STOR 565: Homework 1

Brian N. White

1/10/2022

```
library(diagram)
```

```
## Loading required package: shape
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.5     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Exercise 1

```
x1 <- c(2, .5, 4., 2)
x2 <- c(x1, rep(1, 4))
x3 <- 1:-2
x4 <- c('Hello','', 'World', '!', 'Hello World!' )
x4
```

```
## [1] "Hello"        ""             "World"        "!"            "Hello World!"
```

Exercise 2

```
x0 <- c(1, 0, -1, 2)
x1 <- c(2, .5, 4, 2)


X <- rbind(1:4, x0, x1, rep(1, 4), deparse.level = 0)
X
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1  2.0    3    4
## [2,]    1  0.0   -1    2
## [3,]    2  0.5    4    2
## [4,]    1  1.0    1    1
```

Exercise 3

```
y1 <- X[X < 0]
y2 <- X[X > 0 & X < 2]

y1
```

```
## [1] -1
```

```
y2
```

```
## [1] 1.0 1.0 1.0 0.5 1.0 1.0 1.0
```

Exercise 4

```
students <- data.frame( id      = factor(c("001", "002", "003")),
                        score_A = c(95, 97, 90),
                        score_B = c(80, 75, 84))

students$id == '003'
```

```
## [1] FALSE FALSE  TRUE
```

```
students[students$id == '003', ]
```

```
##    id score_A score_B
## 3 003      90      84
```

Exercise 5

```
r0 <- c('Sun', 'Mon', 'Tues', 'Weds', 'Thurs', 'Fri', 'Sat')
r1 <- c('', 'NY', 2:6)
r2 <- 7:13
r3 <- c(14, 'MLK', 16:20)
r4 <- 21:27
r5 <- c(28:31, rep('', 3))

cal <- rbind(r0, r1, r2, r3, r4, r5)
cal <- as.data.frame(cal)
cal
```

```
##     V1  V2   V3   V4    V5  V6  V7
## r0 Sun Mon Tues Weds Thurs Fri Sat
## r1      NY    2    3     4   5   6
## r2   7   8    9   10    11  12  13
## r3  14 MLK   16   17    18  19  20
## r4  21  22   23   24    25  26  27
## r5  28  29   30   31
```

Exercise 6

```r
id        <- factor(rep(c("001","002","003"), 2))
subj      <- rep(c("A","B"), each = 3)
score     <- c(95, 97, 90, 80, 75, 84)
students3 <- data.frame(id, subj, score)

#using cut
students3 %>%
  mutate(grade = cut(score, c(0, 80, 90, 100), labels = c('C', 'B', 'A'), right = F))
```

```
##    id subj score grade
## 1 001    A    95     A
## 2 002    A    97     A
## 3 003    A    90     A
## 4 001    B    80     B
## 5 002    B    75     C
## 6 003    B    84     B
```

```r
#directly
students3 %>%
  mutate(grade = ifelse(score < 80, 'C', ifelse(score < 90, 'B', 'A')))
```

```
##    id subj score grade
## 1 001    A    95     A
## 2 002    A    97     A
## 3 003    A    90     A
## 4 001    B    80     B
## 5 002    B    75     C
## 6 003    B    84     B
```

Exercise 7

```r
mu <- apply(X, 2, mean)

X.cent <- t(X) - mu
X.var <- (1/3)*X.cent%*%t(X.cent)
X.var
```

```
##             [,1]       [,2]      [,3]        [,4]
## [1,]  0.25000000 -0.1250000 0.7500000 -0.08333333
## [2,] -0.12500000  0.7291667 0.9583333  0.70833333
## [3,]  0.75000000  0.9583333 4.9166667  1.08333333
## [4,] -0.08333333  0.7083333 1.0833333  1.58333333
```

```r
var(X)
```

```
##             [,1]       [,2]      [,3]        [,4]
## [1,]  0.25000000 -0.1250000 0.7500000 -0.08333333
## [2,] -0.12500000  0.7291667 0.9583333  0.70833333
## [3,]  0.75000000  0.9583333 4.9166667  1.08333333
## [4,] -0.08333333  0.7083333 1.0833333  1.58333333
```

Exercise 8

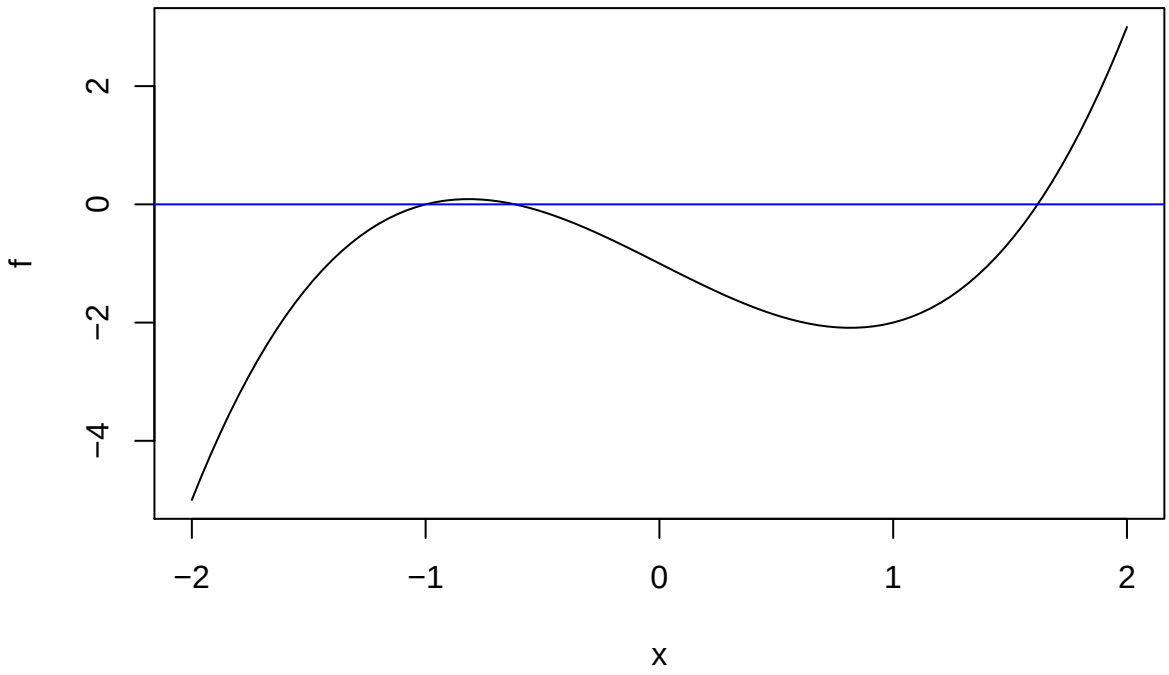
```
students3 %>%
  group_by(subj) %>%
  mutate(score.mean = mean(score))
```

```
## # A tibble: 6 x 4
## # Groups:   subj [2]
##   id    subj  score score.mean
##   <fct> <chr> <dbl>      <dbl>
## 1 001   A        95       94
## 2 002   A        97       94
## 3 003   A        90       94
## 4 001   B        80       79.7
## 5 002   B        75       79.7
## 6 003   B        84       79.7
```

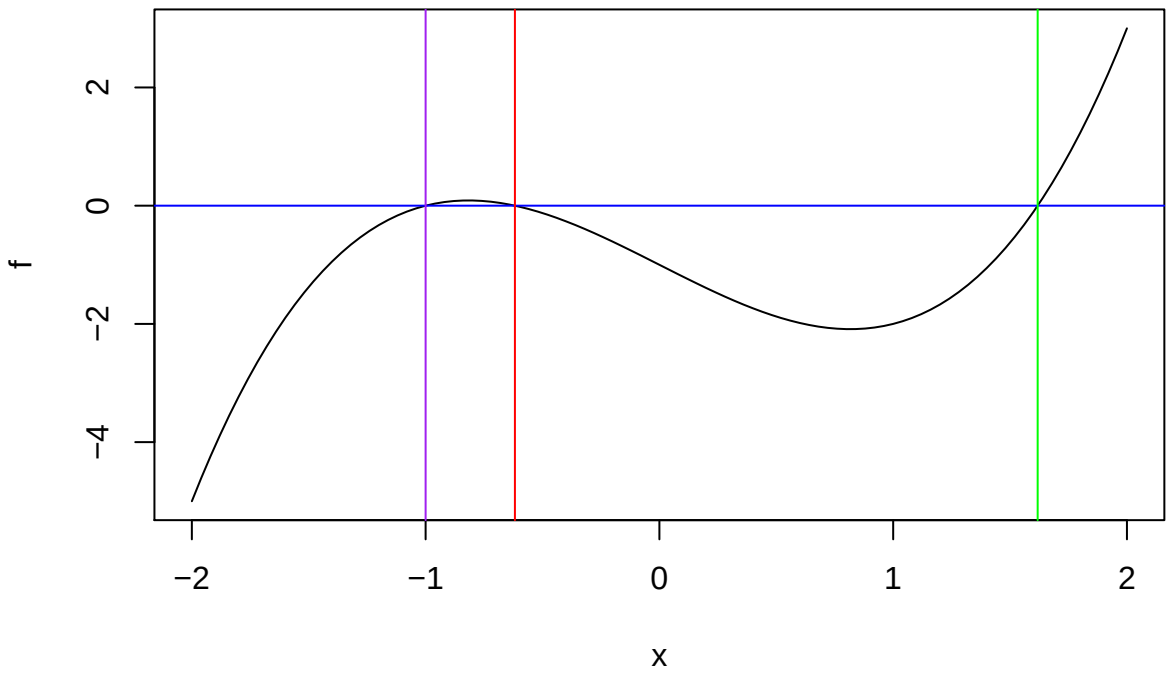
Exercise 9

```
bisect <- function(f, lower, upper, tol = 10^-6, maxit = 1000){
  #algorithm conditions
  if(lower >= upper ) {
    stop('lower must be strictly less than upper')
  }

  if(sign(f(lower)) == sign(f(upper))) {
    stop('f(lower) and f(upper) must have different signs)')
  }

  #begin algorithm

  #set counter
  i <- 1
  while(i <= maxit) {
    c <- (lower + upper)/2
    if(f(c) == 0 | (upper - lower)/2 < tol) {
      return(c)
      break
    }
    #increment counter
     i <- i + 1
    #modify lower and upper
      if(sign(f(c)) == sign(f(lower))) {
      lower <- c
    } else {
      upper <- c
    }
  }
  stop('method failed: maximum number of iterations reached')
}
```

```r
#function to test bisect on
f <- function(x) x^3 - 2*x - 1
plot(f, xlim = c(-2, 2))
abline(0, 0, col='blue')
```




```r
root1 <- bisect(f, 1, 2)
root2 <- bisect(f, -.8, 0)
root3 <- bisect(f, -2, -.9)
plot(f, xlim = c(-2, 2))
abline(0, 0, col='blue')

#confirm roots
abline(v = root1, col = 'green')
abline(v = root2, col= 'red')
abline(v = root3, col= 'purple')
```



Exercise 10

```
salaries <- read.csv('data/unc_salary_data.csv')

index <- str_which(salaries$dept, 'Stat*')

STOR <- salaries[index,]

STOR %>%
  select(name, age, totalsal)
```

```
##                         name age totalsal
## 309          ARGON, SUKRIYE N  40   100900
## 813   BHAMIDI, SREEKALYANI S  34    76800
## 1339    BUDHIRAJA, AMARJIT S  46   149402
## 1607       CARLSTEIN, EDWARD  56   125364
## 2400     CUNNINGHAM, ROBIN J  49    60000
## 2929         DUNN, CHARLES W  67    29000
## 4370             HANNIG, JAN  41   104800
## 5359            JI, CHUANSHU  65    92600
## 5706        KELLY, DOUGLAS G  74    70310
## 5766        KIEBER, ALISON J  52    40097
## 6029   KULKARNI, VIDYADHAR G  59   133700
## 6219   LEADBETTER, MALCOLM R  82   140800
## 6491             LIU, YUFENG  37   170000
## 6616                 LU, SHU  35    81800
```

```
## 6862         MARRON, JAMES S  59   158000
## 7090     MCDANIEL, DENNISE P  59    51466
## 8007         NOBEL, ANDREW B  51   130200
## 8371           PATAKI, GABOR  48    95600
## 8470      PENNINGTON, LORI A  51    36715
## 8649         PIPIRAS, VLADAS  39   106246
## 10146       SMITH, RICHARD L  61   176800
## 12044               XIA, YIN  27    86000
## 12225             ZHANG, KAI  33    78000
## 12269           ZIYA, SERHAN  39   116700
```

```
salaries %>%
  select(name, dept, totalsal) %>%
  filter(totalsal > 200000) %>%
  head()
```

```
##                 name                       dept totalsal
## 1     ADAMS, SASHA D                    Surgery   271000
## 2 ADAMSON, WILLIAM T                    Surgery   410000
## 3  ADIMORA, ADAORA A                   Medicine   230614
## 4   AHALT, STANLEY C Renaissance Computing Inst   257940
## 5    AKINTEMI, OLA B                 Pediatrics   205919
## 6   AKULIAN, JASON A                   Medicine   230000
```

```
salaries %>%
  select(name, dept, totalsal) %>%
  group_by(dept) %>%
  summarise(mean_salary = mean(totalsal)) %>%
  filter(grepl('Statistics*', dept))
```

```
## # A tibble: 1 x 2
##   dept                           mean_salary
##   <chr>                                <dbl>
## 1 Statistics and Operations Res     100471.
```

Exercise 11

```
data(iris)

iris %>%
  mutate(id = row_number()) %>%
  group_by(Species) %>%
  mutate(subset = cut(sample(row_number()), breaks = 5, labels = 1:5)) %>%
  group_by(subset) %>%
  group_split() -> iris_subsets

lapply(iris_subsets, head)
```

```
## [[1]]
## # A tibble: 6 x 7
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species    id subset
##          <dbl>       <dbl>        <dbl>       <dbl> <fct>   <int> <fct>
```

```
## 1          4.9         3            1.4         0.2 setosa      2 1
## 2          5.4         3.9          1.7         0.4 setosa      6 1
## 3          5           3.4          1.5         0.2 setosa      8 1
## 4          5.4         3.7          1.5         0.2 setosa     11 1
## 5          5.8         4            1.2         0.2 setosa     15 1
## 6          5.4         3.9          1.3         0.4 setosa     17 1
##
## [[2]]
## # A tibble: 6 x 7
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species    id subset
##          <dbl>       <dbl>        <dbl>       <dbl> <fct>   <int> <fct>
## 1          4.3         3            1.1         0.1 setosa     14 2
## 2          5.7         4.4          1.5         0.4 setosa     16 2
## 3          5.1         3.7          1.5         0.4 setosa     22 2
## 4          5.1         3.3          1.7         0.5 setosa     24 2
## 5          5           3.4          1.6         0.4 setosa     27 2
## 6          4.7         3.2          1.6         0.2 setosa     30 2
##
## [[3]]
## # A tibble: 6 x 7
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species    id subset
##          <dbl>       <dbl>        <dbl>       <dbl> <fct>   <int> <fct>
## 1          4.7         3.2          1.3         0.2 setosa      3 3
## 2          4.8         3            1.4         0.1 setosa     13 3
## 3          5.7         3.8          1.7         0.3 setosa     19 3
## 4          4.8         3.4          1.9         0.2 setosa     25 3
## 5          5.2         3.5          1.5         0.2 setosa     28 3
## 6          5.2         3.4          1.4         0.2 setosa     29 3
##
## [[4]]
## # A tibble: 6 x 7
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species    id subset
##          <dbl>       <dbl>        <dbl>       <dbl> <fct>   <int> <fct>
## 1          4.4         2.9          1.4         0.2 setosa      9 4
## 2          4.9         3.1          1.5         0.1 setosa     10 4
## 3          5.1         3.8          1.5         0.3 setosa     20 4
## 4          5.4         3.4          1.7         0.2 setosa     21 4
## 5          4.6         3.6          1           0.2 setosa     23 4
## 6          5           3            1.6         0.2 setosa     26 4
##
## [[5]]
## # A tibble: 6 x 7
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species    id subset
##          <dbl>       <dbl>        <dbl>       <dbl> <fct>   <int> <fct>
## 1          5.1         3.5          1.4         0.2 setosa      1 5
## 2          4.6         3.1          1.5         0.2 setosa      4 5
## 3          5           3.6          1.4         0.2 setosa      5 5
## 4          4.6         3.4          1.4         0.3 setosa      7 5
## 5          4.8         3.4          1.6         0.2 setosa     12 5
## 6          5.2         4.1          1.5         0.1 setosa     33 5
```

```
lapply(iris_subsets, function(x) x$id)
```

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

```
##  [1]   2   6   8  11  15  17  18  39  43  46  57  74  76  77  87  89  91  93  98
## [20] 100 101 104 115 116 128 131 138 140 149 150
##
## [[2]]
##  [1]  14  16  22  24  27  30  34  44  45  47  58  59  61  62  64  67  69  72  73
## [20]  83 107 108 110 113 117 121 123 136 137 139
##
## [[3]]
##  [1]   3  13  19  25  28  29  32  37  42  49  53  54  65  70  71  79  82  84  86
## [20]  94 103 109 114 118 119 127 132 135 145 148
##
## [[4]]
##  [1]   9  10  20  21  23  26  31  35  48  50  55  56  60  66  68  75  78  80  88
## [20]  95 102 105 111 120 122 124 125 133 134 146
##
## [[5]]
##  [1]   1   4   5   7  12  33  36  38  40  41  51  52  63  81  85  90  92  96  97
## [20]  99 106 112 126 129 130 141 142 143 144 147
```

```
lapply(iris_subsets, function(x) table(x$Species))
```

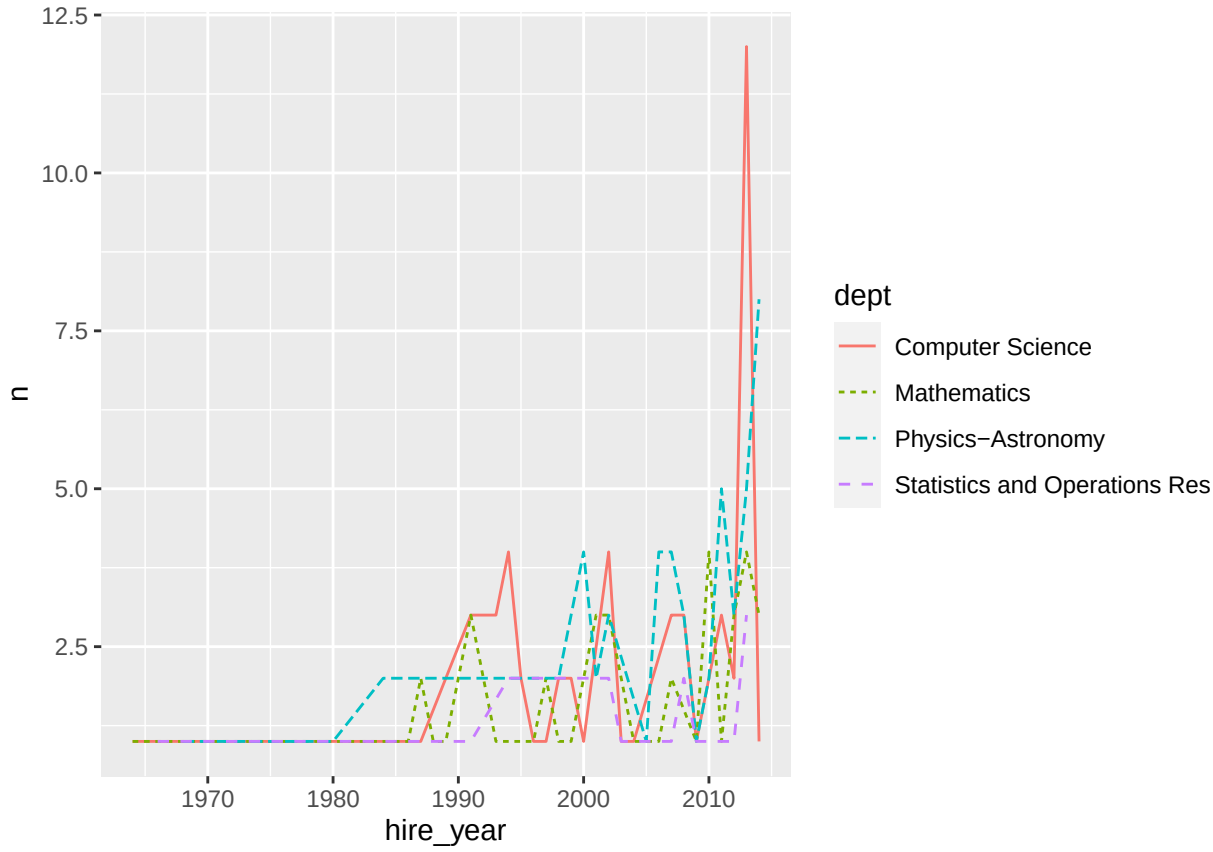
```
## [[1]]
##
##     setosa versicolor  virginica
##         10         10         10
##
## [[2]]
##
##     setosa versicolor  virginica
##         10         10         10
##
## [[3]]
##
##     setosa versicolor  virginica
##         10         10         10
##
## [[4]]
##
##     setosa versicolor  virginica
##         10         10         10
##
## [[5]]
##
##     setosa versicolor  virginica
##         10         10         10
```

Exercise 12

```
salaries %>%
  filter(dept %in% c('Computer Science', 'Mathematics', 'Statistics and Operations Res', 'Physics-Astro
  mutate(hire_year = as.numeric(str_sub(hiredate, 1, 4))) %>%
  group_by(hire_year, dept) %>%
  summarise(n = n()) %>%
```

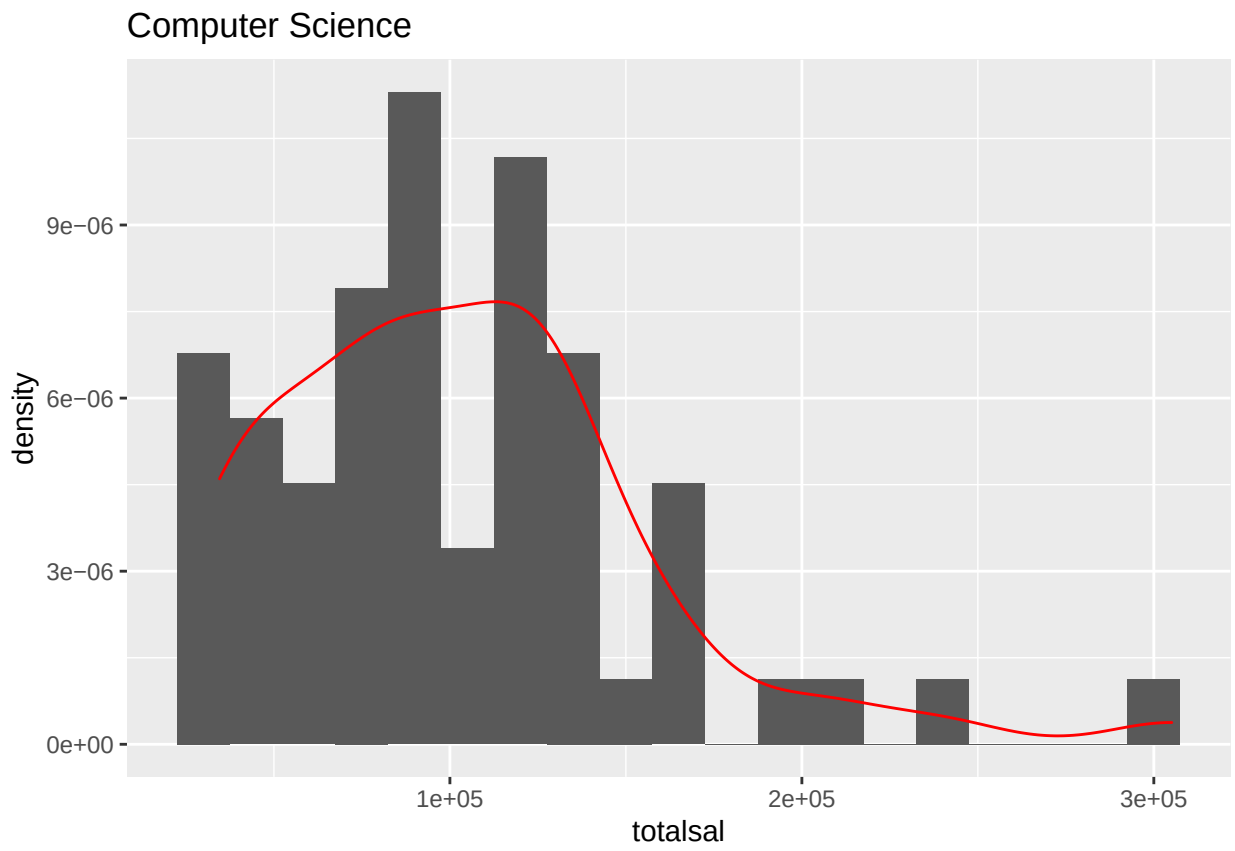
```
  ggplot(aes(hire_year, n, col = dept, linetype = dept)) +
  geom_line()
```

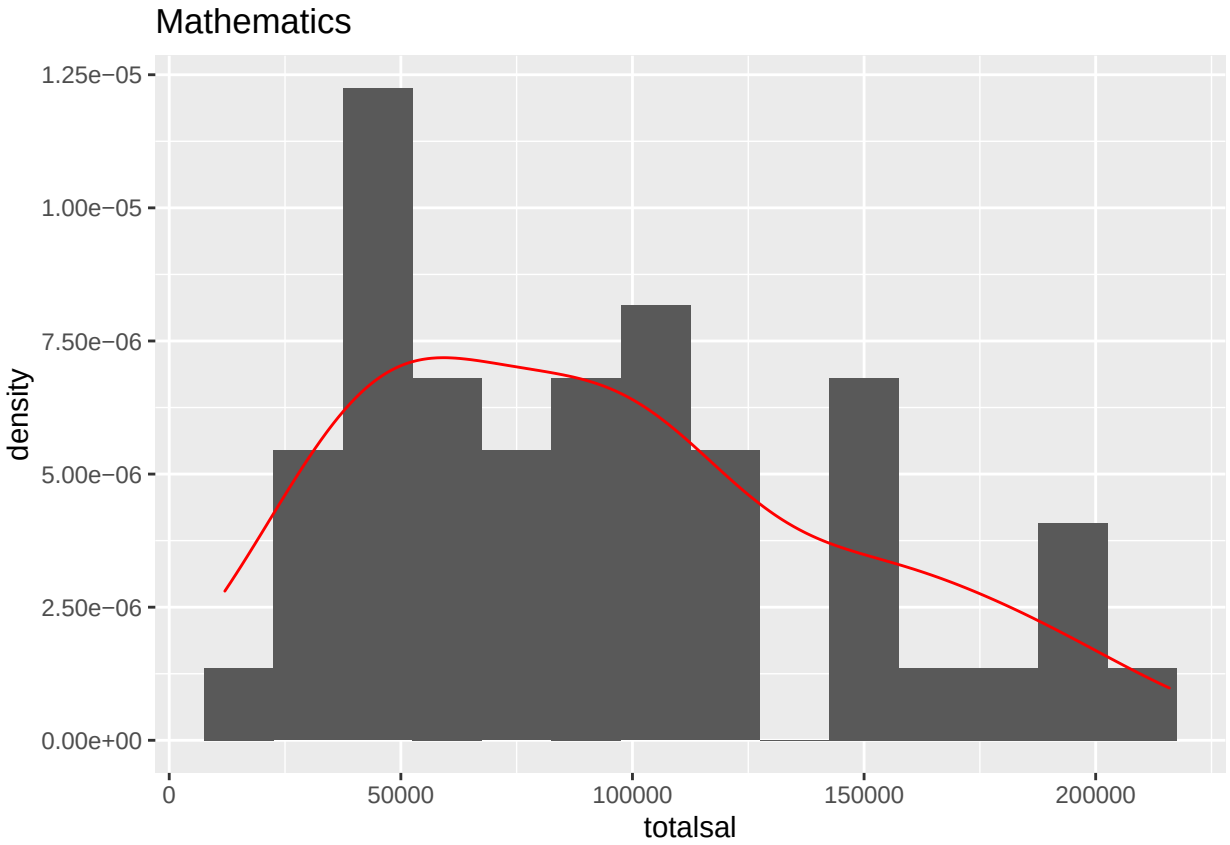
```
## `summarise()` has grouped output by 'hire_year'. You can override using the `.groups` argument.
```

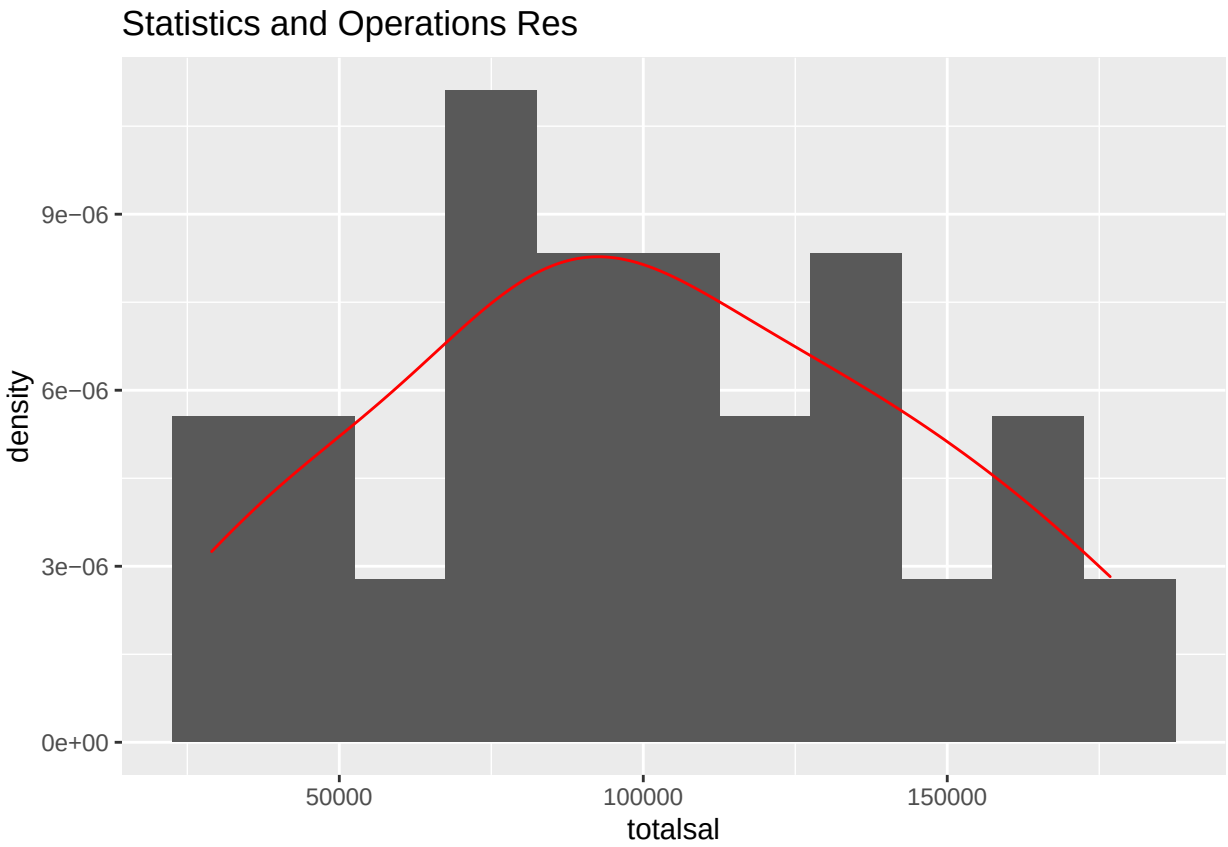


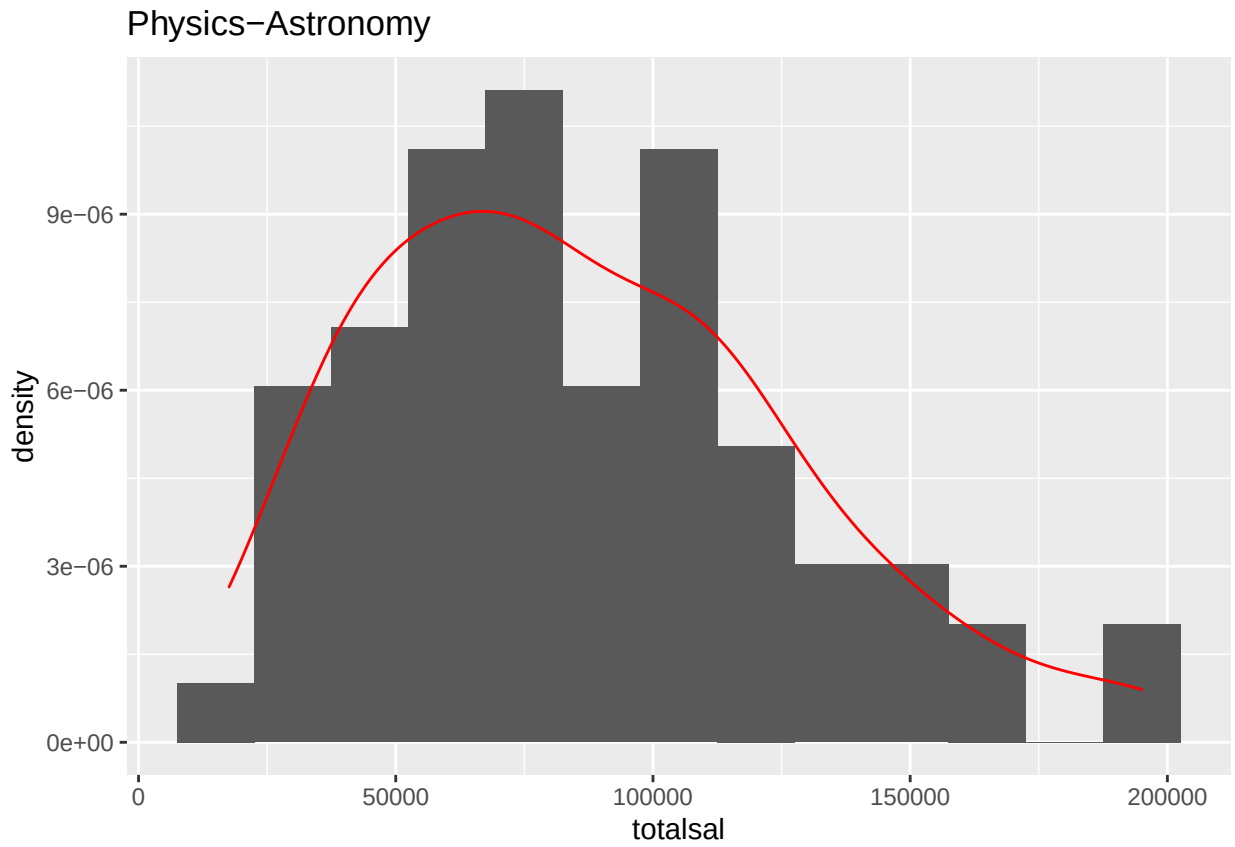
Exercise 13

```
index <- c('Computer Science', 'Mathematics', 'Statistics and Operations Res', 'Physics-Astronomy')

for(i in index){
print(salaries %>%
  filter(dept == i) %>%
  ggplot(aes(x = totalsal)) +
  labs(title = i) +
  geom_histogram(aes(y = ..density..), binwidth = 15000) +
  geom_density(col = 'red'))
}
```


Computer Science
density
9e−06
6e−06
3e−06
0e+00
1e+05
2e+05
3e+05
totalsal

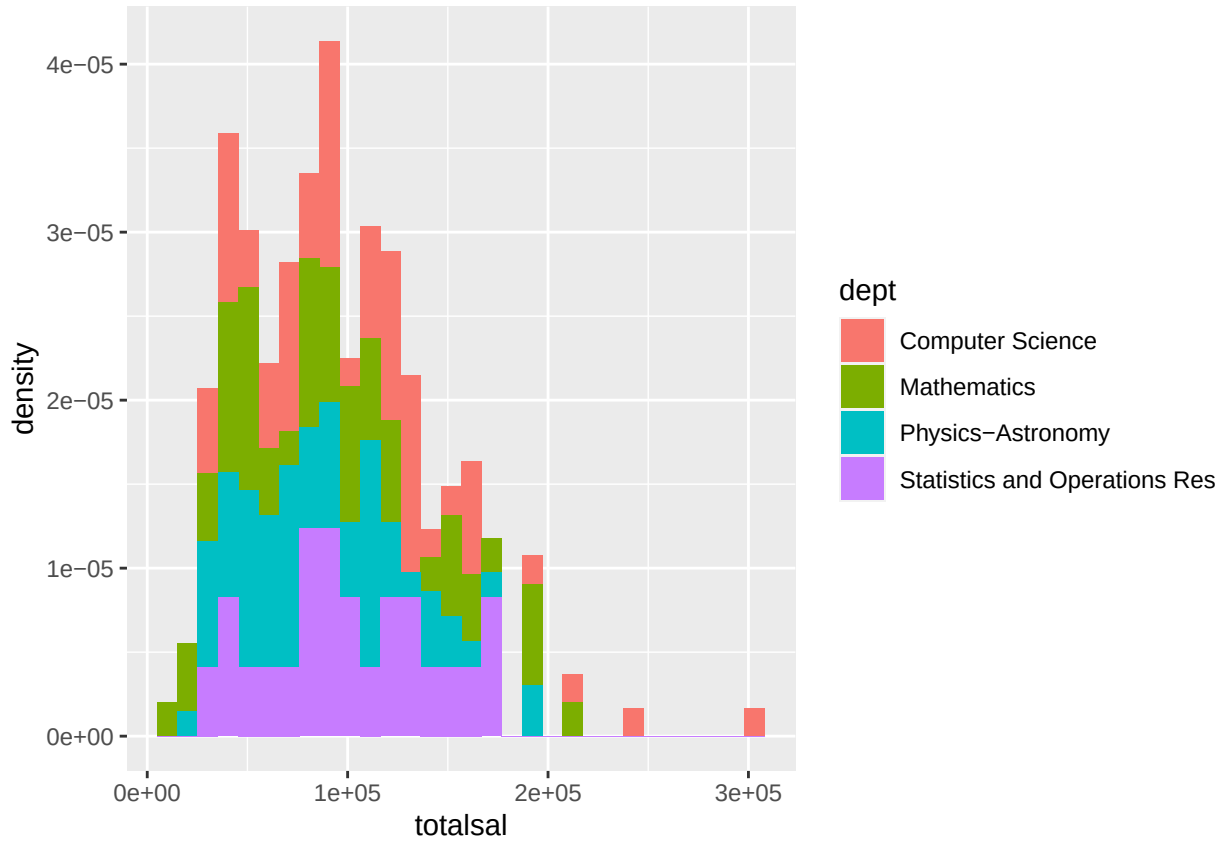

Mathematics
1.25e−05
1.00e−05
7.50e−06
5.00e−06
2.50e−06
0.00e+00
density
0
50000
100000
150000
200000
totalsal


Statistics and Operations Res
density
9e−06
6e−06
3e−06
0e+00
50000
100000
150000
totalsal

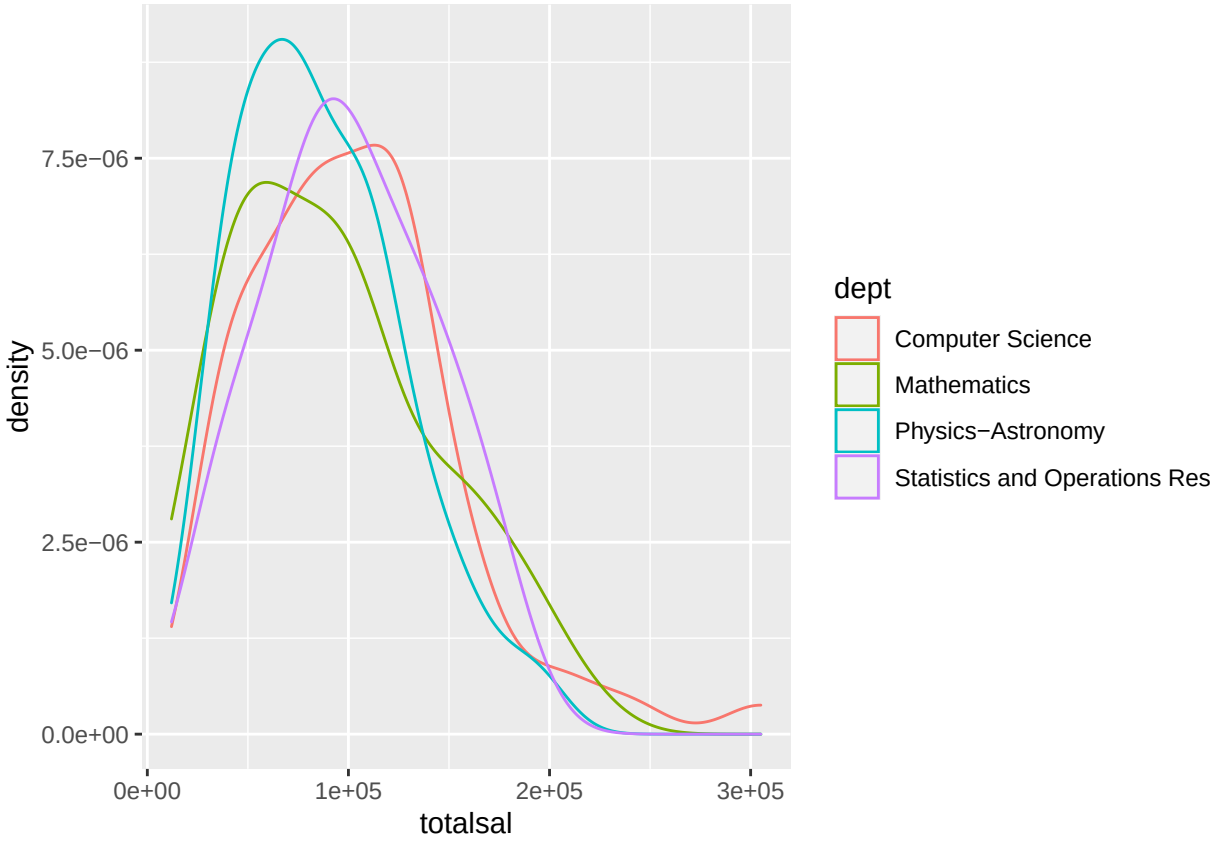



```
salaries %>%
  filter(dept %in% c('Computer Science', 'Mathematics', 'Statistics and Operations Res', 'Physics-Astro
  ggplot(aes(x = totalsal)) +
  geom_histogram(aes(y = ..density.., fill = dept))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
salaries %>%
  filter(dept %in% c('Computer Science', 'Mathematics', 'Statistics and Operations Res', 'Physics-Astro
  ggplot(aes(x = totalsal, col = dept)) +
  geom_density()
```


density
7.5e−06
5.0e−06
2.5e−06
0.0e+00
0e+00
1e+05
2e+05
3e+05
totalsal
dept
Computer Science
Mathematics
Physics−Astronomy
Statistics and Operations Res