Assignment1

Jordan Whitaker

6/6/2020

options(tidyverse.quiet = TRUE)  
library(tidyverse)

## Warning: package 'tidyverse' was built under R version 3.6.3

## Warning: package 'ggplot2' was built under R version 3.6.3

## Warning: package 'tidyr' was built under R version 3.6.3

## Warning: package 'readr' was built under R version 3.6.3

## Warning: package 'dplyr' was built under R version 3.6.3

## Warning: package 'stringr' was built under R version 3.6.3

library(VIM)

## Warning: package 'VIM' was built under R version 3.6.3

## Loading required package: colorspace

## Loading required package: grid

## Loading required package: data.table

##   
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':  
##   
## between, first, last

## The following object is masked from 'package:purrr':  
##   
## transpose

## VIM is ready to use.   
## Since version 4.0.0 the GUI is in its own package VIMGUI.  
##   
## Please use the package to use the new (and old) GUI.

## Suggestions and bug-reports can be submitted at: https://github.com/alexkowa/VIM/issues

##   
## Attaching package: 'VIM'

## The following object is masked from 'package:datasets':  
##   
## sleep

library(mice)

## Warning: package 'mice' was built under R version 3.6.3

##   
## Attaching package: 'mice'

## The following objects are masked from 'package:base':  
##   
## cbind, rbind

grades <- read\_csv("class-grades.csv")

## Parsed with column specification:  
## cols(  
## Prefix = col\_double(),  
## Assignment = col\_double(),  
## Tutorial = col\_double(),  
## Midterm = col\_double(),  
## TakeHome = col\_double(),  
## Final = col\_double()  
## )

summary(grades)

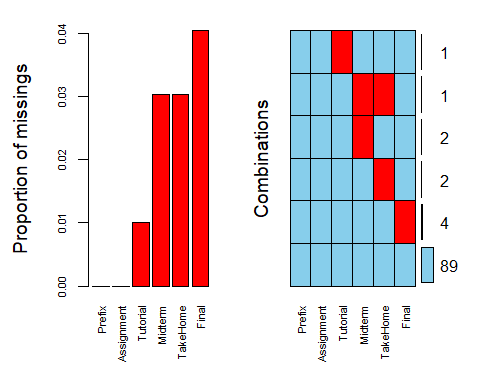
## Prefix Assignment Tutorial Midterm   
## Min. :4.000 Min. : 28.14 Min. : 34.09 Min. : 28.12   
## 1st Qu.:7.000 1st Qu.: 80.88 1st Qu.: 83.93 1st Qu.: 52.50   
## Median :8.000 Median : 89.94 Median : 93.37 Median : 69.38   
## Mean :7.313 Mean : 85.49 Mean : 89.79 Mean : 67.70   
## 3rd Qu.:8.000 3rd Qu.: 95.00 3rd Qu.:100.56 3rd Qu.: 81.56   
## Max. :8.000 Max. :100.83 Max. :112.58 Max. :110.00   
## NA's :1 NA's :3   
## TakeHome Final   
## Min. : 16.91 Min. : 28.06   
## 1st Qu.: 69.91 1st Qu.: 52.91   
## Median : 88.42 Median : 66.11   
## Mean : 81.12 Mean : 68.23   
## 3rd Qu.: 99.07 3rd Qu.: 83.61   
## Max. :108.89 Max. :108.89   
## NA's :3 NA's :4

str(grades)

## Classes 'spec\_tbl\_df', 'tbl\_df', 'tbl' and 'data.frame': 99 obs. of 6 variables:  
## $ Prefix : num 5 8 8 7 8 7 8 7 8 7 ...  
## $ Assignment: num 57.1 95 83.7 81.2 91.3 ...  
## $ Tutorial : num 34.1 105.5 83.2 96.1 93.6 ...  
## $ Midterm : num 64.4 67.5 30 49.4 95 ...  
## $ TakeHome : num 51.5 99.1 63.1 105.9 107.4 ...  
## $ Final : num 52.5 68.3 48.9 80.6 73.9 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. Prefix = col\_double(),  
## .. Assignment = col\_double(),  
## .. Tutorial = col\_double(),  
## .. Midterm = col\_double(),  
## .. TakeHome = col\_double(),  
## .. Final = col\_double()  
## .. )

Task 1: There are a total of 11 NA’s in the dataset. These NA’s exist in Tutorial, Midterm, TakeHome, and Final.

vim\_plot = aggr(grades, numbers = TRUE, prop=c(TRUE, FALSE), cex.axis=.7)

 Task 2: The above VIM plot shows that there is not systematic missingness. There is only one occurance of a student missing multiple pieces of data. In that instance, the two missing pieces of data are from the midterm and the takehome variables. The proportion of missingness is quite low as well with the highest proportion being .004

row\_wise\_grades <- grades %>%  
 drop\_na(Tutorial, Midterm, TakeHome, Final)

Task 3: After removing NA’s using row-wise deletion, there are now 89 observations in the newly created dataframe, row\_wise\_grades.

column\_wise\_grades <- grades %>%  
 select(-Tutorial, -Midterm, -TakeHome, -Final)

Task 4: After nuking the columns that had NA values, once the fallout cleared, only two variables were left standing. Those are Prefix and Assignment.

Task 5: When comparing the implementation of row-wise deletion versus column-wise deletion, for this specific dataset, my opinion is that row-wise deletion is the best way to manage missingness. The dataframe did not have copious amounts of variables that would make up for the removal of column-wise deletion.

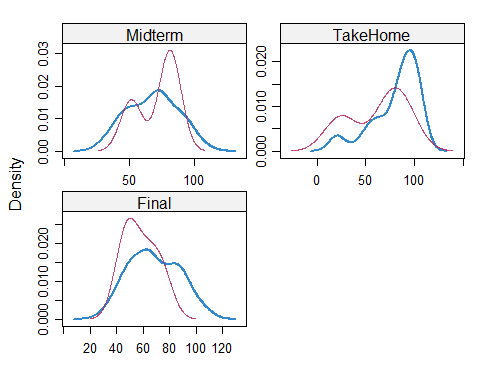
grades\_imp = mice(grades, m=1, method = "pmm", seed = 12345)

##   
## iter imp variable  
## 1 1 Tutorial Midterm TakeHome Final  
## 2 1 Tutorial Midterm TakeHome Final  
## 3 1 Tutorial Midterm TakeHome Final  
## 4 1 Tutorial Midterm TakeHome Final  
## 5 1 Tutorial Midterm TakeHome Final

#in line above: m=1 -> runs one imputation, seed sets the random number seed to get repeatable results  
summary(grades\_imp)

## Class: mids  
## Number of multiple imputations: 1   
## Imputation methods:  
## Prefix Assignment Tutorial Midterm TakeHome Final   
## "" "" "pmm" "pmm" "pmm" "pmm"   
## PredictorMatrix:  
## Prefix Assignment Tutorial Midterm TakeHome Final  
## Prefix 0 1 1 1 1 1  
## Assignment 1 0 1 1 1 1  
## Tutorial 1 1 0 1 1 1  
## Midterm 1 1 1 0 1 1  
## TakeHome 1 1 1 1 0 1  
## Final 1 1 1 1 1 0

densityplot(grades\_imp)



#red imputed, blue original, only shows density plots when more than 1 value the variable was imputed  
#note that the density plots are fairly uninteresting given the small amount of missing data  
grades\_complete = complete(grades\_imp)  
summary(grades\_complete)

## Prefix Assignment Tutorial Midterm   
## Min. :4.000 Min. : 28.14 Min. : 34.09 Min. : 28.12   
## 1st Qu.:7.000 1st Qu.: 80.88 1st Qu.: 84.69 1st Qu.: 52.50   
## Median :8.000 Median : 89.94 Median : 93.10 Median : 69.38   
## Mean :7.313 Mean : 85.49 Mean : 89.76 Mean : 67.80   
## 3rd Qu.:8.000 3rd Qu.: 95.00 3rd Qu.:100.55 3rd Qu.: 81.88   
## Max. :8.000 Max. :100.83 Max. :112.58 Max. :110.00   
## TakeHome Final   
## Min. : 16.91 Min. : 28.06   
## 1st Qu.: 67.96 1st Qu.: 52.09   
## Median : 87.96 Median : 65.56   
## Mean : 80.54 Mean : 67.81   
## 3rd Qu.: 98.42 3rd Qu.: 83.19   
## Max. :108.89 Max. :108.89

Task 7: Dealing with missing data is common and typically requires the deployment of multiple approaches to determine an appropriate solution. Issues that can occur when dealing with missing data is that the data scientist may inject data that will further prove their hypothesis. This bias could lead to a misrepresentation of the data. Although, there are some more objective ways to manage missing data, a technique that comes to mind is using KNN (K Nearest Neighbors), to measure some distance and their average to impute the missing data.