Q46-WilliamKennedy-300015367

William Kennedy

2023-12-04

4. Cluster the Wine dataset using DBSCAN, for various algorithm parameters. Validate your results.

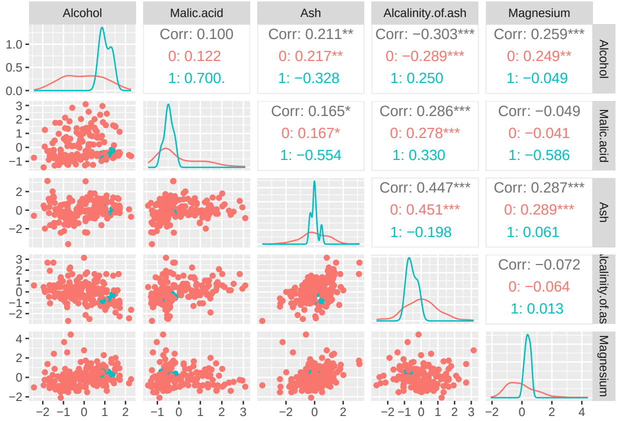
I will only run DCScan on the first 5 features, for the sake of simplicity and computational complexity.

```
library("fpc")
```

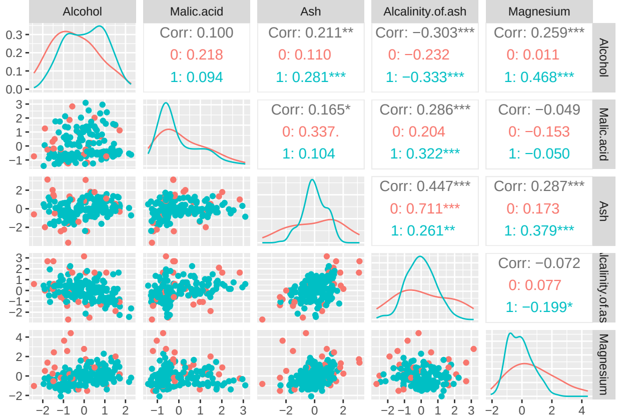
```
## Warning: package 'fpc' was built under R version 4.3.2
```

```
library(ggplot2)
my_dens <- function(data, mapping, ..., low = "#132B43",
  high = "#56B1F7") {
  ggplot(data = data, mapping=mapping) +
  geom_density(..., alpha=0.7)
}

wine = read.csv("wine.csv")
wine.sc = data.frame(scale(wine[-1,]))
dbscan1 = fpc::dbscan(wine.sc[,2:6], eps = 0.75, MinPts = 6)
dbscan2 = fpc::dbscan(wine.sc[,2:6], eps = 1.25, MinPts = 6)
dbscan3 = fpc::dbscan(wine.sc[,2:6], eps = 0.75, MinPts = 10)
dbscan4 = fpc::dbscan(wine.sc[,2:6], eps = 1.25, MinPts = 10)
dbscan5 = fpc::dbscan(wine.sc[,2:6], eps = 0.75, MinPts = 15)
dbscan6 = fpc::dbscan(wine.sc[,2:6], eps = 1.25, MinPts = 15)


GGally::ggpairs(wine.sc[,2:6],
ggplot2::aes(color=as.factor(dbscan1$cluster)),
diag=list(continuous=my_dens))
```

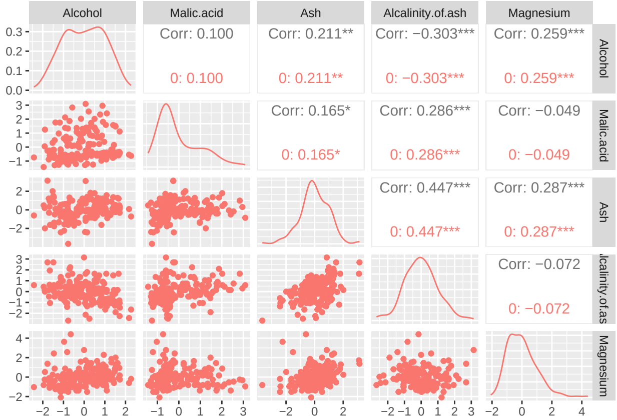
```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```



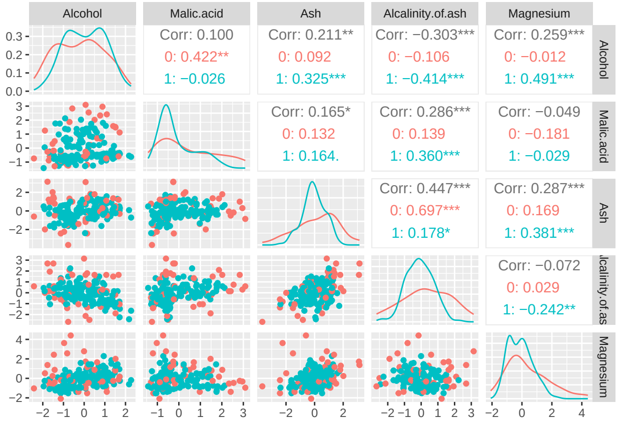
```
GGally::ggpairs(wine.sc[,2:6],
ggplot2::aes(color=as.factor(dbscan2$cluster)),
diag=list(continuous=my_dens))
```



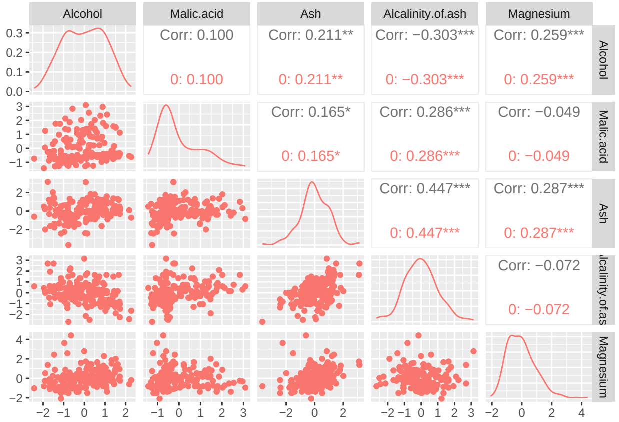
```
GGally::ggpairs(wine.sc[,2:6],
ggplot2::aes(color=as.factor(dbscan3$cluster)),
diag=list(continuous=my_dens))
```



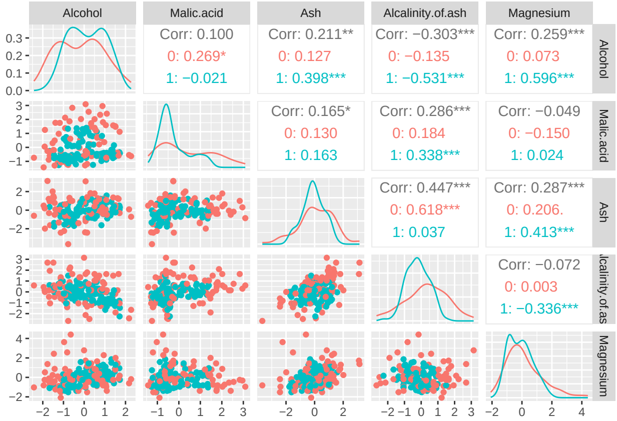
```
GGally::ggpairs(wine.sc[,2:6],
ggplot2::aes(color=as.factor(dbscan4$cluster)),
diag=list(continuous=my_dens))
```



```
GGally::ggpairs(wine.sc[,2:6],
ggplot2::aes(color=as.factor(dbscan5$cluster)),
diag=list(continuous=my_dens))
```



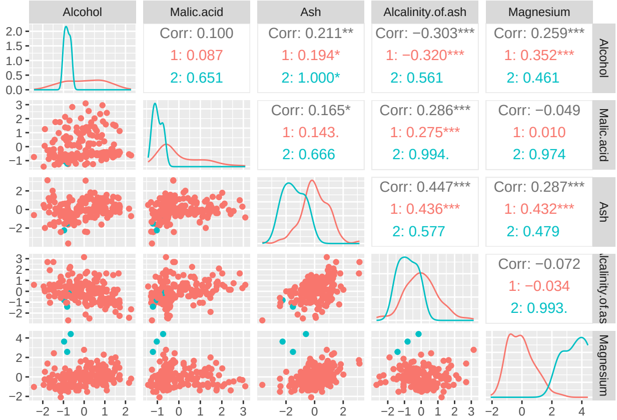
```
GGally::ggpairs(wine.sc[,2:6],
ggplot2::aes(color=as.factor(dbscan6$cluster)),
diag=list(continuous=my_dens))
```



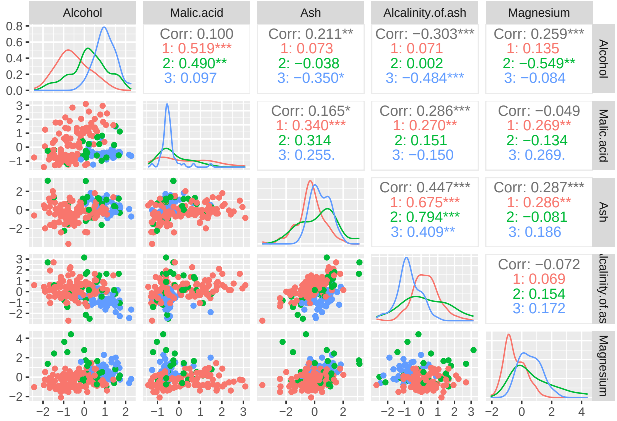
As describe in Chapter 22, as ϵ^* increases, this can be seen dramatically in between instances where $\epsilon^* = 0.75$ increases to $\epsilon^* = 1.25$. The total number of red observations decreases dramatically, this is especially true for when we increase the Minpoints to $Min=15%. However there doesn't appear too be many natural groups in this subset of the data, this may be due to my subset of the features being highly correlated or this approach may not be that effective.

5. Wine datasets using spectral clustering, for various algorithm parameters. Validate your results.

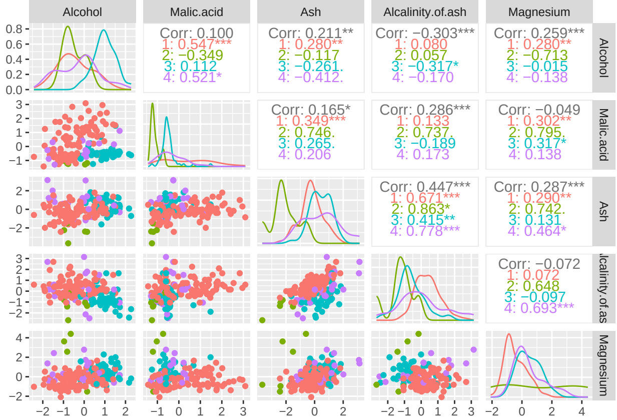
```r
library("ggplot2")
library("GGally")
sc.wine.2 = kernlab::specc(as.matrix(wine.sc[,2:6]), 2)
ggpairs(wine.sc[,2:6],
aes(color=as.factor(sc.wine.2)), diag=list(continuous=my_dens))
```



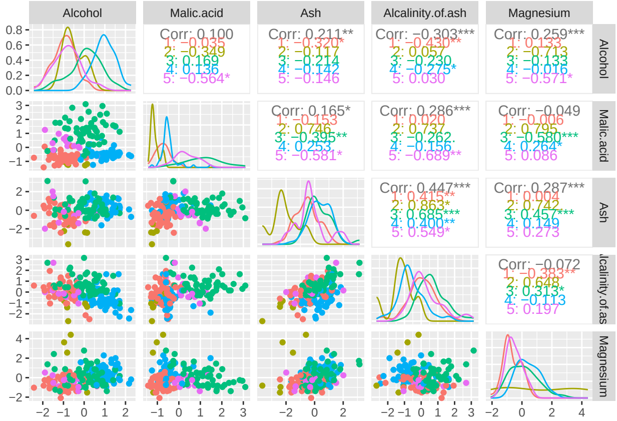
```r
sc.wine.3 = kernlab::specc(as.matrix(wine.sc[,2:6]), 3)
ggpairs(wine.sc[,2:6],
aes(color=as.factor(sc.wine.3)), diag=list(continuous=my_dens))
```



```
sc.wine.4 <- kernlab::specc(as.matrix(wine.sc[,2:6]), 4)
ggpairs(wine.sc[,2:6],
aes(color=as.factor(sc.wine.4)), diag=list(continuous=my_dens))
```



```
sc.wine.5 <- kernlab::specc(as.matrix(wine.sc[,2:6]), 5)
ggpairs(wine.sc[,2:6],
aes(color=as.factor(sc.wine.5)), diag=list(continuous=my_dens))
```



As the number of clusters increases the data becomes more visibly separated but they do not appear to be linearly separable. However, alcohol and Malic.acid seem to have a fair nice separation in each graph with the best being at k=3. It seems that alcohol is the most separable from every other feature in this data subset.