PSTAT 10 — Homework 2

Tejas Patel

Instructions

- Knit to **PDF**(ensure the tinytex package is installed) and upload to Canvas by **Friday, October 17**
- Note these are some **practice problems** for the Mid-term exam 1.

Question 1 — Weather mini-dataset (10 pts)

```r
temps = c(71, 72, 75, 64, 73, 83, 63, 64, 70, 85, 95, 104)
Tmat = matrix(temps, nrow=4,ncol=3,byrow=TRUE)
rownames(Tmat)= c("Santa_Barbara","Reno","Seattle","Phoenix")
colnames(Tmat) = c("Apr","May","Jun")
print(Tmat)
```

```
              Apr May Jun
Santa_Barbara  71  72  75
Reno           64  73  83
Seattle        63  64  70
Phoenix        85  95 104
```

```r
print(Tmat["Reno","Jun"])
```

```
[1] 83
```

```r
print(Tmat["Seattle",1:3])
```

```
Apr May Jun
 63  64  70
```

```
Tmat[Tmat >= 74] = NA
```

It works because it is searching through the matrix for every element above 74 and if it finds a match it turns it into NA

```
rowMeans(Tmat)
```

```
Santa_Barbara          Reno       Seattle       Phoenix 
           NA            NA      65.66667            NA 
```

```
colMeans(Tmat)
```

```
Apr May Jun 
 NA  NA  NA 
```

Question 2 — Inventory binding & dimensions

```
inventory_A = matrix(c(12,8,5,20,0,7),nrow=2, ncol=3, byrow=TRUE)
rownames(inventory_A)=c("Store_A", "Store_B")
inventory_B = matrix(c(6,15,9,1,11,13),nrow=2, ncol=3, byrow=TRUE)
rownames(inventory_B)=c("Store_A", "Store_B")
dim(inventory_A)
```

```
[1] 2 3
```

```
dim(inventory_B)
```

```
[1] 2 3
```

```
inventory_all = rbind(inventory_A, inventory_B)
inventory_expanded = rbind(inventory_all,c(9,4,2))
print(t(inventory_all))
```

```
     Store_A Store_B Store_A Store_B
[1,]      12      20       6       1
[2,]       8       0      15      11
[3,]       5       7       9      13
```

```
print(dim(t(inventory_all)))
```

```
[1] 3 4
```

Question 3 — Course quiz scores

```
scores = matrix(c(9,8,10,7,9,9,6,10,8), nrow=3,ncol=3,byrow = TRUE)
quiz_names=rownames(scores) = c("quiz 1","quiz 2","quiz 3")
colnames(scores)=c("Liam", "Mia","Zoe")
paste("Liam: ", scores["quiz 2", "Liam"], "Zoe: ", scores["quiz 2", "Zoe"])
```

```
[1] "Liam:  7 Zoe:  9"
```

```
for (i in 1:nrow(scores)){
  if (scores[i, "Mia"] >= 9){
    print(quiz_names[i])
  }
}
```

```
[1] "quiz 2"
[1] "quiz 3"
```

```
replace(scores, scores==10, 99)
```

```
       Liam Mia Zoe
quiz 1    9   8  99
quiz 2    7   9   9
quiz 3    6  99   8
```

```
scores = scores[,-1]
print(scores)
```

```
       Mia Zoe
quiz 1   8  10
quiz 2   9   9
quiz 3  10   8
```

Question 4 — 3D arrays: Three-day sensor readings

```r
co2=array(201:236, dim=c(3,4,3))
dimnames(co2)=list(c("room1","room2", "room3"), c("h1","h2","h3","h4"), c("day1","day2","day3
replace(co2, co2["day1"]>212, 212)
```

```
, , day1

       h1  h2  h3  h4
room1 201 204 207 210
room2 202 205 208 211
room3 203 206 209 212

, , day2

       h1  h2  h3  h4
room1 213 216 219 222
room2 214 217 220 223
room3 215 218 221 224

, , day3

       h1  h2  h3  h4
room1 225 228 231 234
room2 226 229 232 235
room3 227 230 233 236
```

```r
print(co2[,,"day2"])
```

```
       h1  h2  h3  h4
room1 213 216 219 222
room2 214 217 220 223
room3 215 218 221 224
```

```r
print(co2[,"h3",])
```

```
      day1 day2 day3
room1  207  219  231
room2  208  220  232
room3  209  221  233
```

```r
print(co2["room3","h4","day1"])
```

```
[1] 212
```

```r
apply(co2[,,"day1"], MARGIN = 1, mean)
```

```
room1 room2 room3 
205.5 206.5 207.5 
```

```r
apply(co2[,,"day2"], MARGIN = 1, mean)
```

```
room1 room2 room3 
217.5 218.5 219.5 
```

```r
apply(co2[,,"day3"], MARGIN = 1, mean)
```

```
room1 room2 room3 
229.5 230.5 231.5 
```

Question 5 — Project info as a list

```r
proj = list(title = "Campus Air Study", year = 2025, n_sensors = 12, lead = "J. Doe", hours =
hours = c("8am","11am","2pm","5pm")
lead = list(first = "John", last = "Doe", email = "jdoe@gmail.com")
status = "pilot"
proj["lead"] = list(lead)
proj = list(status = "pilot", proj)
print(proj[2]$year)
```

```
NULL
```

```r
print(proj[[2]]$year)
```

```
[1] 2025
```

```r
print(proj[[2]]$hours[length(hours)-1:length(hours)])
```

```
[1] "2pm"  "11am" "8am"
```

The difference is the amount of subsetting you're asking R to do. With just one it'll extract a sub-list. With two, itll actually pull the content of the element.

Question 6 - Working with Data Frames state.x77 is a double precision floating point matrix

```r
typeof(state.x77)
```

```
[1] "double"
```

```r
is.matrix(state.x77)
```

```
[1] TRUE
```

```r
df1= data.frame(state.x77)
print(length((df1$Income < 4500)[(df1$Income < 4500)== TRUE]))
```

```
[1] 24
```

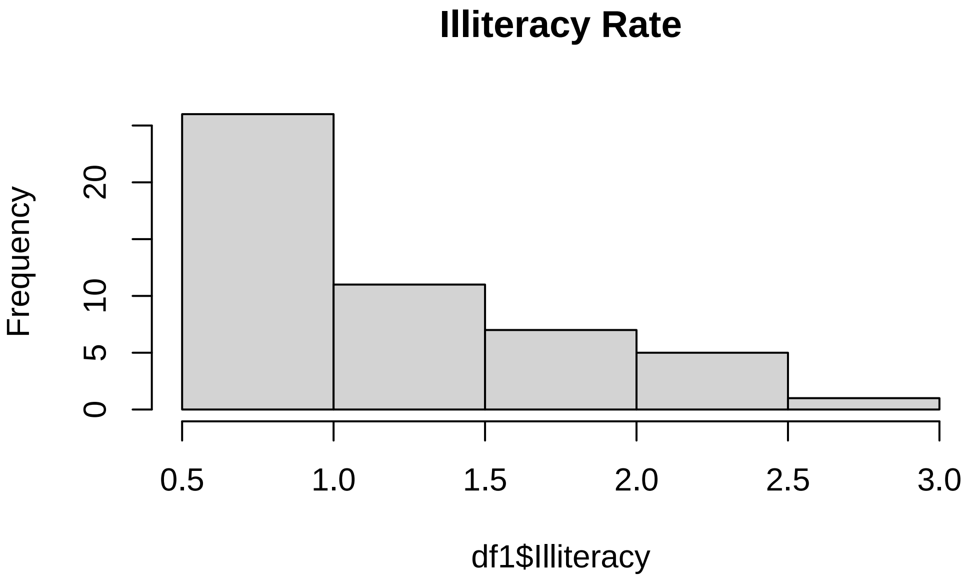
```r
state_names <- if ("State" %in% names(df1)) as.character(df1$State[which(df1$Income < 4500)])
print(state_names)
```

```
 [1] "Alabama"        "Arkansas"       "Georgia"        "Idaho"
 [5] "Indiana"        "Kentucky"       "Louisiana"      "Maine"
 [9] "Mississippi"    "Missouri"       "Montana"        "New Hampshire"
[13] "New Mexico"     "North Carolina" "Oklahoma"       "Pennsylvania"
[17] "South Carolina" "South Dakota"   "Tennessee"      "Texas"
[21] "Utah"           "Vermont"        "West Virginia"  "Wisconsin"
```

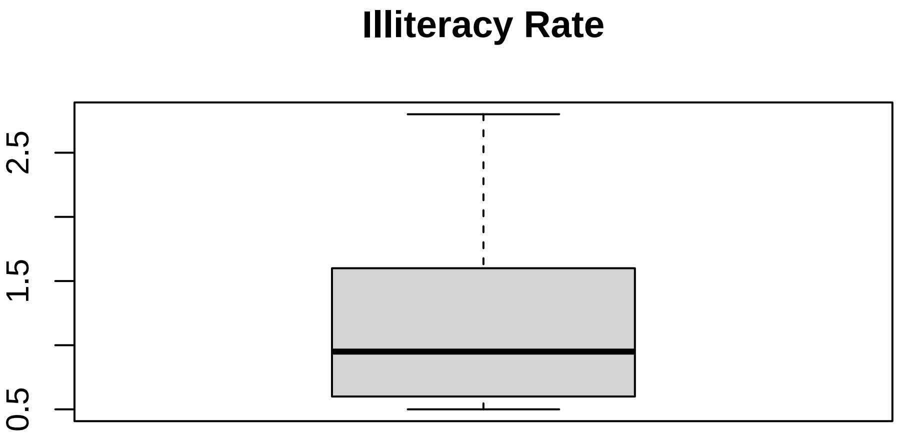
```r
df1$loglifeExp = log(df1$Life.Exp)
print(df1$loglifeExp)
```

```
 [1] 4.234831 4.238589 4.256322 4.257880 4.272630 4.277499 4.283311 4.249352
 [9] 4.257880 4.227418 4.298645 4.274859 4.250493 4.260988 4.284414 4.284689
[17] 4.249923 4.230622 4.254051 4.251633 4.274302 4.257455 4.289911 4.220830
[25] 4.258304 4.256463 4.284965 4.234541 4.265914 4.261693 4.253056 4.256322
[33] 4.237145 4.287441 4.260141 4.268578 4.278470 4.254619 4.275276 4.218919
[41] 4.277777 4.250065 4.261270 4.289089 4.271654 4.249637 4.272770 4.241039
[49] 4.283311 4.252630
```

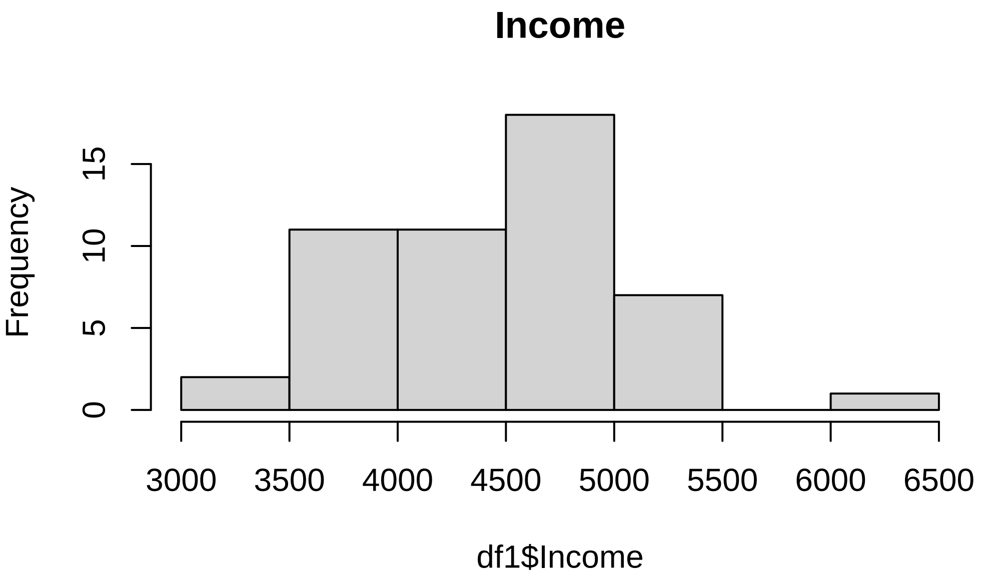
```
hist(df1$Illiteracy, main = "Illiteracy Rate")
```



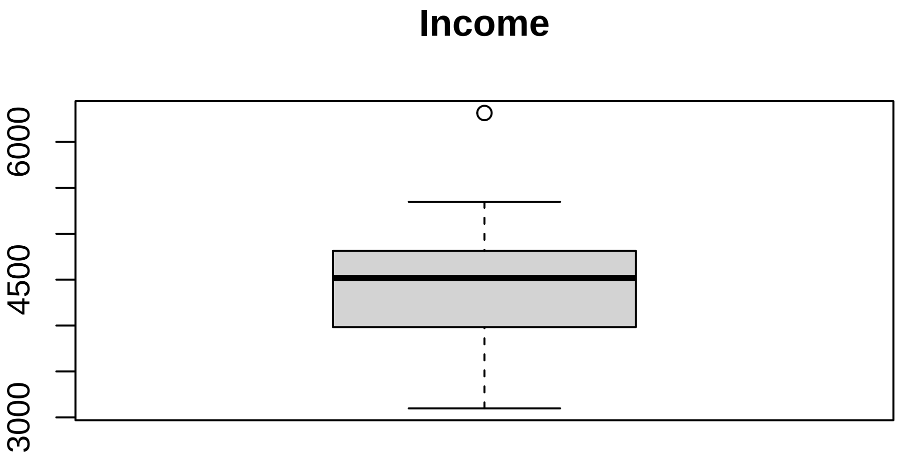
```
boxplot(df1$Illiteracy, main = "Illiteracy Rate")
```



```r
hist(df1$Income, main = "Income")
```



```r
boxplot(df1$Income, main = "Income")
```


Income