Warm-up exercise

Kálmán Abari

2021-09-12

Data structures

Problems

Consider the following set of attributes about the American Film Institute's top-five movies ever from their 2007 list.

1. What code would you use to create a vector named `Movie` with the values `Citizen Kane`, `The Godfather`, `Casablanca`, `Raging Bull`, and `Singing in the Rain`? (Hints: `object <- c()`, Working with character in R)

```
# Solution ----
Movie <- c("Citizen Kane", "The Godfather", "Casablanca", "Raging Bull",
    "Singing in the Rain")
Movie
#> [1] "Citizen Kane"        "The Godfather"
#> [3] "Casablanca"          "Raging Bull"
#> [5] "Singing in the Rain"
```

2. What code would you use to create a vector — giving the year that the movies in Problem 1 were made — named `Year` with the values 1941, 1972, 1942, 1980, and 1952?

```
# Solution ----
Year <- c(1941, 1972, 1942, 1980, 1952)
Year
#> [1] 1941 1972 1942 1980 1952
```

3. What code would you use to create a vector — giving the run times in minutes of the movies in Problem 1 — named `RunTime` with the values 119, 177, 102, 129, and 103?

```
# Solution ----
RunTime <- c(119, 177, 102, 129, 103)
RunTime
#> [1] 119 177 102 129 103
```

4. What code would you use to find the run times of the movies in hours and save them in a vector called `RunTimeHours`? (Hints: Numeric tranformation)

```r
# Solution ----
RunTimeHours <- RunTime/60
RunTimeHours
#> [1] 1.983333 2.950000 1.700000 2.150000 1.716667
```

5. What code would you use to create a data frame named `MovieInfo` containing the vectors created in Problem 1, Problem 2, and Problem 3? (Hints: `data.frame()`)

```r
# Solution ----
MovieInfo <- data.frame(Movie, Year, RunTime)
MovieInfo
#>                  Movie Year RunTime
#> 1         Citizen Kane 1941     119
#> 2        The Godfather 1972     177
#> 3           Casablanca 1942     102
#> 4          Raging Bull 1980     129
#> 5 Singing in the Rain 1952     103
str(MovieInfo)
#> 'data.frame':    5 obs. of  3 variables:
#>  $ Movie  : chr  "Citizen Kane" "The Godfather" "Casablanca" "Raging Bull" ...
#>  $ Year   : num  1941 1972 1942 1980 1952
#>  $ RunTime: num  119 177 102 129 103
```

Manipulation

Problems

Suppose we have the following data frame named `colleges` (download here):

College	Employees	TopSalary	MedianSalary
William and Mary	2104	425000	56496
Christopher Newport	922	381486	47895
George Mason	4043	536714	63029
James Madison	2833	428400	53080
Longwood	746	328268	52000
Norfolk State	919	295000	49605
Old Dominion	2369	448272	54416
Radford	1273	312080	51000
Mary Washington	721	449865	53045
Virginia	7431	561099	60048
Virginia Commonwealth	5825	503154	55000
Virginia Military Institute	550	364269	44999

College	Employees	TopSalary	MedianSalary
Virginia Tech	7303	500000	51656
Virginia State	761	356524	55925

1. What code would you use to select the first, third, tenth, and twelfth entries in the `TopSalary` vector from the `Colleges` data frame? (Hints: Indexing with `[]` operator)

```
# Solution ----
library(rio)
colleges <- import(file = "data/colleges.xlsx")
str(colleges)
#> 'data.frame':	14 obs. of  4 variables:
#>  $ College     : chr  "William and Mary" "Christopher Newport" "George Mason" "James Madison" ...
#>  $ Employees   : num  2104 922 4043 2833 746 ...
#>  $ TopSalary   : num  425000 381486 536714 428400 328268 ...
#>  $ MedianSalary: num  56496 47895 63029 53080 52000 ...
colleges$College <- factor(colleges$College)  # convert to factor
colleges$TopSalary[c(1, 3, 10, 12)]
#> [1] 425000 536714 561099 364269
```

2. What code would you use to select the elements of the `MedianSalary` vector where the `TopSalary` is greater than $400,000? (Hints: `d$MedianSalary[d$TopSalary>400000]`)

```
# Solution ----
colleges$MedianSalary[colleges$TopSalary > 4e+05]
#> [1] 56496 63029 53080 54416 53045 60048 55000 51656
```

3. What code would you use to select the rows of the data frame for colleges with less than or equal to 1000 employees? (Hints: `d[condition, ]`)

```
# Solution ----
colleges[colleges$Employees <= 1000, ]
#>                        College Employees TopSalary
#> 2          Christopher Newport       922    381486
#> 5                     Longwood       746    328268
#> 6                Norfolk State       919    295000
#> 9              Mary Washington       721    449865
#> 12 Virginia Military Institute       550    364269
#> 14              Virginia State       761    356524
#>    MedianSalary
#> 2         47895
#> 5         52000
```

```
#> 6          49605
#> 9          53045
#> 12         44999
#> 14         55925
```

4. What code would you use to select a sample of 5 colleges from this data frame (there are 14 rows)? (Hints: `d[sample(x = 1:14, size = 5, replace = F),]`)

```
# Solution ----
colleges[sample(x = 1:14, size = 5, replace = F), ]
#>            College Employees TopSalary MedianSalary
#> 3    George Mason      4043    536714        63029
#> 8         Radford      1273    312080        51000
#> 13 Virginia Tech       7303    500000        51656
#> 4   James Madison      2833    428400        53080
#> 6   Norfolk State       919    295000        49605
```

Suppose we have the following data frame named Countries (download here):

Nation	Region	Population	PctIncrease	GDPcapita
China	Asia	1409517397	0.4	8582
India	Asia	1339180127	1.1	1852
United States	North America	324459463	0.7	57467
Indonesia	Asia	263991379	1.1	3895
Brazil	South America	209288278	0.8	10309
Pakistan	Asia	197015955	2.0	1629
Nigeria	Africa	190886311	2.6	2640
Bangladesh	Asia	164669751	1.1	1524
Russia	Europe	143989754	0.0	10248
Mexico	North America	129163276	1.3	8562

5. What could would you use to select the rows of the data frame that have GDP per capita less than 10000 and are not in the Asia region?

```
# Solution ----
library(rio)
Countries <- import(file = "data/countries.xlsx")
Countries$Region <- factor(Countries$Region)
Countries[Countries$GDPcapita < 10000 & !(Countries$Region %in%
    "Asia"), ]
#>       Nation        Region Population PctIncrease GDPcapita
```

```
#> 7  Nigeria        Africa  190886311         2.6       2640
#> 10  Mexico North America  129163276         1.3       8562
```

6. What code would you use to select a sample of three nations from this data frame (There are 10 rows)?

```
# Solution ----
Countries[sample(x = 1:10, size = 3, replace = F), ]
#>        Nation Region Population PctIncrease GDPcapita
#> 7     Nigeria Africa  190886311         2.6      2640
#> 8 Bangladesh   Asia  164669751         1.1      1524
#> 4  Indonesia   Asia  263991379         1.1      3895
```

7. What code would you use to select which nations saw a population percent increase greater that 1.5%?

```
# Solution ----
Countries[Countries$PctIncrease > 1.5, ]
#>      Nation Region Population PctIncrease GDPcapita
#> 6 Pakistan   Asia  197015955         2.0      1629
#> 7  Nigeria Africa  190886311         2.6      2640
```

Suppose we have the following data frame named Olympics (download here):

Year	Type	Host	Competitors	Events	Nations	Leader
1992	Summer	Spain	9356	257	169	Unified Team
1992	Winter	France	1801	57	64	Germany
1994	Winter	Norway	1737	61	67	Russia
1996	Summer	United States	10318	271	197	United States
1998	Winter	Japan	2176	68	72	Germany
2000	Summer	Australia	10651	300	199	United States
2002	Winter	United States	2399	78	78	Norway
2004	Summer	Greece	10625	301	201	United States
2006	Winter	Italy	2508	84	80	Germany
2008	Summer	China	10942	302	204	China
2010	Winter	Canada	2566	86	82	Canada
2012	Summer	United Kingdom	10768	302	204	United States
2014	Winter	Russia	2873	98	88	Russia
2016	Summer	Brazil	11238	306	207	United States
2018	Winter	South Korea	2922	102	92	Norway

8. What code would you use to select the rows of the data frame where the

host nation was also the medal leader?

```
# Solution ----
library(rio)
Olympics <- import(file = "data/olympics.xlsx")
Olympics$Type <- factor(Olympics$Type)
Olympics$Host <- factor(Olympics$Host)
Olympics$Leader <- factor(Olympics$Leader)
Olympics[as.character(Olympics$Host) == as.character(Olympics$Leader),
    ]
#>     Year   Type          Host Competitors Events Nations
#> 4  1996 Summer United States       10318    271     197
#> 10 2008 Summer         China       10942    302     204
#> 11 2010 Winter        Canada        2566     86      82
#> 13 2014 Winter        Russia        2873     98      88
#>           Leader
#> 4  United States
#> 10         China
#> 11        Canada
#> 13        Russia
```

9. What code would you use to select the rows of the data frame where the number of competitors per event is greater than 35?

```
# Solution ----
Olympics[Olympics$Competitors/Olympics$Events > 35, ]
#>     Year   Type           Host Competitors Events Nations
#> 1  1992 Summer          Spain        9356    257     169
#> 4  1996 Summer  United States       10318    271     197
#> 6  2000 Summer      Australia       10651    300     199
#> 8  2004 Summer         Greece       10625    301     201
#> 10 2008 Summer          China       10942    302     204
#> 12 2012 Summer United Kingdom       10768    302     204
#> 14 2016 Summer         Brazil       11238    306     207
#>           Leader
#> 1   Unified Team
#> 4  United States
#> 6  United States
#> 8  United States
#> 10         China
#> 12 United States
#> 14 United States
```

10. What code would you use to select the rows of the data frame where the number of competing nations in the Winter Olympics is at least 80?

```r
# Solution ----
Olympics[Olympics$Nations >= 80 & Olympics$Type == "Winter",
    ]
#>    Year   Type        Host Competitors Events Nations
#> 9  2006 Winter       Italy        2508     84      80
#> 11 2010 Winter      Canada        2566     86      82
#> 13 2014 Winter      Russia        2873     98      88
#> 15 2018 Winter South Korea        2922    102      92
#>     Leader
#> 9  Germany
#> 11  Canada
#> 13  Russia
#> 15  Norway
```

Packages

Problems

1. Install the **Ecdat** package. (Hints: `install.packages()`)

```r
# Solution ----
install.packages("Ecdat")
```

2. Say that we previously installed the **Ecdat** library into R and wanted to call the library to access datasets from it. What code would we use to call the library? (Hints: `library()`)

```r
# Solution ----
library("Ecdat")
```

3. Say that we then wanted to call the dataset `Diamond` from the **Ecdat** library. What code would we use to load this dataset into R? (Hints: `data()`)

```r
# Solution ----
data("Diamond")
str(Diamond)
#> 'data.frame':    308 obs. of  5 variables:
#>  $ carat        : num  0.3 0.3 0.3 0.3 0.31 0.31 0.31 0.31 0.31 0.31 ...
#>  $ colour       : Factor w/ 6 levels "D","E","F","G",..: 1 2 4 4 1 2 3 4 5 6 ...
#>  $ clarity      : Factor w/ 5 levels "IF","VS1","VS2",..: 3 2 4 2 2 2 2 5 3 2 ...
#>  $ certification: Factor w/ 3 levels "GIA","HRD","IGI": 1 1 1 1 1 1 1 1 1 1 ...
#>  $ price        : int  1302 1510 1510 1260 1641 1555 1427 1427 1126 1126 ...
```

Frequency and numerical exploratory analyses

Problems

Load the `leuk` dataset from the *MASS* library. This dataset is the survival times (`time`), white blood cell count (`wbc`), and the presence of a morphologic characteristic of white blood cells (`ag`).

1. Generate the frequency table for the presence of the morphologic characteristic.

```
# Solution ----
data("leuk", package = "MASS")
str(leuk)
#> 'data.frame':	33 obs. of  3 variables:
#>  $ wbc : int  2300 750 4300 2600 6000 10500 10000 17000 5400 7000 ...
#>  $ ag  : Factor w/ 2 levels "absent","present": 2 2 2 2 2 2 2 2 2 2 ...
#>  $ time: int  65 156 100 134 16 108 121 4 39 143 ...
table(leuk$ag)
#> 
#>  absent present 
#>      16      17
DescTools::Desc(leuk$ag, plotit = F, )
#> ---------------------------------------------------------- 
#> leuk$ag (factor - dichotomous)
#> 
#>   length      n    NAs unique
#>       33     33      0      2
#>          100.0%   0.0%       
#> 
#>          freq   perc  lci.95  uci.95'
#> absent     16  48.5%   32.5%   64.8%
#> present    17  51.5%   35.2%   67.5%
#> 
#> ' 95%-CI (Wilson)
```

2. Find the median and mean for survival time.

```
# Solution ----
median(leuk$time)
#> [1] 22
```

3. Find the range, IQR, variance, and standard deviation for white blood cell count.

```
# Solution ----
diff(range(leuk$wbc))   # range
#> [1] 99250
IQR(leuk$wbc)
#> [1] 26700
var(leuk$wbc)
#> [1] 1189517888
sd(leuk$wbc)
#> [1] 34489.39
```

4. Find the correlation between white blood cell count and survival time.

```
# Solution ----
cor(leuk$wbc, leuk$time)
#> [1] -0.3294525
```

Load the `survey` dataset from the *MASS* library. This dataset contains the survey responses of a class of college students.

5. Create the contingency table of whether or not the student smoked (`Smoke`) and the student's exercise regimen (`Exer`). (Hints: `table()`, `DescTools::Desc()`)

```
# Solution ----
data("survey", package = "MASS")
str(survey)
#> 'data.frame':    237 obs. of  12 variables:
#>  $ Sex   : Factor w/ 2 levels "Female","Male": 1 2 2 2 2 1 2 1 2 2 ...
#>  $ Wr.Hnd: num  18.5 19.5 18 18.8 20 18 17.7 17 20 18.5 ...
#>  $ NW.Hnd: num  18 20.5 13.3 18.9 20 17.7 17.7 17.3 19.5 18.5 ...
#>  $ W.Hnd : Factor w/ 2 levels "Left","Right": 2 1 2 2 2 2 2 2 2 2 ...
#>  $ Fold  : Factor w/ 3 levels "L on R","Neither",..: 3 3 1 3 2 1 1 3 3 3 ...
#>  $ Pulse : int  92 104 87 NA 35 64 83 74 72 90 ...
#>  $ Clap  : Factor w/ 3 levels "Left","Neither",..: 1 1 2 2 3 3 3 3 3 3 ...
#>  $ Exer  : Factor w/ 3 levels "Freq","None",..: 3 2 2 2 3 3 1 1 3 3 ...
#>  $ Smoke : Factor w/ 4 levels "Heavy","Never",..: 2 4 3 2 2 2 2 2 2 2 ...
#>  $ Height: num  173 178 NA 160 165 ...
#>  $ M.I   : Factor w/ 2 levels "Imperial","Metric": 2 1 NA 2 2 1 1 2 2 2 ...
#>  $ Age   : num  18.2 17.6 16.9 20.3 23.7 ...
# recode factor Smoke
levels(survey$Smoke)
#> [1] "Heavy" "Never" "Occas" "Regul"
survey$Smoke <- car::recode(survey$Smoke, "c(\"Heavy\",\"Occas\",\"Regul\")=\"Yes\";\"Never\"=\"No\"")
table(survey$Smoke, survey$Exer)
#>
```

```
#>       Freq None Some
#>   No     87   18   84
#>   Yes    28    5   14
DescTools::Desc(Smoke ~ Exer, data = survey, plotit = F, )
#> ------------------------------------------------------------
#> Smoke ~ Exer (survey)
#>
#> Summary:
#> n: 236, rows: 2, columns: 3
#>
#> Pearson's Chi-squared test:
#>   X-squared = 3.412, df = 2, p-value = 0.1816
#> Log likelihood ratio (G-test) test of independence:
#>   G = 3.5037, X-squared df = 2, p-value = 0.1735
#> Mantel-Haenszel Chi-squared:
#>   X-squared = 3.3215, df = 1, p-value = 0.06838
#>
#> Warning message:
#>   Exp. counts < 5: Chi-squared approx. may be incorrect!!
#>
#>
#> Phi-Coefficient         0.120
#> Contingency Coeff.      0.119
#> Cramer's V              0.120
#>
#>
#>         Exer    Freq   None   Some    Sum
#> Smoke
#>
#> No      freq      87     18     84    189
#>         perc   36.9%   7.6%  35.6%  80.1%
#>         p.row  46.0%   9.5%  44.4%      .
#>         p.col  75.7%  78.3%  85.7%      .
#>
#> Yes     freq      28      5     14     47
#>         perc   11.9%   2.1%   5.9%  19.9%
#>         p.row  59.6%  10.6%  29.8%      .
#>         p.col  24.3%  21.7%  14.3%      .
#>
#> Sum     freq     115     23     98    236
#>         perc   48.7%   9.7%  41.5% 100.0%
#>         p.row      .      .      .      .
#>         p.col      .      .      .      .
```

```
#>
```

6. Find the mean and median of the student's heart rate (`Pulse`). (Hints: `summary()`, `DescTools::Desc()`, `psych::describe()`)

```
# Solution ----
mean(survey$Pulse, na.rm = T)
#> [1] 74.15104
median(survey$Pulse, na.rm = T)
#> [1] 72.5
summary(survey$Pulse)
#>    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
#>   35.00   66.00   72.50   74.15   80.00  104.00      45
psych::describe(survey$Pulse)
#>    vars   n  mean    sd median trimmed   mad min max range
#> X1    1 192 74.15 11.69   72.5   74.02 11.12  35 104    69
#>     skew kurtosis   se
#> X1 -0.02     0.33 0.84
DescTools::Desc(survey$Pulse, plotit = F)
#> ----------------------------------------------------------
#> survey$Pulse (integer)
#>
#>   length      n    NAs  unique     0s   mean  meanCI'
#>      237    192     45      43      0  74.15   72.49
#>           81.0%  19.0%           0.0%          75.81
#>
#>      .05    .10    .25  median    .75    .90     .95
#>    59.55  60.00  66.00   72.50  80.00  90.00   92.00
#>
#>    range     sd  vcoef     mad    IQR   skew    kurt
#>    69.00  11.69   0.16   11.12  14.00  -0.02    0.33
#>
#> lowest : 35, 40, 48 (2), 50 (2), 54
#> highest: 96 (3), 97, 98, 100 (2), 104 (2)
#>
#> heap(?): remarkable frequency (9.4%) for the mode(s) (= 80)
#>
#> ' 95%-CI (classic)
```

7. Find the range, IQR, variance, and standard deviation for student age (`Age`).

```
# Solution ----
diff(range(survey$Age))  # range
#> [1] 56.25
```

```
IQR(survey$Age)
#> [1] 2.5
var(survey$Age)
#> [1] 41.91701
sd(survey$Age)
#> [1] 6.474335
```

8. Find the correlation between the span of the student's writing hand (`Wr.Hnd`) and nonwriting hand (`NW.Hnd`). (Hints: `cor()`, `DescTools::Desc()`)

```
# Solution ----
cor(survey$Wr.Hnd, survey$NW.Hnd, use = "complete.obs")
#> [1] 0.9483103
DescTools::Desc(Wr.Hnd ~ NW.Hnd, data = survey, plotit = F)
#> ------------------------------------------------------------
#> Wr.Hnd ~ NW.Hnd (survey)
#>
#> Summary:
#> n pairs: 237, valid: 236 (99.6%), missings: 1 (0.4%)
#>
#>
#> Pearson corr. : 0.948
#> Spearman corr.: 0.952
#> Kendall corr. : 0.842
```

Load the `Housing` dataset from the *Ecdat* library. This dataset looks at the variables that affect the sales price of houses.

9. Create the contingency table of whether or not the house has a recreation room (`recroom`) and whether or not the house had a full basement (`fullbase`).

```
# Solution ----
data("Housing", package = "Ecdat")
str(Housing)
#> 'data.frame':	546 obs. of  12 variables:
#>  $ price   : num  42000 38500 49500 60500 61000 66000 66000 69000 83800 88500 ...
#>  $ lotsize : num  5850 4000 3060 6650 6360 4160 3880 4160 4800 5500 ...
#>  $ bedrooms: num  3 2 3 3 2 3 3 3 3 3 ...
#>  $ bathrms : num  1 1 1 1 1 1 2 1 1 2 ...
#>  $ stories : num  2 1 1 2 1 1 2 3 1 4 ...
#>  $ driveway: Factor w/ 2 levels "no","yes": 2 2 2 2 2 2 2 2 2 2 ...
#>  $ recroom : Factor w/ 2 levels "no","yes": 1 1 1 2 1 2 1 1 2 2 ...
#>  $ fullbase: Factor w/ 2 levels "no","yes": 2 1 1 1 1 2 2 1 2 1 ...
```

```
#>  $ gashw   : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
#>  $ airco   : Factor w/ 2 levels "no","yes": 1 1 1 1 1 2 1 1 1 2 ...
#>  $ garagepl: num  1 0 0 0 0 0 2 0 0 1 ...
#>  $ prefarea: Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
table(Housing$recroom, Housing$fullbase)
#>
#>        no yes
#>   no  329 120
#>   yes  26  71
DescTools::Desc(recroom ~ fullbase, data = Housing, plotit = F,
    )
#> ------------------------------------------------------------
#> recroom ~ fullbase (Housing)
#>
#> Summary:
#> n: 546, rows: 2, columns: 2
#>
#> Pearson's Chi-squared test (cont. adj):
#>   X-squared = 73.705, df = 1, p-value < 2.2e-16
#> Fisher's exact test p-value < 2.2e-16
#> McNemar's chi-squared = 59.24, df = 1, p-value = 1.396e-14
#>
#>                     estimate lwr.ci upr.ci'
#>
#> odds ratio             7.487  4.561 12.289
#> rel. risk (col1)       2.734  1.958  3.816
#> rel. risk (col2)       0.365  0.300  0.444
#>
#>
#> Phi-Coefficient        0.372
#> Contingency Coeff.     0.349
#> Cramer's V             0.372
#>
#>
#>          fullbase     no    yes    Sum
#> recroom
#>
#> no       freq        329    120    449
#>          perc      60.3%  22.0%  82.2%
#>          p.row     73.3%  26.7%      .
#>          p.col     92.7%  62.8%      .
#>
#> yes      freq         26     71     97
```

```
#>            perc       4.8%  13.0%  17.8%
#>            p.row     26.8%  73.2%      .
#>            p.col      7.3%  37.2%      .
#>
#> Sum        freq        355    191    546
#>            perc      65.0%  35.0% 100.0%
#>            p.row         .      .      .
#>            p.col         .      .      .
#>
#>
#> ----------
#> ' 95% conf. level
```

10. Find the mean and median of the house's lot size (`lotsize`).

```
# Solution ----
mean(Housing$lotsize)
#> [1] 5150.266
median(Housing$lotsize)
#> [1] 4600
```

11. Find the range, IQR, variance, and standard deviation for the sales price (`price`).

```
# Solution ----
DescTools::Desc(Housing$price, plotit = F)
#> ------------------------------------------------------------
#> Housing$price (numeric)
#>
#>       length          n        NAs     unique         0s'
#>          546        546          0        219          0
#>                  100.0%       0.0%                  0.0%
#>
#>          .05        .10        .25     median        .75
#>    35'000.00  40'500.00  49'125.00  62'000.00  82'000.00
#>
#>        range         sd      vcoef        mad        IQR
#>   165'000.00  26'702.67       0.39  22'239.00  32'875.00
#>
#>         mean     meanCI
#>    68'121.60  65'876.83
#>               70'366.37
#>
#>          .90        .95
```

```
#>   105'000.00  120'000.00
#>
#>         skew        kurt
#>         1.20        1.91
#>
#> lowest : 25'000.0 (3), 25'245.0, 26'000.0, 26'500.0, 27'000.0 (2)
#> highest: 155'000.0, 163'000.0, 174'500.0, 175'000.0 (2), 190'000.0
#>
#> ' 95%-CI (classic)
```

12. Find the correlation between the sales price of the house (`price`) and the number of bedrooms (`bedrooms`).

```
# Solution ----
cor(Housing$price, Housing$bedrooms)
#> [1] 0.3664474
DescTools::Desc(price ~ bedrooms, data = Housing, plotit = F)
#> ----------------------------------------------------------
#> price ~ bedrooms (Housing)
#>
#> Summary:
#> n pairs: 546, valid: 546 (100.0%), missings: 0 (0.0%)
#>
#>
#> Pearson corr. : 0.366
#> Spearman corr.: 0.390
#> Kendall corr. : 0.307
```

Graphical exploratory analyses

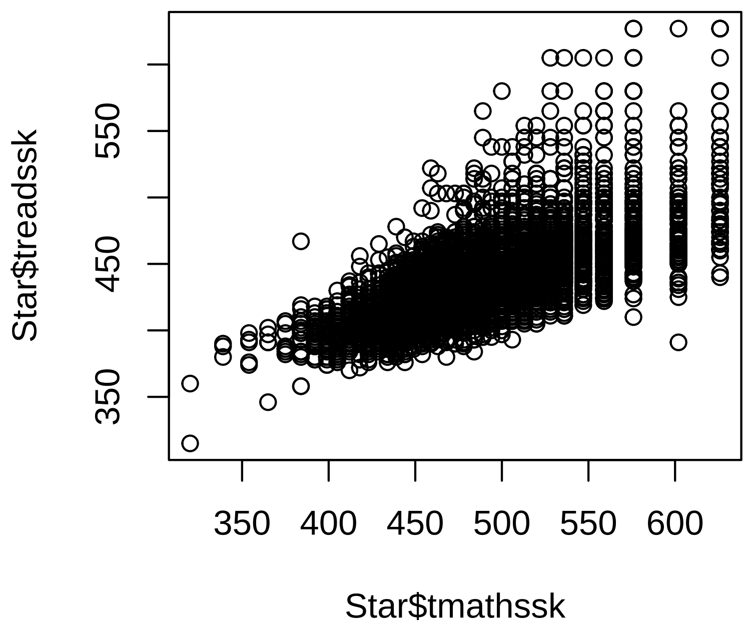
Load the `Star` dataset from the *Ecdat* library. This dataset looks at the affect on class sizes on student learning.

1. Generate the scatterplot of the student's math score `tmathssk` and reading score `treadssk`. (Hints: `plot()`, `ggplot() + geom_point()`)

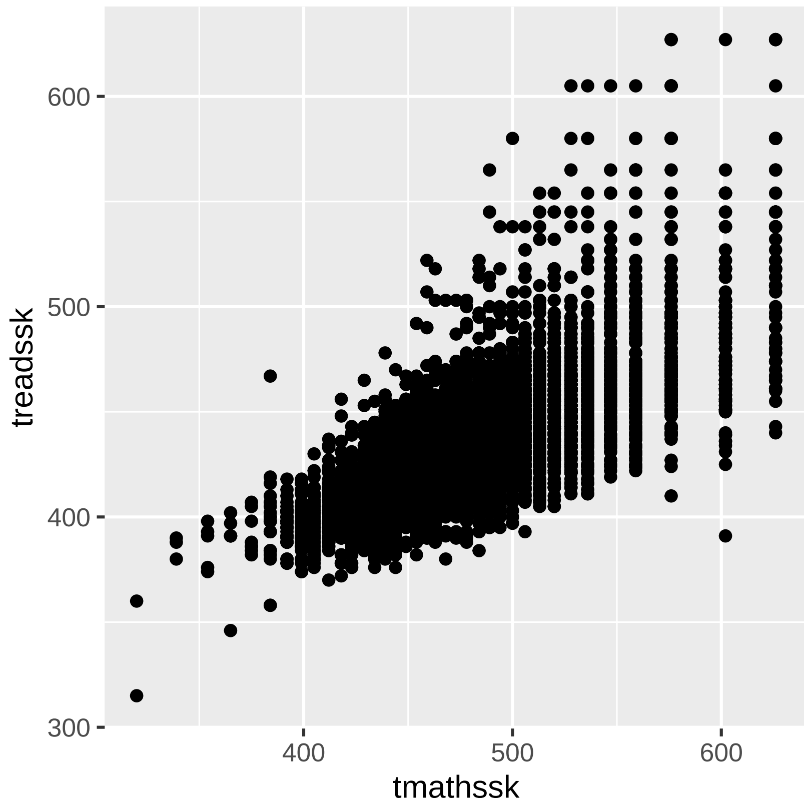
```
# Solution ----
data("Star", package = "Ecdat")
str(Star)
#> 'data.frame':    5748 obs. of  8 variables:
#>  $ tmathssk: int  473 536 463 559 489 454 423 500 439 528 ...
#>  $ treadssk: int  447 450 439 448 447 431 395 451 478 455 ...
#>  $ classk  : Factor w/ 3 levels "regular","small.class",..: 2 2 3 1 2 1 3 1 2 2 ...
#>  $ totexpk : int  7 21 0 16 5 8 17 3 11 10 ...
```

```
#>  $ sex     : Factor w/ 2 levels "girl","boy": 1 1 2 2 2 2 1 1 1 1 ...
#>  $ freelunk: Factor w/ 2 levels "no","yes": 1 1 2 1 2 2 2 1 1 1 ...
#>  $ race    : Factor w/ 3 levels "white","black",..: 1 2 2 1 1 1 2 1 2 1 ...
#>  $ schidkn : int  63 20 19 69 79 5 16 56 11 66 ...
#>  - attr(*, "na.action")= 'omit' Named int [1:5850] 1 4 6 7 8 9 10 15 16 17 ...
#>   ..- attr(*, "names")= chr [1:5850] "1" "4" "6" "7" ...
```

```
# Solution ----
plot(Star$tmathssk, Star$treadssk)
```

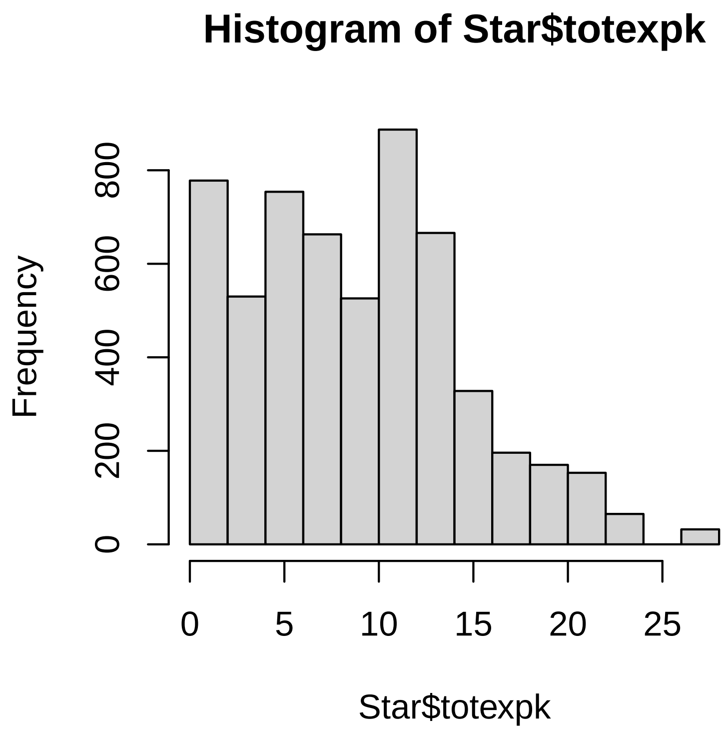



```
# Solution ----
library(ggplot2)
ggplot(data = Star, mapping = aes(x = tmathssk, y = treadssk)) +
    geom_point()
```

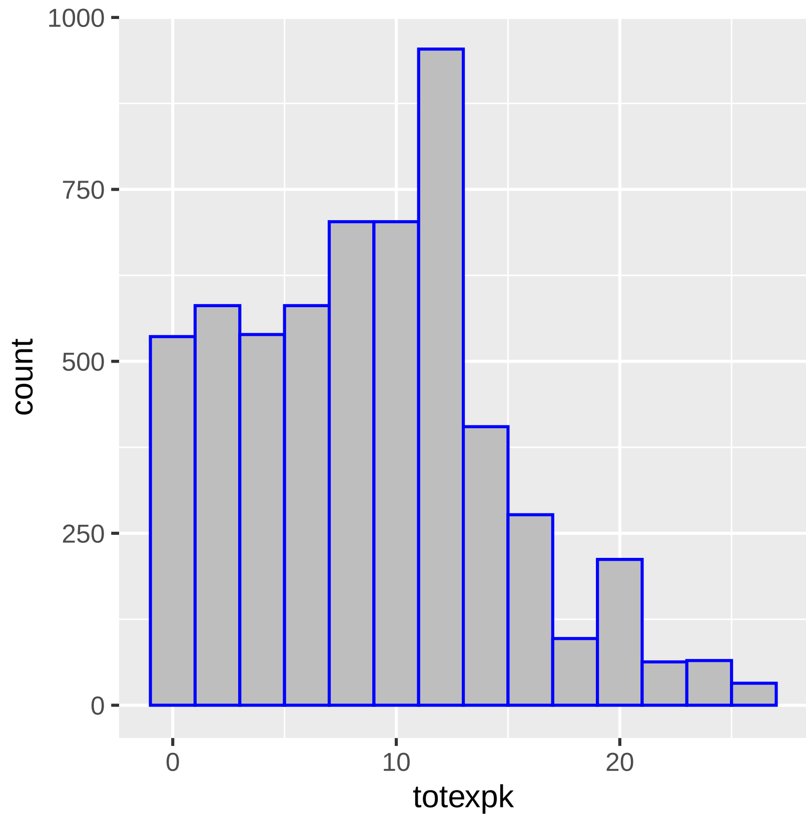


2. Generate the histogram of the years of teaching experience `totexpk`.
   (Hints: `hist()`, `ggplot() + geom_histogram()`)

```
# Solution ----
hist(Star$totexpk)
```




```
# Solution ----
library(ggplot2)
ggplot(data = Star, mapping = aes(x = totexpk)) + geom_histogram(binwidth = 2,
    fill = "grey", col = "blue")
```

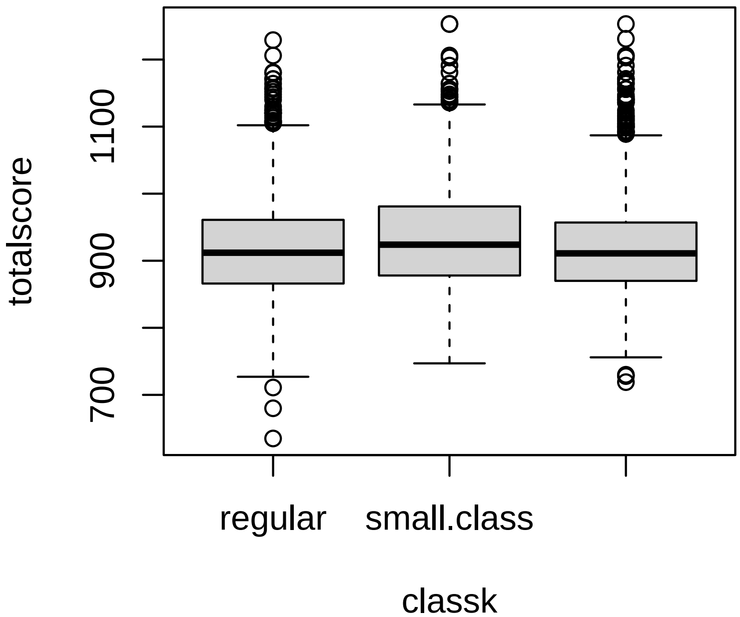


3. Create a new variable in the `Star` dataset called `totalscore` that is the sum of the student's math score `tmathssk` and reading score `treadssk`. (Hints: tranformation)

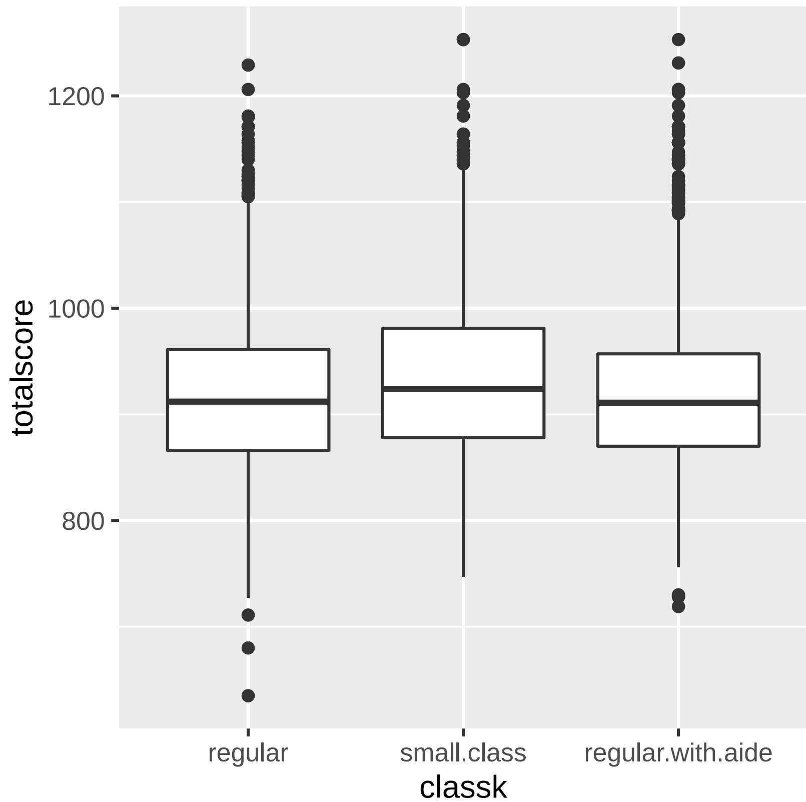
```
# Solution ----
Star$totalscore <- Star$tmathssk + Star$treadssk
```

4. Generate a boxplot of the student's total score `totalscore` split out by the class size type `classk`. (Hints: `boxplot()`, `ggplot() + geom_boxplot()`)

```
# Solution ----
boxplot(totalscore ~ classk, data = Star)
```



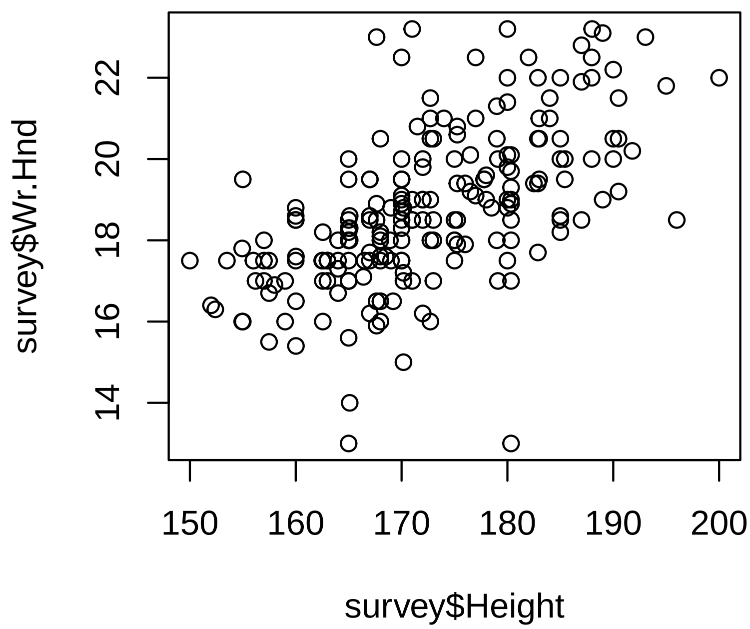

```
# Solution ----
library(ggplot2)
ggplot(data = Star, mapping = aes(x = classk, y = totalscore)) +
    geom_boxplot()
```



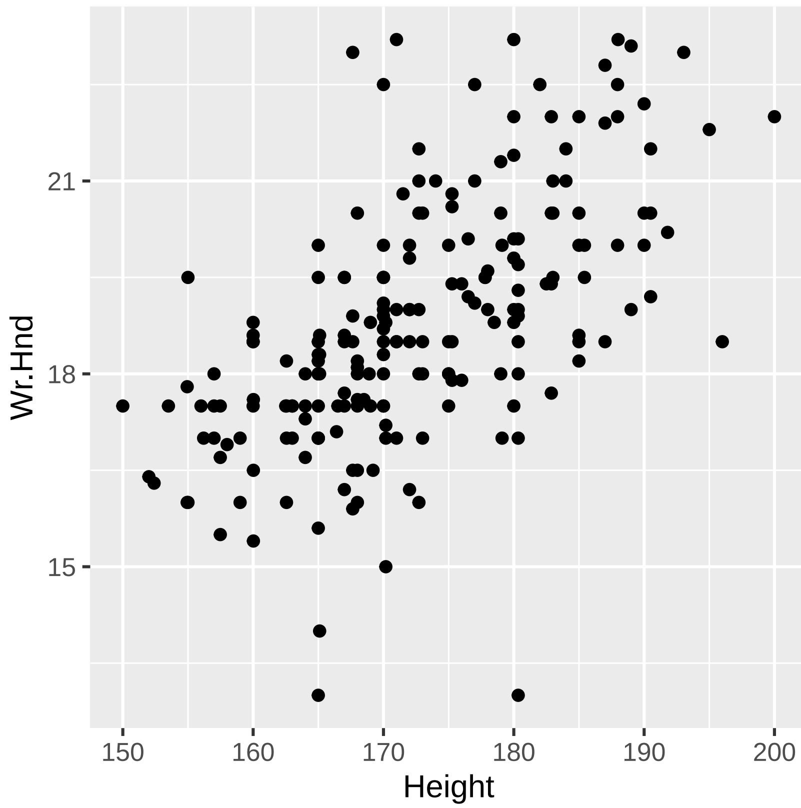
Load the `survey` dataset from the *MASS* library. This dataset contains the survey responses of a class of college students.

5. Generate the scatterplot of the student's height `Height` and writing hand span `Wr.Hnd`.

```
# Solution ----
plot(survey$Height, survey$Wr.Hnd)
```

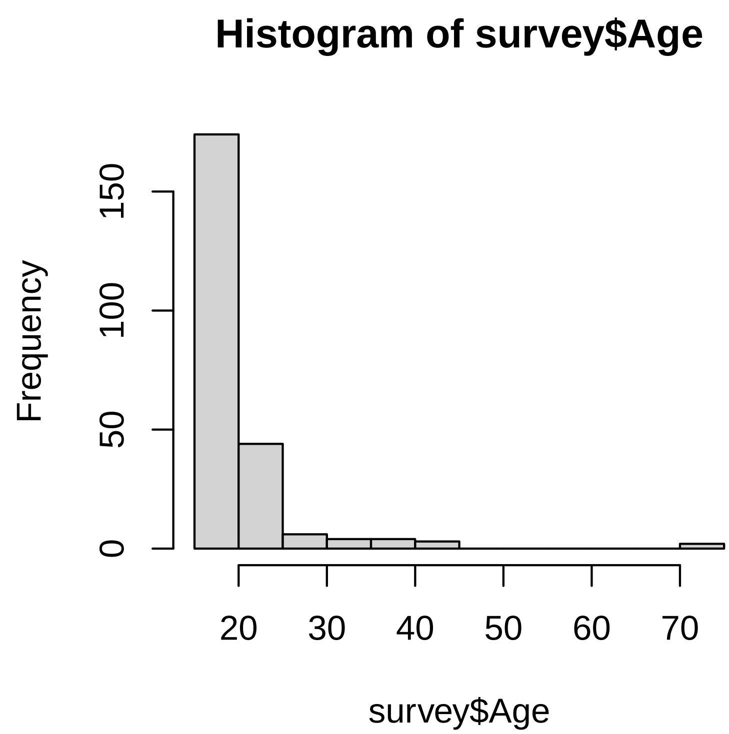



```
# Solution ----
library(ggplot2)
ggplot(data = survey, mapping = aes(x = Height, y = Wr.Hnd)) +
    geom_point()
```

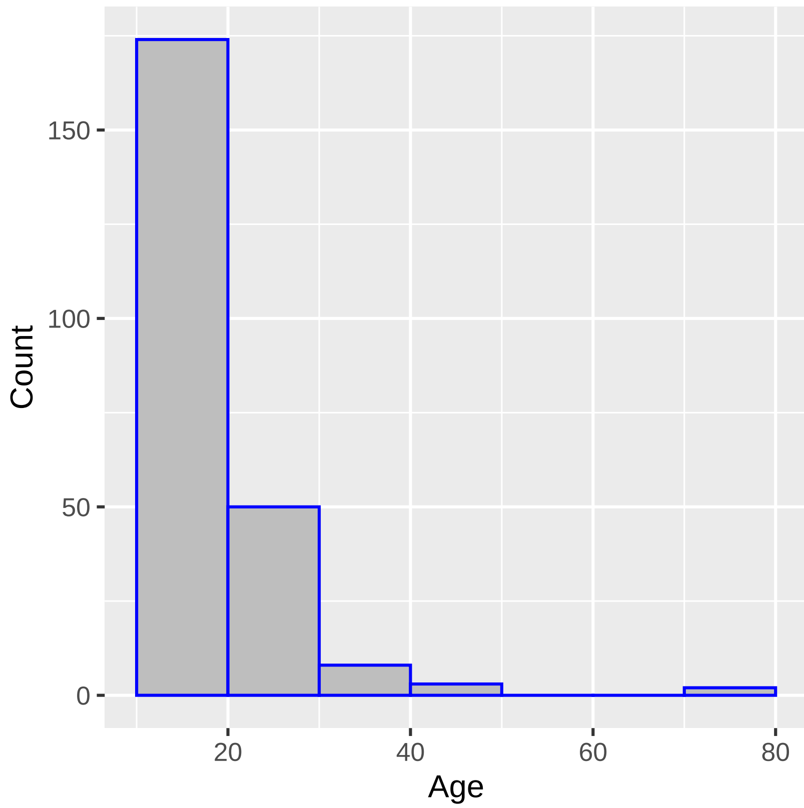


6. Generate the histogram of student age `Age`.

```
# Solution ----
hist(survey$Age)
```

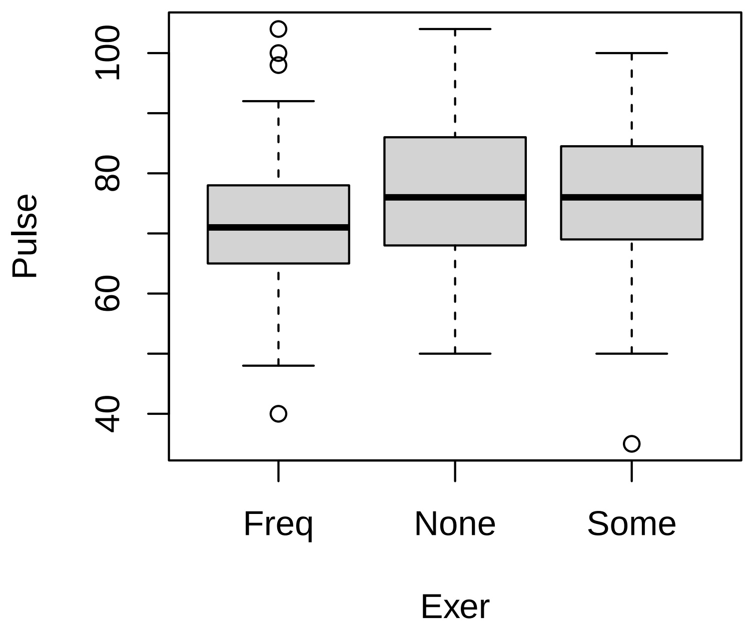



```
# Solution ----
library(ggplot2)
ggplot(data = survey, mapping = aes(x = Age)) + geom_histogram(binwidth = 10,
    fill = "grey", col = "blue", boundary = 10) + labs(x = "Age",
    y = "Count")
```

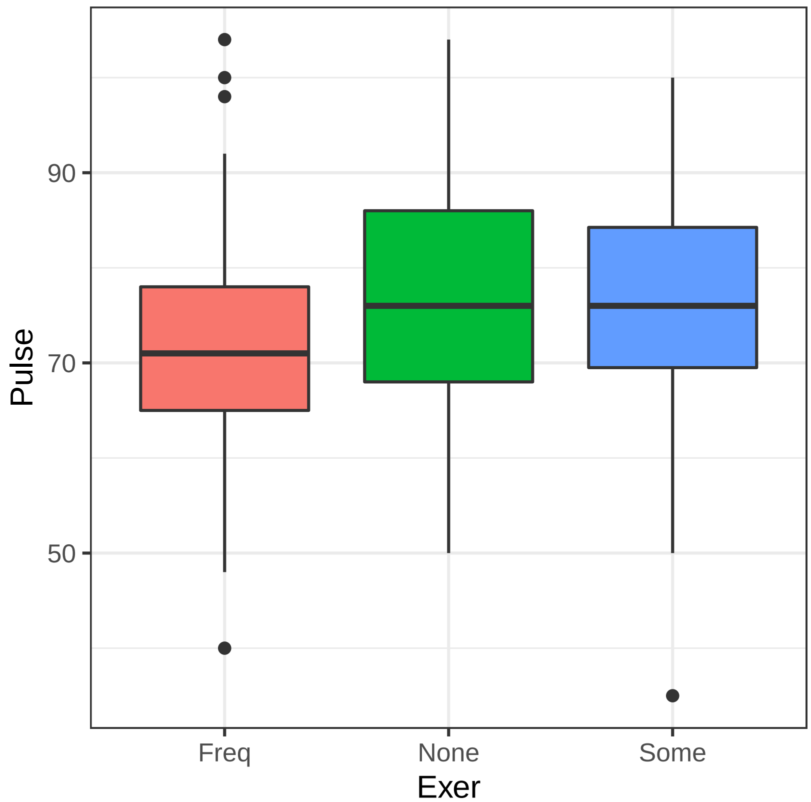


7. Generate a boxplot of the student's heart rate `Pulse` split out by the student's exercise regimen `Exer`.

```
# Solution ----
boxplot(Pulse ~ Exer, data = survey)
```




```r
# Solution ----
library(ggplot2)
ggplot(data = survey, mapping = aes(x = Exer, y = Pulse, fill = Exer)) +
    geom_boxplot() + theme_bw() + theme(legend.position = "none")
```


90
70
50
Pulse
Freq
None
Some
Exer