Glass Dataset

1 Introduction

The glass identification dataset was created by B. German, Central Research Establishment, Home Office Forensic Science Service, Aldermaston, Reading, Berkshire RG7 4PNA. The data frame with 214 observations has examples of the chemical analysis of 7 varieties of glass. The target is to predict the kind of glass type based on the chemical analysis.

Figure 1: Source: macrovector (link)

Dataset basic information:

- `Type` (target): 7 types of glass:
  - `1`: building_windows_float_processed
  - `2`: building_windows_non_float_processed
  - `3`: vehicle_windows_float_processed
  - `4`: vehicle_windows_non_float_processed
  - `5`: containers
  - `6`: tableware
  - `7`: headlamps
- `RI`: refractive index
- `Na`: Sodium (unit measurement: weight percent in corresponding oxide, as are the following attributes)
- `Mg`: Magnesium
- `Al`: Aluminum
- `Si`: Silicon
- `K`: Potasium
- `Ca`: Calcium
- `Ba`: Barium
- `Fe`: Iron

To load the dataset, we use `mlbench`:

```
# load the dataset from mlbench
data(Glass)
glass <- Glass %>% as_tibble()
skimmed_glass <- skimr::skim(glass)
print(glass)
```

```
## # A tibble: 214 x 10
##       RI    Na    Mg    Al    Si     K    Ca    Ba    Fe Type
##    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <fct>
##  1  1.52  13.6  4.49  1.1   71.8  0.06  8.75     0  0    1
##  2  1.52  13.9  3.6   1.36  72.7  0.48  7.83     0  0    1
##  3  1.52  13.5  3.55  1.54  73.0  0.39  7.78     0  0    1
##  4  1.52  13.2  3.69  1.29  72.6  0.57  8.22     0  0    1
##  5  1.52  13.3  3.62  1.24  73.1  0.55  8.07     0  0    1
##  6  1.52  12.8  3.61  1.62  73.0  0.64  8.07     0  0.26 1
##  7  1.52  13.3  3.6   1.14  73.1  0.58  8.17     0  0    1
##  8  1.52  13.2  3.61  1.05  73.2  0.57  8.24     0  0    1
##  9  1.52  14.0  3.58  1.37  72.1  0.56  8.3      0  0    1
## 10  1.52  13    3.6   1.36  73.0  0.57  8.4      0  0.11 1
## # ... with 204 more rows
```

2 Exploratory Data Analysis (EDA)

In this part, we will walk through a few characteristics of glass dataset using library `skimr` and `DataExplorer`.

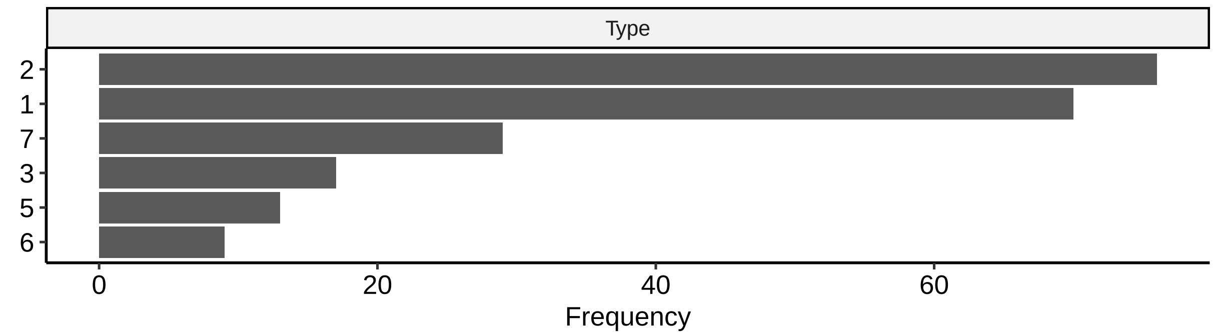
2.1 Factor variables

General statistics about factor variables from glass dataset:

```
skimr::partition(skimmed_glass)$factor %>%
        knitr::kable(format = 'latex', booktabs = TRUE) %>%
        kableExtra::kable_styling(latex_options = 'HOLD_position')
```

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
Type	0	1	FALSE	6	2: 76, 1: 70, 7: 29, 3: 17

```
DataExplorer::plot_bar(glass, ggtheme = ggpubr::theme_pubr(base_size = 10))
```



```
dplyr::count(glass, Type, sort = TRUE)
```

```
## # A tibble: 6 x 2
##   Type      n
##   <fct> <int>
## 1 2        76
## 2 1        70
## 3 7        29
## 4 3        17
## 5 5        13
## 6 6         9
```

The dataset has one factor variable, that is the target `Type`. This variable does not have missing values and only contains 6 unique values, i.e. 6 types of glass (as opposed to 7 as expected from the dataset description). From the bar plot, it can be seen that `Type`'s discrete distribution is highly imbalanced, with the majority of data points belongs to class `1` or `2`. Types `3`, `5`, `6` have the fewest observations with 17, 13, and 9 observations, respectively.

2.2 Numerical variables

General statistics about numerical variables from glass dataset:

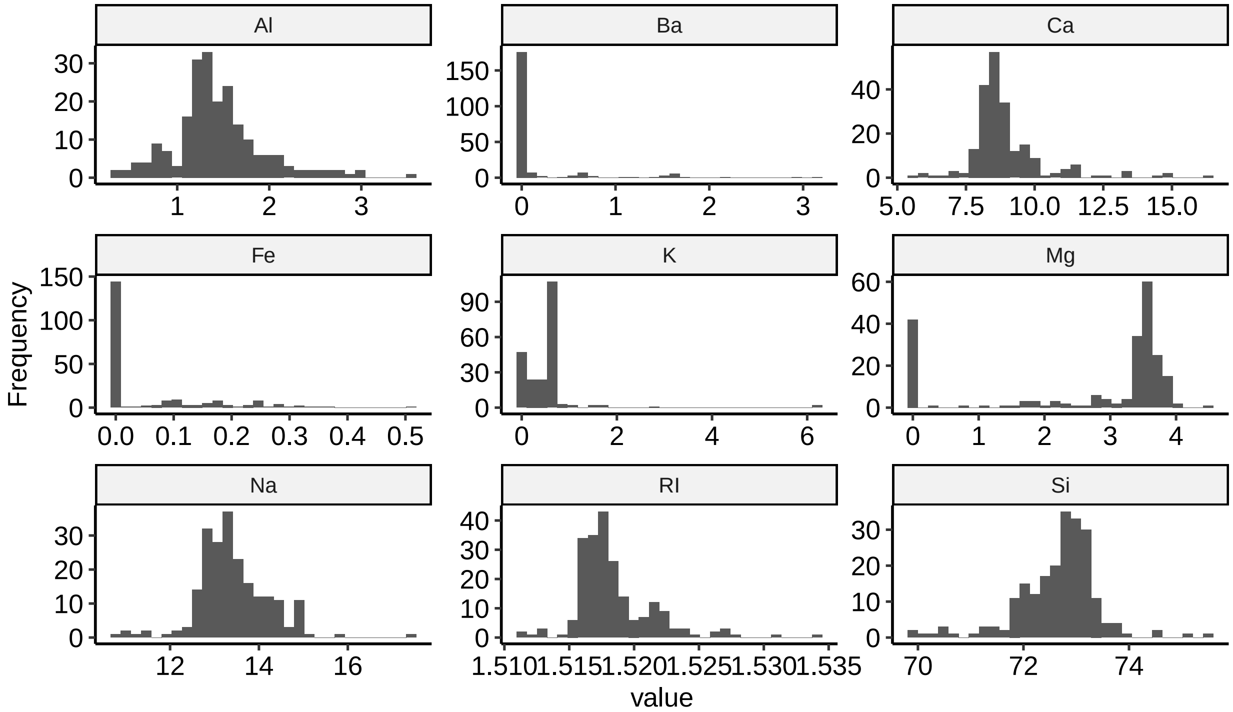
```
skimr::partition(skimmed_glass)$numeric %>%
        knitr::kable(format = 'latex', booktabs = TRUE, digits = 2) %>%
        kableExtra::kable_styling(latex_options = 'HOLD_position')
```

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
RI	0	1	1.52	0.00	1.51	1.52	1.52	1.52	1.53	
Na	0	1	13.41	0.82	10.73	12.91	13.30	13.83	17.38	
Mg	0	1	2.68	1.44	0.00	2.11	3.48	3.60	4.49	
Al	0	1	1.44	0.50	0.29	1.19	1.36	1.63	3.50	
Si	0	1	72.65	0.77	69.81	72.28	72.79	73.09	75.41	
K	0	1	0.50	0.65	0.00	0.12	0.56	0.61	6.21	
Ca	0	1	8.96	1.42	5.43	8.24	8.60	9.17	16.19	
Ba	0	1	0.18	0.50	0.00	0.00	0.00	0.00	3.15	
Fe	0	1	0.06	0.10	0.00	0.00	0.00	0.10	0.51	

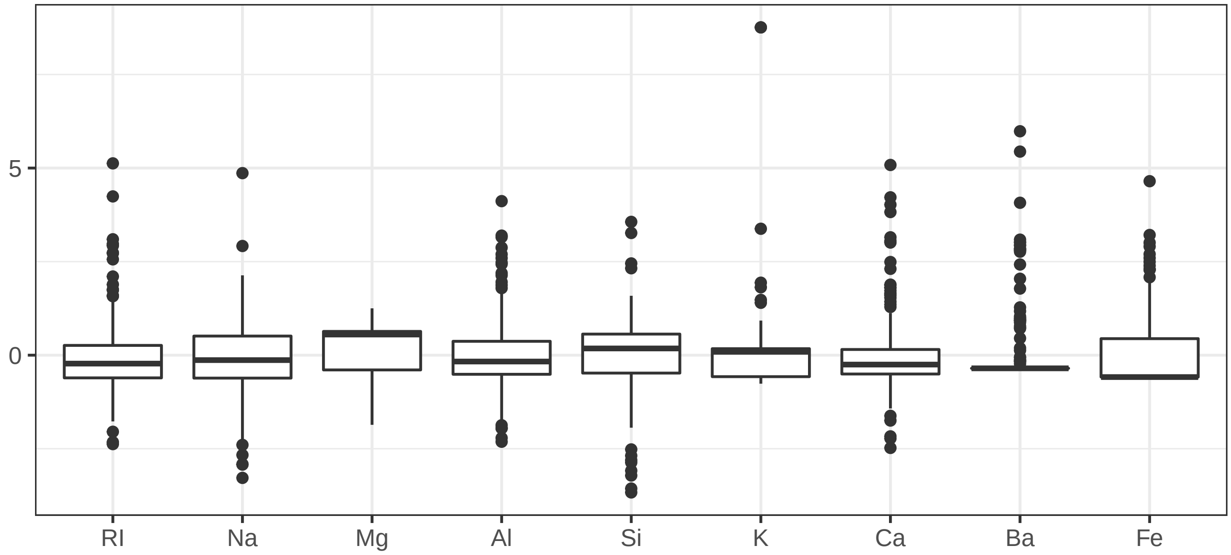
As can be seen from the statistics, similar to the factor variable, numerical variables in this dataset don't have missing values. Except for `RI`, the values of other numerical variables are on the same scale, i.e. weight percent in corresponding oxide, as mentioned in the introduction. A quick look at the percentiles of these variables shows that they have very different distribution.

For more detailed view into the distribution of these variables, we can plot their histograms and (scaled) boxplots:

```
DataExplorer::plot_histogram(
        glass,
        ncol = 3,
        ggtheme = ggpubr::theme_pubr(base_size = 10)
)
```



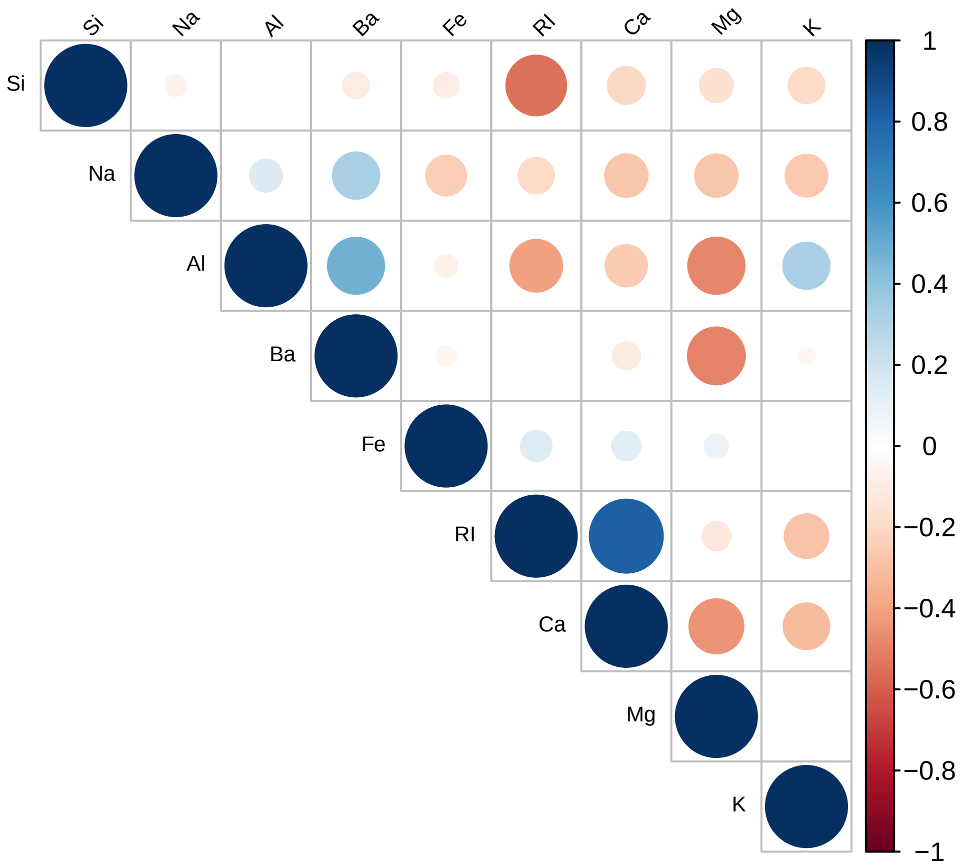
```
glass_numerical <- glass %>% select(where(is.numeric))
glass_numerical_scale <- glass_numerical %>% mutate_all(~(scale(.) %>% as.vector))
glass_numerical_scale_melt <- melt(glass_numerical_scale)
ggplot(data = glass_numerical_scale_melt, aes(x=variable, y=value)) +
        geom_boxplot() +
        theme(axis.title.x=element_blank(),
              axis.title.y=element_blank())
```



According to the plots above, `Ba`, `Fe`, `K` and `Mg` have highly skewed distributions. Furthermore, apart from `Mg`, other numerical features have a lot of outliers.

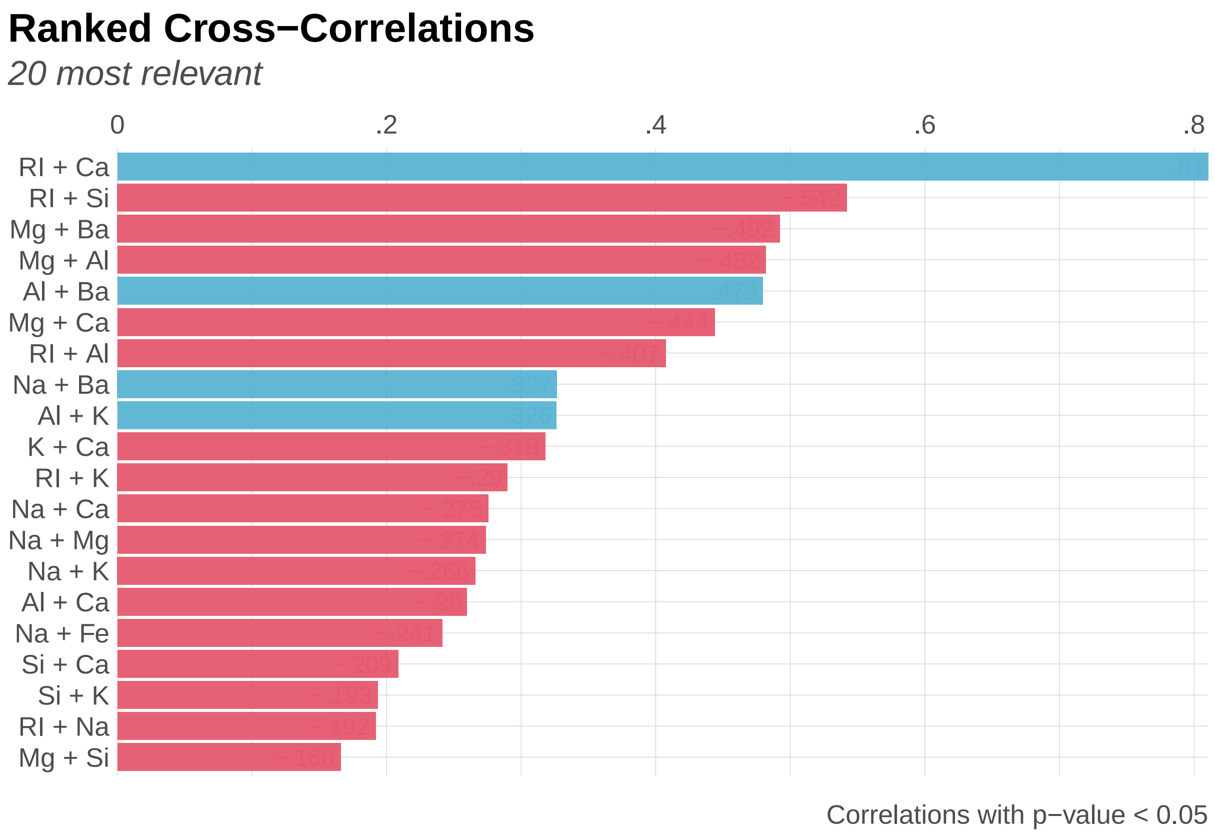
To understand more the linear relationship between the pairs of numerical variables, we create a correlation matrix:

```r
glass_numerical %>%
        cor() %>%
        corrplot(
           type = "upper",
           order = "hclust",
           tl.col = "black",
           tl.srt = 45,
           tl.cex = 0.7
        )
```



We can also create a ranking of top 20 pairs of variables by the magnitude of correlation to interpret the result with `corr_cross` function from library `lares`:

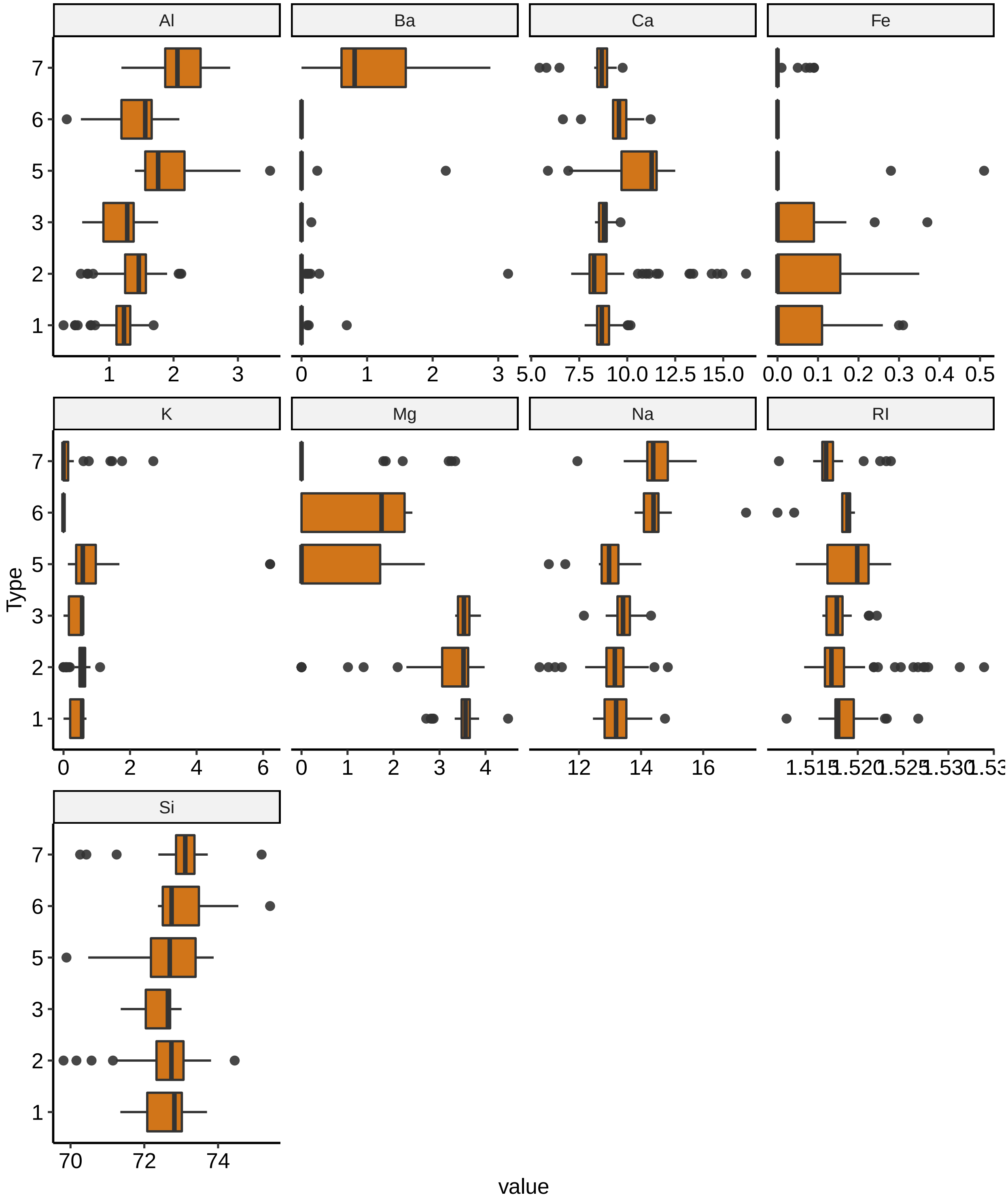
```
corr_cross(glass_numerical,
  max_pvalue = 0.05, # display only significant correlations (at 5% level)
  top = 20 # display top 20 couples of variables (by correlation coefficient)
)
```



From the correlation plot and the cross-correlation ranking, the linear relationship between `Ri` and `Ca` stands out with correlation greater than 0.8.

It might also be worth taking a look at the relationship between each numerical variable and the target by plotting the boxplots broken down by class type:

```
DataExplorer::plot_boxplot(
        glass,
        by = "Type",
        ggtheme = ggpubr::theme_pubr(base_size = 10),
        nrow = 5
)
```



From the boxplots, there are a few interesting points. First, feature `Ba` appears to be able to easily distinguish glass type `7` from the others. Second, higher values of `Na` (> 14) might be a useful indicator for the class `6` or `7`. Furthermore, `Mg` values might be used to differentiate class `5`, `6` and `7` from `1`, `2` and `3`.