Stats 102A - HW 2

Zooey Nguyen

2/8/2021

I would like to preface this assignment by saying I am sorry, I somehow lost track of this homework assignment and was only able to do a little bit of it, which is why it is clearly incomplete and rushed. It is very much not my best work. Thank you for grading, whoever is doing so.

```
source("105195172_stats102a_hw2.R")
library(tidyverse)
```

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

```
## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.0.4     v dplyr   1.0.4
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0
```

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

Problem 1

Part a

This first example is from the Fermi-LAT catalog I worked with in my research. The data is messy because not all cells have a single value. For example, they include "Other Observatory Detections" as a column, and not all GRBs have comments on them. In the redshift column there may be multiple values listed in a single cell because of conflicting calculations of redshift, and in fact, most GRBs have no spectral analysis done at all to calculate redshift.

```
grbs <- read_tsv("grb_table.txt")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_character()
## )
## i Use `spec()` for the full column specifications.
```

```
## Warning: 200 parsing failures.
## row col   expected     actual            file
##   3  -- 35 columns 33 columns 'grb_table.txt'
##  17  -- 35 columns 33 columns 'grb_table.txt'
##  19  -- 35 columns 33 columns 'grb_table.txt'
##  21  -- 35 columns 33 columns 'grb_table.txt'
##  26  -- 35 columns 33 columns 'grb_table.txt'
## ... ... .......... .......... ................
## See problems(...) for more details.
```

```
grbs_example <- grbs[c("GRB", "Time [UT]", "Other Observatory Detections", "Redshift")]
head(grbs_example, 10)
```

```
## # A tibble: 10 x 4
##    GRB     `Time [UT]` `Other Observatory Detections`                    Redshift
##    <chr>   <chr>       <chr>                                            <chr>
##  1 210207B 21:52:08    "Devasthal Optical Telescope (3.6m), Konus-Wind~ <NA>
##  2 210205A 11:11:17     <NA>                                            <NA>
##  3 210204A 06:29:25     <NA>                                            <NA>
##  4 210119A 02:54:09    "Fermi (GBM), MASTER-OAFA, Insight-HXMT/HE, GEC~ <NA>
##  5 210112A 01:37:03    "AGILE, OSN (1.5m), Konus-Wind, CAHA (2.2m), In~ <NA>
##  6 210104B 21:10:10     <NA>                                            <NA>
##  7 210104A 11:26:59    "Fermi (GBM), Xinglong GWAC-F60A, BOOTES-4/MET ~ <NA>
##  8 210102C 20:38:11    "Fermi (GBM)"                                    <NA>
##  9 210102B 09:03:48     <NA>                                            <NA>
## 10 201229A 10:59:30    "SAO RAS (1m), AbAO AS-32"                       <NA>
```

Part b

This data might be better arranged if flexible or blank observations were excluded from the dataset completely, since it creates messiness when you need to conduct analysis on, say, GRBs with redshift, or when you need statistics on actually relevant columns to analysis, which the commentary columns do not help. Here is how we might fix the example dataset:

```
clean_df <- grbs_example %>%
  drop_na("Redshift") %>%
  separate("Redshift", into=c("Redshift", "Technique"), sep="\\(") %>%
  select(-c("Other Observatory Detections", "Technique"))
```

```
## Warning: Expected 2 pieces. Additional pieces discarded in 104 rows [8, 13, 17,
## 21, 23, 35, 47, 50, 52, 56, 57, 62, 68, 70, 77, 80, 86, 101, 106, 118, ...].
```

```
## Warning: Expected 2 pieces. Missing pieces filled with `NA` in 3 rows [59, 212,
## 374].
```

```
head(clean_df, 10)
```

```
## # A tibble: 10 x 3
##    GRB     `Time [UT]` Redshift
##    <chr>   <chr>       <chr>
##  1 201221D 23:06:34    "1.046 "
##  2 201221A 07:09:01    "5.70 "
##  3 201216C 23:07:31    "1.10 "
##  4 201104B 17:33:46    "1.954 "
##  5 201024A 02:48:59    "0.999 "
##  6 201021C 20:27:18    "1.070 "
##  7 201020A 05:47:26    "2.903 "
##  8 201015A 22:50:13    "0.426 "
##  9 201014A 22:48:38    "4.56 "
## 10 200829A 13:59:34    "1.25 "
```

Now we have a nice clean dataset for gamma-ray bursts and their corresponding calculated redshifts.

Problem 2

Part a

Simulating the gradebook dataset.

```
set.seed(105195172)
gradebook_df <- data.frame(matrix(ncol=14, nrow=150))
colnames(gradebook_df) <- c("UID", paste("Homework", 1:10, sep="_"), paste("Exam", 1:3, sep="_"))
UID <- 123456789
for (i in 1:150) {
  UID <- UID + 1
  grades <- rnorm(13, mean=70, sd=15)
  grades[grades > 100] <- 100
  grades[grades < 0] <- 0
  grades <- round(grades, digits=0)
  gradebook_df[i,] <- c(UID, grades)
}
gradebook <- as_tibble(gradebook_df)
```

Part b

Randomly replacing 10% of Homework_10 and 5% of Exam_3 by NA.

```
hw_na <- sample(1:150, 15)
exam_na <- sample(1:150, 8)
gradebook[hw_na, "Homework_10"] <- NA
gradebook[exam_na, "Exam_3"] <- NA
```

Now to verify that we did indeed replace the right numbers of each column.

```
hw_na_count <- sum(is.na(gradebook$Homework_10))
exam_na_count <- sum(is.na(gradebook$Exam_3))
print(hw_na_count / 150)
```

```
## [1] 0.1
```

```
print(exam_na_count / 150)
```

```
## [1] 0.05333333
```

Yep, that is 10% of the Homework 10 column and 5% of the Exam 3 column that is NA.

Part c

Pseudocode for messy_impute.

Part d

```
new_gradebook <- gradebook
messy_impute(new_gradebook, mean, 1)
```

```
## # A tibble: 150 x 14
##       UID Homework_1 Homework_2 Homework_3 Homework_4 Homework_5 Homework_6
##     <dbl>      <dbl>      <dbl>      <dbl>      <dbl>      <dbl>      <dbl>
##  1 1.23e8         55         56         55         90         63         54
##  2 1.23e8         74         85         97         82         72         71
##  3 1.23e8         35         75         71         74         77         55
##  4 1.23e8         87         69         79         67         72         75
```

```
##  5 1.23e8         59         61         71         48         54         58
##  6 1.23e8         69         74         55         52         62         74
##  7 1.23e8         75         56         96         66         64         59
##  8 1.23e8         64         84         66         70         70         75
##  9 1.23e8         82         68         74         51         59         48
## 10 1.23e8         98         79         62         59         42         53
## # ... with 140 more rows, and 7 more variables: Homework_7 <dbl>,
## #   Homework_8 <dbl>, Homework_9 <dbl>, Homework_10 <dbl>, Exam_1 <dbl>,
## #   Exam_2 <dbl>, Exam_3 <dbl>
```