DATA624: Homework 4

Donald Butler

2023-10-01

Contents

Homework 4 **1**

Exercise 3.1 1

Exercise 3.2 4

Homework 4

```
library(fpp3)
library(tidyverse)
library(corrplot)
library(mlbench)
library(inspectdf)    # factor plots
library(naniar)       # missing values by factor
```

Exercise 3.1

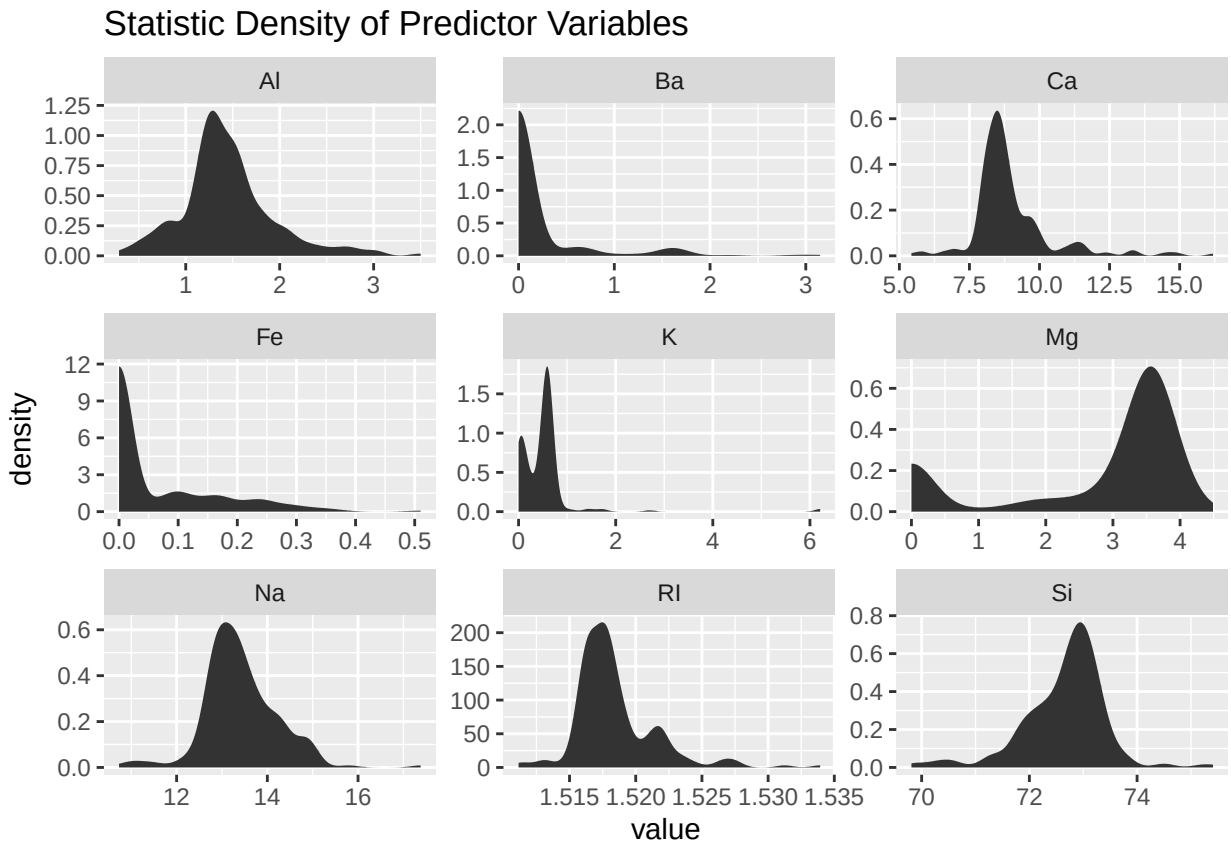
The UC Irvine Machine Learning Repository contains a data set related to glass identification. The data consist of 214 glass samples labeled as one of seven class categories. There are nine predictors, including the refractive index and percentages of eight elements: Na, Mg, Al, Si, K, Ca, Ba, and Fe.

```
data(Glass)
str(Glass)
```

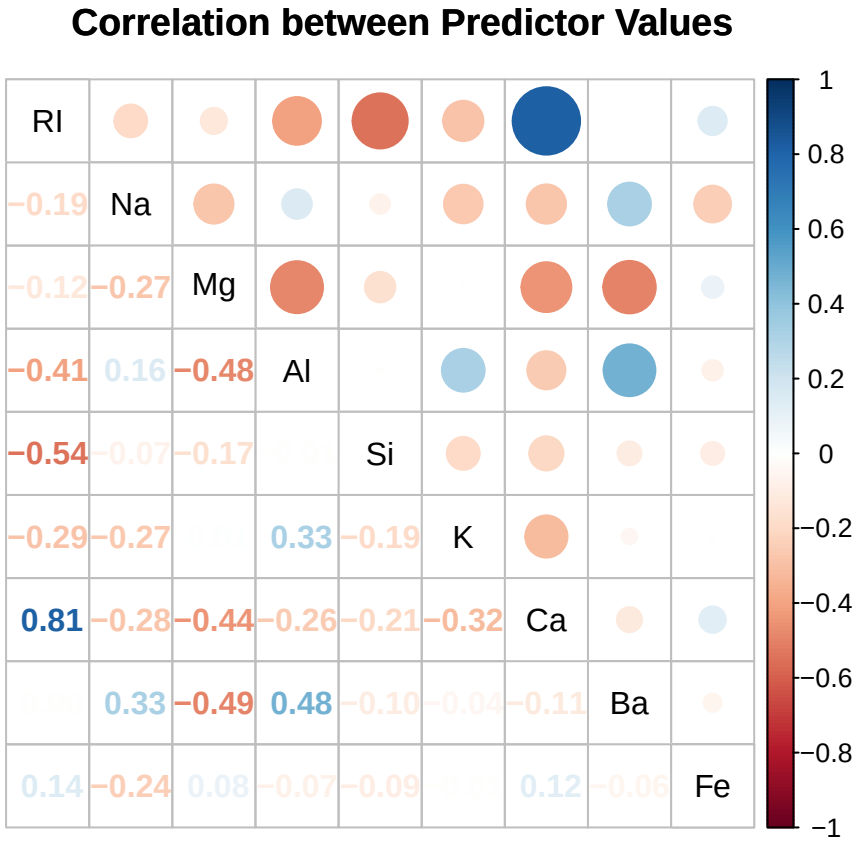
```
## 'data.frame':    214 obs. of  10 variables:
##  $ RI  : num  1.52 1.52 1.52 1.52 1.52 ...
##  $ Na  : num  13.6 13.9 13.5 13.2 13.3 ...
##  $ Mg  : num  4.49 3.6 3.55 3.69 3.62 3.61 3.6 3.61 3.58 3.6 ...
##  $ Al  : num  1.1 1.36 1.54 1.29 1.24 1.62 1.14 1.05 1.37 1.36 ...
##  $ Si  : num  71.8 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 ...
```

a. Using visualizations, explore the predictor variables to understand their distributions as well as the relationships between predictors.

```
Glass |>
  select(-Type) |>
  gather() |>
  ggplot(aes(value)) +
  stat_density() +
  labs(title = "Statistic Density of Predictor Variables") +
  facet_wrap(~key, scales = 'free')
```

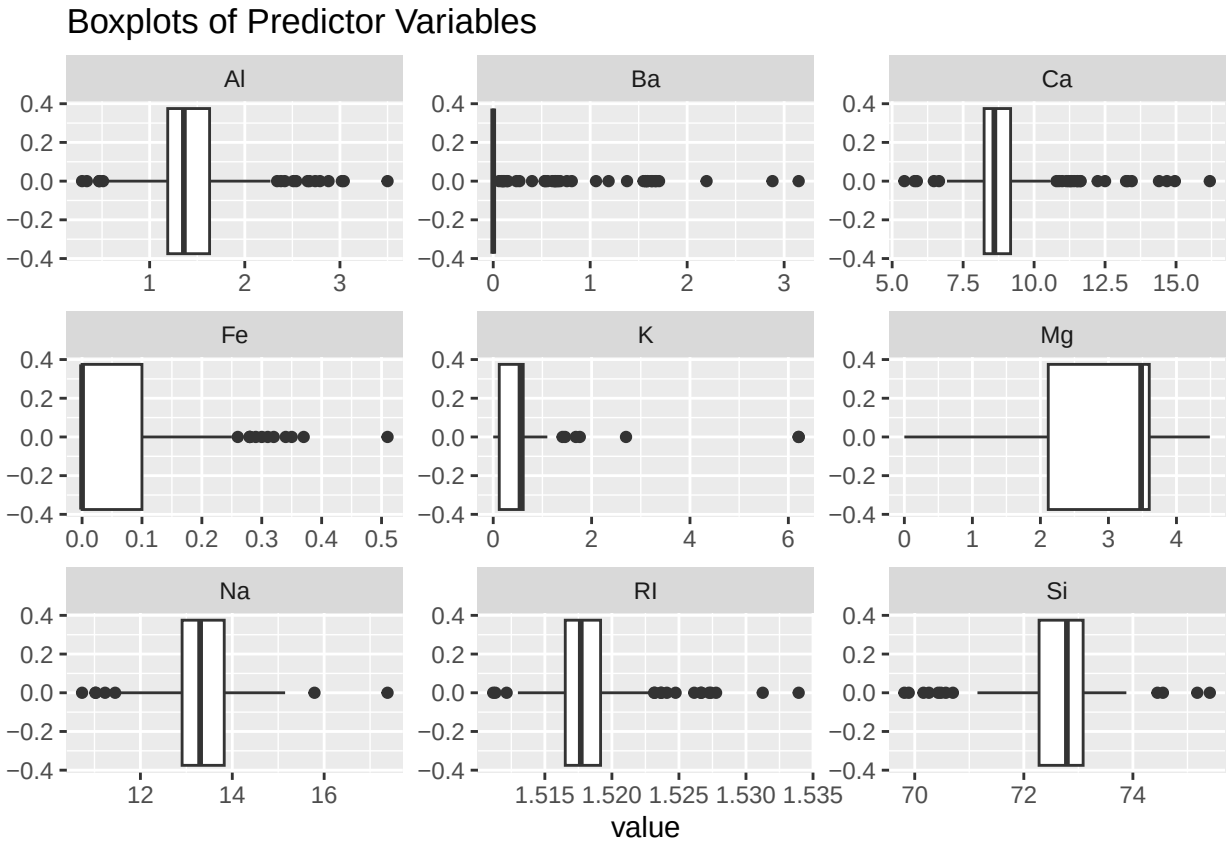



```
Glass |>
  select(-Type) |>
  cor() |>
  corrplot.mixed(tl.pos = 'd', tl.col = 'black',
                 title = 'Correlation between Predictor Values',
                 mar = c(0,0,2,0))
```



b. Do there appear to be any outliers in the data? Are any predictors skewed?

```
Glass |>
  select(-Type) |>
  gather() |>
  ggplot(aes(value)) +
  geom_boxplot() +
  facet_wrap(~key, scales = 'free') +
  labs(title = 'Boxplots of Predictor Variables')
```



Many of the chemical elements have outliers.

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

- Elements with significant right skews: Iron, Barium, and Potassium, may be good candidates for a log transform.
- Elements with near normal distributions: Aluminium, Calcium, Silicon, and Sodium, may be good candidates for z-score normalization.
- Magnesium has a left skew and may benefit from a root transformation.

Exercise 3.2

The soybean data can also be found at the UC Irvine Machine Learning Repository. Data were collected to predict disease in 683 soybeans. The 35 predictors are mostly categorical and include information on the environmen- tal conditions (e.g., temperature, precipitation) and plant conditions (e.g., left spots, mold growth). The outcome labels consist of 19 distinct classes.

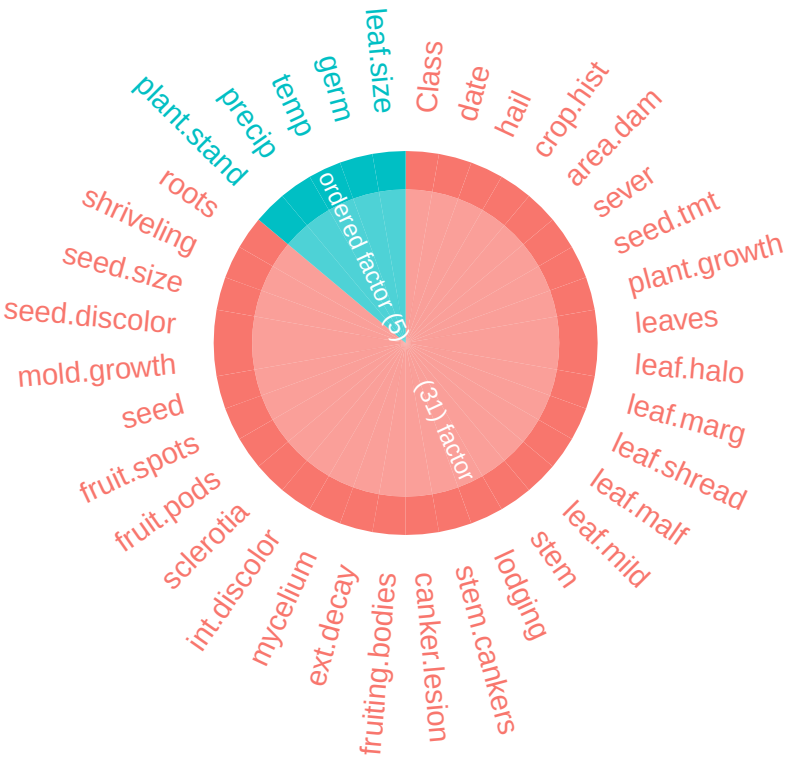
```
data(Soybean)
str(Soybean)
```

```
## 'data.frame':    683 obs. of  36 variables:
##  $ Class          : Factor w/ 19 levels "2-4-d-injury",..: 11 11 11 11 11 11 11 11 11 11 ...
##  $ date           : Factor w/ 7 levels "0","1","2","3",..: 7 5 4 4 7 6 6 5 7 5 ...
##  $ plant.stand    : Ord.factor w/ 2 levels "0"<"1": 1 1 1 1 1 1 1 1 1 1 ...
```

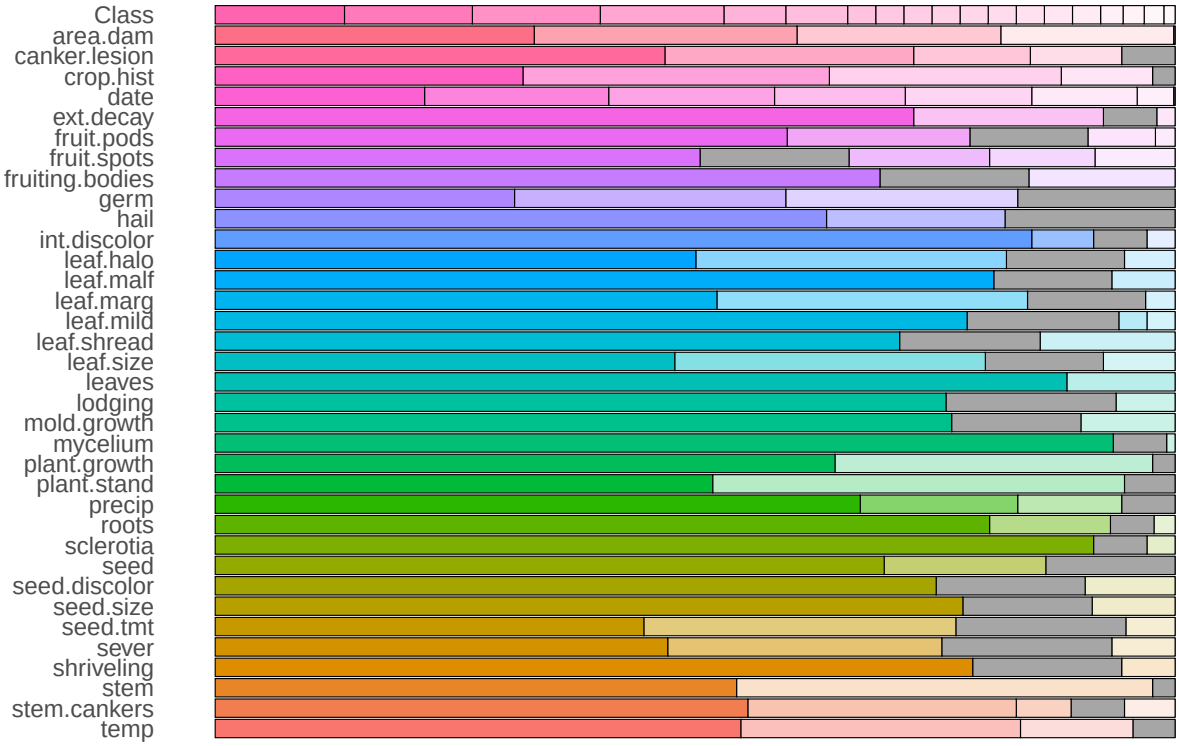
```
##  $ precip         : Ord.factor w/ 3 levels "0"<"1"<"2": 3 3 3 3 3 3 3 3 3 3 ...
##  $ temp           : Ord.factor w/ 3 levels "0"<"1"<"2": 2 2 2 2 2 2 2 2 2 2 ...
##  $ hail           : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 2 1 1 ...
##  $ crop.hist      : Factor w/ 4 levels "0","1","2","3": 2 3 2 2 3 4 3 2 4 3 ...
##  $ area.dam       : Factor w/ 4 levels "0","1","2","3": 2 1 1 1 1 1 1 1 1 1 ...
##  $ sever          : Factor w/ 3 levels "0","1","2": 2 3 3 3 2 2 2 2 2 3 ...
##  $ seed.tmt       : Factor w/ 3 levels "0","1","2": 1 2 2 1 1 1 2 1 2 1 ...
##  $ germ           : Ord.factor w/ 3 levels "0"<"1"<"2": 1 2 3 2 3 2 1 3 2 3 ...
##  $ plant.growth   : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
##  $ leaves         : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
##  $ leaf.halo      : Factor w/ 3 levels "0","1","2": 1 1 1 1 1 1 1 1 1 1 ...
##  $ leaf.marg      : Factor w/ 3 levels "0","1","2": 3 3 3 3 3 3 3 3 3 3 ...
##  $ leaf.size      : Ord.factor w/ 3 levels "0"<"1"<"2": 3 3 3 3 3 3 3 3 3 3 ...
##  $ leaf.shread    : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ leaf.malf      : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ leaf.mild      : Factor w/ 3 levels "0","1","2": 1 1 1 1 1 1 1 1 1 1 ...
##  $ stem           : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
##  $ lodging        : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 2 1 1 1 ...
##  $ stem.cankers   : Factor w/ 4 levels "0","1","2","3": 4 4 4 4 4 4 4 4 4 4 ...
##  $ canker.lesion  : Factor w/ 4 levels "0","1","2","3": 2 2 1 1 2 1 2 2 2 2 ...
##  $ fruiting.bodies: Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
##  $ ext.decay      : Factor w/ 3 levels "0","1","2": 2 2 2 2 2 2 2 2 2 2 ...
##  $ mycelium       : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ int.discolor   : Factor w/ 3 levels "0","1","2": 1 1 1 1 1 1 1 1 1 1 ...
##  $ sclerotia      : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ fruit.pods     : Factor w/ 4 levels "0","1","2","3": 1 1 1 1 1 1 1 1 1 1 ...
##  $ fruit.spots    : Factor w/ 4 levels "0","1","2","4": 4 4 4 4 4 4 4 4 4 4 ...
##  $ seed           : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ mold.growth    : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ seed.discolor  : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ seed.size      : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ shriveling     : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ roots          : Factor w/ 3 levels "0","1","2": 1 1 1 1 1 1 1 1 1 1 ...
```

a. Investigate the frequency distributions for the categorical predictors. Are any of the distributions degenerate in the ways discussed earlier in this chapter?

```
Soybean |>
  inspect_types() |>
  show_plot()
```




```
Soybean |>
  inspect_cat() |>
  show_plot()
```



Both `mycelium` and `sclerotia` are nearly degenerate with a single large value after excluding missing values.

b. Roughly 18% of the data are missing. Are there particular predictors that are more likely to be missing? Is the pattern of missing data related to the classes?

There seems to be a pattern of missing values with the leaf and seed predictors.

```
Soybean |>
  gg_miss_fct(Class) +
  labs(title = 'Missing Values by Class')
```

Three of the classes: `2-4-d-injury`, `cyst-nematode`, and `herbicide-injury`, are missing 100% of the data for most of the variables. There are two other classes: `diaporthe-pod-&-stem-blight` and `phytophthora-rot`, with missing values for some of the variables.

```
Soybean |>
  filter(Class %in% c('2-4-d-injury',
                      'cyst-nematode',
                      'herbicide-injury',
                      'diaporthe-pod-&-stem-blight',
                      'phytophthora-rot')) |>
  count(Class)
```

```
##                         Class  n
## 1                 2-4-d-injury 16
## 2                cyst-nematode 14
## 3 diaporthe-pod-&-stem-blight 15
## 4             herbicide-injury  8
## 5             phytophthora-rot 88
```

c. Develop a strategy for handling missing data, either by eliminating predictors or imputation.

I would eliminate the classes: `2-4-d-injury`, `cyst-nematode`, `diaporthe-pod-&-stem-blight`, and `herbicide-injury`. These classes represent less than 8% of the overall dataset and have significant missing values for most of the variables.

The `phytophthora-rot` class represents nearly 13% of the dataset, so it seems most appropriate to use imputation to supply the missing values based on knowledge of the class and the missing variables.