eda

Daniel Hill

Table of contents

```
library(tidyverse)
library(grid)
library(ggplot2)
```

```
data <- read_csv("Data(2).csv")
summary(data)
```

```
       X1               X2               X3               X4        
 Min.   : 6.840   Min.   : 6.538   Min.   : 6.424   Min.   : 3.875  
 1st Qu.: 9.356   1st Qu.: 9.445   1st Qu.: 8.243   1st Qu.: 8.088  
 Median : 9.872   Median :10.138   Median : 8.847   Median : 8.926  
 Mean   : 9.876   Mean   :10.151   Mean   : 8.861   Mean   : 8.939  
 3rd Qu.:10.399   3rd Qu.:10.855   3rd Qu.: 9.445   3rd Qu.: 9.805  
 Max.   :12.355   Max.   :14.021   Max.   :12.216   Max.   :13.351  
 NA's   :2                         NA's   :1        NA's   :2       
       X5               X6               X7                  X8          
 Min.   :10.53   Min.   : 4.815   Min.   :0.0001708   Min.   :0.0003989  
 1st Qu.:13.24   1st Qu.: 7.447   1st Qu.:0.1852258   1st Qu.:0.1015754  
 Median :13.86   Median : 8.134   Median :0.3749279   Median :0.1701996  
 Mean   :13.85   Mean   : 8.151   Mean   :0.4259452   Mean   :0.2337453  
 3rd Qu.:14.49   3rd Qu.: 8.856   3rd Qu.:0.6775852   3rd Qu.:0.3004369  
 Max.   :16.56   Max.   :11.871   Max.   :1.3009246   Max.   :1.2303619  
                                  NA's   :1           NA's   :3          
       X9                X10                X11              X12        
 Min.   :0.006076   Min.   :0.0003681   Min.   : 6.031   Min.   : 8.046  
 1st Qu.:0.531936   1st Qu.:0.2808408   1st Qu.: 8.412   1st Qu.:11.290  
 Median :0.751335   Median :0.3717752   Median : 9.077   Median :11.913  
 Mean   :0.717052   Mean   :0.3784706   Mean   : 9.175   Mean   :11.930  
```

```
 3rd Qu.:0.888398   3rd Qu.:0.4688587   3rd Qu.: 9.860   3rd Qu.:12.594
 Max.   :1.678634   Max.   :1.1991323   Max.   :13.027   Max.   :15.478
                                        NA's   :1
      X13              X14              X15              X16
 Min.   : 4.919   Min.   : 3.574   Min.   : 7.572   Min.   : 3.801
 1st Qu.: 7.719   1st Qu.: 7.022   1st Qu.:10.054   1st Qu.: 7.132
 Median : 8.244   Median : 7.884   Median :10.701   Median : 7.830
 Mean   : 8.228   Mean   : 7.846   Mean   :10.701   Mean   : 7.814
 3rd Qu.: 8.746   3rd Qu.: 8.689   3rd Qu.:11.344   3rd Qu.: 8.521
 Max.   :11.226   Max.   :12.413   Max.   :14.037   Max.   :11.668
 NA's   :1        NA's   :2        NA's   :1        NA's   :1
      X17                 X18                 X19                 X20
 Min.   :0.0007101   Min.   :0.004206   Min.   :0.0002235   Min.   :0.0118
 1st Qu.:0.3479476   1st Qu.:0.543775   1st Qu.:0.3633119   1st Qu.:0.4337
 Median :0.5016488   Median :0.686125   Median :0.5448244   Median :0.5875
 Mean   :0.5035825   Mean   :0.681698   Mean   :0.5438105   Mean   :0.5894
 3rd Qu.:0.6527304   3rd Qu.:0.819033   3rd Qu.:0.7096470   3rd Qu.:0.7457
 Max.   :1.3151299   Max.   :1.390482   Max.   :1.5176973   Max.   :1.3530
 NA's   :2           NA's   :3
    label
 Length:3000
 Class :character
 Mode  :character
```

```
head(data)
```

```
# A tibble: 6 x 21
     X1    X2    X3    X4    X5    X6     X7     X8    X9    X10   X11   X12
  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>  <dbl> <dbl>  <dbl> <dbl> <dbl>
1  9.63 10.9   7.69  6.95  13.2  9.52 0.670  0.0382 0.806 0.395   9.31  11.3
2 10.6   9.57  7.70  8.63  13.0  8.35 0.489  0.0843 0.678 0.512   6.45  12.5
3 10.7  11.6   8.99  9.33  14.5  8.27 0.0900 0.214  0.524 0.692   7.98  10.4
4  9.47  9.87  9.48 10.6   14.9  7.66 0.0538 0.173  0.566 0.557   8.63  11.3
5 10.4   8.75  9.34  8.72  13.2  9.09 0.597  0.0710 0.987 0.0857 10.1   11.6
6  8.87 11.0   8.69  7.77  15.8  8.28 0.383  0.114  0.748 0.372   8.18  11.5
# i 9 more variables: X13 <dbl>, X14 <dbl>, X15 <dbl>, X16 <dbl>, X17 <dbl>,
#   X18 <dbl>, X19 <dbl>, X20 <dbl>, label <chr>
```

```
library(psych)
describe(data)
```

```
       vars    n  mean   sd median trimmed  mad   min   max range  skew
X1        1 2998  9.88 0.76   9.87    9.88 0.77  6.84 12.36  5.52 -0.08
X2        2 3000 10.15 1.04  10.14   10.15 1.05  6.54 14.02  7.48  0.01
X3        3 2999  8.86 0.87   8.85    8.85 0.89  6.42 12.22  5.79  0.10
X4        4 2998  8.94 1.28   8.93    8.95 1.27  3.87 13.35  9.48 -0.04
X5        5 3000 13.85 0.94  13.86   13.86 0.93 10.53 16.56  6.03 -0.08
X6        6 3000  8.15 1.03   8.13    8.15 1.04  4.81 11.87  7.06  0.05
X7        7 2999  0.43 0.28   0.37    0.41 0.33  0.00  1.30  1.30  0.35
X8        8 2997  0.23 0.20   0.17    0.20 0.12  0.00  1.23  1.23  1.54
X9        9 3000  0.72 0.25   0.75    0.72 0.24  0.01  1.68  1.67 -0.01
X10      10 3000  0.38 0.15   0.37    0.37 0.14  0.00  1.20  1.20  0.37
X11      11 2999  9.18 1.09   9.08    9.13 1.06  6.03 13.03  7.00  0.41
X12      12 3000 11.93 0.98  11.91   11.93 0.97  8.05 15.48  7.43  0.00
X13      13 2999  8.23 0.81   8.24    8.24 0.76  4.92 11.23  6.31 -0.09
X14      14 2998  7.85 1.24   7.88    7.85 1.23  3.57 12.41  8.84 -0.06
X15      15 2999 10.70 0.96  10.70   10.70 0.96  7.57 14.04  6.46  0.02
X16      16 2999  7.81 1.05   7.83    7.81 1.03  3.80 11.67  7.87  0.00
X17      17 2998  0.50 0.22   0.50    0.50 0.23  0.00  1.32  1.31  0.15
X18      18 2997  0.68 0.20   0.69    0.68 0.20  0.00  1.39  1.39 -0.03
X19      19 3000  0.54 0.25   0.54    0.54 0.26  0.00  1.52  1.52  0.18
X20      20 3000  0.59 0.23   0.59    0.59 0.23  0.01  1.35  1.34  0.07
label*   21 3000  2.49 1.11   2.00    2.49 1.48  1.00  4.00  3.00  0.01
       kurtosis   se
X1         0.05 0.01
X2        -0.04 0.02
X3        -0.18 0.02
X4         0.08 0.02
X5        -0.06 0.02
X6        -0.02 0.02
X7        -1.07 0.01
X8         2.19 0.00
X9         0.15 0.00
X10        0.61 0.00
X11        0.03 0.02
X12        0.09 0.02
X13        0.47 0.01
X14        0.02 0.02
X15       -0.06 0.02
X16        0.13 0.02
```

```
X17       -0.22 0.00
X18        0.02 0.00
X19       -0.24 0.00
X20       -0.20 0.00
label*    -1.35 0.02
```

```
library(skimr)
skim(data)
```

Table 1: Data summary

Name	data
Number of rows	3000
Number of columns	21
_____	
Column type frequency:	
character	1
numeric	20
_____	
Group variables	None

Variable type: character

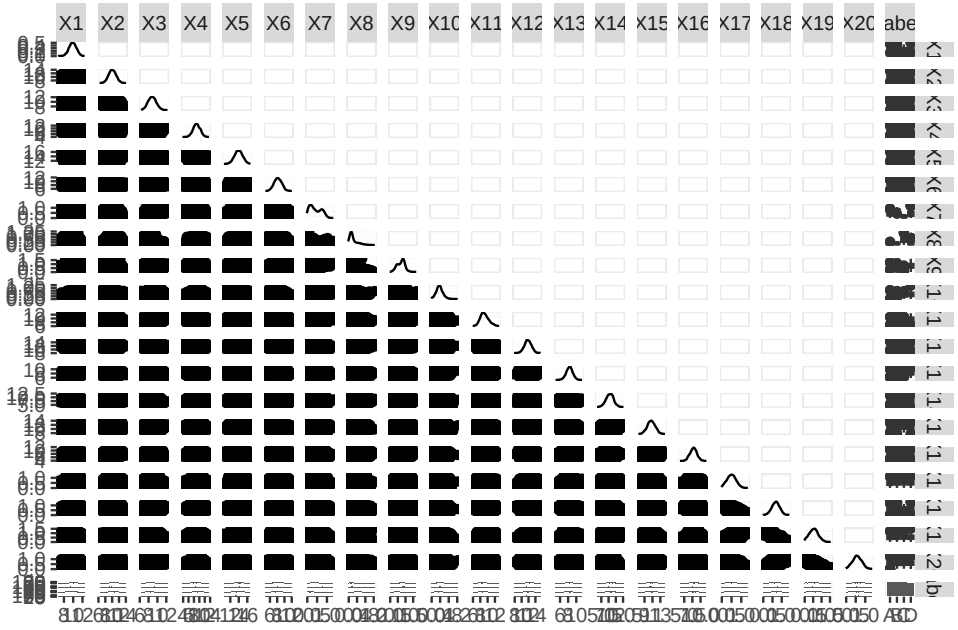
skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
label	0	1	1	1	0	4	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
X1	2	1	9.88	0.76	6.84	9.36	9.87	10.40	12.36	
X2	0	1	10.15	1.04	6.54	9.44	10.14	10.86	14.02	
X3	1	1	8.86	0.87	6.42	8.24	8.85	9.44	12.22	
X4	2	1	8.94	1.28	3.87	8.09	8.93	9.80	13.35	
X5	0	1	13.85	0.94	10.53	13.24	13.86	14.49	16.56	
X6	0	1	8.15	1.03	4.81	7.45	8.13	8.86	11.87	
X7	1	1	0.43	0.28	0.00	0.19	0.37	0.68	1.30	
X8	3	1	0.23	0.20	0.00	0.10	0.17	0.30	1.23	
X9	0	1	0.72	0.25	0.01	0.53	0.75	0.89	1.68	
X10	0	1	0.38	0.15	0.00	0.28	0.37	0.47	1.20	

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
X11	1	1	9.18	1.09	6.03	8.41	9.08	9.86	13.03	
X12	0	1	11.93	0.98	8.05	11.29	11.91	12.59	15.48	
X13	1	1	8.23	0.81	4.92	7.72	8.24	8.75	11.23	
X14	2	1	7.85	1.24	3.57	7.02	7.88	8.69	12.41	
X15	1	1	10.70	0.96	7.57	10.05	10.70	11.34	14.04	
X16	1	1	7.81	1.05	3.80	7.13	7.83	8.52	11.67	
X17	2	1	0.50	0.22	0.00	0.35	0.50	0.65	1.32	
X18	3	1	0.68	0.20	0.00	0.54	0.69	0.82	1.39	
X19	0	1	0.54	0.25	0.00	0.36	0.54	0.71	1.52	
X20	0	1	0.59	0.23	0.01	0.43	0.59	0.75	1.35	

```
library(GGally)
ggpairs(data)
```




```
one_hot_data <- model.matrix(~ label - 1, data = data)
one_hot_data <- data %>%
  select(-label) %>%
  bind_cols(one_hot_data)
head(one_hot_data)
```

```
# A tibble: 6 x 24
```

```
     X1    X2    X3    X4    X5    X6     X7     X8    X9    X10   X11   X12
  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>  <dbl> <dbl>  <dbl> <dbl> <dbl>
1  9.63 10.9   7.69  6.95  13.2  9.52 0.670  0.0382 0.806 0.395   9.31  11.3
2 10.6   9.57  7.70  8.63  13.0  8.35 0.489  0.0843 0.678 0.512   6.45  12.5
3 10.7  11.6   8.99  9.33  14.5  8.27 0.0900 0.214  0.524 0.692   7.98  10.4
4  9.47  9.87  9.48 10.6   14.9  7.66 0.0538 0.173  0.566 0.557   8.63  11.3
5 10.4   8.75  9.34  8.72  13.2  9.09 0.597  0.0710 0.987 0.0857 10.1   11.6
6  8.87 11.0   8.69  7.77  15.8  8.28 0.383  0.114  0.748 0.372   8.18  11.5
# i 12 more variables: X13 <dbl>, X14 <dbl>, X15 <dbl>, X16 <dbl>, X17 <dbl>,
#   X18 <dbl>, X19 <dbl>, X20 <dbl>, labelA <dbl>, labelB <dbl>, labelC <dbl>,
#   labelD <dbl>
```

```r
cor_matrix <- cor(one_hot_data, use = "complete.obs", method = "pearson")
head(cor_matrix)
```

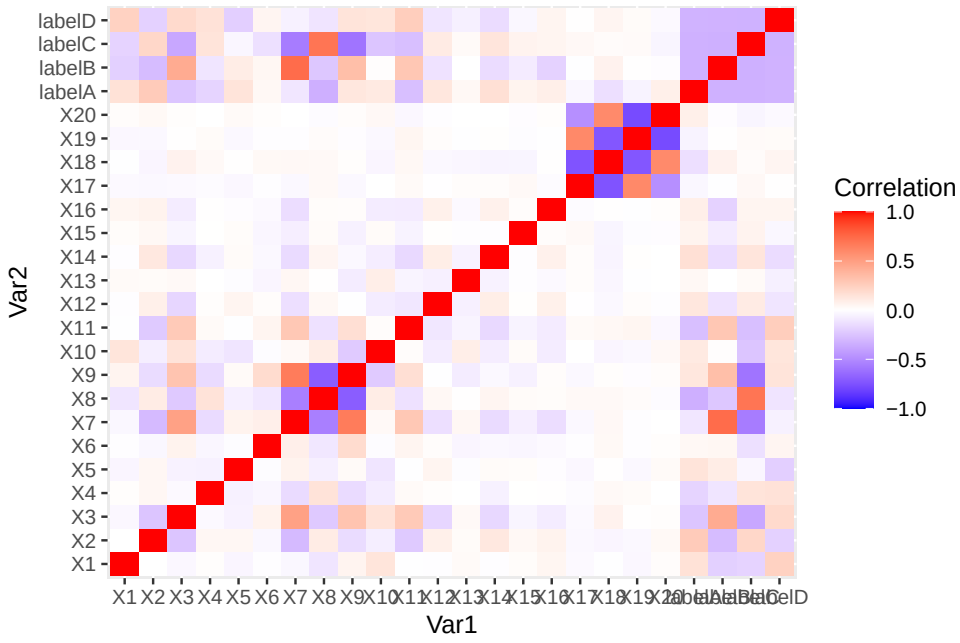
```
              X1            X2          X3          X4          X5           X6
X1  1.0000000000  0.0004839928 -0.03309062  0.01284993 -0.04145908 -0.007821135
X2  0.0004839928  1.0000000000 -0.24822101  0.04218052  0.04156405 -0.034915433
X3 -0.0330906237 -0.2482210119  1.00000000 -0.02572921 -0.05388457  0.064554867
X4  0.0128499291  0.0421805209 -0.02572921  1.00000000 -0.05851801 -0.038662183
X5 -0.0414590756  0.0415640508 -0.05388457 -0.05851801  1.00000000 -0.014620363
X6 -0.0078211350 -0.0349154326  0.06455487 -0.03866218 -0.01462036  1.000000000
           X7          X8          X9         X10          X11          X12
X1 -0.03547230 -0.11199972  0.05635645  0.13723767 -0.006326587 -0.008889542
X2 -0.29474627  0.09925496 -0.15082033 -0.07341833 -0.227344706  0.079796240
X3  0.48299876 -0.23144478  0.30838416  0.14736896  0.275397278 -0.175158129
X4 -0.15250755  0.14536419 -0.15407826 -0.07875387  0.025400841  0.010567198
X5  0.06432839 -0.06637011  0.02623828 -0.10911550 -0.004144906  0.052633626
X6  0.08151158 -0.10345713  0.18419313 -0.01330354  0.052513788  0.013545881
            X13         X14          X15          X16          X17
X1  0.0283227842 -0.01005859  0.017766538  0.047930273 -0.023324503
X2  0.0215100233  0.12064871  0.036820754  0.063760924 -0.029315071
X3  0.0308139912 -0.16786182 -0.042954966 -0.079959173 -0.024278934
X4 -0.0007546634 -0.06033578  0.002137233  0.005629295 -0.016673270
X5 -0.0129216615  0.02324719  0.017587211 -0.010643522 -0.031892588
X6 -0.0406354335 -0.02915716 -0.035919754 -0.026593708 -0.008477601
           X18          X19          X20      labelA      labelB      labelC
X1 -0.006281137 -0.031893972 0.0141295402  0.15264317 -0.20018459 -0.18551655
X2 -0.041581996 -0.031193522 0.0332748075  0.26994307 -0.28790093  0.21096132
X3  0.066312949  0.005606813 0.0102475749 -0.24778320  0.43307358 -0.37608262
X4  0.031392673  0.021718057 0.0004740494 -0.18248084 -0.11087487  0.14210217
X5  0.002812490 -0.027795476 0.0248014826  0.14400892  0.09686318 -0.03983622
```

```
X6  0.032606247 -0.009370526 0.0101575217  0.03794563  0.04452471 -0.13148376
        labelD
X1  0.23682272
X2 -0.19467155
X3  0.19296371
X4  0.15186096
X5 -0.20263772
X6  0.05007463
```

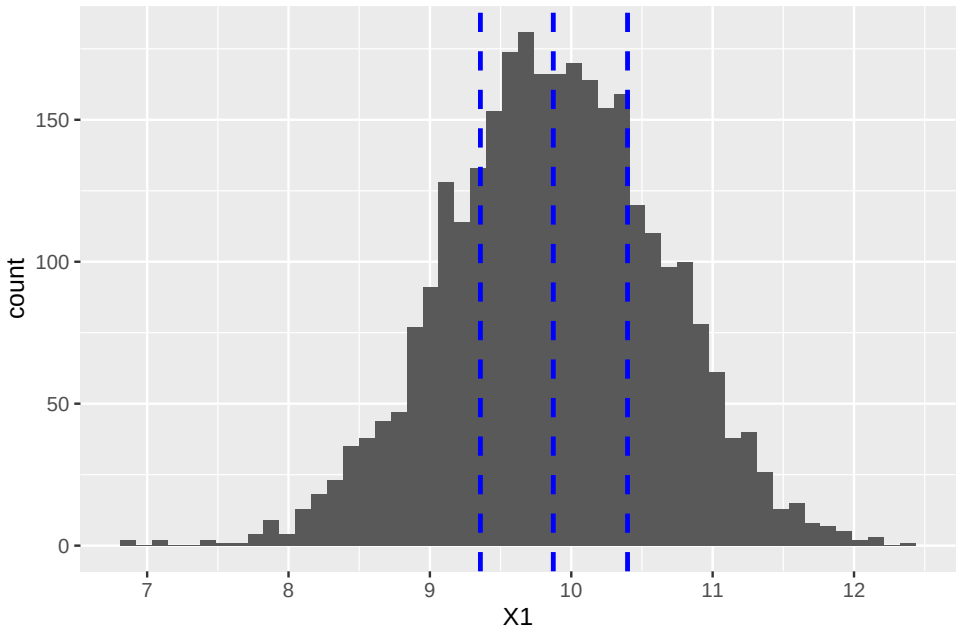
```r
library(reshape2)
cor_data <- melt(cor_matrix)
head(cor_data)
```

```
  Var1 Var2         value
1   X1   X1  1.0000000000
2   X2   X1  0.0004839928
3   X3   X1 -0.0330906237
4   X4   X1  0.0128499291
5   X5   X1 -0.0414590756
6   X6   X1 -0.0078211350
```

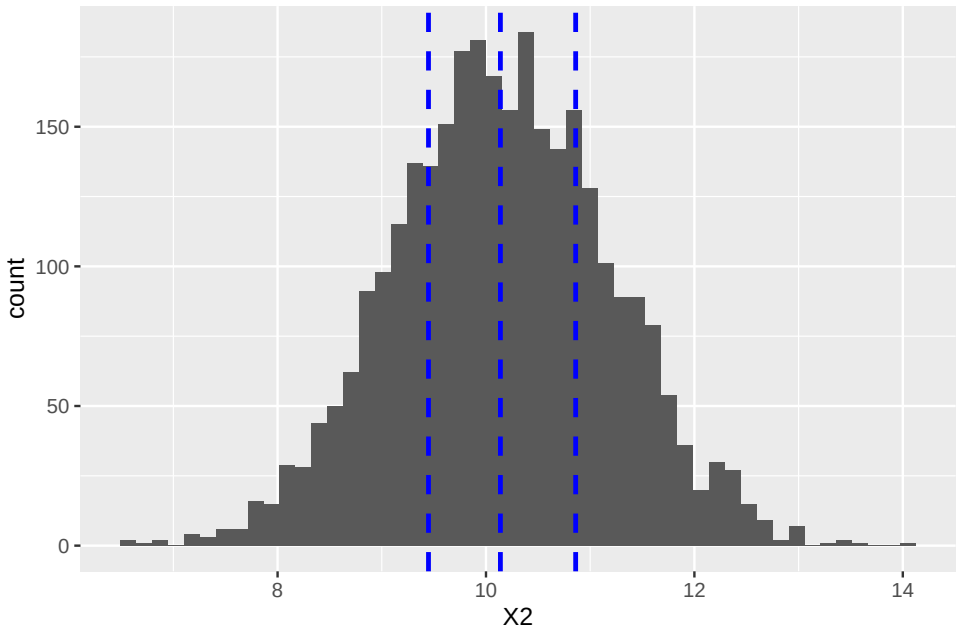
```r
ggplot(cor_data, aes(x = Var1, y = Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient2(low = "blue", high = "red", mid = "white",
                       midpoint = 0, limit = c(-1, 1),
                       name = "Correlation")
```



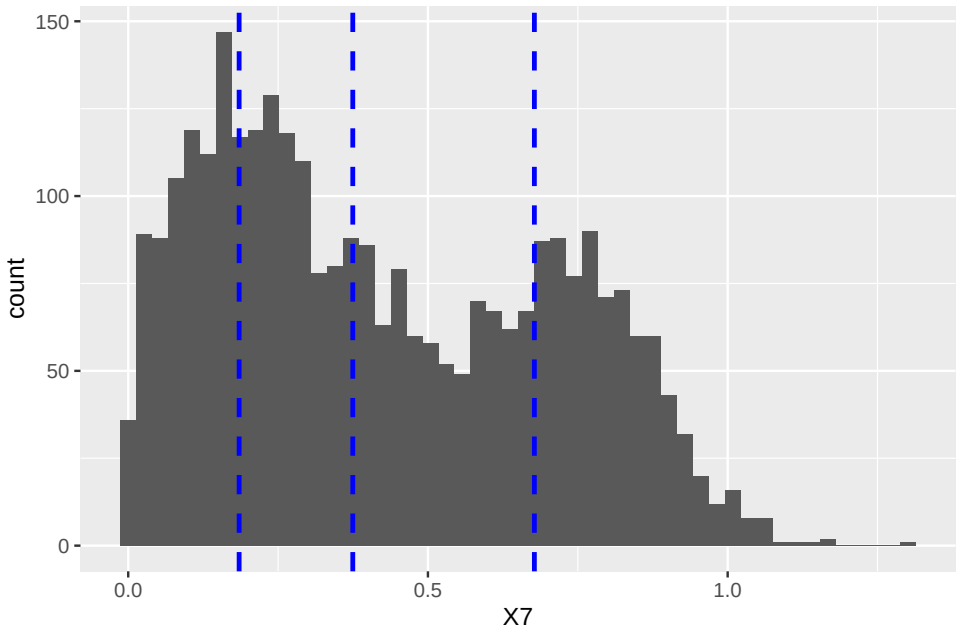

```
quartile_df <- data %>%
  summarize(first=quantile(drop_na(data)$X1, p=1/4),
            second=quantile(drop_na(data)$X1, p=1/2),
            third=quantile(drop_na(data)$X1, p=3/4)) %>%
  tidyr::gather(quartile, value)

data %>%
  ggplot(aes(x=X1)) +
    geom_histogram(bins=50) +
    # (aes(xintercept=median(X1)), size=1.5, color="red") +
    geom_vline(aes(xintercept=value), data=quartile_df,
               size=1,color="blue", linetype=2)
```



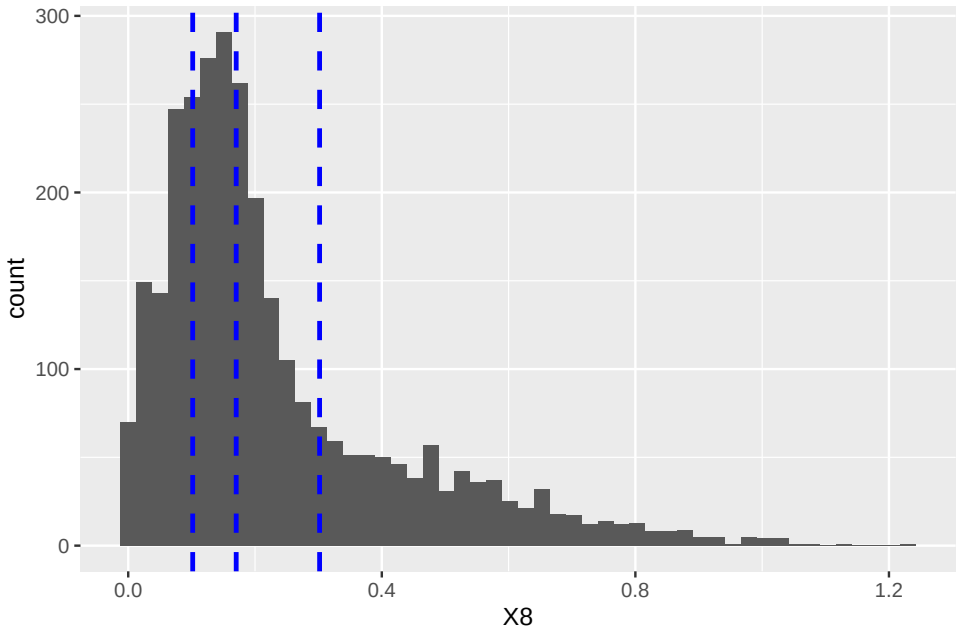

```
quartile_df <- data %>%
  summarize(first=quantile(drop_na(data)$X2, p=1/4),
            second=quantile(drop_na(data)$X2, p=1/2),
            third=quantile(drop_na(data)$X2, p=3/4)) %>%
  tidyr::gather(quartile, value)

data %>%
  ggplot(aes(x=X2)) +
    geom_histogram(bins=50) +
    # (aes(xintercept=median(X2)), size=1.5, color="red") +
    geom_vline(aes(xintercept=value), data=quartile_df,
               size=1,color="blue", linetype=2)
```



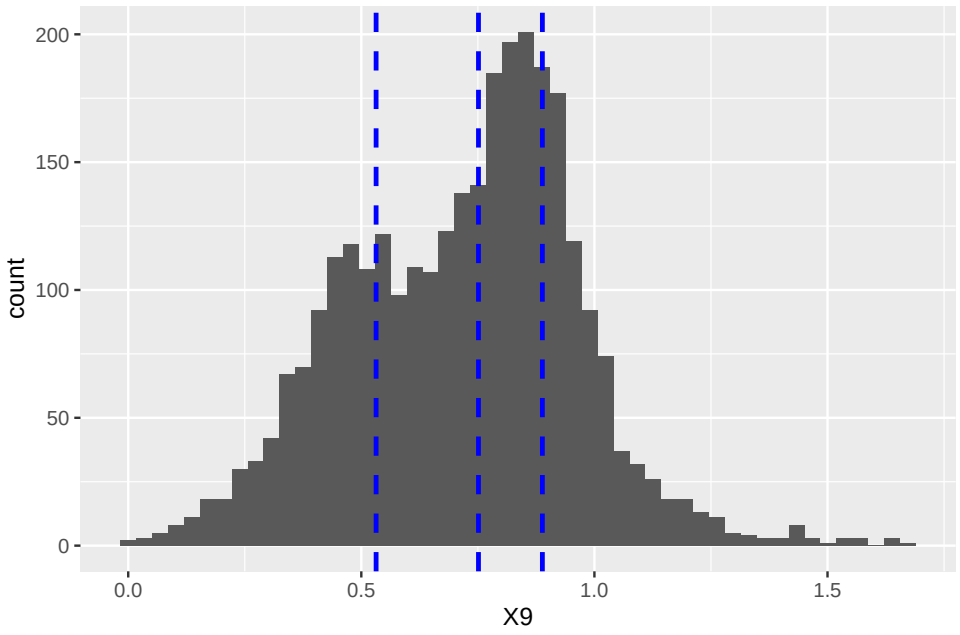

```
quartile_df <- data %>%
  summarize(first=quantile(drop_na(data)$X7, p=1/4),
            second=quantile(drop_na(data)$X7, p=1/2),
            third=quantile(drop_na(data)$X7, p=3/4)) %>%
  tidyr::gather(quartile, value)

data %>%
  ggplot(aes(x=X7)) +
    geom_histogram(bins=50) +
    # (aes(xintercept=median(X7)), size=1.5, color="red") +
    geom_vline(aes(xintercept=value), data=quartile_df,
               size=1,color="blue", linetype=2)
```



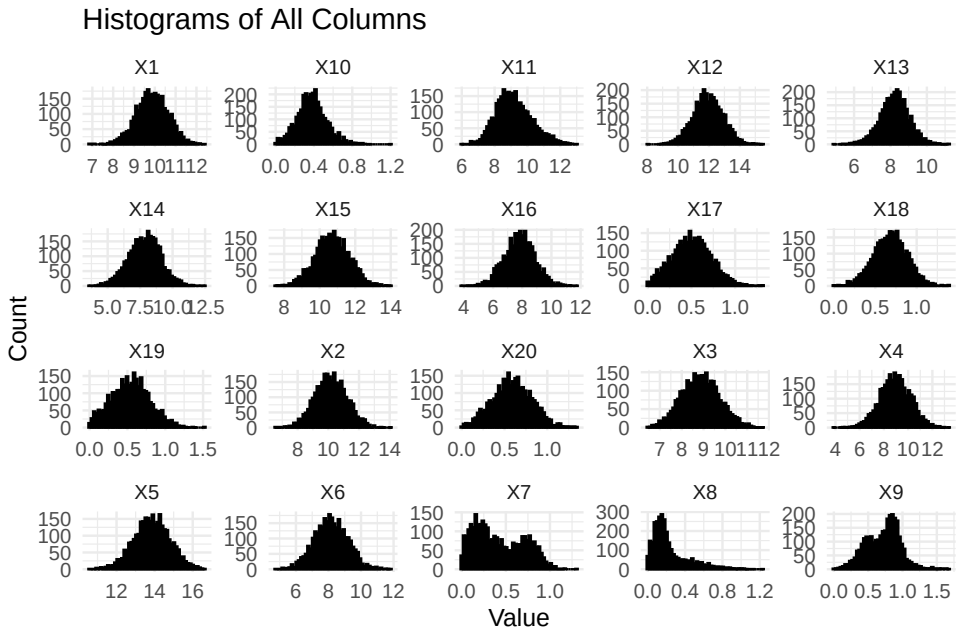

```
quartile_df <- data %>%
  summarize(first=quantile(drop_na(data)$X8, p=1/4),
            second=quantile(drop_na(data)$X8, p=1/2),
            third=quantile(drop_na(data)$X8, p=3/4)) %>%
  tidyr::gather(quartile, value)

data %>%
  ggplot(aes(x=X8)) +
    geom_histogram(bins=50) +
    # (aes(xintercept=median(X8)), size=1.5, color="red") +
    geom_vline(aes(xintercept=value), data=quartile_df,
               size=1,color="blue", linetype=2)
```



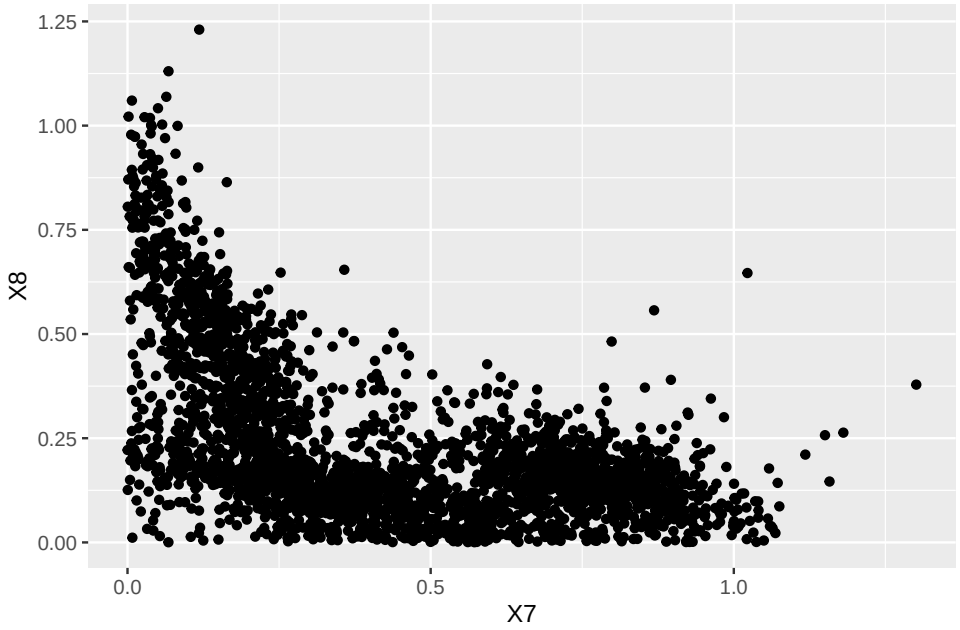

```
quartile_df <- data %>%
  summarize(first=quantile(drop_na(data)$X9, p=1/4),
            second=quantile(drop_na(data)$X9, p=1/2),
            third=quantile(drop_na(data)$X9, p=3/4)) %>%
  tidyr::gather(quartile, value)

data %>%
  ggplot(aes(x=X9)) +
    geom_histogram(bins=50) +
    # (aes(xintercept=median(X9)), size=1.5, color="red") +
    geom_vline(aes(xintercept=value), data=quartile_df,
               size=1,color="blue", linetype=2)
```



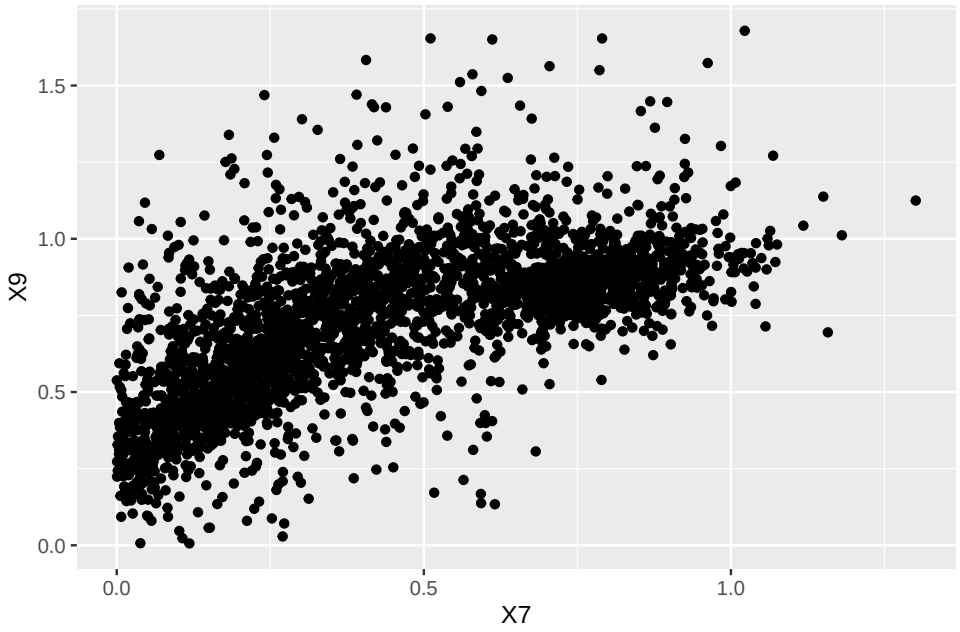

```r
data_long <- drop_na(data) %>%
  select(- starts_with("label")) %>%
  pivot_longer(cols = everything(), names_to = "variable", values_to = "value")

ggplot(data_long, aes(x = value)) +
  geom_histogram(bins = 50, fill = "skyblue", color = "black") +
  facet_wrap(~ variable, scales = "free") +
  theme_minimal() +
  labs(x = "Value", y = "Count", title = "Histograms of All Columns")
```



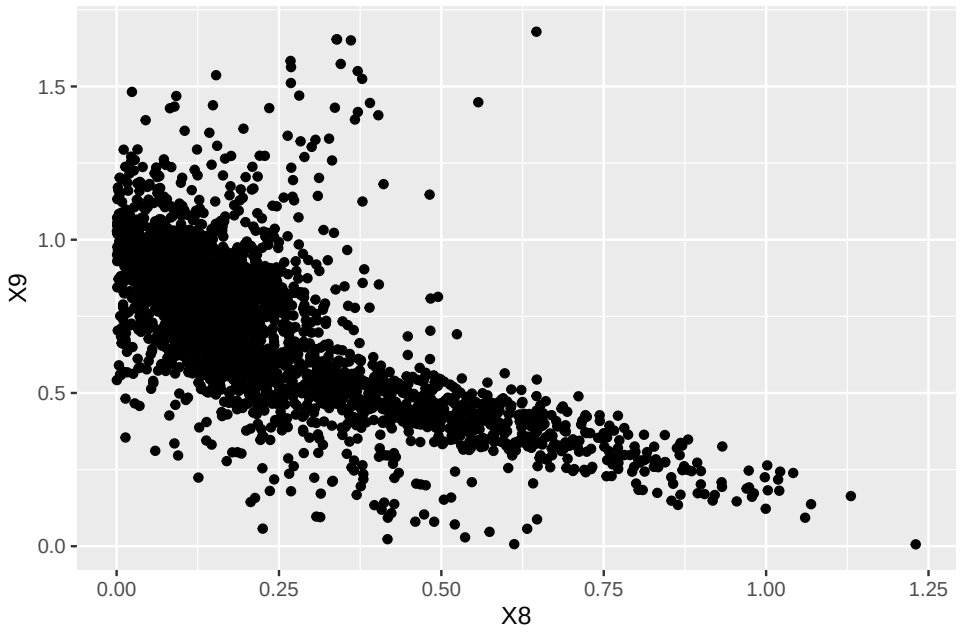

```
data <- read_csv("Data(2).csv")
data %>%
  ggplot(aes(x=X7, y=X8)) +
    geom_point()
```



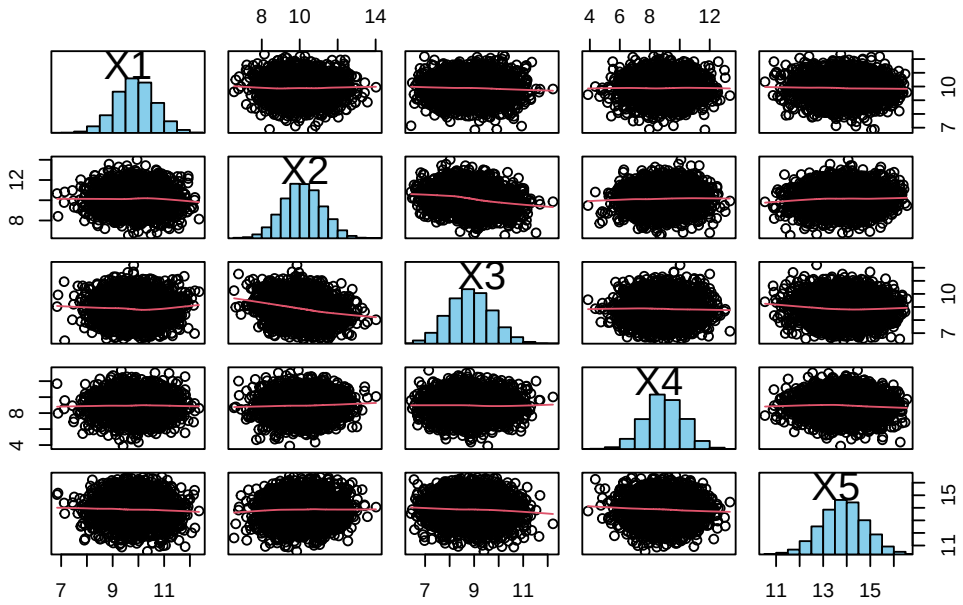
```
data <- read_csv("Data(2).csv")
data %>%
  ggplot(aes(x=X7, y=X9)) +
    geom_point()
```



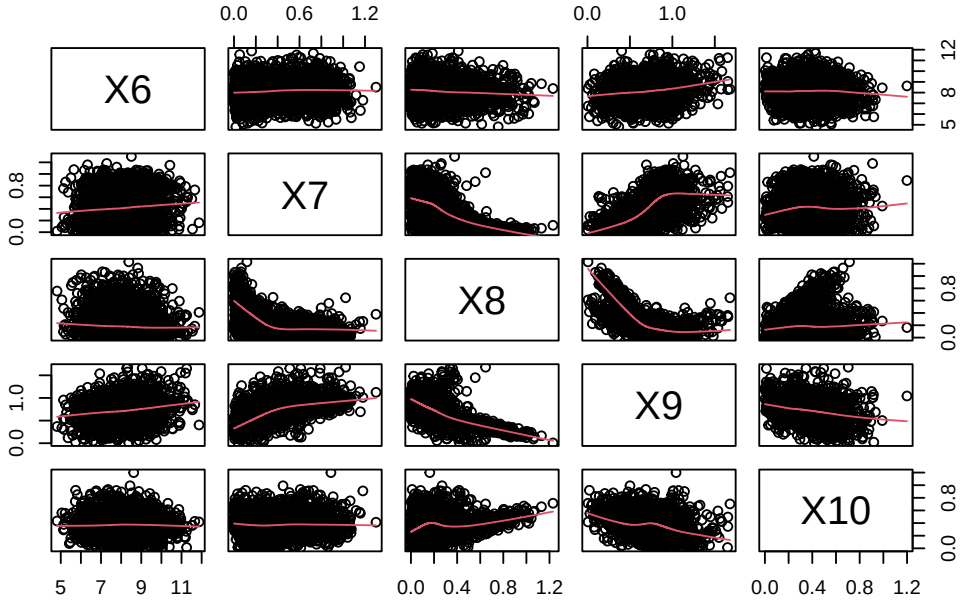

```
data <- read_csv("Data(2).csv")
data %>%
  ggplot(aes(x=X8, y=X9)) +
    geom_point()
```



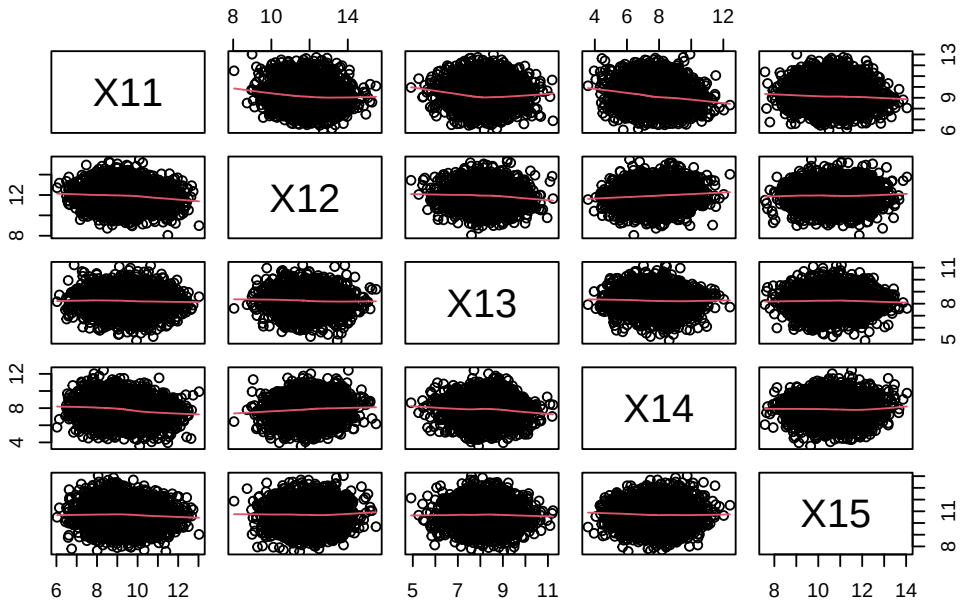

```
panel.hist <- function(x, ...) {
  usr <- par("usr")
  on.exit(par(usr))
  par(usr = c(usr[1:2], 0, 1.5))
  h <- hist(x, plot = FALSE)
  breaks <- h$breaks
  nB <- length(breaks)
  y <- h$counts
  y <- y/max(y)
  rect(breaks[-nB], 0, breaks[-1], y, col = "skyblue", ...)
}
pairs(data[1:5], panel = panel.smooth, diag.panel = panel.hist)
```



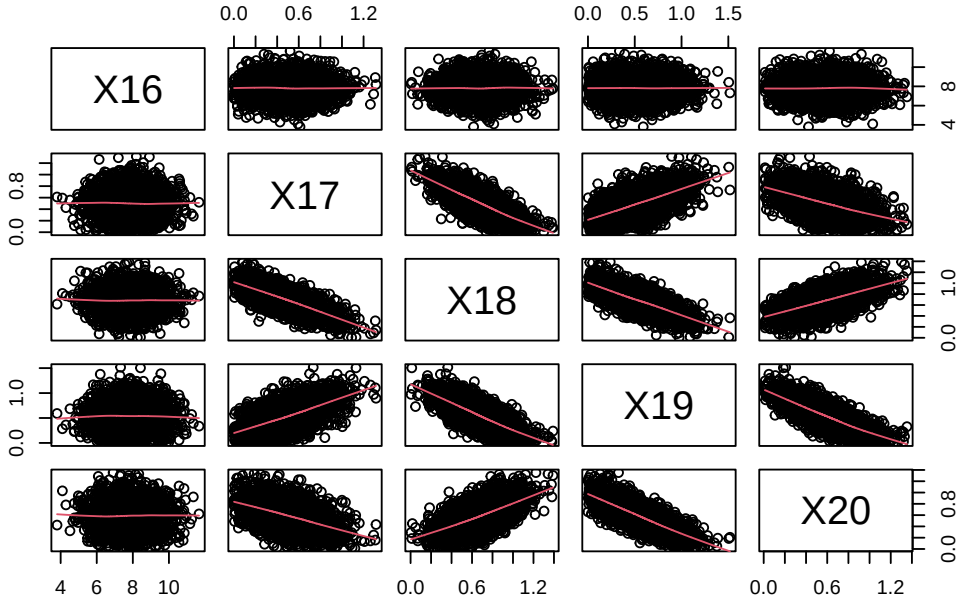

```
pairs(data[6:10], panel = panel.smooth)
```



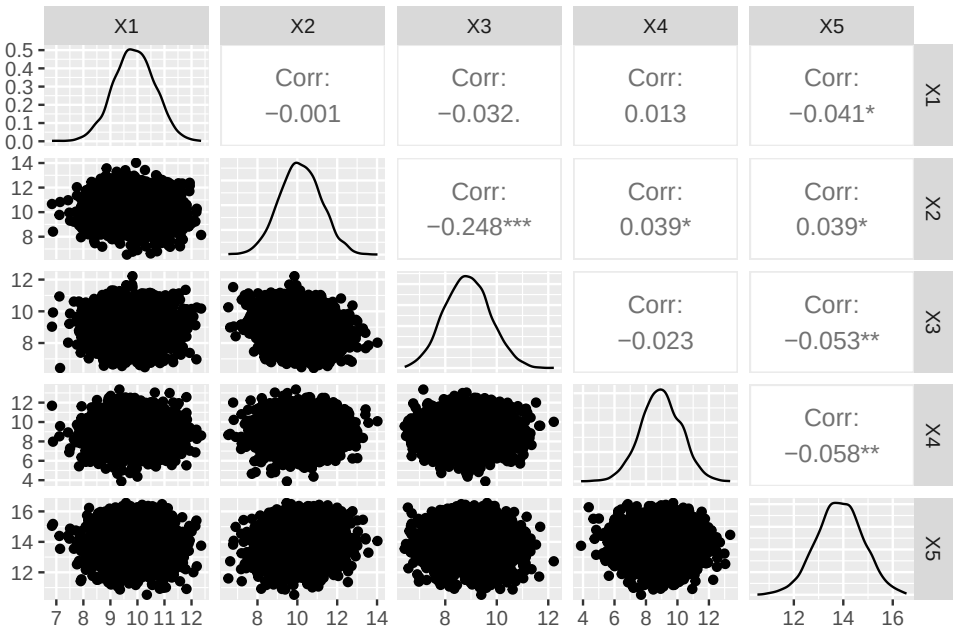

```
pairs(data[11:15], panel = panel.smooth)
```



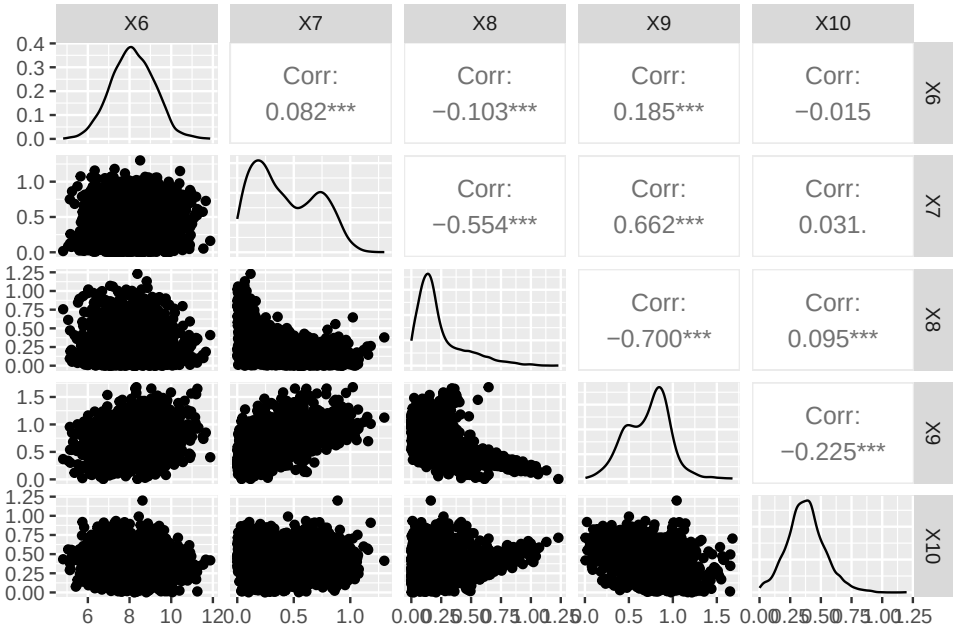

```
pairs(data[16:20], panel = panel.smooth)
```



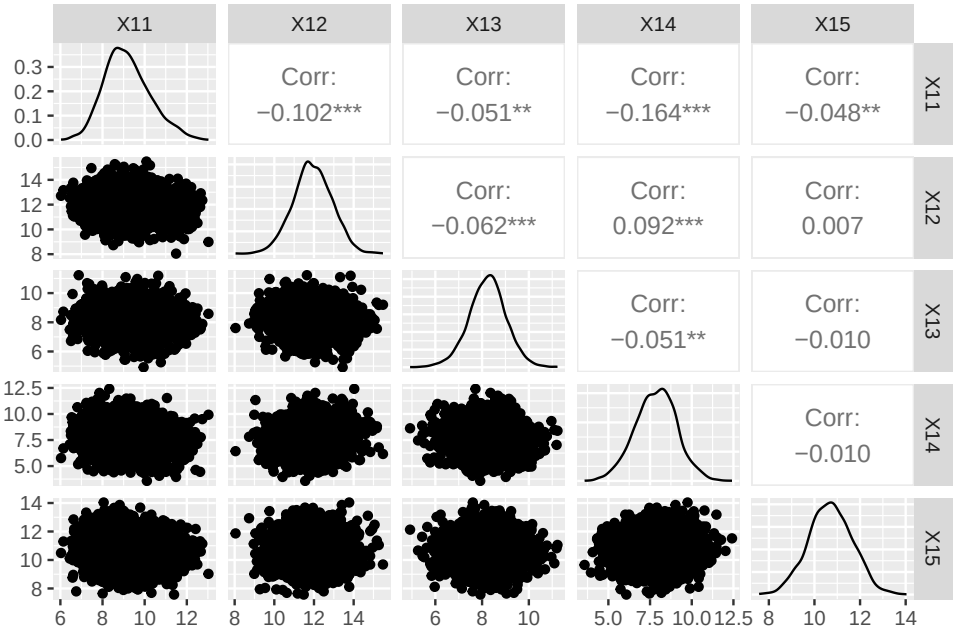

```
library(GGally)
ggpairs(data[1:5])
```



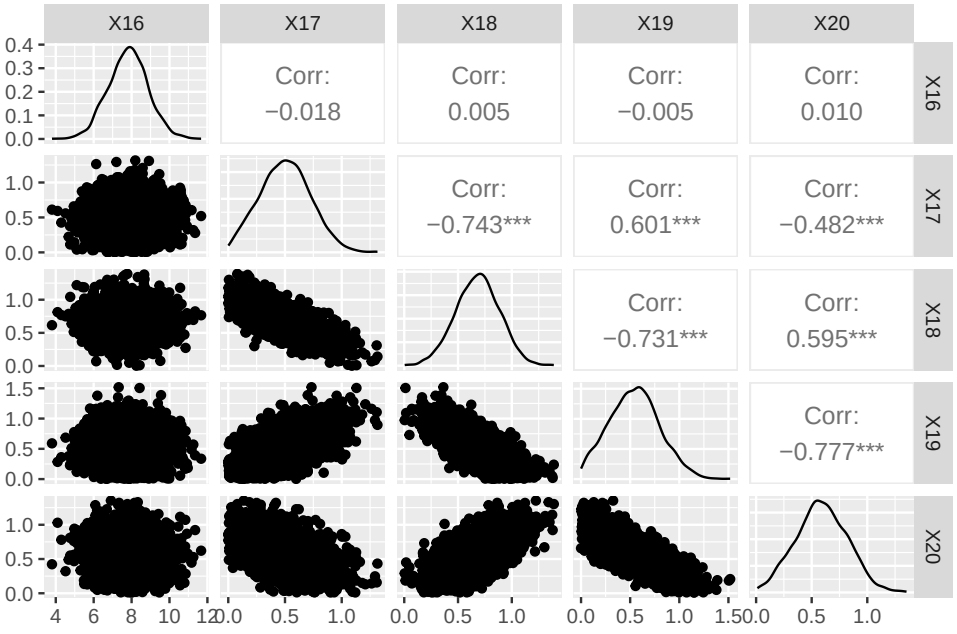

```
library(GGally)
ggpairs(data[6:10])
```



```
library(GGally)
ggpairs(data[11:15])
```



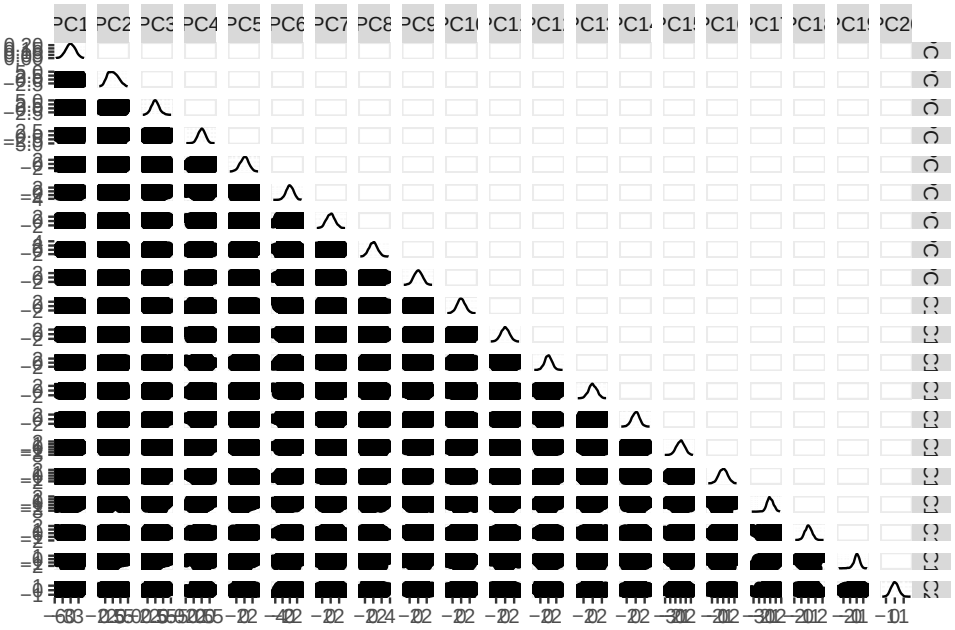

```
library(GGally)
ggpairs(data[16:20])
```



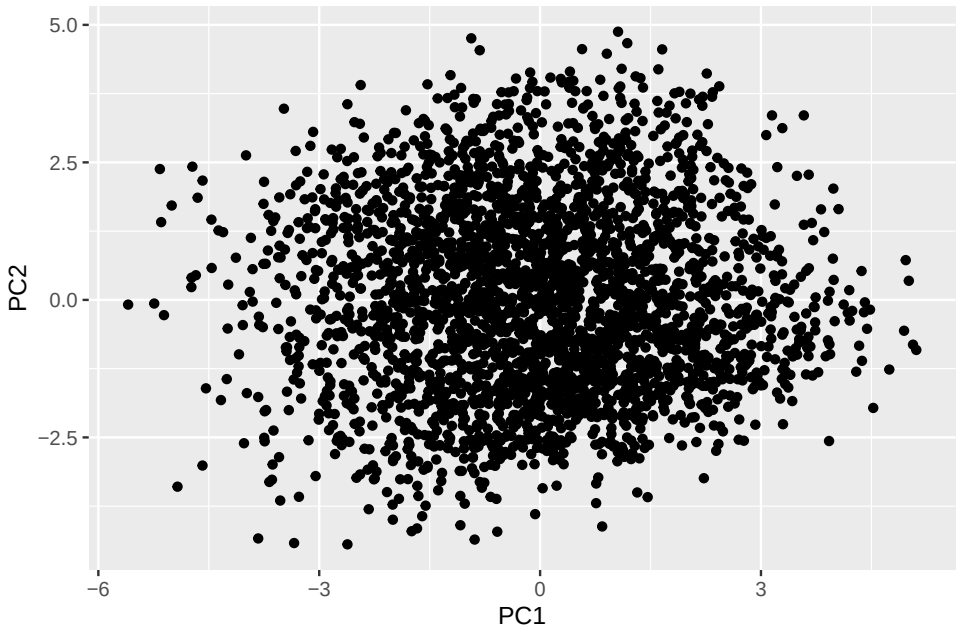
```
data_clean <- data %>% drop_na()
```

```
pc <- prcomp(data_clean[1:20],
             center = TRUE,
             scale = TRUE)
# ggpairs(pc)
```

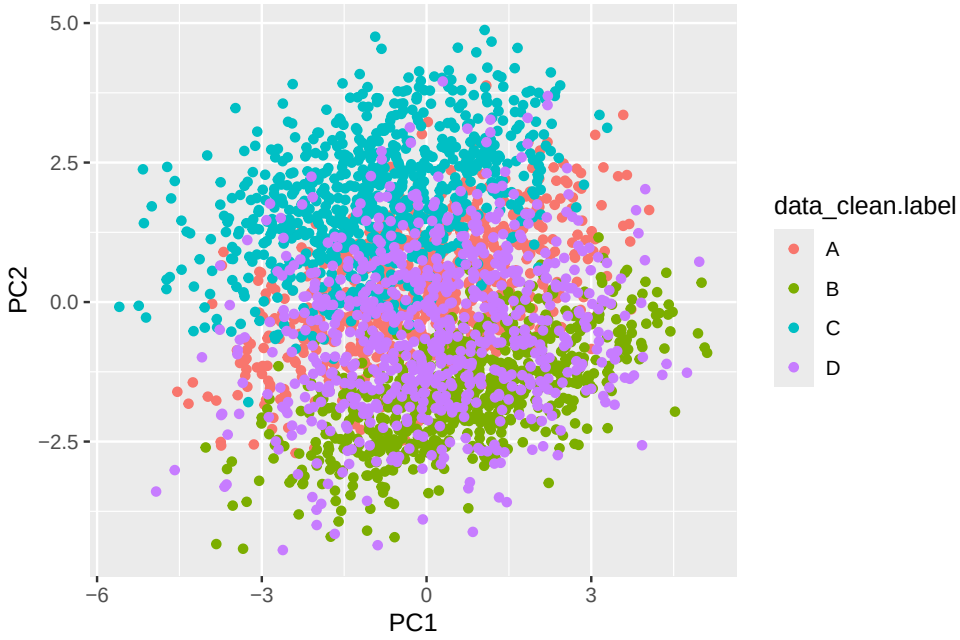
```
pc_data <- data.frame(pc$x)
# pc_data

ggpairs(pc_data)
```



```
ggplot(pc_data, aes(x=PC1, y=PC2)) +
       geom_point()
```

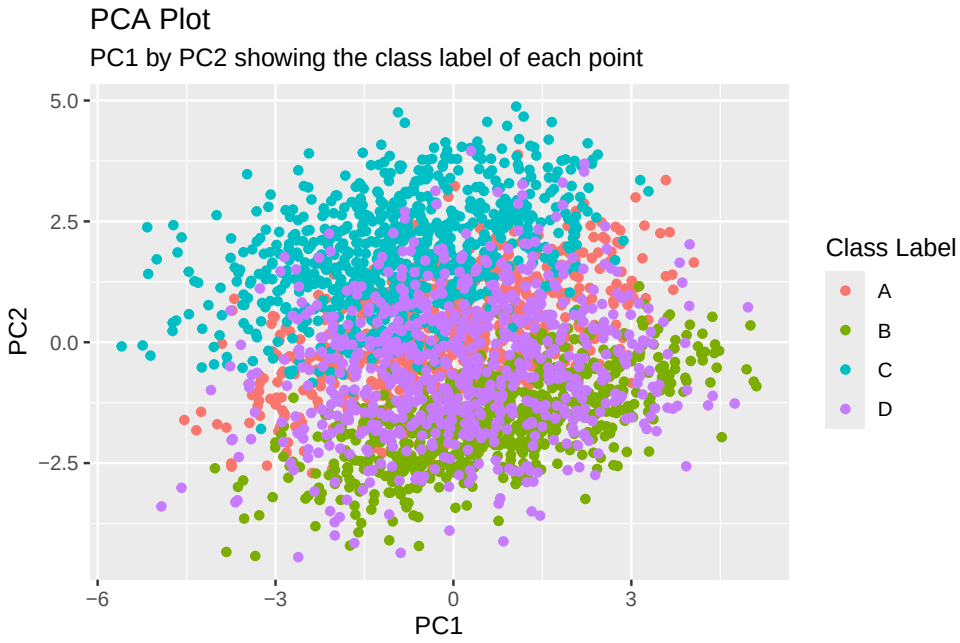



```
labelled_pca <- bind_cols(pc_data[1:2], data.frame(data_clean$label), .name_repair = "univers
# labelled_pca
ggplot(labelled_pca, aes(x=PC1, y=PC2, colour=data_clean.label)) +
       geom_point()
```



definitly a bit of loose clustering tendency here!

```
labelled_pca <- bind_cols(pc_data[1:2], data.frame(data_clean$label), .name_repair = "univers
# labelled_pca
ggplot(labelled_pca, aes(x=PC1, y=PC2, colour=data_clean.label)) +
      geom_point() +
      labs(
          title = "PCA Plot",
          subtitle = "PC1 by PC2 showing the class label of each point",
          colour="Class Label"
        )
```




```
library(hopkins)

hopkins_stat <- hopkins(data[1:20])
# we leave one row out to be the 'reference point' that we measure the distance to the other

# Print result
print(hopkins_stat)
```

```
[1] 0.5903055
```

the clustering tendency of the data using the all the rows of the 20 features is really low. This means the class labels are pretty uniformly distributed across the ranges of all the features.

We will try again only using columns that had migher magnitudes of correlation with the labels in the correlation heatmap.

```
correlated_data <- select(data, columns = c(
  1,
  2,
  3,
  4,
  7,
  8,
  9,
  11,
  12,
  14
  )
)
hopkins_stat_2 <- hopkins(correlated_data)
print(hopkins_stat)
```

```
[1] 0.5903055
```

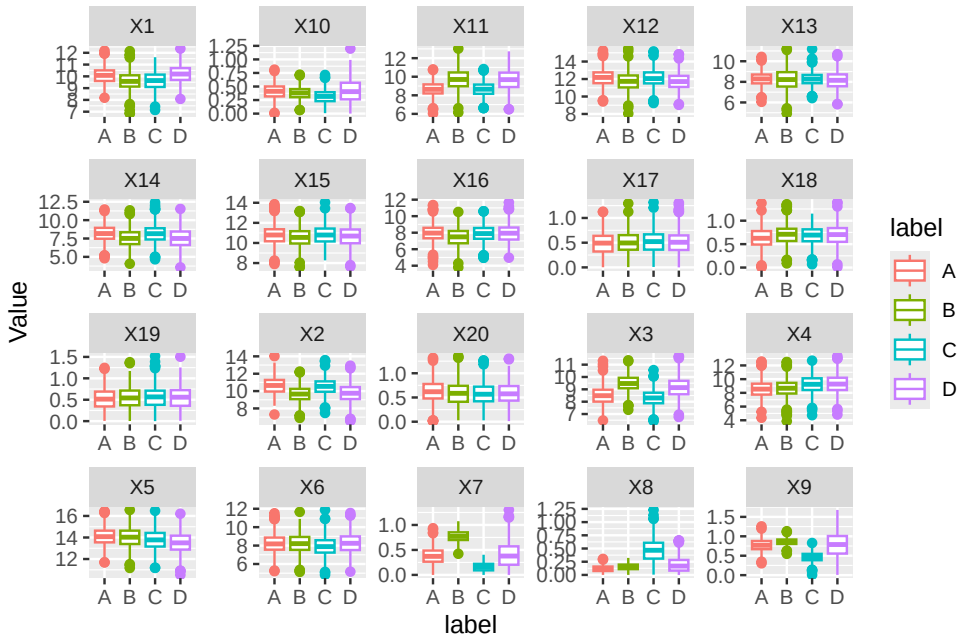
try without the NA columns

```
hopkins_stat <- hopkins(data_clean[1:20])
# we leave one row out to be the 'reference point' that we measure the distance to the other

# Print result
print(hopkins_stat)
```

```
[1] 0.9999994
```

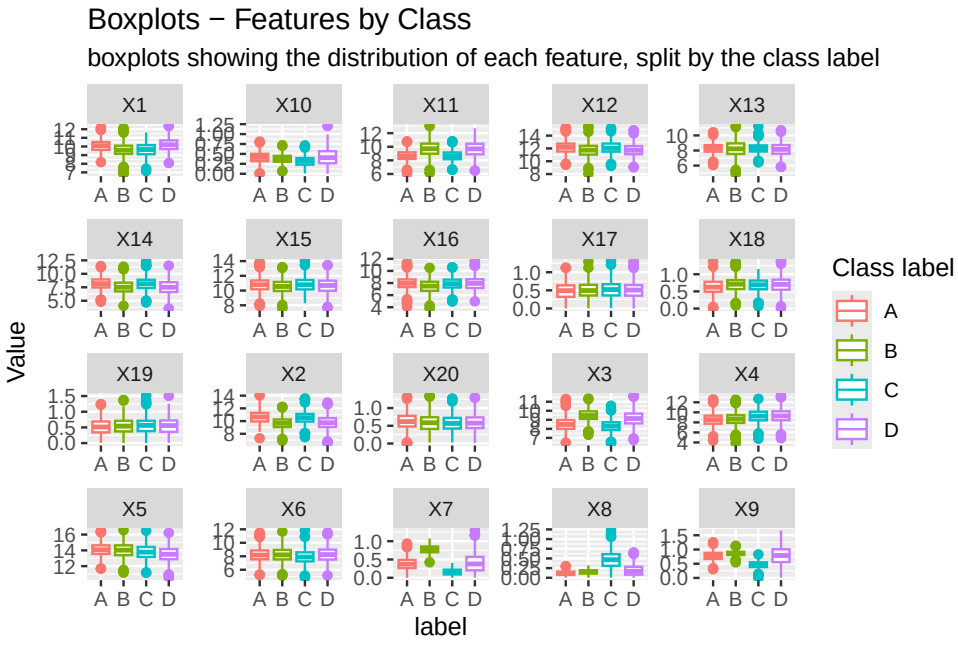
okay nvm, this is really high.

```
clean_and_long <- pivot_longer(data_clean, cols = 1:20, names_to= "Feature", values_to = "Va

boxplot <- ggplot( clean_and_long,  aes(label, Value, colour=label)  ) +
  geom_boxplot() +
  facet_wrap(~Feature, scales="free")

boxplot
```




```
# clean_and_long <- pivot_longer(data_clean, cols = 1:20, names_to= "Feature", values_to = "V
#
# boxplot <- ggplot( clean_and_long,  aes(Feature, Value, colour=label)  ) +
#   geom_boxplot() +
#   facet_wrap(~Value, scales="free")
#
# boxplot
```

```
clean_and_long <- pivot_longer(data_clean, cols = 1:20, names_to= "Feature", values_to = "Va

boxplot <- ggplot( clean_and_long,  aes(label, Value, colour=label)  ) +
  geom_boxplot() +
  facet_wrap(~Feature, scales="free") +
  labs(
    title = "Boxplots - Features by Class",
    subtitle = "boxplots showing the distribution of each feature, split by the class label"
    colour = "Class label"
  )


boxplot
```


Boxplots − Features by Class
boxplots showing the distribution of each feature, split by the class label
X1
X10
X11
X12
X13
X14
X15
X16
X17
X18
X19
X2
X20
X3
X4
X5
X6
X7
X8
X9
A B C D
Value
label
Class label
A
B
C
D