] "Israel"                  "Italy"
##  [89] "Ivory-Coast"             "Jamaica"
##  [91] "Japan"                   "Jordan"
##  [93] "Kampuchea"               "Kenya"
##  [95] "Kiribati"                "Kuwait"
##  [97] "Laos"                    "Lebanon"
##  [99] "Lesotho"                 "Liberia"
## [101] "Libya"                   "Liechtenstein"
## [103] "Luxembourg"              "Malagasy"
## [105] "Malawi"                  "Malaysia"
## [107] "Maldive-Islands"         "Mali"
## [109] "Malta"                   "Marianas"
## [111] "Mauritania"              "Mauritius"
## [113] "Mexico"                  "Micronesia"
## [115] "Monaco"                  "Mongolia"
## [117] "Montserrat"              "Morocco"
## [119] "Mozambique"              "Nauru"
## [121] "Nepal"                   "Netherlands"
## [123] "Netherlands-Antilles"    "New-Zealand"
## [125] "Nicaragua"               "Niger"
## [127] "Nigeria"                 "Niue"
## [129] "North-Korea"             "North-Yemen"
## [131] "Norway"                  "Oman"
## [133] "Pakistan"                "Panama"
## [135] "Papua-New-Guinea"        "Parguay"
## [137] "Peru"                    "Philippines"
## [139] "Poland"                  "Portugal"
## [141] "Puerto-Rico"             "Qatar"
## [143] "Romania"                 "Rwanda"
## [145] "San-Marino"              "Sao-Tome"
## [147] "Saudi-Arabia"            "Senegal"
## [149] "Seychelles"              "Sierra-Leone"
## [151] "Singapore"               "Soloman-Islands"
## [153] "Somalia"                 "South-Africa"
```

```
## [155] "South-Korea"              "South-Yemen"
## [157] "Spain"                    "Sri-Lanka"
## [159] "St-Helena"                "St-Kitts-Nevis"
## [161] "St-Lucia"                 "St-Vincent"
## [163] "Sudan"                    "Surinam"
## [165] "Swaziland"                "Sweden"
## [167] "Switzerland"              "Syria"
## [169] "Taiwan"                   "Tanzania"
## [171] "Thailand"                 "Togo"
## [173] "Tonga"                    "Trinidad-Tobago"
## [175] "Tunisia"                  "Turkey"
## [177] "Turks-Cocos-Islands"      "Tuvalu"
## [179] "UAE"                      "Uganda"
## [181] "UK"                       "Uruguay"
## [183] "US-Virgin-Isles"          "USA"
## [185] "USSR"                     "Vanuatu"
## [187] "Vatican-City"             "Venezuela"
## [189] "Vietnam"                  "Western-Samoa"
## [191] "Yugoslavia"               "Zaire"
## [193] "Zambia"                   "Zimbabwe"
##
## $landmass
## [1] 5 3 4 6 1 2
##
## $zone
## [1] 1 3 2 4
##
## $area
##   [1]   648    29  2388     0  1247  2777  7690    84    19     1   143    31
##  [13]    23   113    47  1099   600  8512     6   111   274   678    28   474
##  [25]  9976     4   623  1284   757  9561  1139     2   342    51   115     9
##  [37]   128    43    22    49   284  1001    21  1222    12    18   337   547
##  [49]    91   268    10   108   249   239   132  2176   109   246    36   215
##  [61]   112    93   103  3268  1904  1648   435    70   301   323    11   372
##  [73]    98   181   583   236    30  1760     3   587   118   333  1240  1031
##  [85]  1973  1566   447   783   140    41  1267   925   121   195   324   212
##  [97]   804    76   463   407  1285   300   313    92   237    26  2150   196
## [109]    72   637  1221    99   288   505    66  2506    63    17   450   185
## [121]   945   514    57     5   164   781   245   178  9363 22402    15   912
## [133]   256   905   753   391
##
## $population
##  [1]   16    3   20    0    7   28   15    8   90   10    1    6  119    9   35
## [16]    4   24    2   11 1008    5   47   31   54   17   61   14  684  157   39
## [31]   57  118   13   77   12   56   18   84   48   36   22   29   38   49   45
## [46]  231  274   60
##
## $language
##  [1] 10  6  8  1  2  4  3  5  7  9
##
## $religion
## [1] 2 6 1 0 5 3 4 7
##
## $bars
```

```
## [1] 0 2 3 1 5
##
## $stripes
##  [1]  3  0  2  1  5  9 11 14  4  6 13  7
##
## $colors
## [1] 5 3 2 8 6 4 7 1
##
## $red
## [1] 1 0
##
## $green
## [1] 1 0
##
## $blue
## [1] 0 1
##
## $gold
## [1] 1 0
##
## $white
## [1] 1 0
##
## $black
## [1] 1 0
##
## $orange
## [1] 0 1
##
## $mainhue
## [1] "green"  "red"    "blue"   "gold"   "white"  "orange" "black"  "brown"
##
## $circles
## [1] 0 1 4 2
##
## $crosses
## [1] 0 1 2
##
## $saltires
## [1] 0 1
##
## $quarters
## [1] 0 1 4
##
## $sunstars
##  [1]  1  0  6 22 14  3  4  5 15 10  7  2  9 50
##
## $crescents
## [1] 0 1
##
## $triangle
## [1] 0 1
##
## $icon
```

```
## [1] 1 0
##
## $animate
## [1] 0 1
##
## $text
## [1] 0 1
##
## $topleft
## [1] "black"  "red"    "green"  "blue"   "white"  "orange" "gold"
##
## $botright
## [1] "green"  "red"    "white"  "black"  "blue"   "gold"   "orange" "brown"
```

```
lapply(unique_vals_list, function(elem) elem[2]) #using a custom function, return second element in the
```

```
## $country
## [1] "Albania"
##
## $landmass
## [1] 3
##
## $zone
## [1] 3
##
## $area
## [1] 29
##
## $population
## [1] 3
##
## $language
## [1] 6
##
## $religion
## [1] 6
##
## $bars
## [1] 2
##
## $stripes
## [1] 0
##
## $colors
## [1] 3
##
## $red
## [1] 0
##
## $green
## [1] 0
##
## $blue
## [1] 1
##
```

```
## $gold
## [1] 0
##
## $white
## [1] 0
##
## $black
## [1] 0
##
## $orange
## [1] 1
##
## $mainhue
## [1] "red"
##
## $circles
## [1] 1
##
## $crosses
## [1] 1
##
## $saltires
## [1] 1
##
## $quarters
## [1] 1
##
## $sunstars
## [1] 0
##
## $crescents
## [1] 1
##
## $triangle
## [1] 1
##
## $icon
## [1] 0
##
## $animate
## [1] 1
##
## $text
## [1] 1
##
## $topleft
## [1] "red"
##
## $botright
## [1] "red"
```

Various examples to learn tapply

tapply allows us to run a function on a dataset and group by a subset

```
populationbyreligion <- tapply(flags$population, flags$religion, sum) #returns sum of population by rel
populationbyreligion_and_lang <- tapply(flags$population, list(flags$religion, flags$language), sum) #g
```