Bios 6301: Assignment 8

Max Rohde

Question 1

Install the `readxl` package and run the following

```
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 = c("tbl_df"))
```

```
##  [1] [             [[            [[<-          [<-           $
##  [6] $<-           arrange_      as.data.frame coerce        distinct_
## [11] filter_       fortify       group_data    initialize    mutate_
## [16] names<-       nest_legacy   nest          Ops           row.names<-
## [21] show          slice_        slotsFromS3   str           summarise_
## [26] type_sum
## see '?methods' for accessing help and source code
```

```
methods(class = c("tbl"))
```

```
##  [1] [[<-        [<-         $<-         as.tbl      coerce      format
##  [7] fortify     glimpse     initialize  Ops         print       show
## [13] slotsFromS3 tbl_sum
## see '?methods' for accessing help and source code
```

```
methods(class = c("data.frame"))
```

```
##   [1] [                [[               [[<-             [<-
##   [5] $<-              aggregate        anti_join        anyDuplicated
##   [9] anyNA            arrange_         arrange          as_factor
```

```
##  [13] as_tibble         as.data.frame     as.list           as.matrix
##  [17] as.tbl            auto_copy         by                cbind
##  [21] coerce            collapse          collect           complete_
##  [25] complete          compute           dim               dimnames
##  [29] dimnames<-        distinct_         distinct          do_
##  [33] do                dplyr_col_modify  dplyr_reconstruct dplyr_row_slice
##  [37] drop_na_          drop_na           droplevels        duplicated
##  [41] edit              expand_           expand            extract_
##  [45] extract           fill_             fill              filter_
##  [49] filter            format            formula           fortify
##  [53] full_join         gather_           gather            ggplot_add
##  [57] glimpse           group_by_         group_by          group_data
##  [61] group_indices_    group_indices     group_keys        group_map
##  [65] group_modify      group_nest        group_size        group_split
##  [69] group_trim        group_vars        groups            head
##  [73] initialize        inner_join        intersect         is.na
##  [77] left_join         Math              merge             mutate_
##  [81] mutate            n_groups          na.exclude        na.omit
##  [85] nest_by           nest_join         nest_legacy       nest
##  [89] Ops               pivot_longer      pivot_wider       plot
##  [93] print             prompt            pull              rbind
##  [97] relocate          rename_           rename_with       rename
## [101] replace_na        right_join        row.names         row.names<-
## [105] rows_delete       rows_insert       rows_patch        rows_update
## [109] rows_upsert       rowsum            rowwise           same_src
## [113] sample_frac       sample_n          select_           select
## [117] semi_join         separate_         separate_rows_    separate_rows
## [121] separate          setdiff           setequal          show
## [125] slice_            slice_head        slice_max         slice_min
## [129] slice_sample      slice_tail        slice             slotsFromS3
## [133] split             split<-           spread_           spread
## [137] stack             str               subset            summarise_
## [141] summarise         summary           Summary           t
## [145] tail              tbl_vars          transform         transmute
## [149] type.convert      ungroup           union_all         union
## [153] unique            unite_            unite             unnest_legacy
## [157] unnest            unstack           within
## see '?methods' for accessing help and source code
```

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

```
sloop::s3_dispatch(print(dat))
```

```
##    print.tbl_df
## => print.tbl
```

```
##  * print.data.frame
##  * print.default
```

```
# We see that print.tbl_df is being dispatched
```

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)

```
sloop::s3_dispatch(print(dat))
```

```
## => print.data.frame
##  * print.default
```

```
# We see that print.tbl is being dispatched
```

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)

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

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

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

```
## [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){
  map(dat, ~nUnique(.x))
}
```

9. Check your function (2 points)

```
nUnique(dat)
```

```
## $`Procedure Code Category`
## [1] 39
##
## $`ICD-10-PCS Codes`
## [1] 9681
##
## $`Procedure Code Descriptions`
## [1] 9681
##
## $`Code Status`
## [1] 6
```

Question 2

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)

```
create_record <- function() {
  record <- makePatient()
  names(record) <- c("name",
                     "gender",
                     "date_of_birth",
                     "date_of_admission",
                     "pulse",
                     "temperature",
                     "fluid_intake")

  class(record) <- "medicalRecord"

  return(record)
}

set.seed(8)
record <- create_record()
```

```
print(record)
```

```
## $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(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 <- mean(x[["pulse"]])
  temperature_mean <- mean(x[["temperature"]])
  fluids_mean <- mean(x[["fluid_intake"]])

  measurements <- list(pulse_mean = pulse_mean,
                       temperature_mean = temperature_mean,
                       fluids_mean = fluids_mean)
  return(measurements)
}

mean(record)
```

```
## $pulse_mean
## [1] 78.5
##
## $temperature_mean
## [1] 97.785
##
## $fluids_mean
## [1] 0.4
```

```
print.medicalRecord <- function(x) {
  print(glue("Name: {x$name}"))

  print(glue("Gender: {x$gender}"))
```

```
  print(glue("DOB: {x$date_of_birth}"))


  df <- tibble(date_of_admission = x$date_of_admission,
               pulse = x$pulse,
               temperature= x$temperature,
               fluid_intake = x$fluid_intake)

  df %>% arrange(date_of_admission) -> df

  print.data.frame(df)
}

print(record)
```

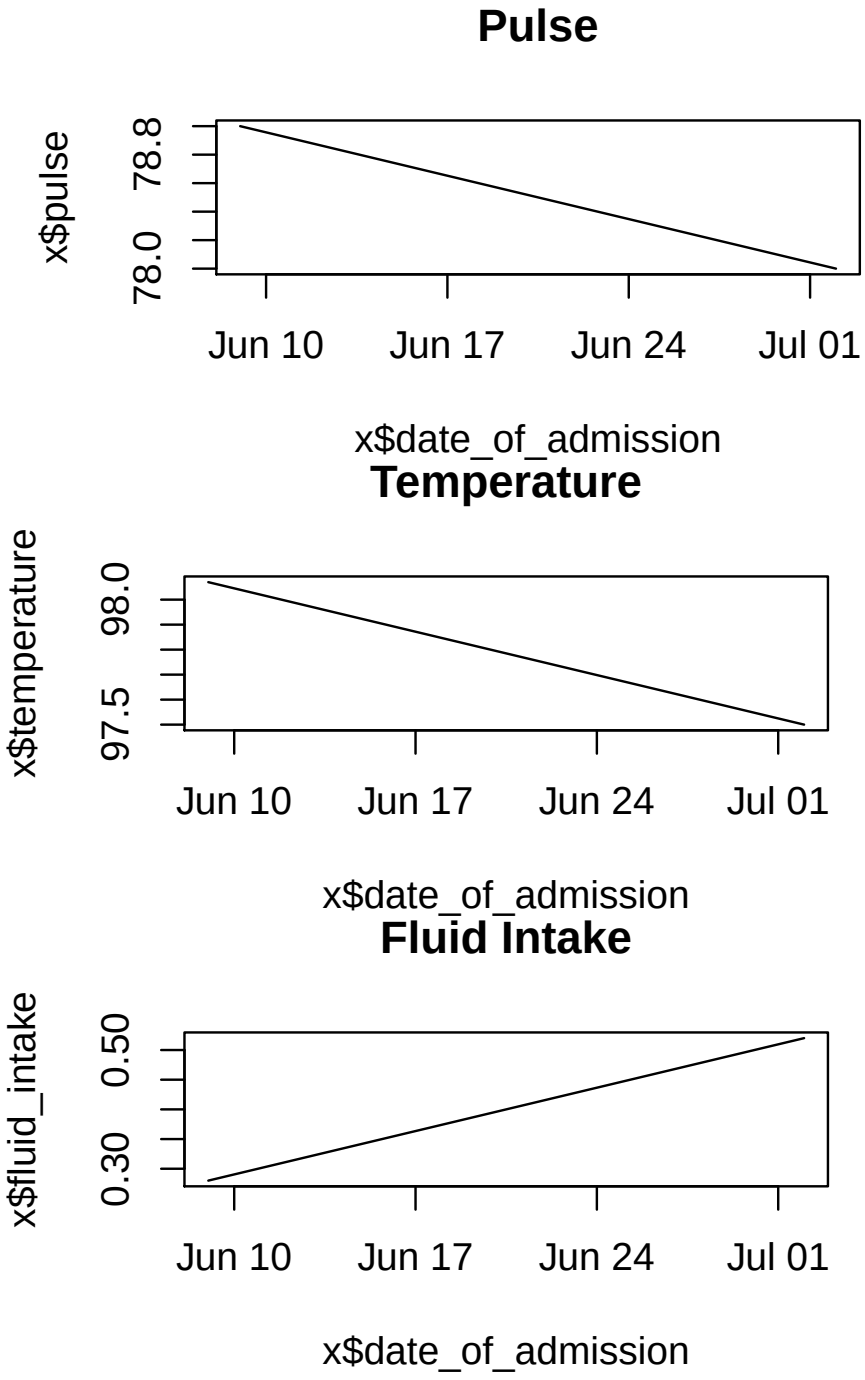
```
## Name: Yes
## Gender: male
## DOB: 1977-05-03
##   date_of_admission pulse temperature fluid_intake
## 1        2013-06-09    79       98.07         0.28
## 2        2013-07-02    78       97.50         0.52
```

```
plot.medicalRecord <- function(x) {
  plot(x$date_of_admission,
       x$pulse,
       type = "l",
       main = "Pulse")

  plot(x$date_of_admission,
       x$temperature,
       type = "l",
       main = "Temperature")

  plot(x$date_of_admission,
       x$fluid_intake,
       type = "l",
       main = "Fluid Intake")
}

plot(record)
```



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)

```r
set.seed(8)

# Create a cohort of 10 patients
cohort <- map(1:10, ~create_record())
class(cohort) <- "cohort"

mean.cohort <- function(x) {
  means <- map(x, ~mean(.x))
  return(means)
}

mean(cohort)
```

```
## [[1]]
## [[1]]$pulse_mean
## [1] 78.5
##
## [[1]]$temperature_mean
## [1] 97.785
##
## [[1]]$fluids_mean
## [1] 0.4
##
##
## [[2]]
## [[2]]$pulse_mean
## [1] 86.33333
##
## [[2]]$temperature_mean
## [1] 98.39667
##
## [[2]]$fluids_mean
## [1] 0.4133333
##
##
## [[3]]
## [[3]]$pulse_mean
## [1] 77
##
## [[3]]$temperature_mean
## [1] 98.6475
##
## [[3]]$fluids_mean
## [1] 0.52
```

```
##
##
## [[4]]
## [[4]]$pulse_mean
## [1] 83.16667
##
## [[4]]$temperature_mean
## [1] 98.485
##
## [[4]]$fluids_mean
## [1] 0.2966667
##
##
## [[5]]
## [[5]]$pulse_mean
## [1] 83.5
##
## [[5]]$temperature_mean
## [1] 98.45
##
## [[5]]$fluids_mean
## [1] 0.4525
##
##
## [[6]]
## [[6]]$pulse_mean
## [1] 84.4
##
## [[6]]$temperature_mean
## [1] 98.484
##
## [[6]]$fluids_mean
## [1] 0.522
##
##
## [[7]]
## [[7]]$pulse_mean
## [1] 76.5
##
## [[7]]$temperature_mean
## [1] 98.38
##
## [[7]]$fluids_mean
## [1] 0.3975
```

```
##
##
## [[8]]
## [[8]]$pulse_mean
## [1] 75
##
## [[8]]$temperature_mean
## [1] 98.3675
##
## [[8]]$fluids_mean
## [1] 0.5225
##
##
## [[9]]
## [[9]]$pulse_mean
## [1] 73
##
## [[9]]$temperature_mean
## [1] 98.36
##
## [[9]]$fluids_mean
## [1] 0.15
##
##
## [[10]]
## [[10]]$pulse_mean
## [1] 77
##
## [[10]]$temperature_mean
## [1] 98.54
##
## [[10]]$fluids_mean
## [1] 0.15
```

```
print.cohort <- function(x) {
  walk(x, ~print(.x))
}

print(cohort)
```

```
## Name: Yes
## Gender: male
## DOB: 1977-05-03
##   date_of_admission pulse temperature fluid_intake
```

```
## 1        2013-06-09    79       98.07         0.28
## 2        2013-07-02    78       97.50         0.52
## Name: Fal
## Gender: male
## DOB: 1988-05-24
##   date_of_admission pulse temperature fluid_intake
## 1        2010-11-16    76       98.23         0.18
## 2        2013-03-24    87       98.21         0.10
## 3        2013-09-12    96       98.75         0.96
## Name: Zog
## Gender: male
## DOB: 1988-12-14
##   date_of_admission pulse temperature fluid_intake
## 1        2010-02-24    84       98.54         0.40
## 2        2013-03-25    69       98.49         0.81
## 3        2013-07-29    75       98.82         0.59
## 4        2013-10-27    80       98.74         0.28
## Name: Yol
## Gender: male
## DOB: 1986-03-11
##   date_of_admission pulse temperature fluid_intake
## 1        2010-02-22    84       98.87         0.39
## 2        2011-12-27    89       98.27         0.97
## 3        2012-03-10    87       98.78         0.12
## 4        2012-11-26    92       98.26         0.14
## 5        2013-03-24    78       98.44         0.13
## 6        2014-01-28    69       98.29         0.03
## Name: Yak
## Gender: female
## DOB: 1983-09-15
##   date_of_admission pulse temperature fluid_intake
## 1        2011-07-19    75       98.58         0.60
## 2        2012-04-07    88       97.53         0.29
## 3        2012-07-11    81       99.11         0.66
## 4        2012-08-30    90       98.58         0.26
## Name: Gaf
## Gender: female
## DOB: 1978-04-27
##   date_of_admission pulse temperature fluid_intake
## 1        2010-07-19    91       98.01         0.47
## 2        2011-05-03    90       98.61         0.36
## 3        2012-04-24    89       98.32         0.42
## 4        2012-08-06    77       98.96         0.74
## 5        2013-08-21    75       98.52         0.62
```

```
## Name: Kuw
## Gender: female
## DOB: 1980-11-07
##   date_of_admission pulse temperature fluid_intake
## 1        2010-10-03    82       98.49         0.12
## 2        2010-10-29    81       98.17         0.93
## 3        2011-09-16    72       98.21         0.29
## 4        2012-07-10    71       98.65         0.25
## Name: Mav
## Gender: female
## DOB: 1989-07-16
##   date_of_admission pulse temperature fluid_intake
## 1        2010-02-08    66       97.95         0.79
## 2        2010-04-19    88       98.00         0.50
## 3        2010-06-11    83       98.45         0.79
## 4        2012-03-02    63       99.07         0.01
## Name: Fel
## Gender: male
## DOB: 1985-08-16
##   date_of_admission pulse temperature fluid_intake
## 1        2010-09-26    81       98.51         0.24
## 2        2012-06-24    65       98.21         0.06
## Name: Say
## Gender: female
## DOB: 1974-09-22
##   date_of_admission pulse temperature fluid_intake
## 1        2010-03-14    77       98.54         0.15
```