

MVVM Code Smell data exploration

In []:

```
# # Installing required Libraries
install.packages("correlation")
install.packages("bayestestR")
install.packages("see")
install.packages("ggplot2")
install.packages("tidyverse")
install.packages("poorman")
```

In []:

```
#Loading the installed libraries
library(readxl)          # reading in data
library(ggplot2)          # visualizing data
library(gridExtra)        # combining multiple plots
library(corrgram)         # visualizing data
library(corrplot)         # visualizing data
library(Hmisc)            # produces correlation matrices with p-values
library(ppcor)            # assesses partial correlations
```

In [40]:

```
#reading MVVM final input file
file_mvvm <- read.csv(file = "MVVM_input.csv", header=TRUE, stringsAsFactors = F)
```

In [41]:

```
head(file_mvvm)
```

ID	Magic.Number	Long.Statement	Missing.Defaults	Complex.Conditional	Unutilized.abstraction	Cyclic.Dependency	Insufficient.Modularization	Count
1	12	18	0	0	18	0	0	0
2	9	7	1	0	19	0	0	2
3	44	23	3	1	9	18	0	0
4	10	6	1	0	6	0	0	0
5	18	65	3	0	9	0	0	1
6	195	60	10	14	61	14	0	10



```
In [42]: str(file_mvvm) # shows the structure of the data frame
summary(file_mvvm) #provides summary statistics on the columns of the data frame

'data.frame': 40 obs. of 11 variables:
 $ ID                  : int 1 2 3 4 5 6 7 8 9 10 ...
 $ Magic.Number         : int 12 9 44 10 18 195 8 15 30 11 ...
 $ Long.Statement       : int 18 7 23 6 65 60 10 19 16 56 ...
 $ Missing.Defaults    : int 0 1 3 1 3 10 0 0 2 0 ...
 $ Complex.Conditional : int 0 0 1 0 0 14 0 0 0 0 ...
 $ Unutilized.abstraction : int 18 19 9 6 9 61 10 13 20 30 ...
 $ Cyclic.Dependency   : int 0 0 18 0 0 14 0 10 10 0 ...
 $ Insufficient.Modularization: int 0 2 0 0 1 10 0 1 1 6 ...
 $ Deficient.Encapsulation : int 18 13 9 0 8 35 4 2 5 10 ...
 $ Total.Code.Smells    : int 30 19 74 17 101 332 27 36 58 84 ...
 $ Total.Architectual.Smells : int 6 3 25 1 1 34 1 15 18 5 ...

      ID      Magic.Number  Long.Statement  Missing.Defaults
Min. : 1.00  Min. : 4.00  Min. : 4.00  Min. : 0.00
1st Qu.:10.75 1st Qu.: 15.00 1st Qu.: 15.50 1st Qu.: 0.00
Median :20.50 Median : 39.50 Median : 29.00 Median : 2.00
Mean   :20.50 Mean   : 79.62 Mean   : 43.25 Mean   : 3.85
3rd Qu.:30.25 3rd Qu.:106.25 3rd Qu.: 57.75 3rd Qu.: 5.25
Max.   :40.00  Max.   :388.00  Max.   :224.00 Max.   :27.00

Complex.Conditional Unutilized.abstraction Cyclic.Dependency
Min.   : 0.0      Min.   : 3.0      Min.   : 0.000
1st Qu.: 0.0      1st Qu.: 10.0     1st Qu.: 0.000
Median : 0.5      Median : 17.0     Median : 3.000
Mean   : 2.8      Mean   : 25.3     Mean   : 7.625
3rd Qu.: 3.0      3rd Qu.: 28.5     3rd Qu.:10.250
Max.   :23.0      Max.   :135.0     Max.   :49.000

Insufficient.Modularization Deficient.Encapsulation Total.Code.Smells
Min.   : 0.000      Min.   : 0.00      Min.   : 17.0
1st Qu.: 0.000      1st Qu.: 4.75      1st Qu.: 50.5
Median : 1.500      Median : 8.50      Median : 90.5
Mean   : 3.075      Mean   :11.18      Mean   :143.1
3rd Qu.: 5.000      3rd Qu.:15.75      3rd Qu.:226.2
Max.   :22.000      Max.   :43.00      Max.   :547.0

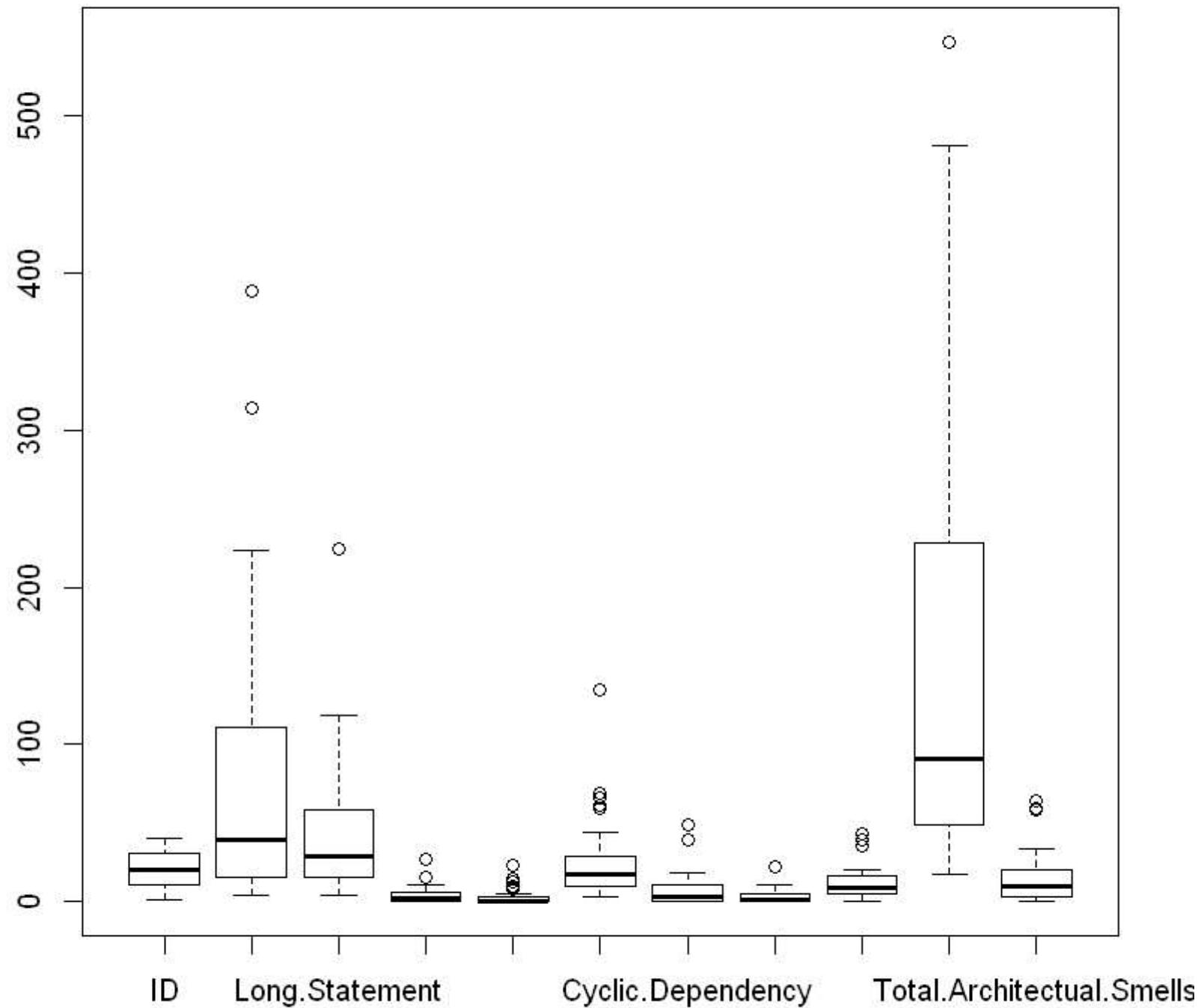
Total.Architectual.Smells
Min.   : 0.00
1st Qu.: 3.00
Median : 9.50
Mean   :14.22
3rd Qu.:19.00
Max.   :64.00
```

checking outliers with box plot

In [7]:

```
# Create a boxplot of the dataset, outliers are shown as two distinct points
boxplot(file_mvvm)$out
```

1.388
2.314
3.224
4.27
5.15
6.14
7.13
8.10
9.23
10.9
11.8
12.61
13.59
14.135
15.66
16.69
17.39
18.49
19.39
20.22
21.35
22.43
23.39
24.547
25.58
26.64
27.59



There are outliers present in all the columns.

In [8]:

```
#making a copy of original file before changing the column name
df_mvvm <- file_mvvm
```

renaming the column name for our ease in furthur calculations

In [9]:

```
#LOAD LIBRARY Library(tidyverse)
library(tidyverse)
#rename the column name
names(file_mvvm)[names(file_mvvm) == "Magic.Number"] <- "MN"
names(file_mvvm)[names(file_mvvm) == "Long.Statement"] <- "LS"
names(file_mvvm)[names(file_mvvm) == "Missing.Defaults"] <- "MD"
names(file_mvvm)[names(file_mvvm) == "Complex.Conditional"] <- "CC"
names(file_mvvm)[names(file_mvvm) == "Unutilized.abstraction"] <- "Ua"
names(file_mvvm)[names(file_mvvm) == "Cyclic.Dependency"] <- "CD"
names(file_mvvm)[names(file_mvvm) == "Insufficient.Modularization"] <- "IM"
names(file_mvvm)[names(file_mvvm) == "Deficient.Encapsulation"] <- "DE"
names(file_mvvm)[names(file_mvvm) == "Total.Architectual.Smells"] <- "TDS"
names(file_mvvm)[names(file_mvvm) == "Total.Code.Smells"] <- "TCS"
```

Registered S3 method overwritten by 'rvest':

```
method           from
read_xml.response xml2
-- Attaching packages ----- tidyverse 1.2.1 --
v tibble    3.1.1    v purrr    0.3.2
v tidyr     1.1.3    v dplyr    1.0.6
v readr     1.3.1    v stringr  1.4.0
v tibble    3.1.1    vforcats  0.4.0
Warning message:
"package 'tibble' was built under R version 3.6.3"Warning message:
"package 'tidyr' was built under R version 3.6.3"Warning message:
"package 'dplyr' was built under R version 3.6.3"-- Conflicts ----- tidyverse_conflicts() --
x dplyr::combine() masks gridExtra::combine()
x dplyr::filter()  masks stats::filter()
x dplyr::lag()     masks stats::lag()
```

```
In [10]: #creating a copy of dataframe
df <- file_mvvm
```

```
In [11]: #column names after renaming
colnames(file_mvvm)
```

1. 'ID'
2. 'MN'
3. 'LS'
4. 'MD'
5. 'CC'
6. 'Ua'
7. 'CD'
8. 'IM'
9. 'DE'
10. 'TCS'
11. 'TDS'

We can see there are lot of outliers present in our dataset.

We are targeting to remove outliers from all the columns for better correlation analysis.

```
In [12]: # finding outliers in our dataset - Long.Statement
Q1 <- quantile(file_mvvm$MN, probs=c(.25, .75), na.rm = FALSE)
Q2 <- quantile(file_mvvm$LS, probs=c(.25, .75), na.rm = FALSE)
Q3 <- quantile(file_mvvm$MD, probs=c(.25, .75), na.rm = FALSE)
Q4 <- quantile(file_mvvm$CC, probs=c(.25, .75), na.rm = FALSE)
Q5 <- quantile(file_mvvm$Ua, probs=c(.25, .75), na.rm = FALSE)
Q6 <- quantile(file_mvvm$CD, probs=c(.25, .75), na.rm = FALSE)
Q7 <- quantile(file_mvvm$IM, probs=c(.25, .75), na.rm = FALSE)
Q8 <- quantile(file_mvvm$DE, probs=c(.25, .75), na.rm = FALSE)
Q9 <- quantile(file_mvvm$TDS, probs=c(.25, .75), na.rm = FALSE)
Q10 <- quantile(file_mvvm$TCS, probs=c(.25, .75), na.rm = FALSE)

# how to find outliers in r - calculate Interquartile Range
iqr1 <- IQR(file_mvvm$MN)
iqr2 <- IQR(file_mvvm$LS)
```

```

iqr3 <- IQR(file_mvvm$MD)
iqr4 <- IQR(file_mvvm$CC)
iqr5 <- IQR(file_mvvm$Ua)
iqr6 <- IQR(file_mvvm$CD)
iqr7 <- IQR(file_mvvm$IM)
iqr8 <- IQR(file_mvvm$DE)
iqr9 <- IQR(file_mvvm$TDS)
iqr10 <- IQR(file_mvvm$TCS)

# how to find outliers in r - upper and Lower range
#up <- Q[2]+1.5*iqr # Upper Range
#low<- Q[1]-1.5*iqr # Lower Range

# how to remove outliers in (the removal)
file_mvvm<- subset(file_mvvm, file_mvvm$MN > (Q1[1] - 1.5*iqr1) & file_mvvm$MN <(Q1[2]+1.5*iqr1))
file_mvvm<- subset(file_mvvm, file_mvvm$LS > (Q2[1] - 1.5*iqr2) & file_mvvm$LS< (Q2[2]+1.5*iqr2))
file_mvvm<- subset(file_mvvm, file_mvvm$MD > (Q3[1] - 1.5*iqr3) & file_mvvm$MD< (Q3[2]+1.5*iqr3))
file_mvvm<- subset(file_mvvm, file_mvvm$CC > (Q4[1] - 1.5*iqr4) & file_mvvm$CC< (Q4[2]+1.5*iqr4))
file_mvvm<- subset(file_mvvm, file_mvvm$Ua > (Q5[1] - 1.5*iqr5) & file_mvvm$Ua< (Q5[2]+1.5*iqr5))
file_mvvm<- subset(file_mvvm, file_mvvm$CD > (Q6[1] - 1.5*iqr6) & file_mvvm$CD< (Q6[2]+1.5*iqr6))
file_mvvm<- subset(file_mvvm, file_mvvm$IM > (Q7[1] - 1.5*iqr7) & file_mvvm$IM< (Q7[2]+1.5*iqr7))
file_mvvm<- subset(file_mvvm, file_mvvm$DE > (Q8[1] - 1.5*iqr8) & file_mvvm$DE< (Q8[2]+1.5*iqr8))
file_mvvm<- subset(file_mvvm, file_mvvm$TDS >(Q9[1] - 1.5*iqr9) & file_mvvm$TDS<(Q9[2]+1.5*iqr9))
file_mvvm<- subset(file_mvvm, file_mvvm$TCS >(Q10[1] - 1.5*iqr10) & file_mvvm$TCS<(Q10[2]+1.5*iqr10))

```

In [13]:

```
#checking the data post removal of outliers
file_mvvm
```

	ID	MN	LS	MD	CC	Ua	CD	IM	DE	TCS	TDS
1	1	12	18	0	0	18	0	0	18	30	6
2	2	9	7	1	0	19	0	2	13	19	3
3	3	44	23	3	1	9	18	0	9	74	25
4	4	10	6	1	0	6	0	0	0	17	1
5	5	18	65	3	0	9	0	1	8	101	1
7	7	8	10	0	0	10	0	0	4	27	1
8	8	15	19	0	0	13	10	1	2	36	15
9	9	30	16	2	0	20	10	1	5	58	18

ID	MN	LS	MD	CC	Ua	CD	IM	DE	TCS	TDS	
10	10	11	56	0	0	30	0	6	10	84	5
11	11	119	25	6	2	16	2	2	11	163	8
14	14	35	35	11	3	28	0	0	6	94	2
15	15	42	24	3	0	12	8	0	5	87	4
16	16	36	12	0	0	18	8	0	10	54	12
17	17	61	41	0	2	6	2	5	4	115	3
20	20	32	6	2	0	20	0	0	6	46	6
22	22	43	32	2	0	44	0	2	4	83	4
25	25	36	57	0	0	25	13	0	19	95	27
27	27	128	14	0	0	28	7	6	1	157	12
29	29	98	73	1	2	44	2	5	19	181	10
30	30	6	20	0	1	17	0	4	6	27	3
31	31	19	10	3	0	17	0	0	7	32	2
32	32	37	12	1	1	15	3	1	12	59	8
35	35	212	52	11	3	7	12	6	7	335	10
36	36	53	43	6	0	4	8	2	5	110	14
37	37	135	19	3	1	8	3	4	10	167	9
38	38	4	10	3	0	11	0	2	3	18	2
39	39	15	4	1	0	3	0	0	7	23	0
40	40	8	43	4	0	9	0	2	3	79	4

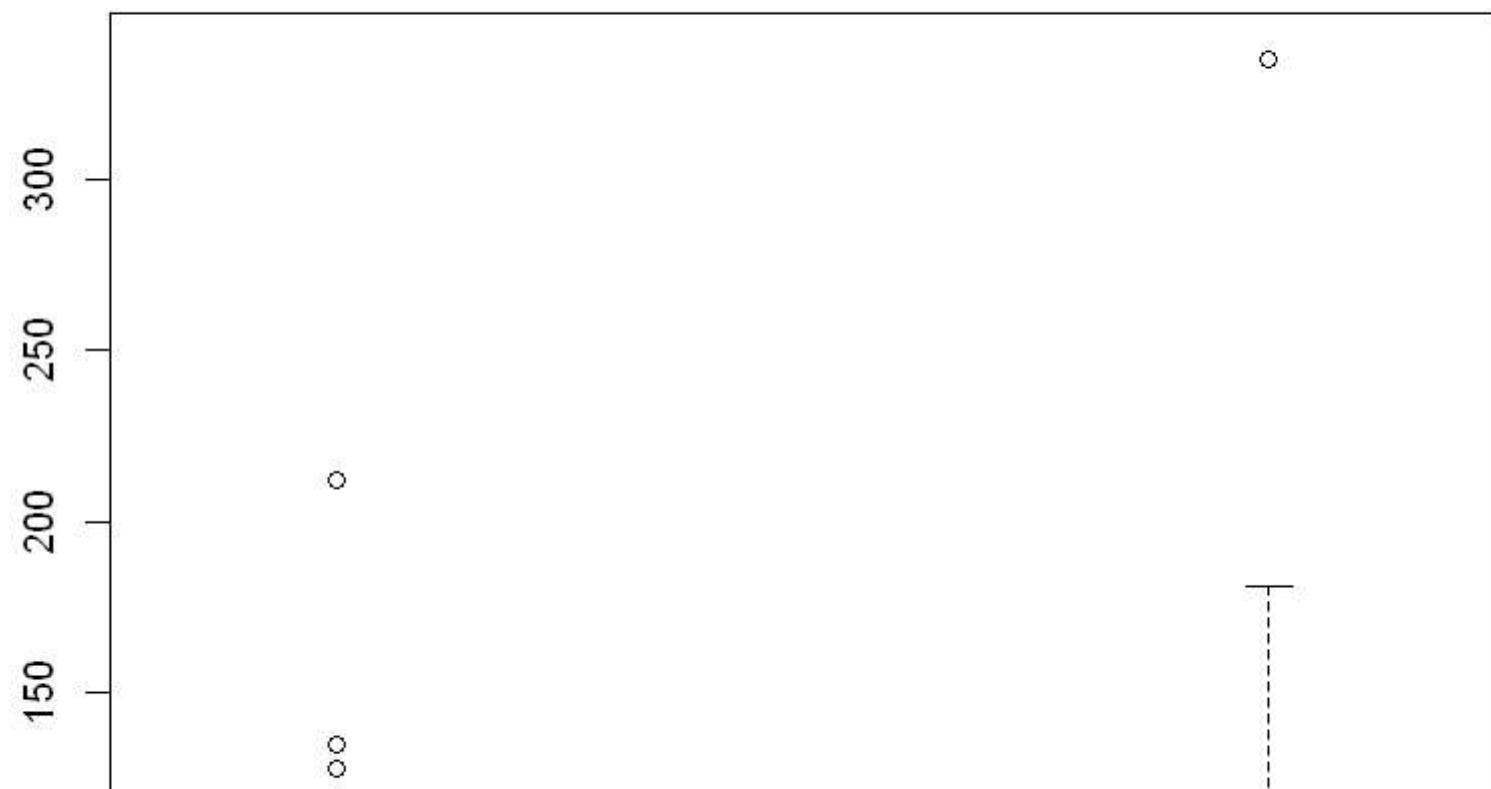
In [14]:

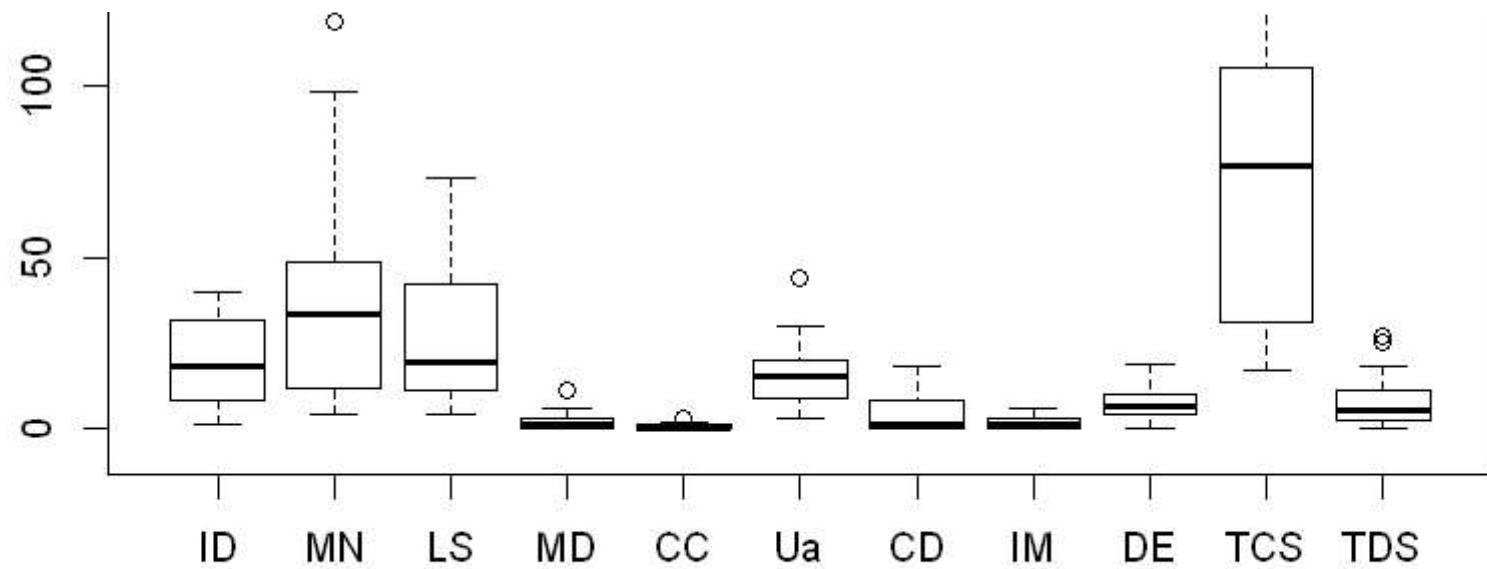
```
# Create a boxplot of the dataset post removal of some outliers
boxplot(file_mvvm)$out
```

1.119

2.128

3.212
4.135
5.11
6.11
7.3
8.3
9.44
10.44
11.335
12.25
13.27





We can see above that the outliers has been decreased in a certain amount and we can go for finding correlation between columns.

In [18]:

```
# Correlation test between multiple variables
round(cor(file_mvvm, use = "pairwise.complete.obs"), 2)
```

	ID	MN	LS	MD	CC	Ua	CD	IM	DE	TCS	TDS
ID	1.00	0.31	0.09	0.22	0.12	-0.08	-0.06	0.32	-0.09	0.30	-0.06
MN	0.31	1.00	0.29	0.47	0.62	0.04	0.36	0.59	0.10	0.94	0.25
LS	0.09	0.29	1.00	0.25	0.36	0.30	0.14	0.44	0.35	0.58	0.20
MD	0.22	0.47	0.25	1.00	0.63	-0.14	0.12	0.05	-0.13	0.56	-0.05
CC	0.12	0.62	0.36	0.63	1.00	0.07	0.09	0.39	0.17	0.66	0.00
Ua	-0.08	0.04	0.30	-0.14	0.07	1.00	-0.13	0.23	0.35	0.09	0.11
CD	-0.06	0.36	0.14	0.12	0.09	-0.13	1.00	-0.03	0.08	0.34	0.88
IM	0.32	0.59	0.44	0.05	0.39	0.23	-0.03	1.00	-0.02	0.64	-0.03

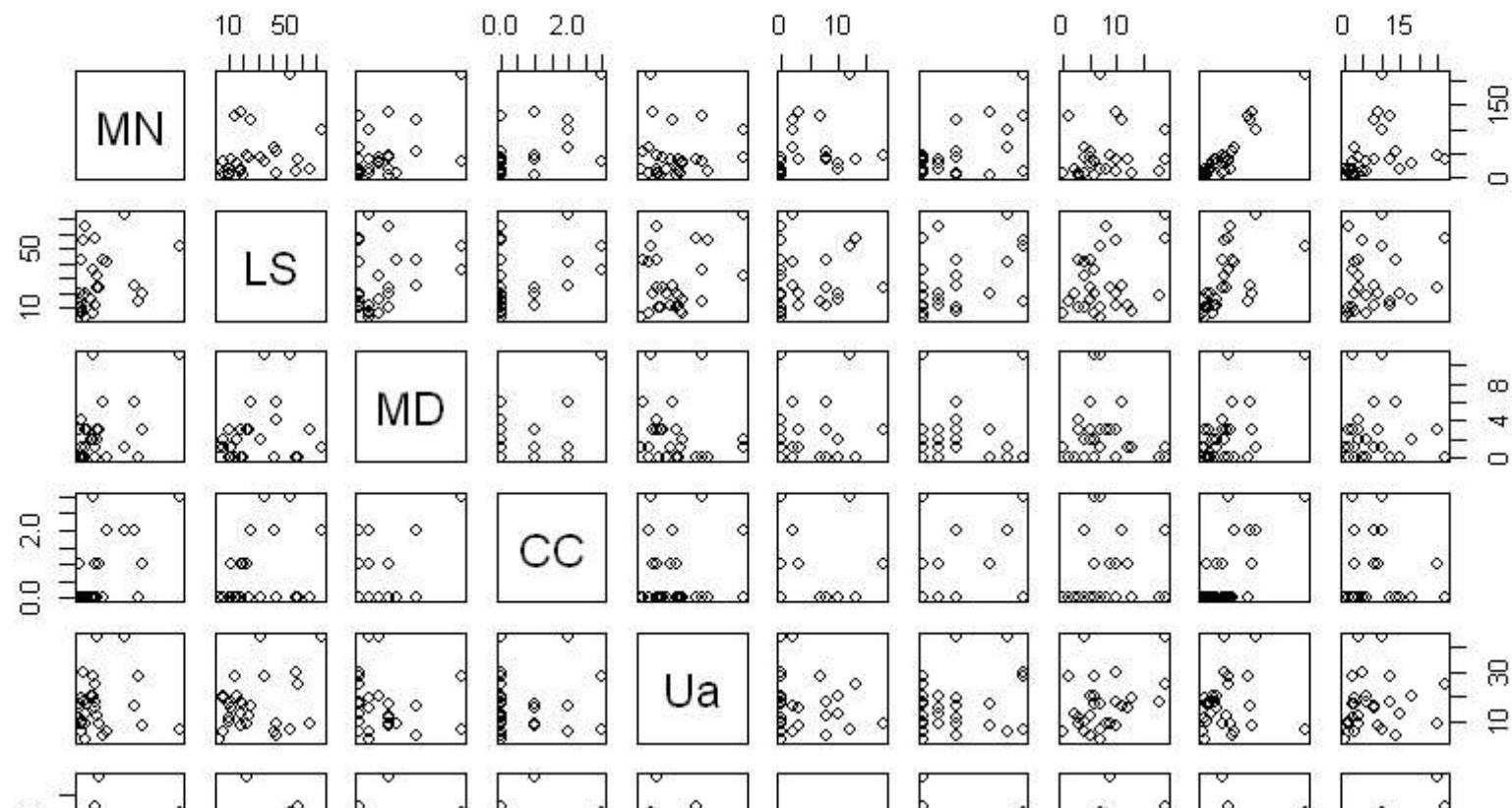
	ID	MN	LS	MD	CC	Ua	CD	IM	DE	TCS	TDS
DE	-0.09	0.10	0.35	-0.13	0.17	0.35	0.08	-0.02	1.00	0.14	0.32
TCS	0.30	0.94	0.58	0.56	0.66	0.09	0.34	0.64	0.14	1.00	0.24
TDS	-0.06	0.25	0.20	-0.05	0.00	0.11	0.88	-0.03	0.32	0.24	1.00

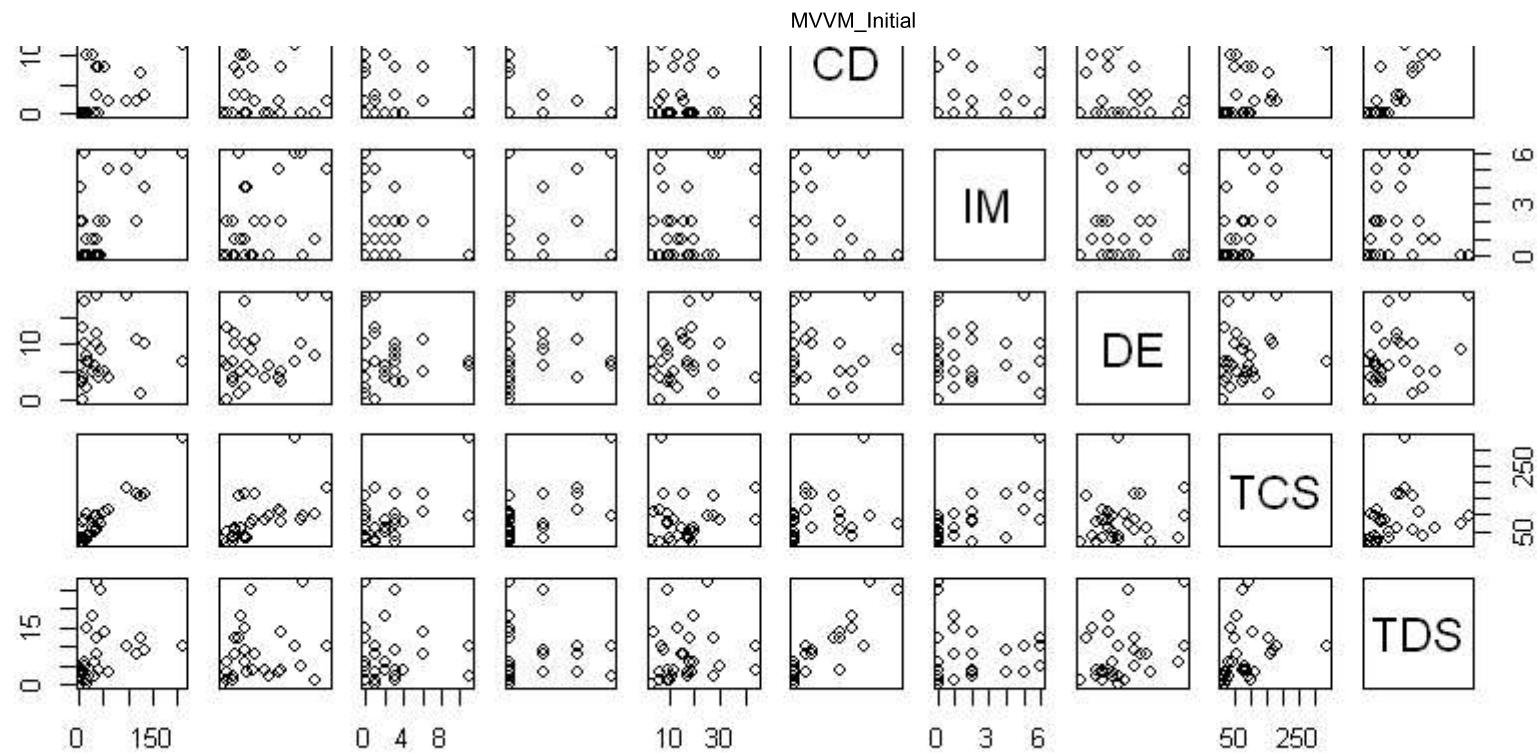
We are plotting three correlation graphs to find our relationship existance between each column present in our dataset.

- 1. Scatter plot matrices to check relationship within each columns of dataset

In [19]:

```
#Scatter plot matrix to check relationship within each columns of dataset
pairs(file_mvvm[, c(2, 3:11)])
```





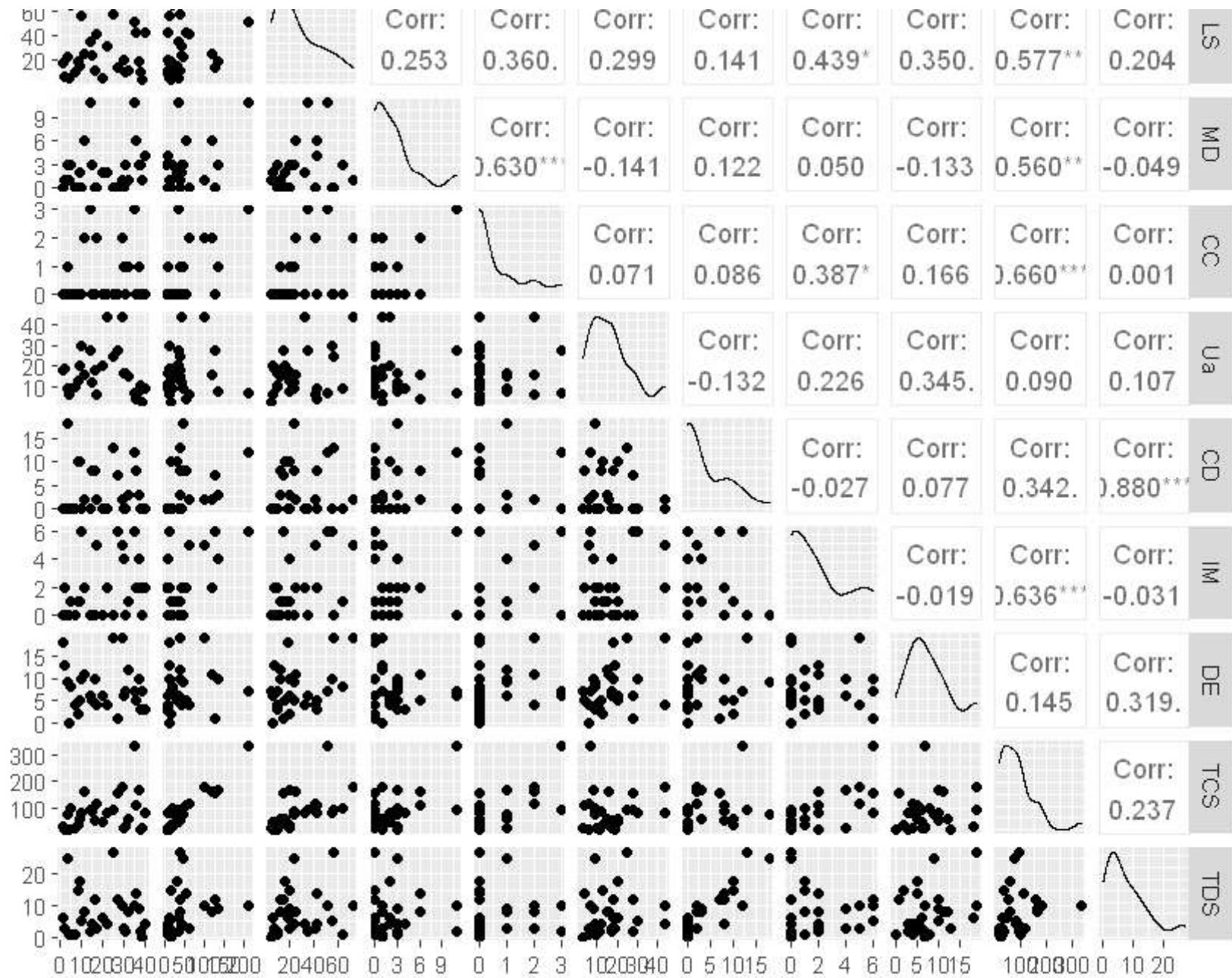
- 1. pairs plot to check correlation pattern between multiple variables

In [22]:

```
#create pairs plot
#install.packages('GGally')
library(GGally)
ggpairs(file_mvvm)
```



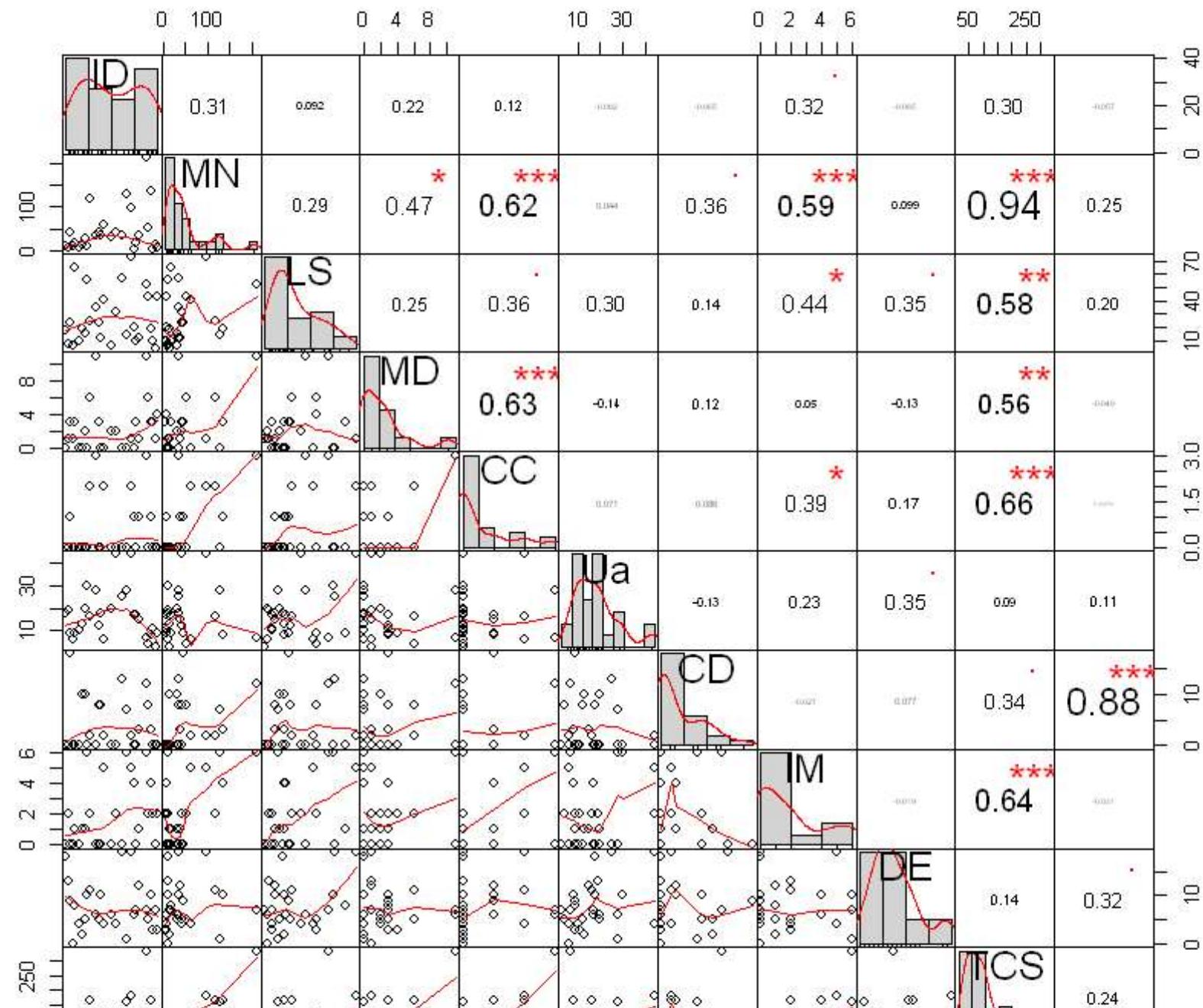
MVVM Initia

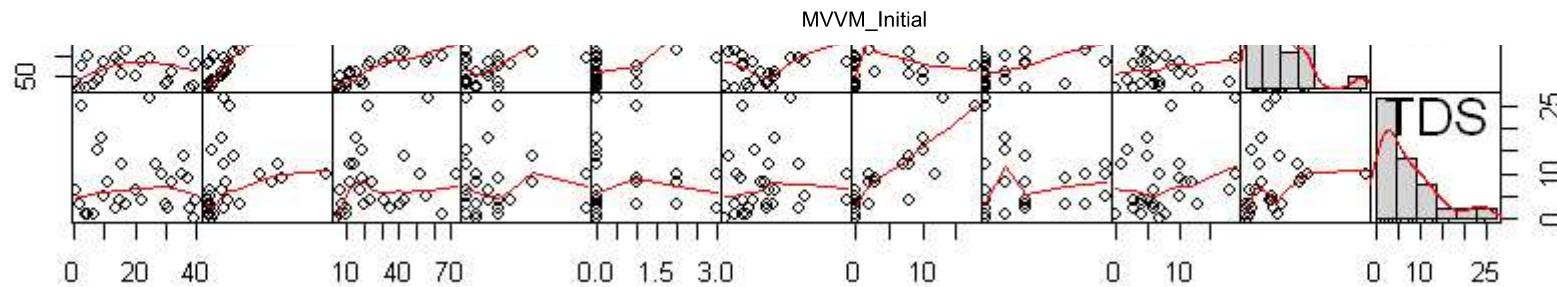


- 1. displaying PerformanceAnalytics chart of correlation matrix between variables

```
In [25]: # install.packages("PerformanceAnalytics")
```

```
# Library(PerformanceAnalytics)
chart.Correlation(file_mvvm, histogram=TRUE, pch=19)
```





In the above plot:

- The distribution of each variable is shown on the diagonal.
- On the bottom of the diagonal : the bivariate scatter plots with a fitted line are displayed
- On the top of the diagonal : the value of the correlation plus the significance level as stars
- Each significance level is associated to a symbol : p-values(0, 0.001, 0.01, 0.05, 0.1, 1) <=> symbols("****", "***", "**", "*", ".", " ")

From the above three plots we can see few positive correlation , which are :

- MN and CC
- MD and CC
- MN and IM
- TDS and TCS are showing high correlation as its a summation column values for the rest.

Hence we will further do the pearson, kendall, spearman on these columns.

- Pearson correlation evaluates the linear relationship between two continuous variables.
- Kendall and spearman correlation evaluates the monotonic relationship between variables.

In [26]:

```
#pearson1 - MN and CC
p1 <- cor(file_mvvm$MN, file_mvvm$CC, method = "pearson")
p1
cor.test(file_mvvm$MN, file_mvvm$CC, use = 'complete.obs')
```

0.616112864035666

Pearson's product-moment correlation

```
data: file_mvvm$MN and file_mvvm$CC
t = 3.9885, df = 26, p-value = 0.0004817
```

```
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.3155727 0.8043125
sample estimates:
cor
0.6161129
```

Observation: Our p-value is significant and the 95% confidence interval confirms this as the range does not contain zero.

This suggests the pearson correlation between MN(Magic.Numbe) and CC(Complex.Conditional) per architecture is r = 0.62 with 95% confidence of being between 0.32 and 0.80.

In [45]:

```
#pearson2 - MD and CC
p2 <- cor(file_mvvm$MD, file_mvvm$CC, method = "pearson")
p2
cor.test(file_mvvm$MD, file_mvvm$CC, use = 'complete.obs')
```

```
Error in cor(file_mvvm$MD, file_mvvm$CC, method = "pearson"): supply both 'x' and 'y' or a matrix-like 'x'
Traceback:
```

1. cor(file_mvvm\$MD, file_mvvm\$CC, method = "pearson")
2. stop("supply both 'x' and 'y' or a matrix-like 'x'")

Observation: Our p-value is significant and the 95% confidence interval confirms this as the range does not contain zero.

This suggests the pearson correlation between MD(Missing.Defaults) and CC(Complex.Conditional) per architecture is r = 0.63 with 95% confidence of being between 0.34 and 0.81.

In [32]:

```
#pearson3 - MN and IM
p3 <- cor(file_mvvm$MN, file_mvvm$IM, method = "pearson")
p3
cor.test(file_mvvm$MN, file_mvvm$IM, use = 'complete.obs')
```

0.588419310898783

Pearson's product-moment correlation

```
data: file_mvvm$MN and file_mvvm$IM
t = 3.7108, df = 26, p-value = 0.0009894
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.2759122 0.7884182
sample estimates:
```

```
cor
0.5884193
```

Observation: Our p-value is significant and the 95% confidence interval confirms this as the range does not contain zero.

This suggests the pearson correlation between MN(Magic.Number) and IM(Insufficient.Modularization) per architecture is $r = 0.59$ with 95% confidence of being between 0.28 and 0.79.

In [27]:

```
#spearman1 - MN and CC
s1 <- cor(file_mvvm$MN, file_mvvm$CC, method = "spearman")
s1
cor.test(file_mvvm$MN, file_mvvm$CC, use = 'complete.obs', method = "spearman")
```

0.521424596320157

Warning message in cor.test.default(file_mvvm\$MN, file_mvvm\$CC, use = "complete.obs", :
 "Cannot compute exact p-value with ties"

Spearman's rank correlation rho

data: file_mvvm\$MN and file_mvvm\$CC
 S = 1748.7, p-value = 0.004434
 alternative hypothesis: true rho is not equal to 0
 sample estimates:
 rho
 0.5214246

Observation: Our p-value is significant .

This suggests the Spearman rank correlation rho between MN(Magic.Number) and CC(Complex.Conditional) per architecture is $r = 0.52$ with P-vale less than 0.05 we can reject the null hypothesis and say there is relatiopn between two variables.

In [30]:

```
#spearman2 - MD and CC
s2 <- cor(file_mvvm$MD, file_mvvm$CC, method = "spearman")
s2
cor.test(file_mvvm$MD, file_mvvm$CC, use = 'complete.obs', method = "spearman")
```

0.298708205235264

Warning message in cor.test.default(file_mvvm\$MD, file_mvvm\$CC, use = "complete.obs", :
 "Cannot compute exact p-value with ties"

Spearman's rank correlation rho

data: file_mvvm\$MD and file_mvvm\$CC
 S = 2562.5, p-value = 0.1226

```
alternative hypothesis: true rho is not equal to 0
sample estimates:
rho
0.2987082
```

Observation: Our p-value is not significant and the 95% confidence interval confirms this as the range does not contain zero.

Cant reject the null hypothesis, no relationship present between variables.

In [33]:

```
#spearman3 - MN and IM
s3 <- cor(file_mvvm$MN, file_mvvm$IM, method = "spearman")
s3
cor.test(file_mvvm$MN, file_mvvm$IM, use = 'complete.obs', method = "spearman")
```

0.296205974575754

```
Warning message in cor.test.default(file_mvvm$MN, file_mvvm$IM, use = "complete.obs", :
"Cannot compute exact p-value with ties"
Spearman's rank correlation rho
```

```
data: file_mvvm$MN and file_mvvm$IM
S = 2571.7, p-value = 0.1259
alternative hypothesis: true rho is not equal to 0
sample estimates:
rho
0.296206
```

Observation: Our p-value is not significant .

Cant reject the null hypothesis, no relationship present between variables.

In [28]:

```
#kendall1 - MN and CC
k1 <- cor(file_mvvm$MN, file_mvvm$CC, method = "kendall")
k1
cor.test(file_mvvm$MN, file_mvvm$CC, use = 'complete.obs', method = "kendall")
```

0.415747548024836

```
Warning message in cor.test.default(file_mvvm$MN, file_mvvm$CC, use = "complete.obs", :
"Cannot compute exact p-value with ties"
Kendall's rank correlation tau
```

```
data: file_mvvm$MN and file_mvvm$CC
z = 2.7167, p-value = 0.006594
alternative hypothesis: true tau is not equal to 0
```

```
sample estimates:
  tau
0.4157475
```

Observation: Our p-value is significant and the 95% confidence interval confirms this as the range does not contain zero.

This suggests the kendall rank correlation rho between MN(Magic.Number) and CC(Complex.Conditional) per architecture is $r = 0.52$ with P-vale less than 0.05 we can reject the null hypothesis and say there is relatiopn between two variables.

In [31]:

```
#kendall2 - MD and CC
k2 <- cor(file_mvvm$MD, file_mvvm$CC, method = "kendall")
k2
cor.test(file_mvvm$MD, file_mvvm$CC, use = 'complete.obs', method = "kendall")
```

0.258148475045959

Warning message in cor.test.default(file_mvvm\$MD, file_mvvm\$CC, use = "complete.obs", :
 "Cannot compute exact p-value with ties"
 Kendall's rank correlation tau

data: file_mvvm\$MD and file_mvvm\$CC
 z = 1.5787, p-value = 0.1144
 alternative hypothesis: true tau is not equal to 0
 sample estimates:
 tau
 0.2581485

Observation: Our p-value is not significant.

Cant reject the null hypothesis, no relationship present between variables.

In [34]:

```
#kendall3 - MN and IM
k3 <- cor(file_mvvm$MN, file_mvvm$IM, method = "kendall")
k3
cor.test(file_mvvm$MN, file_mvvm$IM, use = 'complete.obs', method = "kendall")
```

0.224733287487747

Warning message in cor.test.default(file_mvvm\$MN, file_mvvm\$IM, use = "complete.obs", :
 "Cannot compute exact p-value with ties"
 Kendall's rank correlation tau

data: file_mvvm\$MN and file_mvvm\$IM

```

z = 1.5465, p-value = 0.122
alternative hypothesis: true tau is not equal to 0
sample estimates:
tau
0.2247333

```

Observation: Our p-value is not significant .

Cant reject the null hypothesis, no relationship present between variables.

In [35]:

```
#printing all the correlation values
cat(p1,s1,k1, p2,s2,k2, p3,s3,k3)
```

```
0.6161129 0.5214246 0.4157475 0.6298136 0.2987082 0.2581485 0.5884193 0.296206 0.2247333
```

Observation

In [37]:

```
#summarizing the correlation in a tabular format
corr_df <- data.frame(
  Sl_No = c(1,2,3,4,5,6,7,8,9),
  Corr_Method = c("Pearson", "spearman", "kendall", "Pearson", "spearman", "kendall", "Pearson", "spearman", "kendall"),
  Column_related = c("MN and CC", "MN and CC", "MN and CC", "MD and CC", "MD and CC",
                     "MD and CC", "MN and IM", "MN and IM", "MN and IM"),
  corr_value = c(0.6161129 ,0.5214246 ,0.4157475 ,0.6298136 ,0.2987082 ,0.2581485 ,0.5884193 ,0.296206 ,0.2247333)
)

corr_df
```

Sl_No	Corr_Method	Column_related	corr_value
1	Pearson	MN and CC	0.6161129
2	spearman	MN and CC	0.5214246
3	kendall	MN and CC	0.4157475
4	Pearson	MD and CC	0.6298136
5	spearman	MD and CC	0.2987082
6	kendall	MD and CC	0.2581485
7	Pearson	MN and IM	0.5884193
8	spearman	MN and IM	0.2962060

SI_No	Corr_Method	Column_related	corr_value
9	kendall	MN and IM	0.2247333

9 kendall MN and IM 0.2247333

In [38]:

```
knitr::kable(corr_df, format = "markdown")
```

S1_No	Corr_Method	Column_related	corr_value
1	Pearson	MN and CC	0.6161129
2	spearman	MN and CC	0.5214246
3	kendall	MN and CC	0.4157475
4	Pearson	MD and CC	0.6298136
5	spearman	MD and CC	0.2987082
6	kendall	MD and CC	0.2581485
7	Pearson	MN and IM	0.5884193
8	spearman	MN and IM	0.2962060
9	kendall	MN and IM	0.2247333

- From the above correlation tests we can conclude that there is high correlation present within MVVM architecture between Magic.Number & Complex.Conditional, Missing.Defaults & Complex.Conditional , Magic.Number & Insufficient.Modularization

In []: