

Bios 6301: Assignment 8

Jonathan Lifferth

25/30 good!

Due Tuesday, 14 November, 1:00 PM

$5^{n=\text{day}}$ points taken off for each day late.

30 points total.

Submit a single knitr file (named `homework8.rmd`), along with a valid PDF output file. Inside the file, clearly indicate which parts of your responses go with which problems (you may use the original homework document as a template). Add your name as `author` to the file's metadata section. Raw R code/output or word processor files are not acceptable.

Failure to name file `homework8.rmd` or include author name may result in 5 points taken off.

Question 1

15 points

Install the `readxl` package and run the following

```
# install.packages('readxl')
fn <- 'icd10.xlsx'
if(file.access(fn, mode = 4) == -1) {
  url <- "https://www.cdc.gov/nhsn/xls/icd10-pcs-pcm-nhsn-opc.xlsx"
  download.file(url, destfile = fn, mode = 'wb')
}
dat <- readxl::read_excel(fn, sheet = 2)
```

1. Show the class of `dat`. (1 point)

```
class(dat)
```

```
## [1] "tbl_df"      "tbl"        "data.frame"
```

2. Show the methods available for objects of the given class (if there are multiple classes, show methods for all classes). (3 points)

```
for (class in strsplit(class(dat), " ")) {
  print(class)
  print(methods(class=class))
}
```

```
## [1] "tbl_df"
## [1] [          [[<-          [<-          $
## [6] $<-          as.data.frame coerce          initialize          names<-
```

```
## [11] Ops          row.names<-  show          slotsFromS3  str
## see '?methods' for accessing help and source code
## [1] "tbl"
## [1] [[<-          [<-          $<-          coerce          format          initialize
## [7] Ops          print          show          slotsFromS3
## see '?methods' for accessing help and source code
## [1] "data.frame"
## [1] [            [[            [[<-          [<-          $<-
## [6] aggregate    anyDuplicated anyNA          as.data.frame as.list
## [11] as.matrix     as.vector     by            cbind         coerce
## [16] dim           dimnames      dimnames<-    droplevels    duplicated
## [21] edit          format         formula       head          initialize
## [26] is.na         Math          merge         na.exclude    na.omit
## [31] Ops          plot          print         prompt        rbind
## [36] row.names     row.names<-   rowsum        show          slotsFromS3
## [41] split         split<-       stack         str           subset
## [46] summary       Summary       t            tail          transform
## [51] type.convert  unique        unstack       within        xtfrm
## see '?methods' for accessing help and source code
```

3. If you call `print(dat)`, what print method is being dispatched? (1 point)

```
typeof(dat)
```

```
## [1] "list"          not sure why it says list here, as in question 1 dat is a dataframe or tbl or tbl_df
```

```
?print.listof
```

```
methods(print)[90]
```

```
## [1] "print.dendrogram"
```

4. Set the class of `dat` to be a `data.frame`. (1 point)

```
dat <- as.data.frame(dat)
```

```
class(dat)
```

```
## [1] "data.frame"
```

5. If you call `print(dat)` again, what print method is being dispatched? (1 point)

```
methods(print)[87]
```

```
## [1] "print.data.frame"
```

Define a new generic function `nUnique` with the code below.

```
nUnique <- function(x) {
  UseMethod('nUnique')
}
```

6. Write a default method for `nUnique` to count the number of unique values in an element. (2 points)

```
nUnique.default <- function(x) {
  if(length(x) > 1) {
    return(length(unique(x)))
  } else {
    return(1)
  }
}
nUnique(c(1,2,3,2,2,2))
```

```
## [1] 3
```

7. Check your function (2 points)

output?

```
nUnique(letters) # should return 26
nUnique(sample(10, 100, replace = TRUE)) # should return 10 (probably)
```

8. Write a `data.frame` method for `nUnique` to operate on `data.frame` objects. This version should return counts for each column in a `data.frame`. (2 points)

```
df <- data.frame(
  A = c(1, 2, 3),
  B = c(1, 3, 4),
  C = c(2, 4, 5)
)
names(df)[1]
```

```
## [1] "A"
```

```
df[,2]
```

```
## [1] 1 3 4
```

```
nUnique.data.frame <- function(x) {
  for (i in 1:length(names(x))) {
    print(nUnique(x[,i]))
  }
}
nUnique(df)
```

```
## [1] 3
```

```
## [1] 3
```

```
## [1] 3
```

9. Check your function (2 points)

```
nUnique(dat)          output?
```

Question 2

15 points

Programming with classes. The following function will generate random patient information.

```
makePatient <- function() {  
  vowel <- grep("[aeiou]", letters)  
  cons <- grep("[^aeiou]", letters)  
  name <- paste(sample(LETTERS[cons], 1), sample(letters[vowel], 1), sample(letters[cons], 1), sep='')  
  gender <- factor(sample(0:1, 1), levels=0:1, labels=c('female', 'male'))  
  dob <- as.Date(sample(7500, 1), origin="1970-01-01")  
  n <- sample(6, 1)  
  doa <- as.Date(sample(1500, n), origin="2010-01-01")  
  pulse <- round(rnorm(n, 80, 10))  
  temp <- round(rnorm(n, 98.4, 0.3), 2)  
  fluid <- round(runif(n), 2)  
  list(name, gender, dob, doa, pulse, temp, fluid)  
}
```

```
makePatient()[1]
```

```
## [[1]]  
## [1] "Wev"
```

1. Create an S3 class `medicalRecord` for objects that are a list with the named elements `name`, `gender`, `date_of_birth`, `date_of_admission`, `pulse`, `temperature`, `fluid_intake`. Note that an individual patient may have multiple measurements for some measurements. Set the RNG seed to 8 and create a medical record by taking the output of `makePatient`. Print the medical record, and print the class of the medical record. (5 points)

```
# make a new class  
set.seed(8)  
my_record <- list(name=NULL, gender=NULL, date_of_birth=NULL, date_of_admission=NULL, pulse=NULL,  
  temperature=NULL, fluid_intake=NULL)  
class(my_record) <- 'medicalRecord'  
patient_data <- makePatient()  
  
my_record$name <- patient_data[[1]]  
my_record$gender <- as.character(patient_data[[2]])  
my_record$date_of_birth <- patient_data[[3]]  
my_record$date_of_admission <- patient_data[[4]]  
my_record$pulse <- patient_data[[5]]  
my_record$temperature <- patient_data[[6]]  
my_record$fluid_intake <- patient_data[[7]]  
  
my_record
```

```
## $name
## [1] "Yes"
##
## $gender
## [1] "male"
##
## $date_of_birth
## [1] "1977-05-03"
##
## $date_of_admission
## [1] "2013-06-09" "2013-07-02"
##
## $pulse
## [1] 79 78
##
## $temperature
## [1] 98.07 97.50
##
## $fluid_intake
## [1] 0.28 0.52
##
## attr("class")
## [1] "medicalRecord"
```

```
class(my_record)
```

```
## [1] "medicalRecord"
```

2. Write a `medicalRecord` method for the generic function `mean`, which returns averages for pulse, temperature and fluids. Also write a `medicalRecord` method for `print`, which employs some nice formatting, perhaps arranging measurements by date, and `plot`, that generates a composite plot of measurements over time. Call each function for the medical record created in part 1. (5 points)

```
mean.medicalRecord <- function(x) {
  pulse <- mean(x$pulse)
  temp <- mean(x$temperature)
  fluids <- mean(x$fluid_intake)
  return(list(c("pulse: ", pulse),
               c("temp: ", temp),
               c("fluid intake: ", fluids)))
}

mean(my_record)
```

```
## [[1]]
## [1] "pulse: " "78.5"
##
## [[2]]
## [1] "temp: " "97.785"
##
## [[3]]
## [1] "fluid intake: " "0.4"
```

3. Create a further class for a cohort (group) of patients, and write methods for `mean` and `print` which, when applied to a cohort, apply mean or print to each patient contained in the cohort. Hint: think of this as a “container” for patients. Reset the RNG seed to 8 and create a cohort of ten patients, then show the output for `mean` and `print`. (5 points)

```
make_record <- function() {  
  new_record <- list(name=NULL, gender=NULL, date_of_birth=NULL, date_of_admission=NULL,  
                    pulse=NULL, temperature=NULL, fluid_intake=NULL)  
  class(new_record) <- 'medicalRecord'  
  new_patient_data <- makePatient()  
  
  new_record$name <- new_patient_data[[1]]  
  new_record$gender <- as.character(new_patient_data[[2]])  
  new_record$date_of_birth <- new_patient_data[[3]]  
  new_record$date_of_admission <- new_patient_data[[4]]  
  new_record$pulse <- new_patient_data[[5]]  
  new_record$temperature <- new_patient_data[[6]]  
  new_record$fluid_intake <- new_patient_data[[7]]  
  return(new_record)  
}
```

```
class(make_record())
```

```
## [1] "medicalRecord"
```

```
set.seed(8)  
my_cohort <- list()  
  
for (i in 1:10) {  
  new_record <- make_record()  
  my_cohort[[i]] <- new_record  
}  
  
class(my_cohort) <- 'cohort'  
my_cohort
```

```
## [[1]]  
## $name  
## [1] "Yes"  
##  
## $gender  
## [1] "male"  
##  
## $date_of_birth  
## [1] "1977-05-03"  
##  
## $date_of_admission  
## [1] "2013-06-09" "2013-07-02"  
##  
## $pulse  
## [1] 79 78  
##
```

```

## $temperature
## [1] 98.07 97.50
##
## $fluid_intake
## [1] 0.28 0.52
##
## attr("class")
## [1] "medicalRecord"
##
## [[2]]
## $name
## [1] "Fal"
##
## $gender
## [1] "male"
##
## $date_of_birth
## [1] "1988-05-24"
##
## $date_of_admission
## [1] "2010-11-16" "2013-09-12" "2013-03-24"
##
## $pulse
## [1] 76 96 87
##
## $temperature
## [1] 98.23 98.75 98.21
##
## $fluid_intake
## [1] 0.18 0.96 0.10
##
## attr("class")
## [1] "medicalRecord"
##
## [[3]]
## $name
## [1] "Zog"
##
## $gender
## [1] "male"
##
## $date_of_birth
## [1] "1988-12-14"
##
## $date_of_admission
## [1] "2013-03-25" "2013-07-29" "2013-10-27" "2010-02-24"
##
## $pulse
## [1] 69 75 80 84
##
## $temperature
## [1] 98.49 98.82 98.74 98.54
##
## $fluid_intake

```

```

## [1] 0.81 0.59 0.28 0.40
##
## attr(,"class")
## [1] "medicalRecord"
##
## [[4]]
## $name
## [1] "Yol"
##
## $gender
## [1] "male"
##
## $date_of_birth
## [1] "1986-03-11"
##
## $date_of_admission
## [1] "2014-01-28" "2013-03-24" "2012-03-10" "2010-02-22" "2011-12-27"
## [6] "2012-11-26"
##
## $pulse
## [1] 69 78 87 84 89 92
##
## $temperature
## [1] 98.29 98.44 98.78 98.87 98.27 98.26
##
## $fluid_intake
## [1] 0.03 0.13 0.12 0.39 0.97 0.14
##
## attr(,"class")
## [1] "medicalRecord"
##
## [[5]]
## $name
## [1] "Yak"
##
## $gender
## [1] "female"
##
## $date_of_birth
## [1] "1983-09-15"
##
## $date_of_admission
## [1] "2012-08-30" "2012-04-07" "2011-07-19" "2012-07-11"
##
## $pulse
## [1] 90 88 75 81
##
## $temperature
## [1] 98.58 97.53 98.58 99.11
##
## $fluid_intake
## [1] 0.26 0.29 0.60 0.66
##
## attr(,"class")

```



```

## [1] "medicalRecord"
##
## [[6]]
## $name
## [1] "Gaf"
##
## $gender
## [1] "female"
##
## $date_of_birth
## [1] "1978-04-27"
##
## $date_of_admission
## [1] "2012-04-24" "2010-07-19" "2012-08-06" "2013-08-21" "2011-05-03"
##
## $pulse
## [1] 89 91 77 75 90
##
## $temperature
## [1] 98.32 98.01 98.96 98.52 98.61
##
## $fluid_intake
## [1] 0.42 0.47 0.74 0.62 0.36
##
## attr("class")
## [1] "medicalRecord"
##
## [[7]]
## $name
## [1] "Kuw"
##
## $gender
## [1] "female"
##
## $date_of_birth
## [1] "1980-11-07"
##
## $date_of_admission
## [1] "2011-09-16" "2010-10-29" "2012-07-10" "2010-10-03"
##
## $pulse
## [1] 72 81 71 82
##
## $temperature
## [1] 98.21 98.17 98.65 98.49
##
## $fluid_intake
## [1] 0.29 0.93 0.25 0.12
##
## attr("class")
## [1] "medicalRecord"
##
## [[8]]
## $name

```

```

## [1] "Mav"
##
## $gender
## [1] "female"
##
## $date_of_birth
## [1] "1989-07-16"
##
## $date_of_admission
## [1] "2012-03-02" "2010-06-11" "2010-02-08" "2010-04-19"
##
## $pulse
## [1] 63 83 66 88
##
## $temperature
## [1] 99.07 98.45 97.95 98.00
##
## $fluid_intake
## [1] 0.01 0.79 0.79 0.50
##
## attr(,"class")
## [1] "medicalRecord"
##
## [[9]]
## $name
## [1] "Fel"
##
## $gender
## [1] "male"
##
## $date_of_birth
## [1] "1985-08-16"
##
## $date_of_admission
## [1] "2012-06-24" "2010-09-26"
##
## $pulse
## [1] 65 81
##
## $temperature
## [1] 98.21 98.51
##
## $fluid_intake
## [1] 0.06 0.24
##
## attr(,"class")
## [1] "medicalRecord"
##
## [[10]]
## $name
## [1] "Say"
##
## $gender
## [1] "female"

```

```
##
## $date_of_birth
## [1] "1974-09-22"
##
## $date_of_admission
## [1] "2010-03-14"
##
## $pulse
## [1] 77
##
## $temperature
## [1] 98.54
##
## $fluid_intake
## [1] 0.15
##
## attr("class")
## [1] "medicalRecord"
##
## attr("class")
## [1] "cohort"
```

```
mean.cohort <- function(x) {
  for (patient in x) {
    print(patient$name)
    print(mean(patient))
  }
}

print.cohort <- function(x) {
  for (patient in x) {
    print(patient)
  }
}
```

```
mean(my_cohort)
```

```
## [1] "Yes"
## [[1]]
## [1] "pulse: " "78.5"
##
## [[2]]
## [1] "temp: " "97.785"
##
## [[3]]
## [1] "fluid intake: " "0.4"
##
## [1] "Fal"
## [[1]]
## [1] "pulse: " "86.3333333333333"
##
## [[2]]
## [1] "temp: " "98.3966666666667"
```

```

##
## [[3]]
## [1] "fluid intake: "      "0.4133333333333333"
##
## [1] "Zog"
## [[1]]
## [1] "pulse: " "77"
##
## [[2]]
## [1] "temp: " "98.6475"
##
## [[3]]
## [1] "fluid intake: " "0.52"
##
## [1] "Yol"
## [[1]]
## [1] "pulse: "      "83.16666666666667"
##
## [[2]]
## [1] "temp: " "98.485"
##
## [[3]]
## [1] "fluid intake: "      "0.2966666666666667"
##
## [1] "Yak"
## [[1]]
## [1] "pulse: " "83.5"
##
## [[2]]
## [1] "temp: " "98.45"
##
## [[3]]
## [1] "fluid intake: " "0.4525"
##
## [1] "Gaf"
## [[1]]
## [1] "pulse: " "84.4"
##
## [[2]]
## [1] "temp: " "98.484"
##
## [[3]]
## [1] "fluid intake: " "0.522"
##
## [1] "Kuw"
## [[1]]
## [1] "pulse: " "76.5"
##
## [[2]]
## [1] "temp: " "98.38"
##
## [[3]]
## [1] "fluid intake: " "0.3975"
##

```

```

## [1] "Mav"
## [[1]]
## [1] "pulse: " "75"
##
## [[2]]
## [1] "temp: " "98.3675"
##
## [[3]]
## [1] "fluid intake: " "0.5225"
##
## [1] "Fel"
## [[1]]
## [1] "pulse: " "73"
##
## [[2]]
## [1] "temp: " "98.36"
##
## [[3]]
## [1] "fluid intake: " "0.15"
##
## [1] "Say"
## [[1]]
## [1] "pulse: " "77"
##
## [[2]]
## [1] "temp: " "98.54"
##
## [[3]]
## [1] "fluid intake: " "0.15"

```

```
print(my_cohort)
```

```

## $name
## [1] "Yes"
##
## $gender
## [1] "male"
##
## $date_of_birth
## [1] "1977-05-03"
##
## $date_of_admission
## [1] "2013-06-09" "2013-07-02"
##
## $pulse
## [1] 79 78
##
## $temperature
## [1] 98.07 97.50
##
## $fluid_intake
## [1] 0.28 0.52
##
## attr(,"class")

```

```

## [1] "medicalRecord"
## $name
## [1] "Fal"
##
## $gender
## [1] "male"
##
## $date_of_birth
## [1] "1988-05-24"
##
## $date_of_admission
## [1] "2010-11-16" "2013-09-12" "2013-03-24"
##
## $pulse
## [1] 76 96 87
##
## $temperature
## [1] 98.23 98.75 98.21
##
## $fluid_intake
## [1] 0.18 0.96 0.10
##
## attr("class")
## [1] "medicalRecord"
## $name
## [1] "Zog"
##
## $gender
## [1] "male"
##
## $date_of_birth
## [1] "1988-12-14"
##
## $date_of_admission
## [1] "2013-03-25" "2013-07-29" "2013-10-27" "2010-02-24"
##
## $pulse
## [1] 69 75 80 84
##
## $temperature
## [1] 98.49 98.82 98.74 98.54
##
## $fluid_intake
## [1] 0.81 0.59 0.28 0.40
##
## attr("class")
## [1] "medicalRecord"
## $name
## [1] "Yol"
##
## $gender
## [1] "male"
##
## $date_of_birth

```

```

## [1] "1986-03-11"
##
## $date_of_admission
## [1] "2014-01-28" "2013-03-24" "2012-03-10" "2010-02-22" "2011-12-27"
## [6] "2012-11-26"
##
## $pulse
## [1] 69 78 87 84 89 92
##
## $temperature
## [1] 98.29 98.44 98.78 98.87 98.27 98.26
##
## $fluid_intake
## [1] 0.03 0.13 0.12 0.39 0.97 0.14
##
## attr("class")
## [1] "medicalRecord"
## $name
## [1] "Yak"
##
## $gender
## [1] "female"
##
## $date_of_birth
## [1] "1983-09-15"
##
## $date_of_admission
## [1] "2012-08-30" "2012-04-07" "2011-07-19" "2012-07-11"
##
## $pulse
## [1] 90 88 75 81
##
## $temperature
## [1] 98.58 97.53 98.58 99.11
##
## $fluid_intake
## [1] 0.26 0.29 0.60 0.66
##
## attr("class")
## [1] "medicalRecord"
## $name
## [1] "Gaf"
##
## $gender
## [1] "female"
##
## $date_of_birth
## [1] "1978-04-27"
##
## $date_of_admission
## [1] "2012-04-24" "2010-07-19" "2012-08-06" "2013-08-21" "2011-05-03"
##
## $pulse
## [1] 89 91 77 75 90

```

```

##
## $temperature
## [1] 98.32 98.01 98.96 98.52 98.61
##
## $fluid_intake
## [1] 0.42 0.47 0.74 0.62 0.36
##
## attr(,"class")
## [1] "medicalRecord"
## $name
## [1] "Kuw"
##
## $gender
## [1] "female"
##
## $date_of_birth
## [1] "1980-11-07"
##
## $date_of_admission
## [1] "2011-09-16" "2010-10-29" "2012-07-10" "2010-10-03"
##
## $pulse
## [1] 72 81 71 82
##
## $temperature
## [1] 98.21 98.17 98.65 98.49
##
## $fluid_intake
## [1] 0.29 0.93 0.25 0.12
##
## attr(,"class")
## [1] "medicalRecord"
## $name
## [1] "Mav"
##
## $gender
## [1] "female"
##
## $date_of_birth
## [1] "1989-07-16"
##
## $date_of_admission
## [1] "2012-03-02" "2010-06-11" "2010-02-08" "2010-04-19"
##
## $pulse
## [1] 63 83 66 88
##
## $temperature
## [1] 99.07 98.45 97.95 98.00
##
## $fluid_intake
## [1] 0.01 0.79 0.79 0.50
##
## attr(,"class")

```



```

## [1] "medicalRecord"
## $name
## [1] "Fel"
##
## $gender
## [1] "male"
##
## $date_of_birth
## [1] "1985-08-16"
##
## $date_of_admission
## [1] "2012-06-24" "2010-09-26"
##
## $pulse
## [1] 65 81
##
## $temperature
## [1] 98.21 98.51
##
## $fluid_intake
## [1] 0.06 0.24
##
## attr("class")
## [1] "medicalRecord"
## $name
## [1] "Say"
##
## $gender
## [1] "female"
##
## $date_of_birth
## [1] "1974-09-22"
##
## $date_of_admission
## [1] "2010-03-14"
##
## $pulse
## [1] 77
##
## $temperature
## [1] 98.54
##
## $fluid_intake
## [1] 0.15
##
## attr("class")
## [1] "medicalRecord"

```