The UC Irvine Machine Learning Repository contains a data set related to glass identification. The data consists 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

|  |
| --- |
| > #install.packages('mlbench')  > library(mlbench)  > data(Glass)  > str(Glass)  'data.frame': 214 obs. of 10 variables:  $ RI : num 1.52 1.52 1.52 1.52 1.52 ...  $ Na : num 13.6 13.9 13.5 13.2 13.3 ...  $ Mg : num 4.49 3.6 3.55 3.69 3.62 3.61 3.6 3.61 3.58 3.6 ...  $ Al : num 1.1 1.36 1.54 1.29 1.24 1.62 1.14 1.05 1.37 1.36 ...  $ Si : num 71.8 72.7 73 72.6 73.1 ...  $ K : num 0.06 0.48 0.39 0.57 0.55 0.64 0.58 0.57 0.56 0.57 ...  $ Ca : num 8.75 7.83 7.78 8.22 8.07 8.07 8.17 8.24 8.3 8.4 ...  $ Ba : num 0 0 0 0 0 0 0 0 0 0 ...  $ Fe : num 0 0 0 0 0 0.26 0 0 0 0.11 ...  $ Type: Factor w/ 6 levels "1","2","3","5",..: 1 1 1 1 1 1 1 1 1 1 ...  > |

1. Utilize suitable visualizations (employ any types of data visualization you deem appropriate) to explore the predictor variables, aiming to understand their distributions and relationships among them.
2. Do there appear to be any outliers in the data? Are any predictors skewed? Show all the work!
3. Are there any relevant transformations of one or more predictors that might improve the classification model? Show all the work!
4. Fit SVM model (You may refer to Chapter 4 material for details) using the following R codes: (This code will be discussed in detail in the following chapters)

|  |
| --- |
| #install.packages('kernlab')  library(kernlab)  set.seed(231)  sigDist <- sigest(Type~ ., data = Glass, frac = 1)  sigDist  90% 50% 10%  0.02622151 0.09278583 0.84623750  svmTuneGrid <- data.frame(sigma = as.vector(sigDist)[1], C = 2^(-2:10))  svmTuneGrid  sigma C  1 0.02622151 0.25  2 0.02622151 0.50  3 0.02622151 1.00  4 0.02622151 2.00  5 0.02622151 4.00  6 0.02622151 8.00  7 0.02622151 16.00  8 0.02622151 32.00  9 0.02622151 64.00  10 0.02622151 128.00  11 0.02622151 256.00  12 0.02622151 512.00  13 0.02622151 1024.00  #install.packages('AppliedPredictiveModeling')  library(AppliedPredictiveModeling)  library(caret) #access the train function.  set.seed(1056)  #It may take a while to run  svmFit <- train(Type~ ., data = Glass, method = "svmRadial",  preProc = c("center", "scale"),tuneGrid = svmTuneGrid,  trControl = trainControl(method = "repeatedcv", repeats = 5))  plot(svmFit, scales = list(x = list(log = 2))) |