HW7

Duc Le

11/24/2020

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(outliers)
library(naniar)
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
library(mice)
```

```
##
## Attaching package: 'mice'
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

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

```
library(ggplot2)
library(EnvStats)
```

```
##
## Attaching package: 'EnvStats'
```

```
## The following object is masked from 'package:MASS':
##
##     boxcox
```

```
## The following objects are masked from 'package:stats':
```

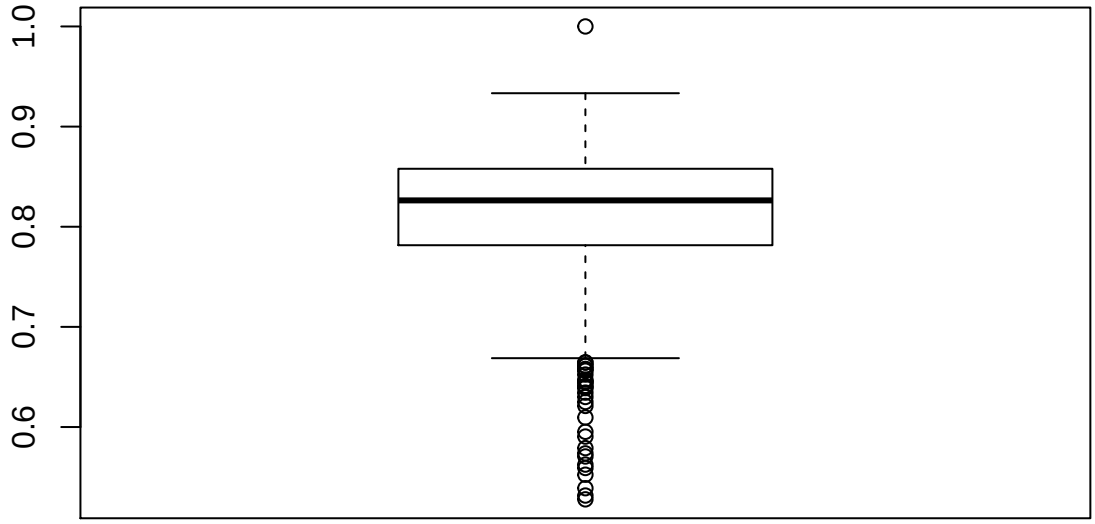
```
##
##     predict, predict.lm

## The following object is masked from 'package:base':
##
##     print.default
```

```
setwd("C:/Users/Duker/Desktop/Fall 2020/CS 614/Homework/HW7")
load("HW7.Rdata")
```

Problem 1

```
attach(demo)
# Boxplot
boxplot(Prof_Score, data = demo)
```




```
bx.o = boxplot.stats(Prof_Score)$out

# Z-score
ol_SD = function(z){
  z = na.omit(z)
  z.score = scale(z)
  w = which(abs(z.score) > 3)
  return(z[w])
}

z.o = ol_SD(Prof_Score)
```

```
# Rosner Test
out = rosnerTest(Prof_Score, 10)
out$n.outliers
```

```
## [1] 6
```

```
length(bx.o)
```

```
## [1] 32
```

```
length(z.o)
```

```
## [1] 12
```

The boxplot method detects almost 3 times as many outliers as the z-score method. The Rosner Test only shows 6 outliers. The discrepancy in outliers detected between the boxplot & z-score methods was justifyable since the boxplot's threshold is the median & z-score's threshold is the mean. I would think it would require ether more domain knowledge of the data or better visualization to approach outliers with the Rosner Test since you do have to start with an arbitrary "k" outliers.

```
paste("The outliers detected by the boxplot range from", round(min(bx.o),3), "to",
      round(max(bx.o),3))
```

```
## [1] "The outliers detected by the boxplot range from 0.528 to 1"
```

```
paste("The outliers detected by the z-score method range from", round(min(z.o),3), "to",
      round(max(z.o),3))
```

```
## [1] "The outliers detected by the z-score method range from 0.528 to 0.61"
```

I would treat future outliers depending on the distribution of the data. If the data has high variance, I would maybe approach outliers using the z-score. The reason is if the outliers are clustered together, the boxplot may not be able to detect them.

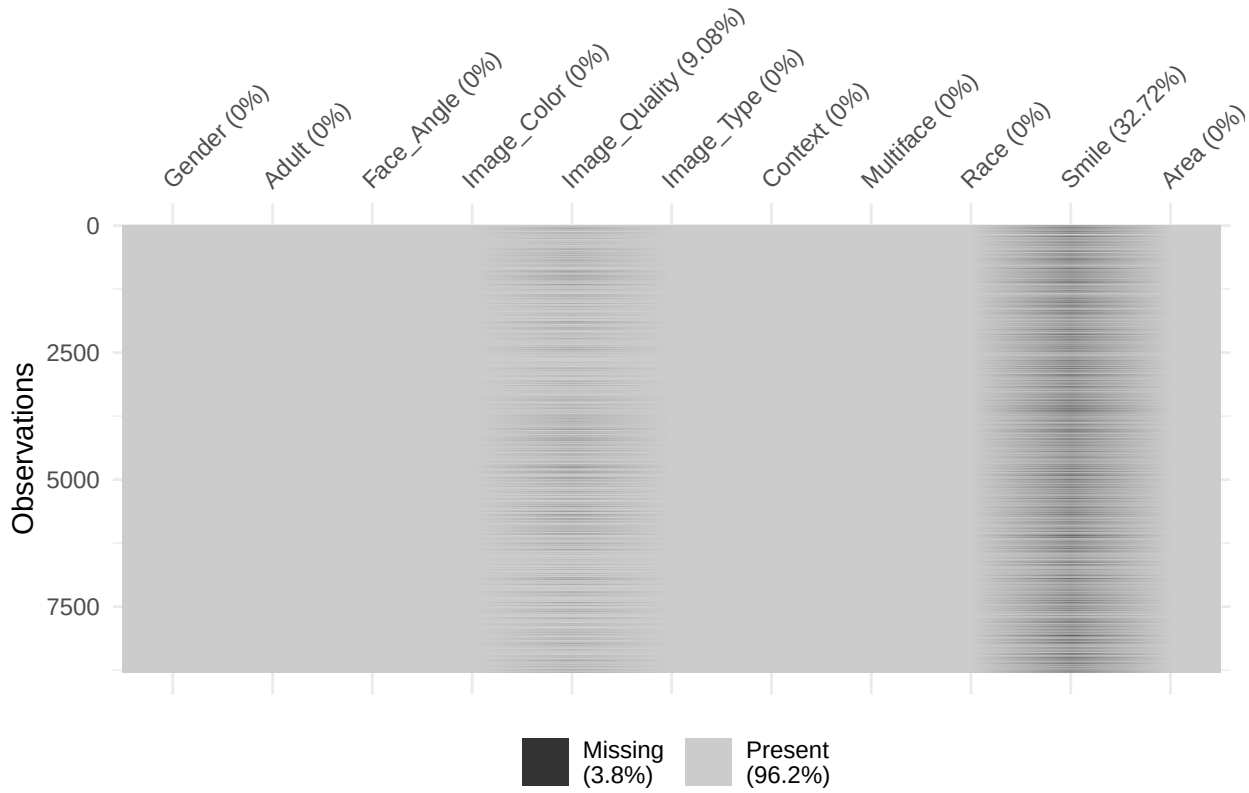
Problem 2

```
summary(df)
```

```
##      Gender          Adult          Face_Angle      Image_Color   Image_Quality
##  female :2300   Min.   :0.0000   Min.   :0.0000   Min.   :0.00   fair:3137
##  male   :6402   1st Qu.:1.0000   1st Qu.:0.0000   1st Qu.:0.00   good:4852
##  unknown:  85   Median :1.0000   Median :1.0000   Median :0.00   NA's: 798
##                 Mean   :0.9332   Mean   :0.7165   Mean   :0.44
##                 3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.00
##                 Max.   :1.0000   Max.   :1.0000   Max.   :1.00
##
##    Image_Type        Context       Multiface                  Race
##  Min.   :0.0000   ad     :2639   Min.   :0.0000   americanindian :  71
##  1st Qu.:1.0000   author : 185   1st Qu.:0.0000   asian          : 413
##  Median :1.0000   cover  : 353   Median :1.0000   black          : 593
##  Mean   :0.8676   feature:5610   Mean   :0.5542   pacificislander:  74
##  3rd Qu.:1.0000                  3rd Qu.:1.0000   unknown        : 155
##  Max.   :1.0000                  Max.   :1.0000   white          :7481
##
##      Smile             Area
##  Min.   :0.0000   Min.   :    594
##  1st Qu.:0.0000   1st Qu.:   8100
```

```
##  Median :0.0000   Median :  14140
##  Mean   :0.4367   Mean   :  35250
##  3rd Qu.:1.0000   3rd Qu.:  31465
##  Max.   :1.0000   Max.   :1576872
##  NA's   :2875
```

```
vis_miss(df)
```




```
IQ_mis = factor(ifelse(is.na(df$Image_Quality), 1, 0))
smile_mis = factor(ifelse(is.na(df$Smile), 1, 0))

chisq.test(df$Image_Type, IQ_mis)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  df$Image_Type and IQ_mis
## X-squared = 129.52, df = 1, p-value < 2.2e-16
```

```
chisq.test(df$Image_Color, smile_mis)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  df$Image_Color and smile_mis
## X-squared = 5.9849, df = 1, p-value = 0.01443
```

There is some statistical significance from Chi-Sq Test when comparing NA's from Image Quality & Smile

vs. Image Type & Image Color. Thus, the NA's can be classified as MAR.

```
lw.df = na.omit(df)
```

List-wise Deletion

```
imp = mice(df, maxit = 5, print =  FALSE)

impute.df = complete(imp, method = logreg, include = F)
```

Multiple Imputation

```
lin.reg1 = lm(Area~., data = lw.df)
final1 = stepAIC(lin.reg1, trace = 0)
summary(final1)
```

Linear Regression Model

```
##
## Call:
## lm(formula = Area ~ Gender + Image_Color + Image_Quality + Image_Type +
##     Context + Multiface + Smile, data = lw.df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -155988  -25093   -9856    8230 1447004
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)          32041       4356   7.355 2.20e-13 ***
## Gendermale            7061       2441   2.893  0.00384 **
## Genderunknown        64108      10209   6.280 3.66e-10 ***
## Image_Color           3691       2124   1.738  0.08224 .
## Image_Qualitygood    23922       2178  10.982  < 2e-16 ***
## Image_Type            5740       3376   1.700  0.08915 .
## Contextauthor       -22429       9795  -2.290  0.02207 *
## Contextcover         68483       5361  12.774  < 2e-16 ***
## Contextfeature      -16009       2656  -6.027 1.78e-09 ***
## Multiface           -25590       2137 -11.973  < 2e-16 ***
## Smile                -4749       2194  -2.165  0.03045 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 76520 on 5380 degrees of freedom
## Multiple R-squared:  0.1144, Adjusted R-squared:  0.1128
## F-statistic: 69.51 on 10 and 5380 DF,  p-value: < 2.2e-16
```

```
lin.reg2 = lm(Area~., data = impute.df)
final2 = stepAIC(lin.reg2, trace = 0)
summary(final2)
```

```
##
## Call:
```

```
## lm(formula = Area ~ Gender + Image_Color + Image_Quality + Image_Type +
##     Context + Multiface + Smile, data = impute.df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -151761  -26392  -10176    7552 1466523
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)          30631       3344   9.159  < 2e-16 ***
## Gendermale            7046       1920   3.670 0.000244 ***
## Genderunknown        70605       8510   8.297  < 2e-16 ***
## Image_Color           5725       1675   3.419 0.000632 ***
## Image_Qualitygood    24538       1718  14.279  < 2e-16 ***
## Image_Type            8974       2560   3.505 0.000458 ***
## Contextauthor       -23102       5908  -3.910 9.29e-05 ***
## Contextcover         54508       4396  12.401  < 2e-16 ***
## Contextfeature      -19219       1963  -9.790  < 2e-16 ***
## Multiface           -24152       1689 -14.299  < 2e-16 ***
## Smile                -5971       1721  -3.470 0.000523 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 76890 on 8776 degrees of freedom
## Multiple R-squared:  0.09951,    Adjusted R-squared:  0.09848
## F-statistic: 96.98 on 10 and 8776 DF,  p-value: < 2.2e-16
```

There isn't a significant difference between the R-squared scores from the 2 models. The R-squared for the list-wise deleted dataset is ~0.11 vs. the imputed dataset's ~0.09.The poor R-squared scores could also be the results of a dataset that's incompatible with linear regression (categorical explanatory variables vs. continuous response variable).