

Bios 6301: Assignment 8

Lan Shi

Due Tuesday, 16 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

```
library(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)

```
methods(class(dat)[1])
```

```
## [1] [          [[          [[<-          [<-          $
## [6] $<-          as.data.frame coerce          initialize names<-
## [11] Ops          row.names<- show          slotsFromS3 str
## see '?methods' for accessing help and source code
```

```
methods(class(dat)[2])
```

```
## [1] [[<-          [<-          $<-          coerce          format          initialize
## [7] Ops          print          show          slotsFromS3
## see '?methods' for accessing help and source code
```

```
methods(class(dat)[3])
```

```
## [1] [
## [6] aggregate anyDuplicated anyNA as.data.frame as.list
## [11] as.matrix by cbind coerce dim
## [16] dimnames dimnames<- droplevels duplicated edit
## [21] format formula head initialize is.na
## [26] Math merge na.exclude na.omit Ops
## [31] plot print prompt rbind row.names
## [36] row.names<- rowsum show slotsFromS3 split
## [41] split<- stack str subset summary
## [46] Summary t tail transform type.convert
## [51] 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)

```
print.tbl_df
```

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

```
class(dat) = 'data.frame'
```

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

```
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) {
  length(unique(x))
}
```

7. Check your function (2 points)

```
nUnique(letters) # should return 26
```

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

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

```
## [1] 10
```

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)

```
nUnique.data.frame <- function(x) {
  sapply(x, \ (y) length(unique(y)))
}
```

9. Check your function (2 points)

```
nUnique(dat)
```

```
## Procedure Code Category ICD-10-PCS Codes
## 39 9697
## Procedure Code Descriptions Code Status
## 9697 4
```

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)  
}
```

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)

```
set.seed(8)  
mr = makePatient()  
names(mr) = c('name', 'gender', 'date_of_birth', 'date_of_admission',  
             'pulse', 'temperature', 'fluid_intake')  
class(mr) = 'medicalRecord'  
print(mr)
```

```
## $name  
## [1] "Yes"  
##  
## $gender  
## [1] male  
## Levels: female 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"
```

```
print(class(mr))
```

```
## [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(mr){  
  y = c(mean(mr$pulse),mean(mr$temperature),mean(mr$fluid_intake))  
  names(y) = c('pulse', 'temperature', 'fluid_intake')  
  y  
}  
mean(mr)
```

```
##      pulse  temperature fluid_intake  
##      78.500      97.785      0.400
```

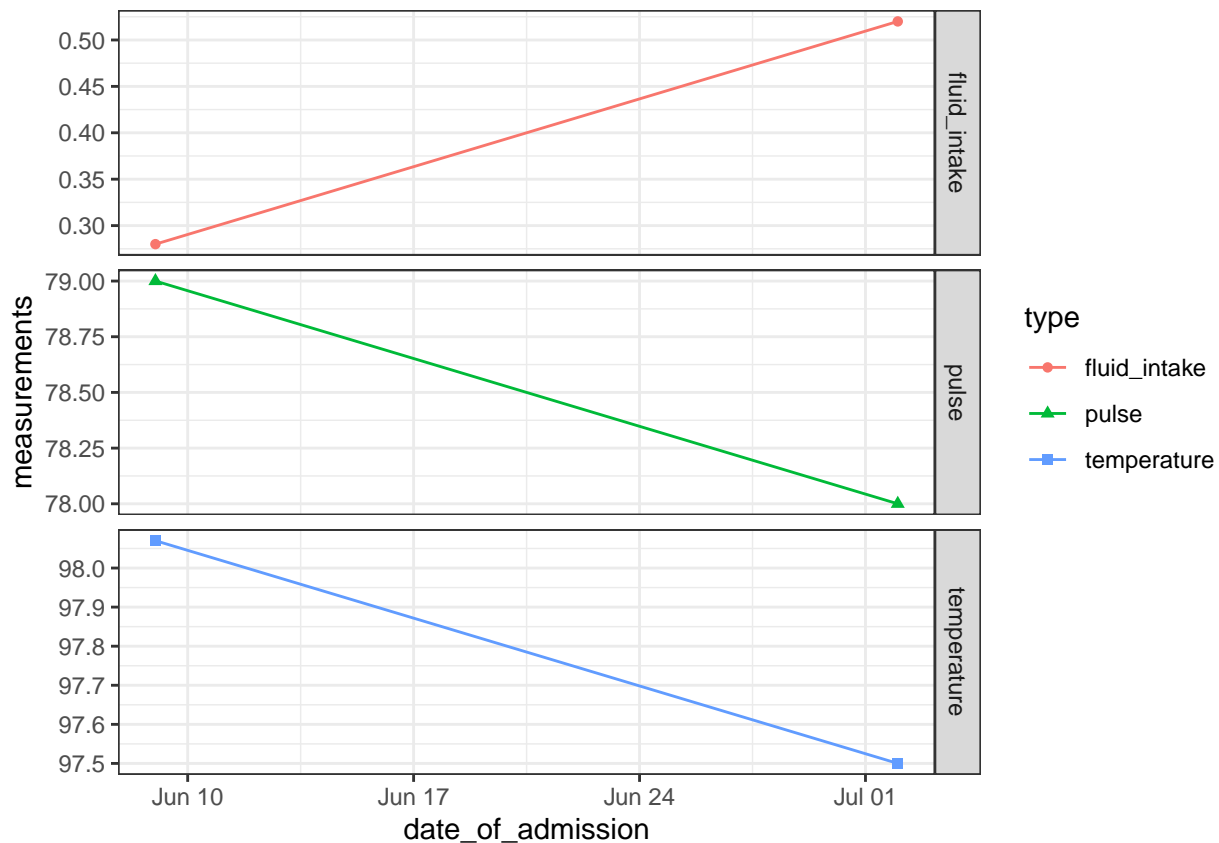
```
#arranging measurements by date, in a decreasing order
```

```
print.medicalRecord <- function(mr){  
  y = do.call(cbind.data.frame, mr)  
  y.arrange = y[order(y$date_of_admission,decreasing = T),]  
  print(y.arrange)  
}  
print(mr)
```

```
##  name gender date_of_birth date_of_admission pulse temperature fluid_intake  
## 2  Yes   male   1977-05-03      2013-07-02    78      97.50      0.52  
## 1  Yes   male   1977-05-03      2013-06-09    79      98.07      0.28
```

```
# composite plot of measurements over time.
```

```
library(tidyr)  
library(ggplot2)  
plot.medicalRecord <- function(mr){  
  y = do.call(cbind.data.frame, mr) %>% as_tibble %>%  
    pivot_longer(5:7,names_to = 'type',values_to = "measurements")  
  p = ggplot(y,aes(x=date_of_admission,  
                  y=measurements,color=type)) +  
    geom_point(aes(shape=type)) +  
    geom_line() + theme_bw() +  
    facet_grid(type ~ .,scales = "free")  
  print(p)  
}  
plot(mr)
```



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)

```
set.seed(8)
cohort = list()
name = NULL
for (i in 1:10){
  mr = makePatient()
  names(mr) = c('name', 'gender', 'date_of_birth',
               'date_of_admission', 'pulse', 'temperature', 'fluid_intake')
  name = c(name, mr$name)
  cohort = c(cohort, list(mr))
}
names(cohort) = name
class(cohort) = 'cohort'
cohort[[5]]
```

```
## $name
## [1] "Yak"
##
## $gender
## [1] female
## Levels: female male
##
## $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
```

```
mean.cohort = function(cohort){
  lapply(cohort,function(x){
    class(x) = 'medicalRecord'
    mean(x)
  })
}
mean(cohort)
```

```
## $Yes
##      pulse  temperature fluid_intake
##      78.500      97.785      0.400
##
## $Fal
##      pulse  temperature fluid_intake
##      86.3333333  98.3966667  0.4133333
##
## $Zog
##      pulse  temperature fluid_intake
##      77.0000      98.6475      0.5200
##
## $Yol
##      pulse  temperature fluid_intake
##      83.1666667  98.4850000  0.2966667
##
## $Yak
##      pulse  temperature fluid_intake
##      83.5000      98.4500      0.4525
##
## $Gaf
##      pulse  temperature fluid_intake
##      84.400      98.484      0.522
##
## $Kuw
##      pulse  temperature fluid_intake
##      76.5000      98.3800      0.3975
##
## $Mav
##      pulse  temperature fluid_intake
##      75.0000      98.3675      0.5225
##
## $Fel
##      pulse  temperature fluid_intake
```

```
##      73.00      98.36      0.15
##
## $Say
##      pulse  temperature fluid_intake
##      77.00      98.54      0.15
```

```
print.cohort <- function(cohort){
  invisible(lapply(cohort,function(x){
    class(x) = 'medicalRecord'
    y = print(x)
  }))
}
print(cohort)
```

```
##  name gender date_of_birth date_of_admission pulse temperature fluid_intake
## 2  Yes  male   1977-05-03      2013-07-02    78      97.50      0.52
## 1  Yes  male   1977-05-03      2013-06-09    79      98.07      0.28
##  name gender date_of_birth date_of_admission pulse temperature fluid_intake
## 2  Fal  male   1988-05-24      2013-09-12    96      98.75      0.96
## 3  Fal  male   1988-05-24      2013-03-24    87      98.21      0.10
## 1  Fal  male   1988-05-24      2010-11-16    76      98.23      0.18
##  name gender date_of_birth date_of_admission pulse temperature fluid_intake
## 3  Zog  male   1988-12-14      2013-10-27    80      98.74      0.28
## 2  Zog  male   1988-12-14      2013-07-29    75      98.82      0.59
## 1  Zog  male   1988-12-14      2013-03-25    69      98.49      0.81
## 4  Zog  male   1988-12-14      2010-02-24    84      98.54      0.40
##  name gender date_of_birth date_of_admission pulse temperature fluid_intake
## 1  Yol  male   1986-03-11      2014-01-28    69      98.29      0.03
## 2  Yol  male   1986-03-11      2013-03-24    78      98.44      0.13
## 6  Yol  male   1986-03-11      2012-11-26    92      98.26      0.14
## 3  Yol  male   1986-03-11      2012-03-10    87      98.78      0.12
## 5  Yol  male   1986-03-11      2011-12-27    89      98.27      0.97
## 4  Yol  male   1986-03-11      2010-02-22    84      98.87      0.39
##  name gender date_of_birth date_of_admission pulse temperature fluid_intake
## 1  Yak  female  1983-09-15      2012-08-30    90      98.58      0.26
## 4  Yak  female  1983-09-15      2012-07-11    81      99.11      0.66
## 2  Yak  female  1983-09-15      2012-04-07    88      97.53      0.29
## 3  Yak  female  1983-09-15      2011-07-19    75      98.58      0.60
##  name gender date_of_birth date_of_admission pulse temperature fluid_intake
## 4  Gaf  female  1978-04-27      2013-08-21    75      98.52      0.62
## 3  Gaf  female  1978-04-27      2012-08-06    77      98.96      0.74
## 1  Gaf  female  1978-04-27      2012-04-24    89      98.32      0.42
## 5  Gaf  female  1978-04-27      2011-05-03    90      98.61      0.36
## 2  Gaf  female  1978-04-27      2010-07-19    91      98.01      0.47
##  name gender date_of_birth date_of_admission pulse temperature fluid_intake
## 3  Kuw  female  1980-11-07      2012-07-10    71      98.65      0.25
## 1  Kuw  female  1980-11-07      2011-09-16    72      98.21      0.29
## 2  Kuw  female  1980-11-07      2010-10-29    81      98.17      0.93
## 4  Kuw  female  1980-11-07      2010-10-03    82      98.49      0.12
##  name gender date_of_birth date_of_admission pulse temperature fluid_intake
## 1  Mav  female  1989-07-16      2012-03-02    63      99.07      0.01
## 2  Mav  female  1989-07-16      2010-06-11    83      98.45      0.79
## 4  Mav  female  1989-07-16      2010-04-19    88      98.00      0.50
## 3  Mav  female  1989-07-16      2010-02-08    66      97.95      0.79
##  name gender date_of_birth date_of_admission pulse temperature fluid_intake
```

##	1	Fel	male	1985-08-16	2012-06-24	65	98.21	0.06
##	2	Fel	male	1985-08-16	2010-09-26	81	98.51	0.24
##		name	gender	date_of_birth	date_of_admission	pulse	temperature	fluid_intake
##	1	Say	female	1974-09-22	2010-03-14	77	98.54	0.15