Module 1 homework

Filipp Krasovsky

5/17/2021

1. (30 points) The UC Irvine Machine Learning Repository contains a data set related to glass identification. The data consist of 214 glass samples labeled as one of seven class categories. There are nine predictors, including the refractive index and percentages of eight elements: Na, Mg, Al, Si, K, Ca, Ba, and Fe. The data can be accessed via: library(mlbench) data(Glass)

a. Using visualizations, explore the predictor variables to understand their distributions as well as the relationships between predictors.
b. Do there appear to be any outliers in the data? Are any predictors skewed?
c. Are there any relevant transformations of one or more predictors that might improve the classification model?

load in our dataset:

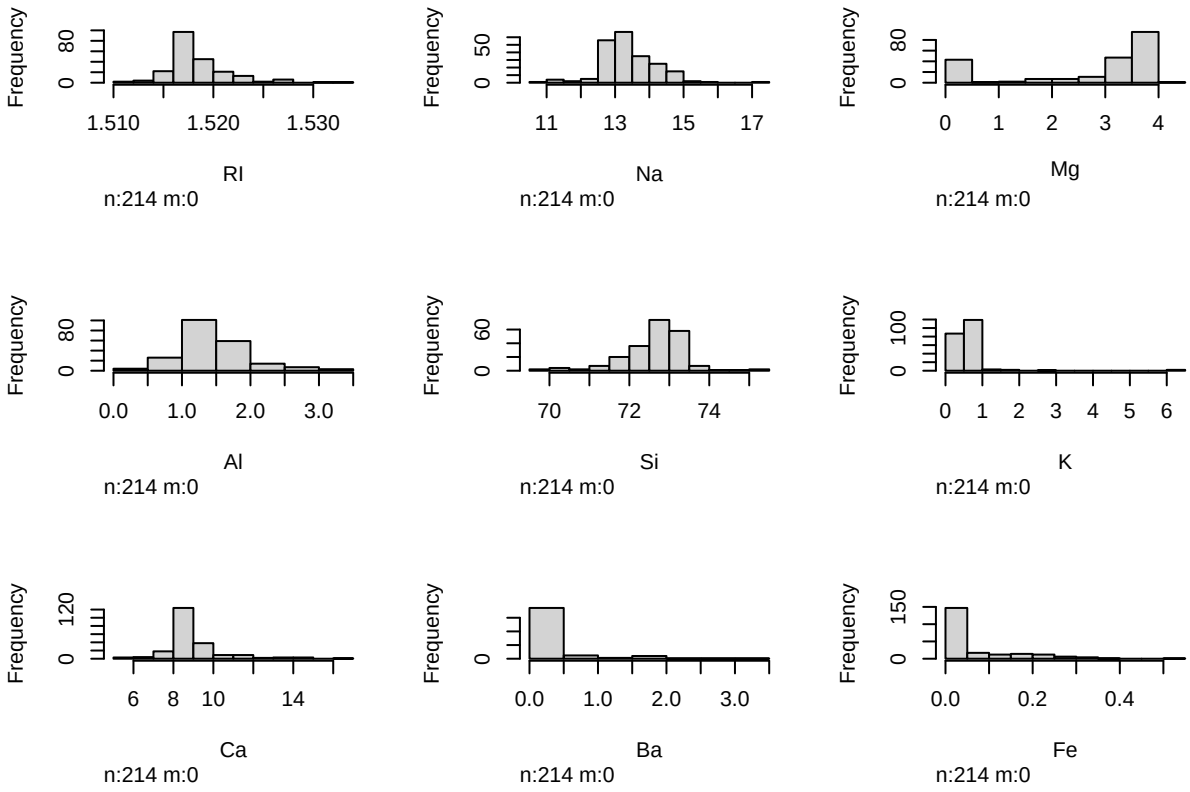
```
library(mlbench)
```

```
## Warning: package 'mlbench' was built under R version 4.0.5
```

```
data(Glass)
df = as.data.frame(Glass)
predictors = subset(df,select=-c(Type))
```

part a: we can begin the visualization process by looking at a histogram of each predictor and their skewness values for validation.

```
library(Hmisc)
hist.data.frame(predictors)
```




```
require(e1071)
```

```
## Loading required package: e1071
```

```
##
## Attaching package: 'e1071'
```

```
## The following object is masked from 'package:Hmisc':
##
##     impute
```

```
svals <- apply(predictors,2,skewness)
print(svals)
```

```
##         RI         Na         Mg         Al         Si          K         Ca
##  1.6027151  0.4478343 -1.1364523  0.8946104 -0.7202392  6.4600889  2.0184463
##         Ba         Fe
##  3.3686800  1.7298107
```

Our findings validate that none of our predictors have a normal distribution, with many of our predictors being either right skewed or left skewed. In particular, RI, NA, AL, K, CA, BA, and FE are right sekewed, with BA, CA, and K being the most skewed among these. Mg and Si are the left skewed predictors. Of all predictors Si and Na appear closest to having a normal distribution.

To further substantiate the claim none of our distributions are normal, we can run the Shapiro-Wilks test, where the null hypothesis is that the distribution is normal. therefore, if our p-value is not greather than 0.05, we reject the null hypothesis in favor of the claim that the distribution isn't normal.

```
for (i in names(predictors)){
    pval <- shapiro.test(predictors[,i])[2]
    result<-ifelse(pval >=0.05,"Normal","Not Normal")
    print(paste("shapiro test for:",i,"=",result,"pvalue:",pval))
}
```

```
## [1] "shapiro test for: RI = Not Normal pvalue: 1.0766713449726e-12"
## [1] "shapiro test for: Na = Not Normal pvalue: 3.4655430546966e-07"
## [1] "shapiro test for: Mg = Not Normal pvalue: 2.39092121727115e-19"
## [1] "shapiro test for: Al = Not Normal pvalue: 2.08315629600399e-07"
## [1] "shapiro test for: Si = Not Normal pvalue: 2.17503176825416e-09"
## [1] "shapiro test for: K = Not Normal pvalue: 2.17218809927206e-25"
## [1] "shapiro test for: Ca = Not Normal pvalue: 4.28658420594697e-16"
## [1] "shapiro test for: Ba = Not Normal pvalue: 5.38330203908191e-26"
## [1] "shapiro test for: Fe = Not Normal pvalue: 1.15666802873704e-20"
```

Next, we can examine the relationship between predictors. First, we start with a heat-cluster correlogram:

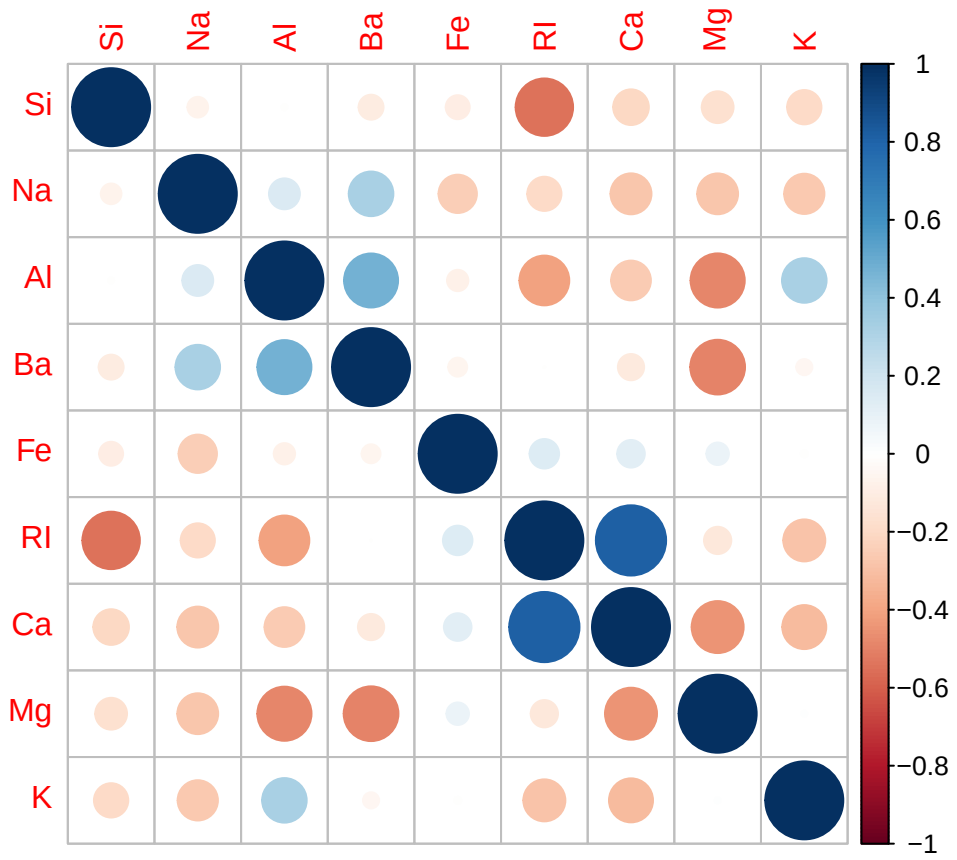
```
#a. visualize.
require(corrplot)
```

```
## Loading required package: corrplot
```

```
## Warning: package 'corrplot' was built under R version 4.0.5
```

```
## corrplot 0.88 loaded
```

```
#extract predictors into correlogram
predictors.cor = cor(predictors)
corrplot(predictors.cor,order="hclust")
```



Face value observation suggests a mildly strong relationship between SI and RI, RI and CA, AL and BA, Mg and Al, Mg and Ba, and Ca and MG. We may have to conduct PCA in order to reduce dimensionality and capture the variance between these predictors in our modeling.

For more detail, let's define a strong correlation as any value over 0.4. In that instance, we find the following relationships to be significant: (RI,AL) (RI,SI) (RI,SA) (MG,AL) (MG,CA) (MG,BA) (AL,BA)

We substantiate this with the matrix below:

```
#we can also pick a threshold such as 0.4 to decide if we have a colinearity issue.
#potentially problematic areas: RI->AL, RI->SI, RI->CA
#                               MG->AL, MG->CA, MG->BA
#                               AL->BA

thresh_cors = ifelse(abs(predictors.cor)>=0.4,1,0)
print(thresh_cors)
```

```
##    RI Na Mg Al Si K Ca Ba Fe
## RI  1  0  0  1  1 0  1  0  0
## Na  0  1  0  0  0 0  0  0  0
## Mg  0  0  1  1  0 0  1  1  0
## Al  1  0  1  1  0 0  0  1  0
## Si  1  0  0  0  1 0  0  0  0
## K   0  0  0  0  0 1  0  0  0
## Ca  1  0  1  0  0 0  1  0  0
## Ba  0  0  1  1  0 0  0  1  0
## Fe  0  0  0  0  0 0  0  0  1
```

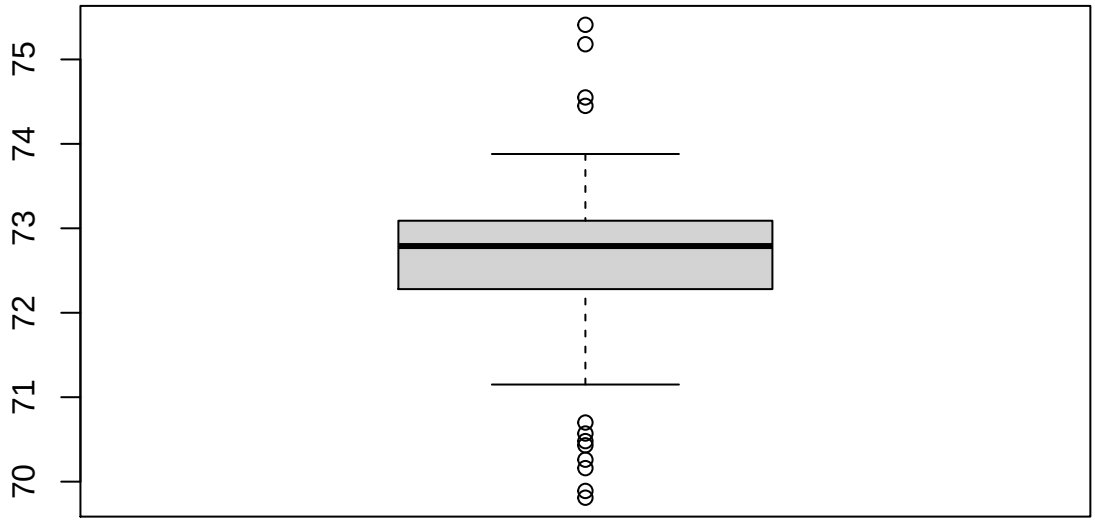
(b) Are any predictors skewed? Are there any outliers?

(see skewness output above)

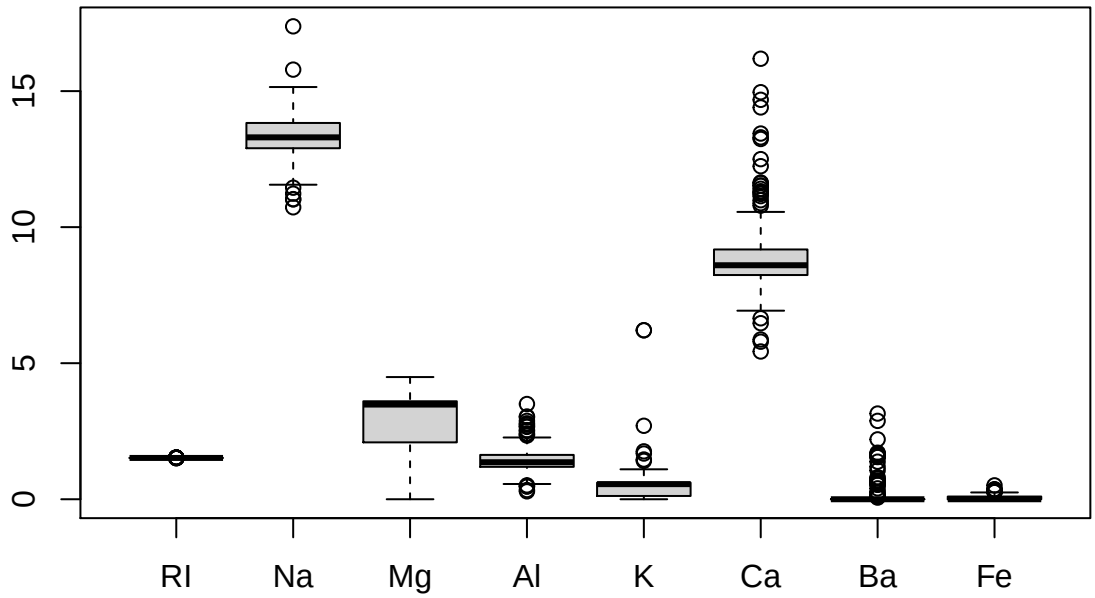
Our findings validate that none of our predictors have a normal distribution, with many of our predictors being either right skewed or left skewed. In particular, RI, NA, AL, K, CA, BA, and FE are right sekewed, with BA, CA, and K being the most skewed among these. Mg and Si are the left skewed predictors. Of all predictors Si and Na appear closest to having a normal distribution.

We can find out if our data has outliers by plotting a box-plot for each of our predictors. Since Si is on a much higher magnitude than all other predictors, we isolate it for more insight.

```
boxplot(predictors$Si)
```




```
boxplot(subset(predictors,select=-c(Si)))
```



Si, Na, Al, K, Ca, and Ba all have considerable a,ounts of outliers.

c. Are there any relevant transformations of one or more predictors that might improve the classification model?

for this, we look to the box-cox transformation module in R to determine what transformation we should apply for each predictors.

```
#(b^lambda)-1)/ lambda
require(caret)
```

```
## Loading required package: caret
```

```
## Warning: package 'caret' was built under R version 4.0.5
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:survival':
##
##     cluster
```

```
for ( i in names(predictors)){

  x = BoxCoxTrans(predictors[,i])
  print(paste(i,",power to raise to:",x$lambda))

}
```

```
## [1] "RI ,power to raise to: -2"
## [1] "Na ,power to raise to: -0.0999999999999999"
## [1] "Mg ,power to raise to: NA"
## [1] "Al ,power to raise to: 0.5"
## [1] "Si ,power to raise to: 2"
## [1] "K ,power to raise to: NA"
## [1] "Ca ,power to raise to: -1.1"
## [1] "Ba ,power to raise to: NA"
## [1] "Fe ,power to raise to: NA"
```

Thus far, we know that the following transformations can help us: 1. raising NA to the power of -2 2. raising NA to the power of -.1 3. square root transformation for Al 4. Squaring Si 5. Raising Ca to the power of -1.1

this leaves us with four nulls: Mg, K, Ba, and Fe - for which there may be a transformation, but it's not clear what kind. Ba, Fa, and K are heavily right-skewed, so much so that a lambda transformation may be infeasible. we can modify this by adding a very small amount to each zero in our dataset.

```
for (i in names(predictors)){

  predictors[,i] = ifelse(predictors[,i]==0,0.00000001,predictors[,i])

}
```

```
p<- preProcess(predictors,method='BoxCox')
predictors.t <- predict(p,predictors)
svals.t <- apply(predictors.t,2,skewness)
print(svals.t)
```

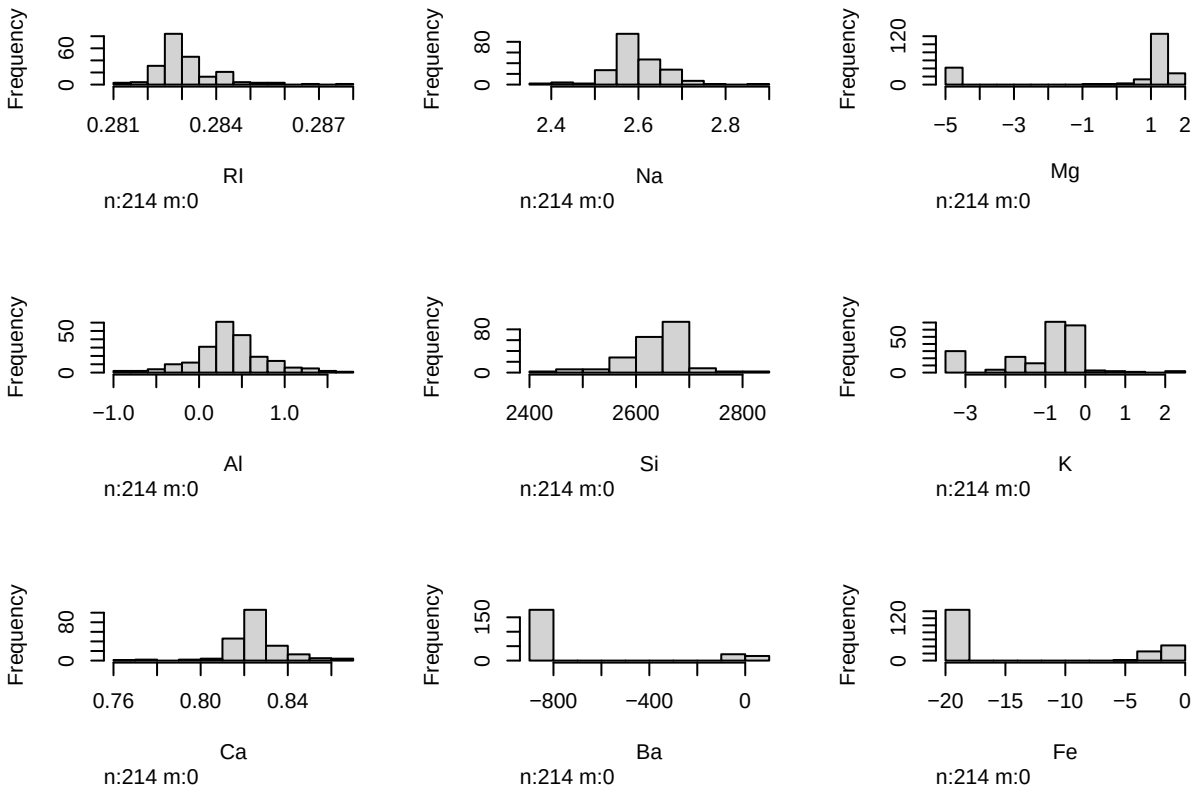
```
##          RI          Na          Mg          Al          Si           K
##  1.56566039  0.03384644 -1.47252731  0.09105899 -0.65090568 -0.80741391
##          Ca          Ba          Fe
## -0.19395573  1.67564949  0.73841949
```

```
print(svals)
```

```
##         RI         Na         Mg         Al         Si          K         Ca
##  1.6027151  0.4478343 -1.1364523  0.8946104 -0.7202392  6.4600889  2.0184463
##         Ba         Fe
##  3.3686800  1.7298107
```

As demonstrated, our Box-Cox transformation significantly reduces skewness for our predictors. We can create a histogram for validation:

```
hist.data.frame(predictors.t)
```



While some predictors remain heavily skewed, we did significantly reduce skewness for others.

2. (20 points) The image below shows a scatter plot matrix of the continuous features of a dataset. Discuss the relationships between the features in the dataset that this scatter plot highlights.

Life expectancy and infant mortality seem to show a strong negative correlation. Education and life expectancy do not seem to have any linear relationship, but it could be argue that a polynomial relationship can be tested for. Life expectancy and health also seem to share a nonlinear relationship, as do infant mortality and health. Health USD exhibits a similar trend with life expectancy, and possibly has a negative nonlinear relationship with infant mortality.

3. (10 points) Discuss the relationships shown in each visualizations:

a. The visualization below illustrates the relationship between Diastolic BP & Tachycardia, left most plot has data where Tachycardia = true & false.

Tachycardia is defined as a condition where the heart rate increases rapidly when sitting up or standing, which we assume can manifest itself in higher values of the Dialostic BP in this dataset. True enough, the distribution for DBP = false is contained primarily within the 60-120 range, while the distribution for individuals with Tachycardia extends well past 120 into the 150 range.

b. The visualization below illustrates the relationship between height & Tachycardia, left most plot has data where Tachycardia = true & false.

the most apparently visual takeaway is that the distribution for individuals with tachycardia tends to be more left skewed, while the opposite is true for regular individuals. this is intuitive given that the proclivity for having a larger height is consistent with the problems that would cause massive fluctuations in blood pressure, so taller individuals are more prone to tachycardia.

4. (30 points) Use the data at the UCI Machine Learning Repository web page (or download the "hcv-dat0.csv" file in Blackboard) and pick the numeric predictors (exclude columns X & Age) to perform the following analysis in R:

a. Are there any missing data in the predictors? If yes, summarize the missing data by each predictor.
b. Are there any predictors with skewed distributions?
c. Plot histograms of all predictors to observe skewness visually
d. Compute skewness using the skewness function from the e1071 package.
e. Apply box cox transformations to the data then recompute the skewness metrics and report the differences and does box cox transformation help?
f. Plot histograms of transformed predictors to observe changes to skewness visually.

```
#load in our dataset
df = read.csv("hcvdat0.csv",TRUE)
df = subset(df,select=-c(X))

#check for missing values
missing_values = df[rowSums(is.na(df)) > 0, ]
print(nrow(missing_values))
```

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

```
for ( i in names(missing_values)){
  print(
    paste(i,
          (nrow(subset(missing_values,is.na(missing_values[,i]))))
    )
  )
}
```

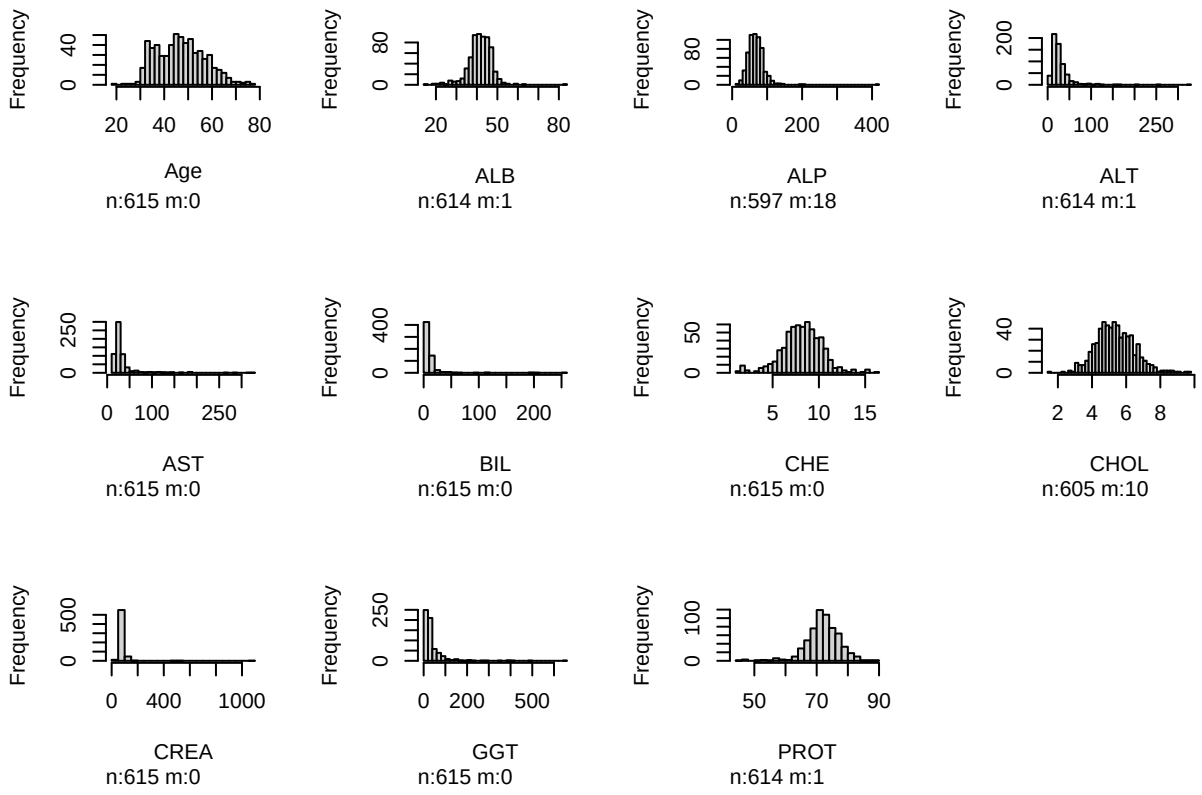
```
## [1] "Category 0"
## [1] "Age 0"
## [1] "Sex 0"
## [1] "ALB 1"
## [1] "ALP 18"
## [1] "ALT 1"
## [1] "AST 0"
## [1] "BIL 0"
## [1] "CHE 0"
## [1] "CHOL 10"
## [1] "CREA 0"
## [1] "GGT 0"
## [1] "PROT 1"
```

We have 26 missing values in total, broken down by the predictors above.

(b) to find predictors with skewed distributions, we can plot each predictor and look at their skewneess.

(c) [this also involves plotting the histograms.]

```
predictors = subset(df,select=-c(Category,Sex))
hist.data.frame(predictors)
```



Age appears to be bimodal and skewed. AST, ALT, ALP, BIL, CREA, and GGT are visibly right skewed. PROT is somewhat left skewed.

(d) Compute Skewness:

```
svals <- apply(predictors,2,na.rm=TRUE,skewness)
svals
```

```
##        Age        ALB        ALP        ALT        AST        BIL        CHE
##  0.2658328 -0.1759048  4.6315552  5.4792399  4.9162540  8.3445765 -0.1096956
##       CHOL       CREA        GGT       PROT
##  0.3739660 15.0953748  5.6052871 -0.9589839
```

Overall, we find that age, ALB, and CHE are all relatively close to normal distribution. ALP, ALT, AST, BIL, CREA, and GGT are heavily right skewed. PROT is left skewed.

(e) Apply box cox transformations to the data then recompute the skewness metrics and report the differences and does box cox transformation help?

```
p<- preProcess(predictors,method='BoxCox')
predictors.t <- predict(p,predictors)
svals.t <- apply(predictors.t,2,na.rm=TRUE,skewness)
print(svals.t)
```

```
##         Age         ALB         ALP         ALT         AST         BIL
## -0.02730266  0.32090260 -0.21816588 -0.42694320  0.05826619  0.05365506
##         CHE        CHOL        CREA         GGT        PROT
##  0.17614896  0.04167873  0.64684828  0.07373092 -0.45312501
```

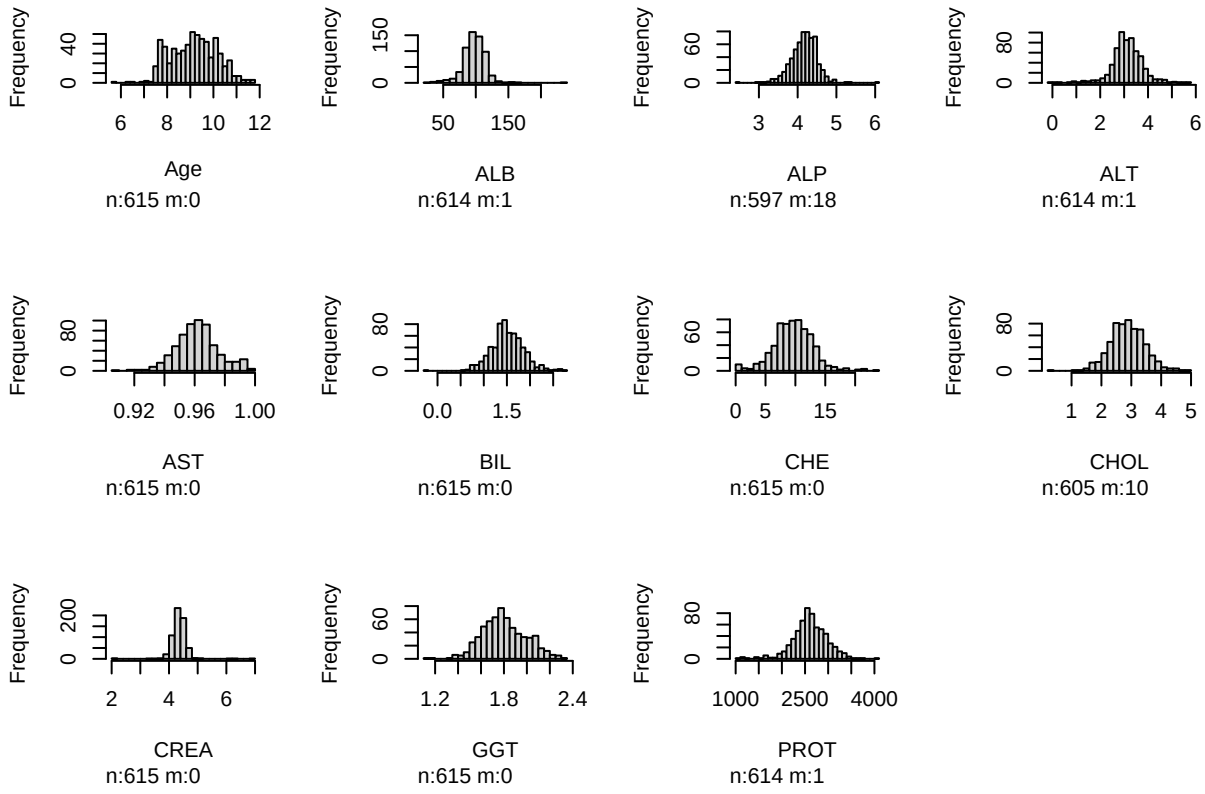
```
print(svals)
```

```
##        Age        ALB        ALP        ALT        AST        BIL        CHE
##  0.2658328 -0.1759048  4.6315552  5.4792399  4.9162540  8.3445765 -0.1096956
##       CHOL       CREA        GGT       PROT
##  0.3739660 15.0953748  5.6052871 -0.9589839
```

Applying the boxcox transformation considerably reduces the skewness of almost all of our predictors, while increasing it for others (ie CHE).

(f) Plot histograms of transformed predictors to observe changes to skewness visually.

```
hist.data.frame(predictors.t)
```



Overall, we can visually confirm notable improvements in skew.