Alex Foss

February 2, 2020

Contents

List of Figures

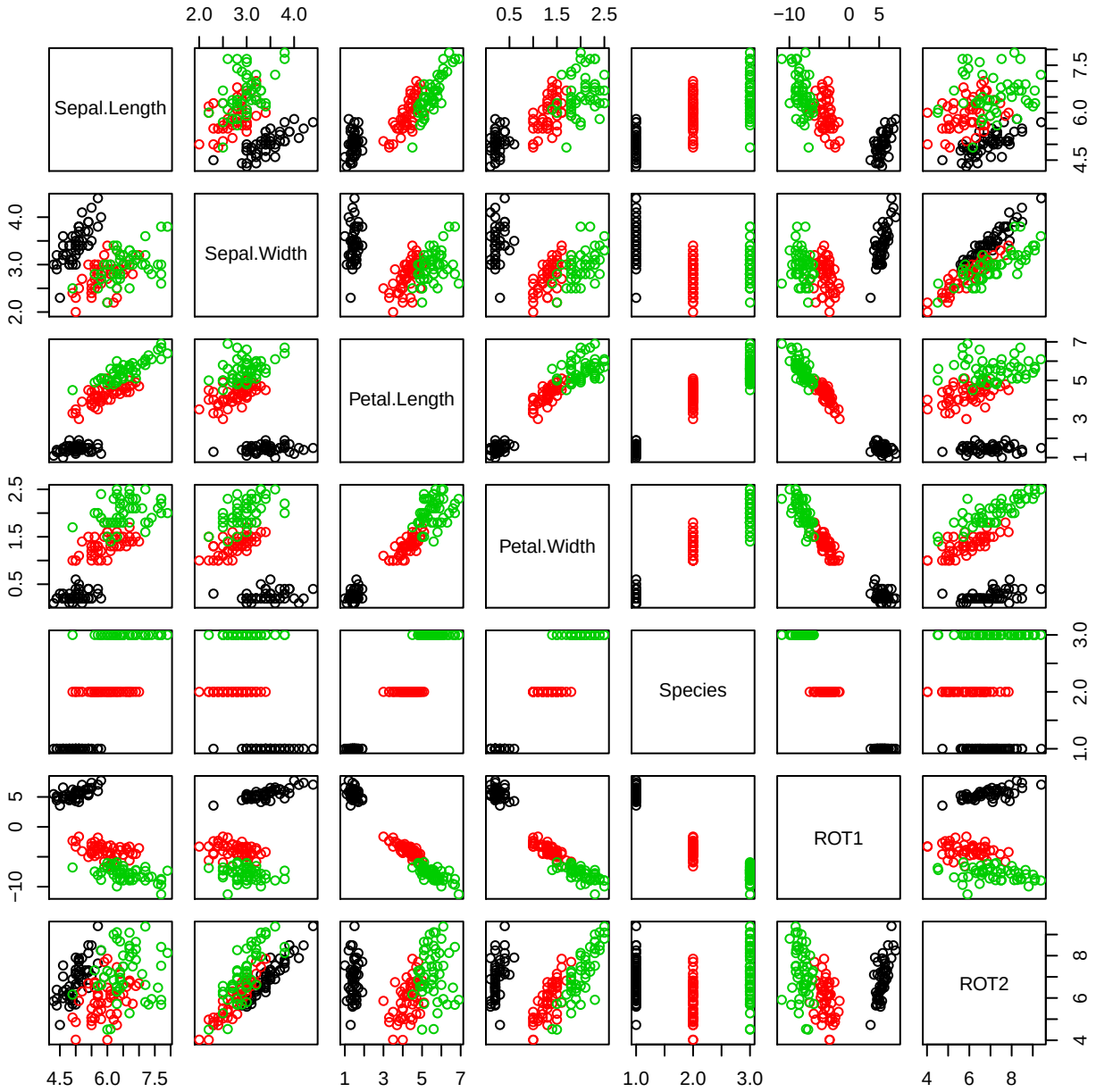
List of Tables

1 Initializations

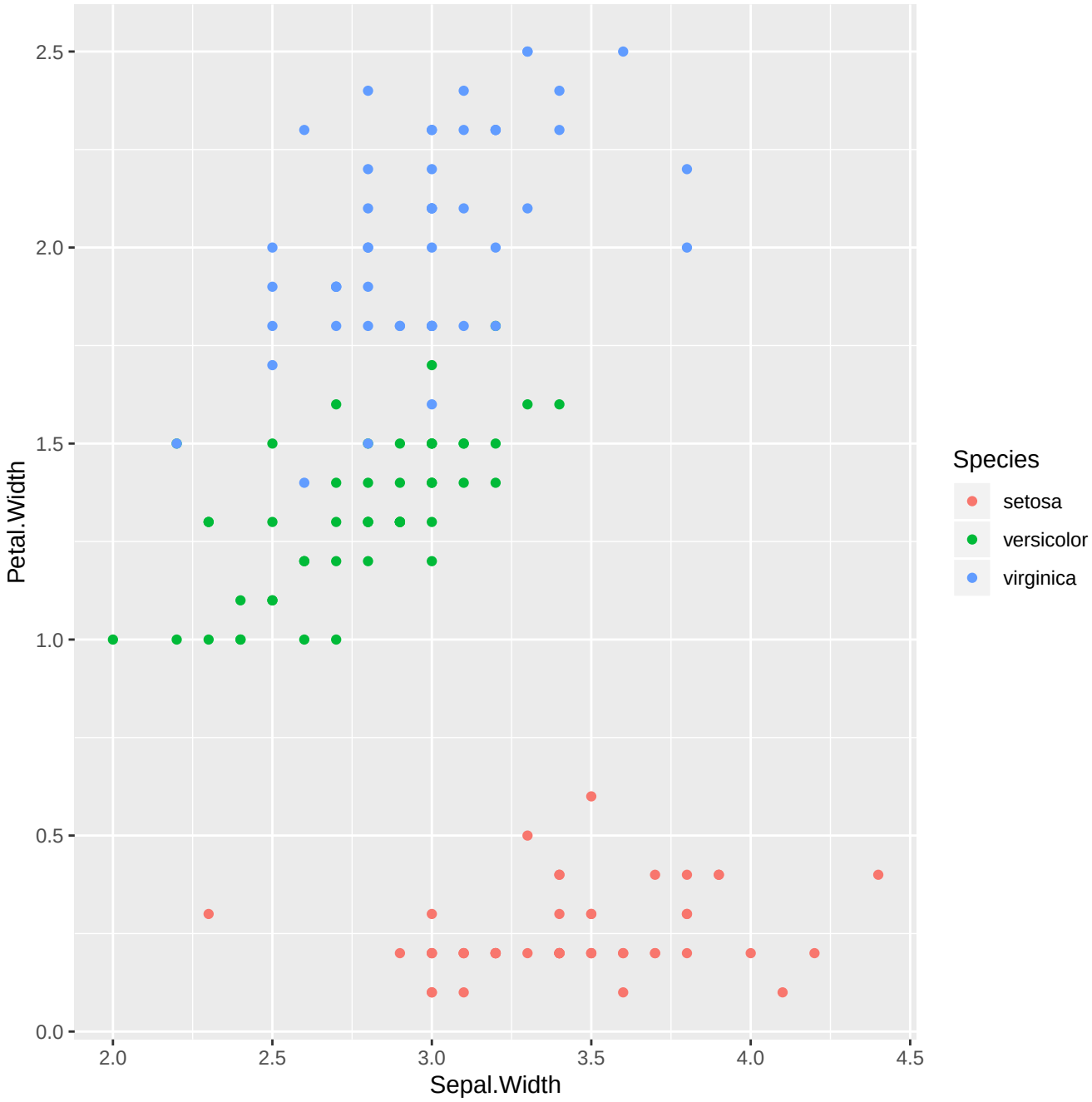
```
#opts_knit$set(concordance=TRUE)
#opts_knit$set(self.contained=FALSE)
#opts_knit$set(tidy=TRUE)
#suppressMessages(library(xtable))
suppressMessages(library(tictoc))
suppressMessages(library(MASS)) # for lda
suppressMessages(library(ggplot2))
suppressMessages(library(dbscan))
sessionInfo()

## R version 3.4.4 (2018-03-15)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 18.04.3 LTS
##
## Matrix products: default
## BLAS: /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.7.1
## LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.7.1
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] methods   stats     graphics  grDevices utils     datasets  base
##
## other attached packages:
## [1] dbscan_1.1-5  ggplot2_3.2.1 MASS_7.3-49   tictoc_1.0
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.3       withr_2.1.2      crayon_1.3.4     grid_3.4.4
##  [5] R6_2.4.1         lifecycle_0.1.0  gtable_0.3.0     magrittr_1.5
##  [9] evaluate_0.14    scales_1.1.0     pillar_1.4.3     rlang_0.4.4
## [13] stringi_1.4.3    lazyeval_0.2.2   tools_3.4.4      stringr_1.4.0
## [17] munsell_0.5.0    xfun_0.11        compiler_3.4.4   pkgconfig_2.0.3
## [21] colorspace_1.4-1 knitr_1.26       tibble_2.1.3
```

2 Iris "Classic"

```
irisOrig <- iris
irisRot <- lda(
  x = iris[,1:4,],
  grouping = iris$Species
)
str(irisRot)

## List of 8
##  $ prior  : Named num [1:3] 0.333 0.333 0.333
##   ..- attr(*, "names")= chr [1:3] "setosa" "versicolor" "virginica"
##  $ counts : Named int [1:3] 50 50 50
##   ..- attr(*, "names")= chr [1:3] "setosa" "versicolor" "virginica"
##  $ means  : num [1:3, 1:4] 5.01 5.94 6.59 3.43 2.77 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:3] "setosa" "versicolor" "virginica"
##   .. ..$ : chr [1:4] "Sepal.Length" "Sepal.Width" "Petal.Length" "Petal.Width"
##  $ scaling: num [1:4, 1:2] 0.8294 1.5345 -2.2012 -2.8105 0.0241 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:4] "Sepal.Length" "Sepal.Width" "Petal.Length" "Petal.Width"
##   .. ..$ : chr [1:2] "LD1" "LD2"
##  $ lev    : chr [1:3] "setosa" "versicolor" "virginica"
##  $ svd    : num [1:2] 48.64 4.58
##  $ N      : int 150
##  $ call   : language lda(x = iris[, 1:4, ], grouping = iris$Species)
##  - attr(*, "class")= chr "lda"

rots <- as.matrix(iris[,1:4]) %*% irisRot$scaling
iris$ROT1 <- rots[,1]
iris$ROT2 <- rots[,2]
pairs(iris, col = iris$Species)
```

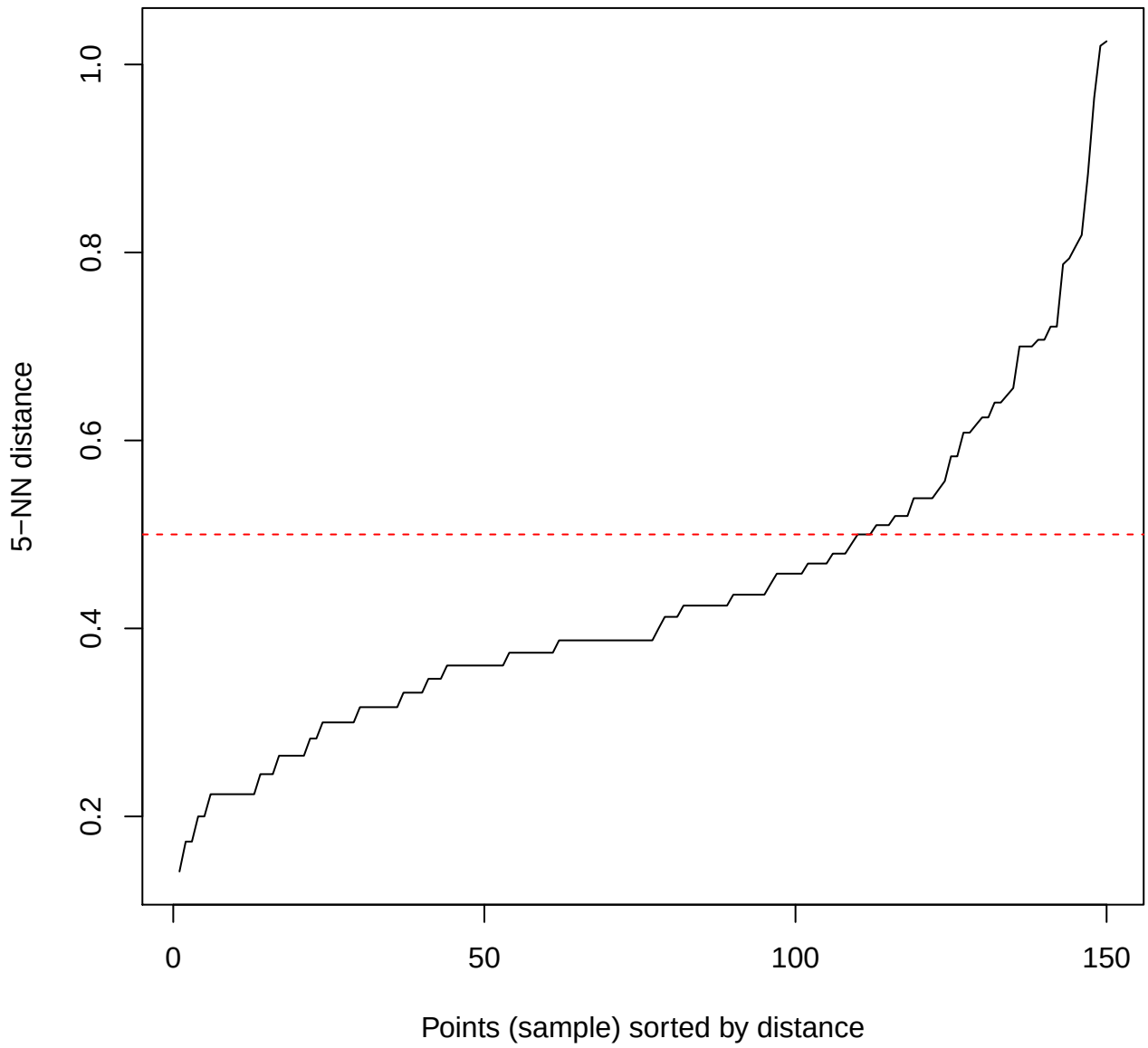



```
(
  ggplot(
    iris,
    aes(x=Sepal.Width, y = Petal.Width, color = Species)
  )
  + geom_point()
)
```

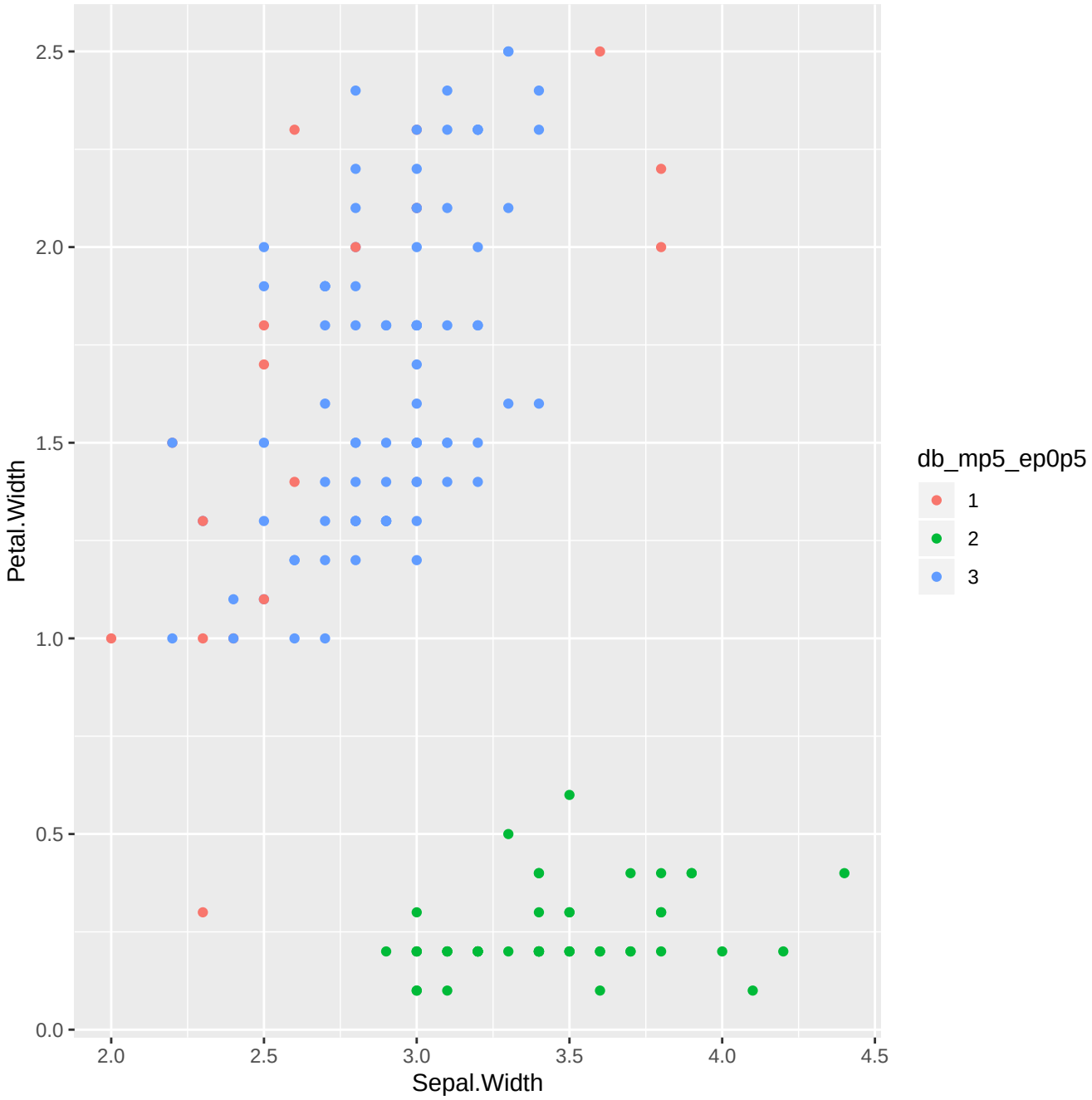

Petal.Width
2.5
2.0
1.5
1.0
0.5
0.0
2.0
2.5
3.0
3.5
4.0
4.5
Sepal.Width
Species
setosa
versicolor
virginica

2.1 DBSCAN

```
# 0.4 -- 0.7 look reasonable
kNNdistplot(iris[,1:4], k = 5)
abline(h=.5, col = "red", lty=2)
```

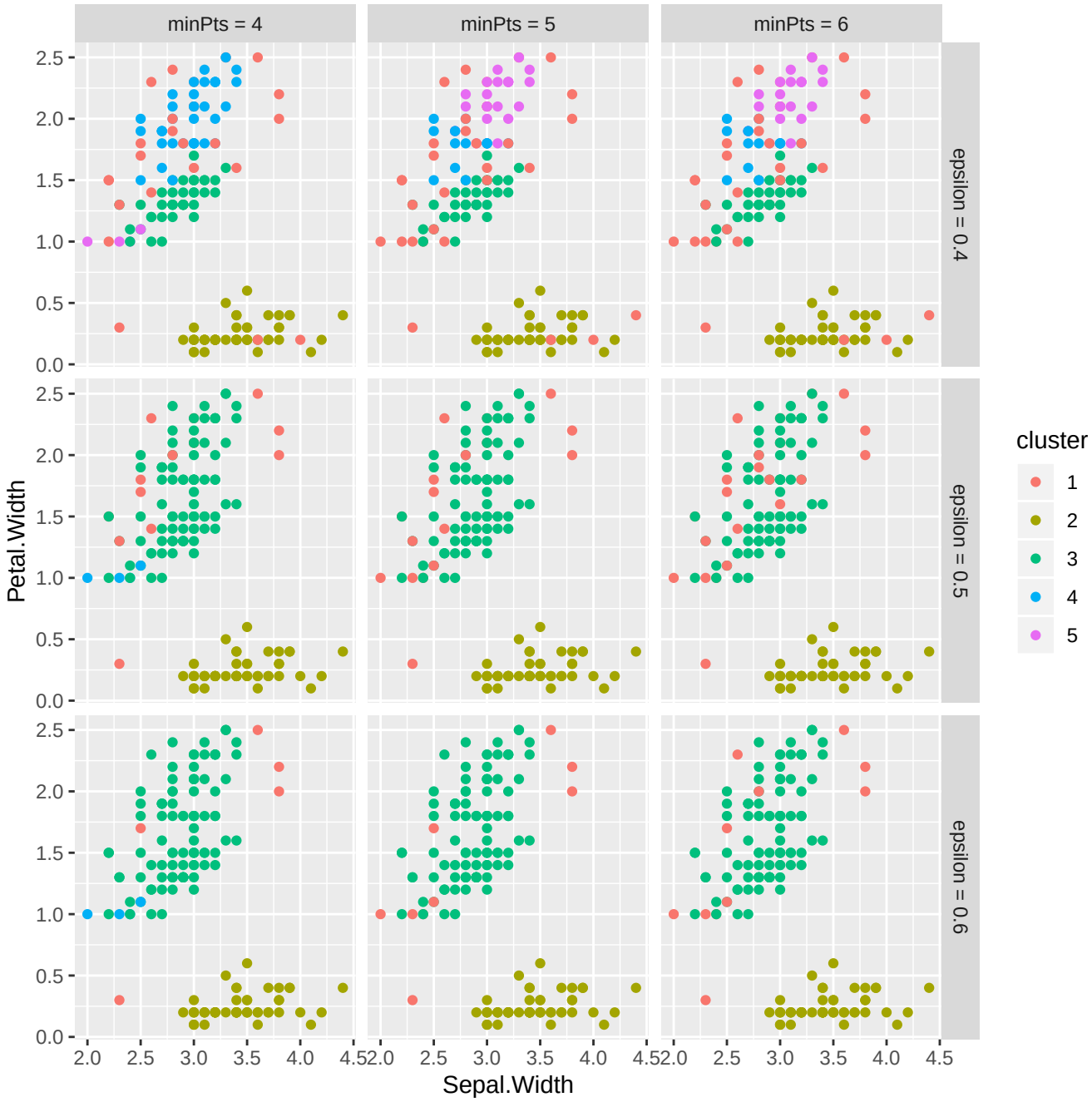



```
res1 <- dbscan(iris[,1:4], eps = .5, minPts = 5)
iris$db_mp5_ep0p5 <- factor(res1$cluster + 1)
(
  ggplot(
    iris,
    aes(x=Sepal.Width, y = Petal.Width, color = db_mp5_ep0p5)
  )
  + geom_point()
)
```


Petal.Width
Sepal.Width
2.5
2.0
1.5
1.0
0.5
0.0
2.0
2.5
3.0
3.5
4.0
4.5
db_mp5_ep0p5
1
2
3

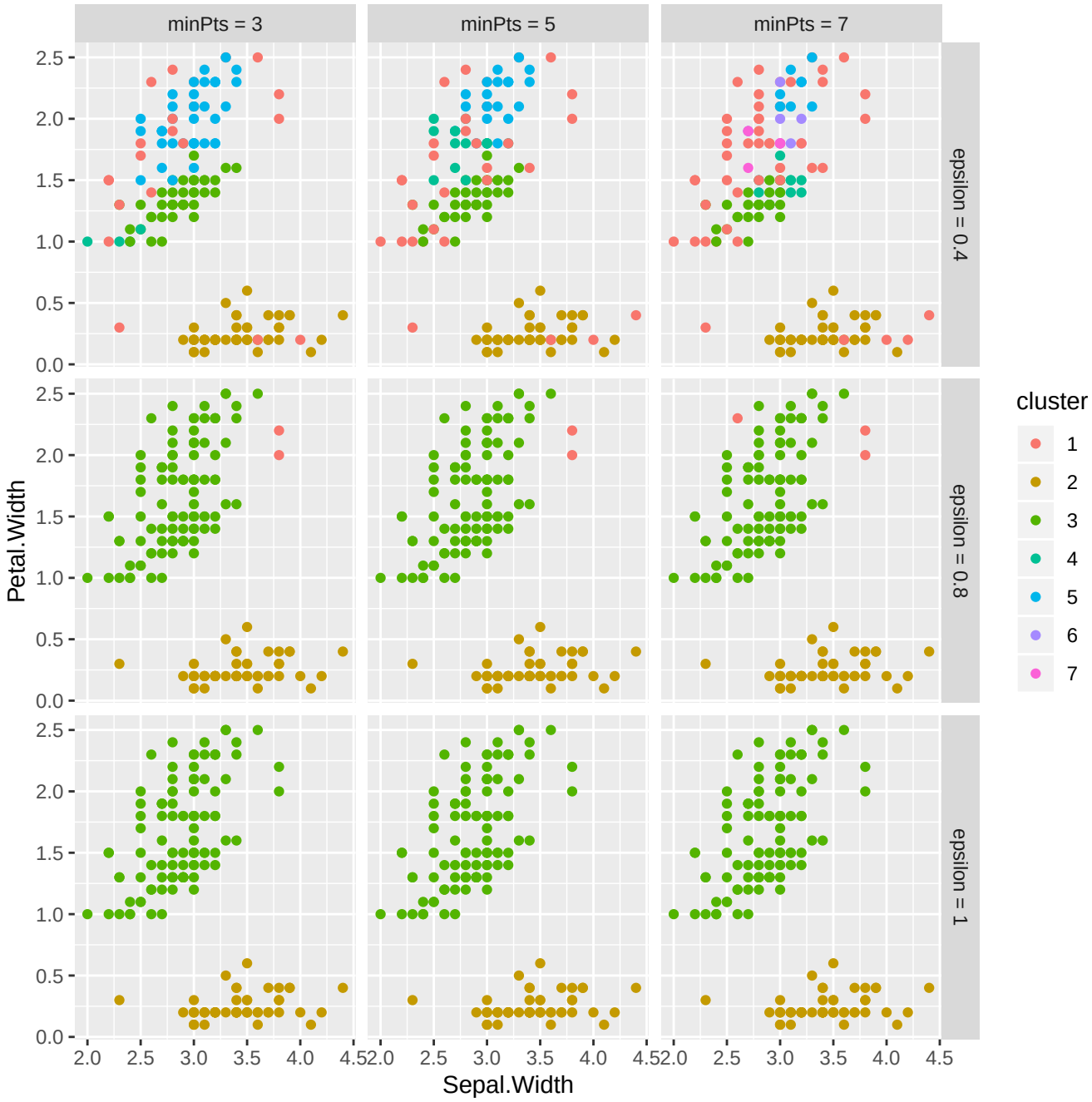
2.1.1 Magic Parameter Knowledge

```
db_iris_parm <- expand.grid(
  eps = seq(0.4, 0.6, 0.1),
  mp  = 4:6
#  mp  = c(3,5,7)
)
tic()
db_iris_list <- lapply(
  1:nrow(db_iris_parm),
  function(i) {
    dres <- dbscan(
      iris[,1:4],
      eps = db_iris_parm$eps[i],
      minPts = db_iris_parm$mp[i]
    )
    return(cbind(
      iris,
      data.frame(
        cluster = factor(dres$cluster + 1),
        eps = paste('epsilon =',db_iris_parm$eps[i]),
        minPts = paste('minPts =',db_iris_parm$mp[i])
      )
    ))
  }
)
toc()

## 0.087 sec elapsed

db_iris_df <- Reduce(f = rbind, x = db_iris_list)

(
  ggplot(
    db_iris_df,
    aes(x=Sepal.Width, y = Petal.Width, color = cluster)
  )
  + geom_point()
  + facet_grid(rows = vars(eps), cols = vars(minPts))
)
```


minPts = 4
minPts = 5
minPts = 6
epsilon = 0.4
epsilon = 0.5
epsilon = 0.6
Petal.Width
Sepal.Width
0.0
0.5
1.0
1.5
2.0
2.5
2.0
2.5
3.0
3.5
4.0
4.5
cluster
1
2
3
4
5

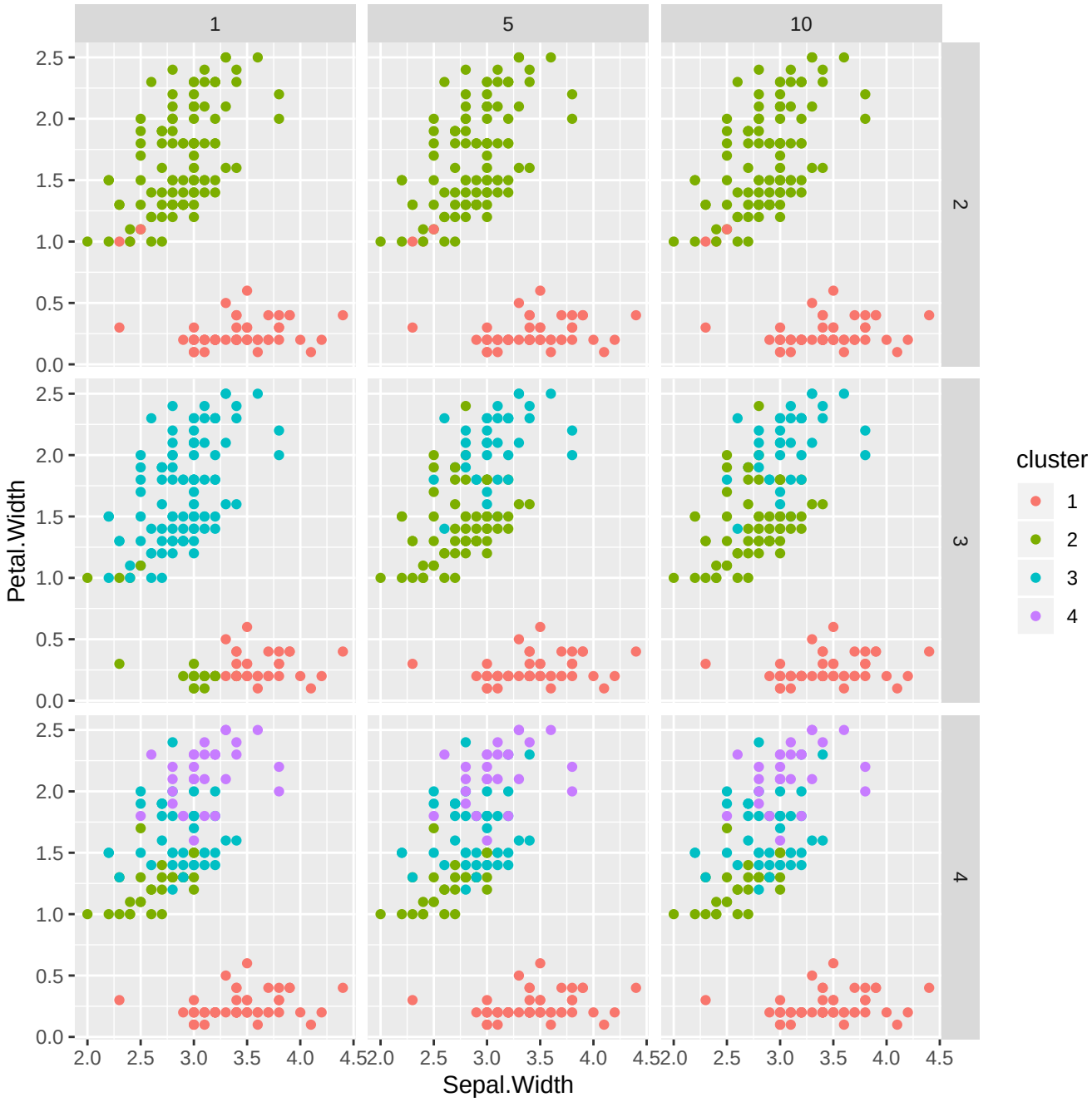
2.1.2 Slightly-Less-Magic Parameter Knowledge

```
db_iris_parm <- expand.grid(
  eps = c(0.4, 0.8, 1.0),
  mp  = c(3,5,7)
)
tic()
db_iris_list <- lapply(
  1:nrow(db_iris_parm),
  function(i) {
    dres <- dbscan(
      iris[,1:4],
      eps = db_iris_parm$eps[i],
      minPts = db_iris_parm$mp[i]
    )
    return(cbind(
      iris,
      data.frame(
        cluster = factor(dres$cluster + 1),
        eps = paste('epsilon =',db_iris_parm$eps[i]),
        minPts = paste('minPts =',db_iris_parm$mp[i])
      )
    ))
  }
)
toc()

## 0.041 sec elapsed

db_iris_df <- Reduce(f = rbind, x = db_iris_list)

(
  ggplot(
    db_iris_df,
    aes(x=Sepal.Width, y = Petal.Width, color = cluster)
  )
  + geom_point()
  + facet_grid(rows = vars(eps), cols = vars(minPts))
)
```


minPts = 3
minPts = 5
minPts = 7
epsilon = 0.4
epsilon = 0.8
epsilon = 1
Petal.Width
Sepal.Width
cluster
1
2
3
4
5
6
7

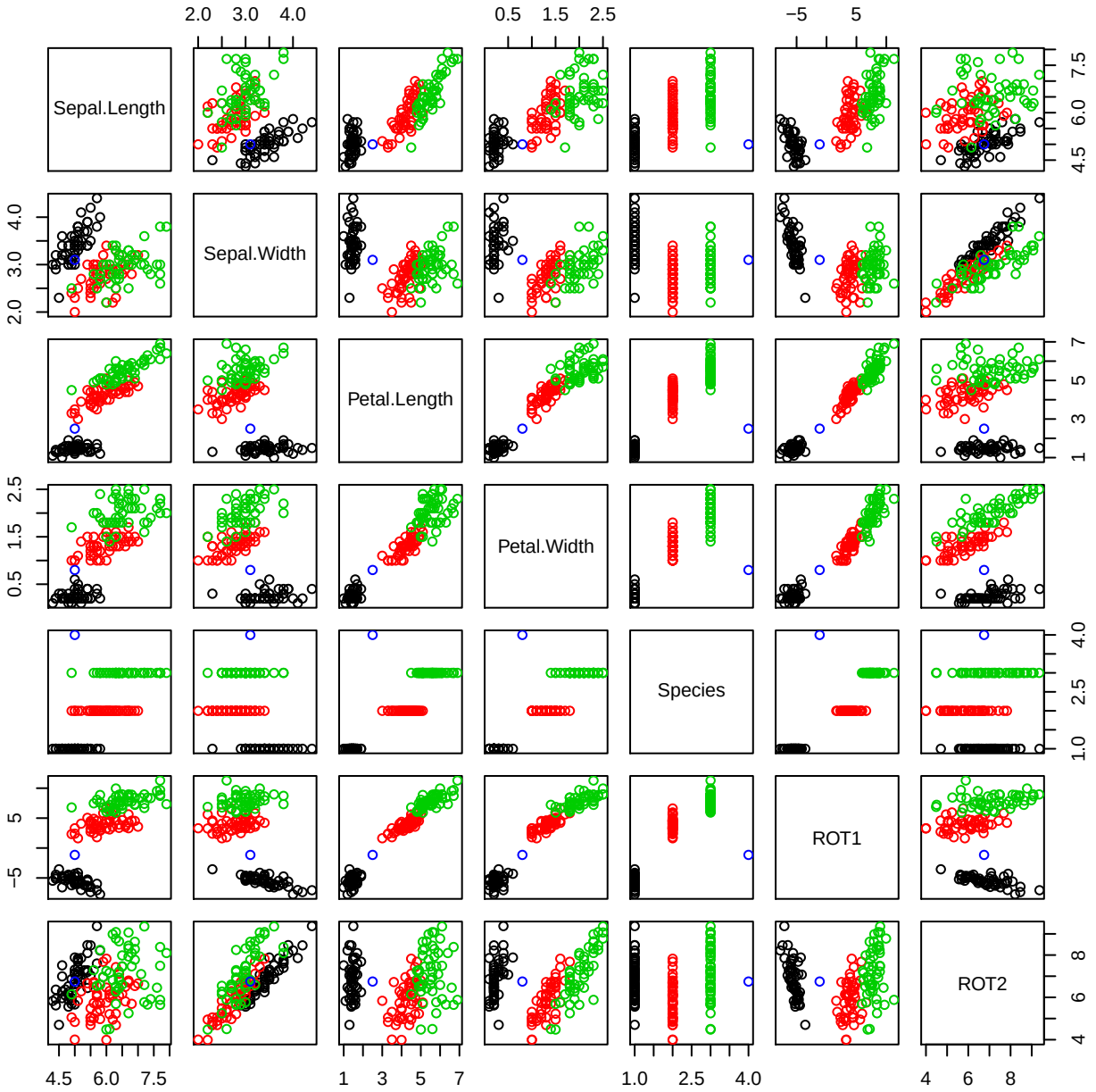
2.2 K-Means: Cake or Death?

```
km_iris_parm <- expand.grid(
  centers = c(2:4),
  nstart = c(1, 5, 10),
  iter.max = c(10, 100, 1000)
)
tic()
km_iris_list <- lapply(
  1:nrow(km_iris_parm),
  function(i) {
    dres <- kmeans(
      iris[,1:4],
      centers = km_iris_parm$centers[i],
      nstart = km_iris_parm$nstart[i],
      iter.max = km_iris_parm$iter.max[i]
    )
    # reorder clusters
    dres$cluster <- order(order(dres$centers[,4]))[dres$cluster]
    return(cbind(
      iris,
      data.frame(
        cluster = factor(dres$cluster + 0),
        centers = km_iris_parm$centers[i],
        nstart = km_iris_parm$nstart[i],
        iter.max = km_iris_parm$iter.max[i]
      )
    ))
  }
)
toc()

## 0.111 sec elapsed

km_iris_df <- Reduce(f = rbind, x = km_iris_list)

(
  ggplot(
    km_iris_df[km_iris_df$iter.max == 1000,],
    aes(x=Sepal.Width, y = Petal.Width, color = cluster)
  )
  + geom_point()
  + facet_grid(rows = vars(centers), cols = vars(nstart))
)
```

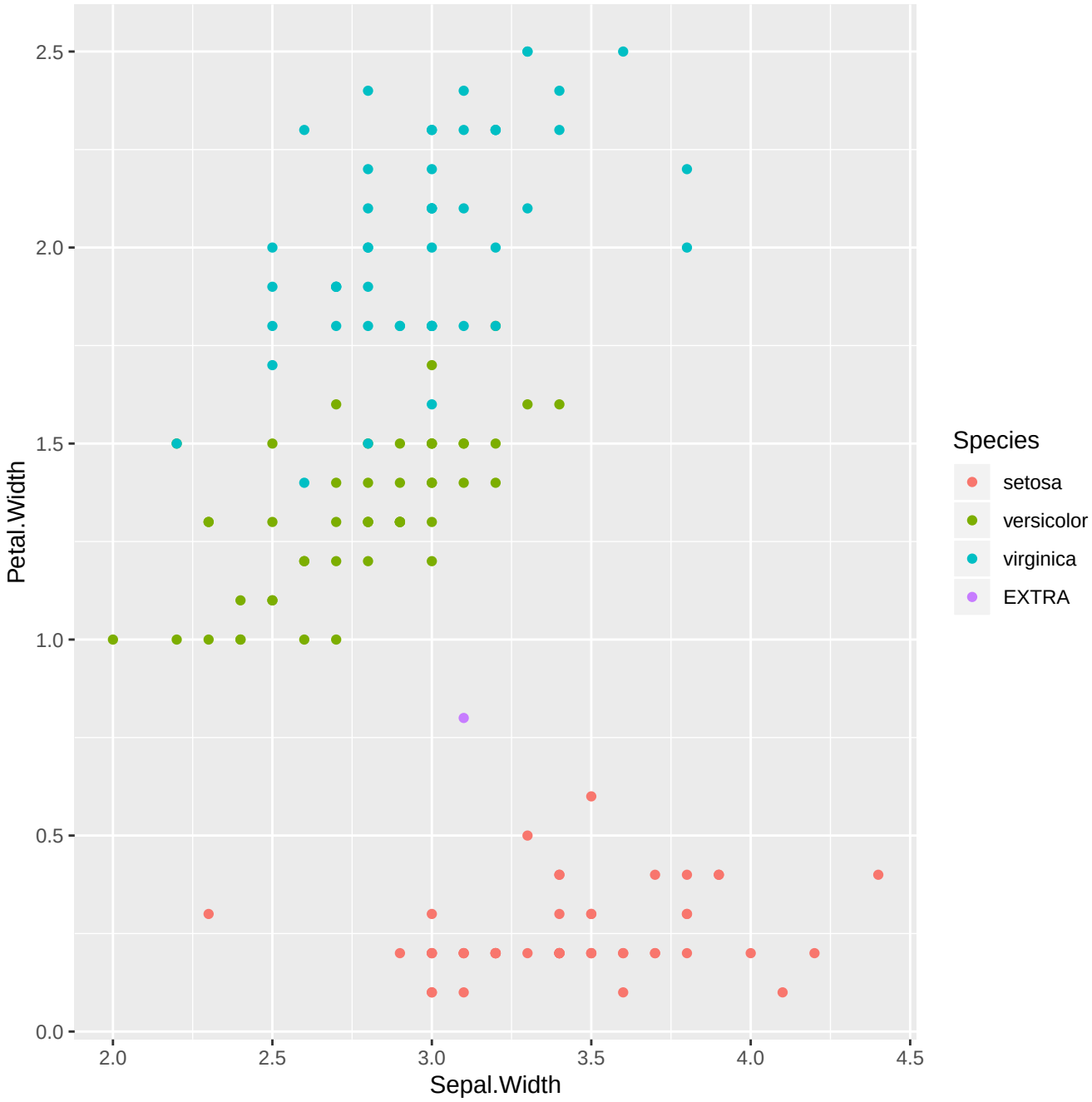

1
5
10
2
3
4
Petal.Width
Sepal.Width
0.0
0.5
1.0
1.5
2.0
2.5
2.0
2.5
3.0
3.5
4.0
4.5
cluster
1
2
3
4

3 Iris Plus One: Hard Mode?

```
extraPt <- data.frame(
  Sepal.Length = 5.0,
  Sepal.Width = 3.1,
  Petal.Length = 2.5,
  Petal.Width = 0.8,
  Species = 'EXTRA'
)
iris <- rbind(irisOrig, extraPt)
#irisExtra <- rbind(iris, extraPt)
irisRot <- lda(
  x = iris[,1:4,],
  grouping = iris$Species
)
str(irisRot)

## List of 8
##  $ prior  : Named num [1:4] 0.33113 0.33113 0.33113 0.00662
##   ..- attr(*, "names")= chr [1:4] "setosa" "versicolor" "virginica" "EXTRA"
##  $ counts : Named int [1:4] 50 50 50 1
##   ..- attr(*, "names")= chr [1:4] "setosa" "versicolor" "virginica" "EXTRA"
##  $ means  : num [1:4, 1:4] 5.01 5.94 6.59 5 3.43 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:4] "setosa" "versicolor" "virginica" "EXTRA"
##   .. ..$ : chr [1:4] "Sepal.Length" "Sepal.Width" "Petal.Length" "Petal.Width"
##  $ scaling: num [1:4, 1:3] -0.8279 -1.5347 2.2013 2.8088 0.0183 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:4] "Sepal.Length" "Sepal.Width" "Petal.Length" "Petal.Width"
##   .. ..$ : chr [1:3] "LD1" "LD2" "LD3"
##  $ lev    : chr [1:4] "setosa" "versicolor" "virginica" "EXTRA"
##  $ svd    : num [1:3] 39.761 3.74 0.592
##  $ N      : int 151
##  $ call   : language lda(x = iris[, 1:4, ], grouping = iris$Species)
##  - attr(*, "class")= chr "lda"

rots <- as.matrix(iris[,1:4]) %*% irisRot$scaling
iris$ROT1 <- rots[,1]
iris$ROT2 <- rots[,2]
pairs(iris, col = iris$Species)
```

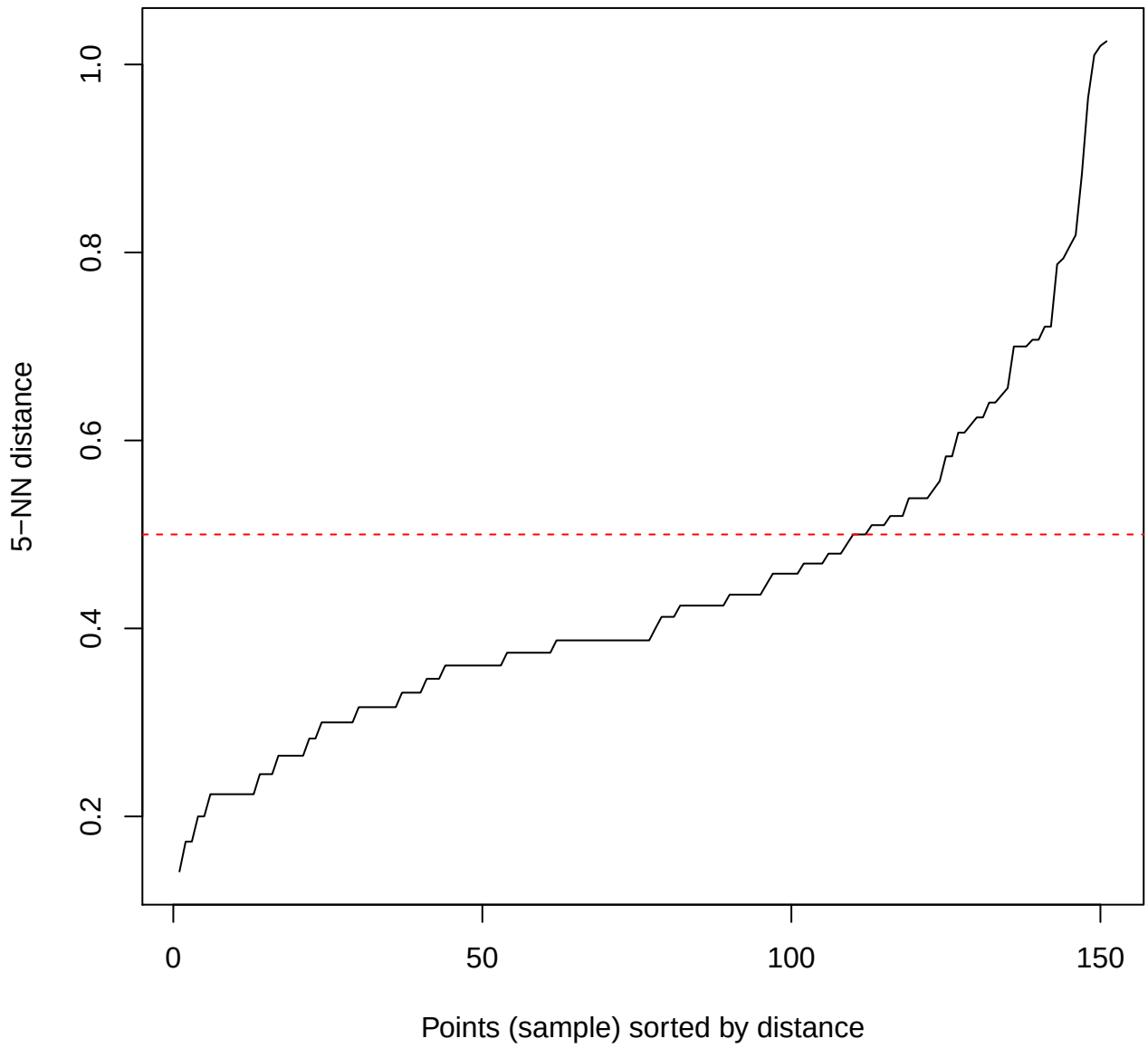



```
(
  ggplot(
    iris,
    aes(x=Sepal.Width, y = Petal.Width, color = Species)
  )
  + geom_point()
)
```

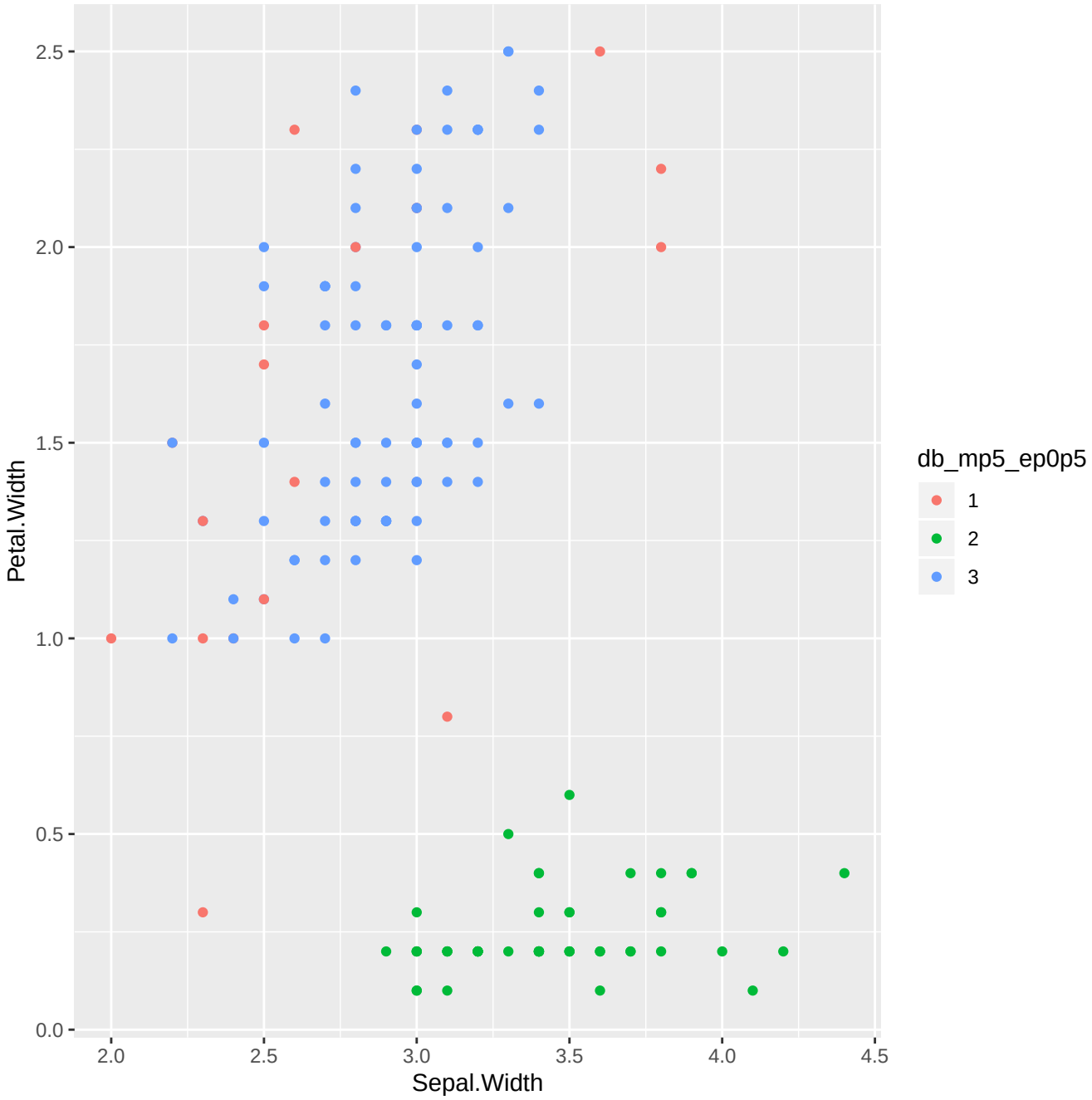

Petal.Width
2.5
2.0
1.5
1.0
0.5
0.0
2.0
2.5
3.0
3.5
4.0
4.5
Sepal.Width
Species
setosa
versicolor
virginica
EXTRA

3.1 DBSCAN

```
# 0.4 -- 0.7 look reasonable
kNNdistplot(iris[,1:4], k = 5)
abline(h=.5, col = "red", lty=2)
```

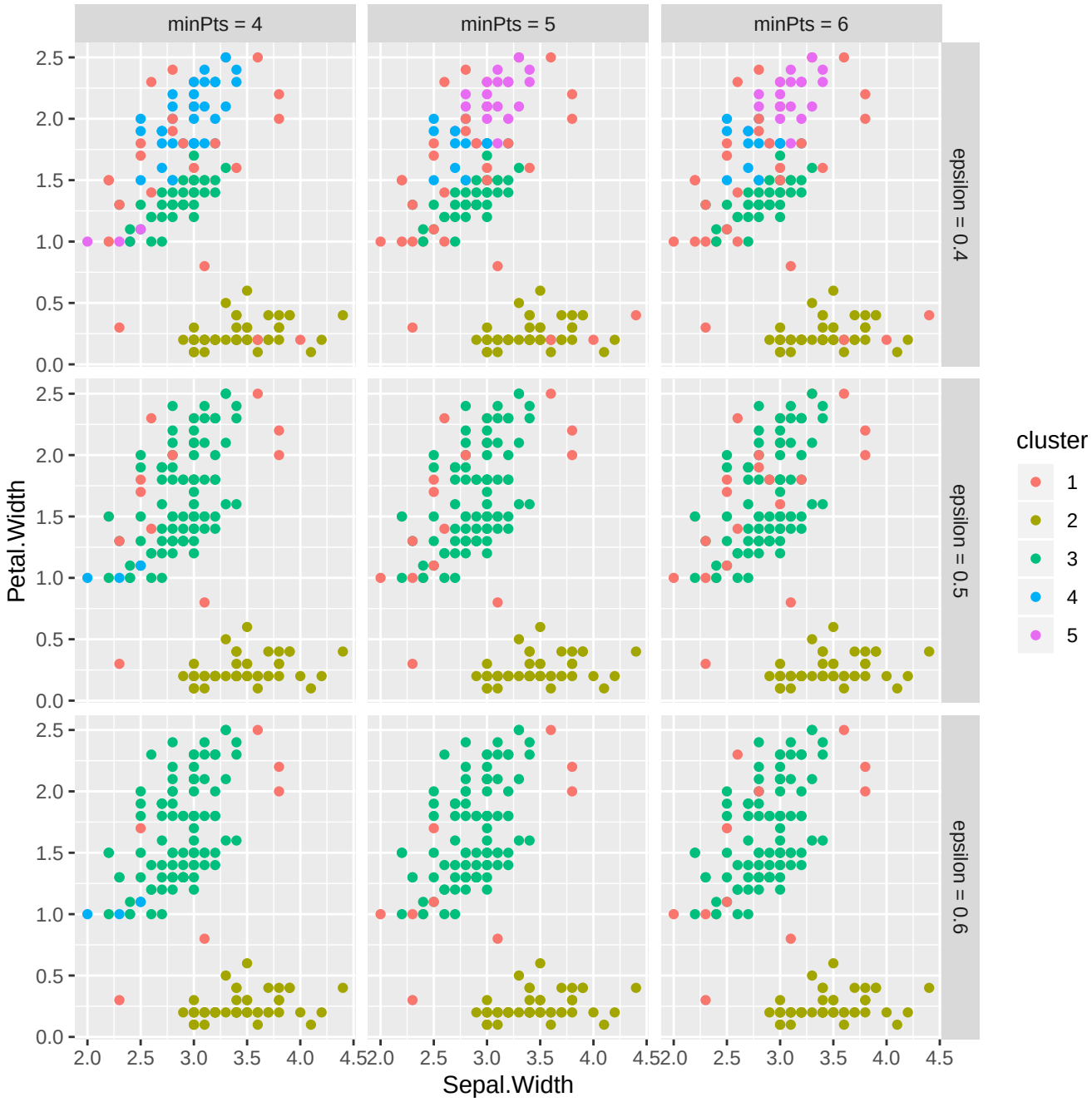



```
res1 <- dbscan(iris[,1:4], eps = .5, minPts = 5)
iris$db_mp5_ep0p5 <- factor(res1$cluster + 1)
(
  ggplot(
    iris,
    aes(x=Sepal.Width, y = Petal.Width, color = db_mp5_ep0p5)
  )
  + geom_point()
)
```


Petal.Width
Sepal.Width
2.5
2.0
1.5
1.0
0.5
0.0
2.0
2.5
3.0
3.5
4.0
4.5
db_mp5_ep0p5
1
2
3

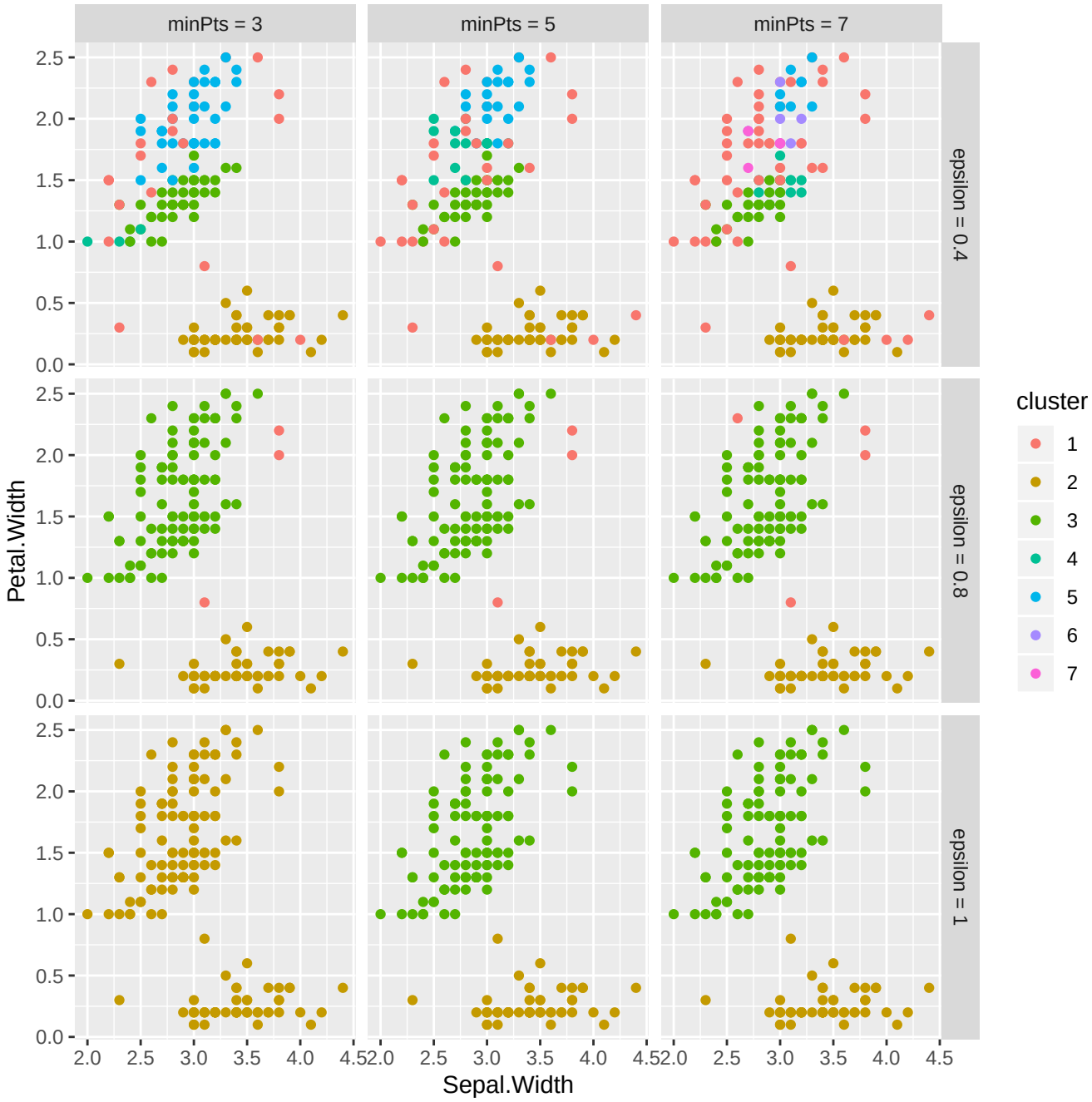
3.1.1 Magic Parameter Knowledge

```
db_iris_parm <- expand.grid(
  eps = seq(0.4, 0.6, 0.1),
  mp  = 4:6
#  mp  = c(3,5,7)
)
tic()
db_iris_list <- lapply(
  1:nrow(db_iris_parm),
  function(i) {
    dres <- dbscan(
      iris[,1:4],
      eps = db_iris_parm$eps[i],
      minPts = db_iris_parm$mp[i]
    )
    return(cbind(
      iris,
      data.frame(
        cluster = factor(dres$cluster + 1),
        eps = paste('epsilon =',db_iris_parm$eps[i]),
        minPts = paste('minPts =',db_iris_parm$mp[i])
      )
    ))
  }
)
toc()

## 0.028 sec elapsed

db_iris_df <- Reduce(f = rbind, x = db_iris_list)

(
  ggplot(
    db_iris_df,
    aes(x=Sepal.Width, y = Petal.Width, color = cluster)
  )
  + geom_point()
  + facet_grid(rows = vars(eps), cols = vars(minPts))
)
```


minPts = 4
minPts = 5
minPts = 6
epsilon = 0.4
epsilon = 0.5
epsilon = 0.6
Petal.Width
Sepal.Width
cluster
1
2
3
4
5

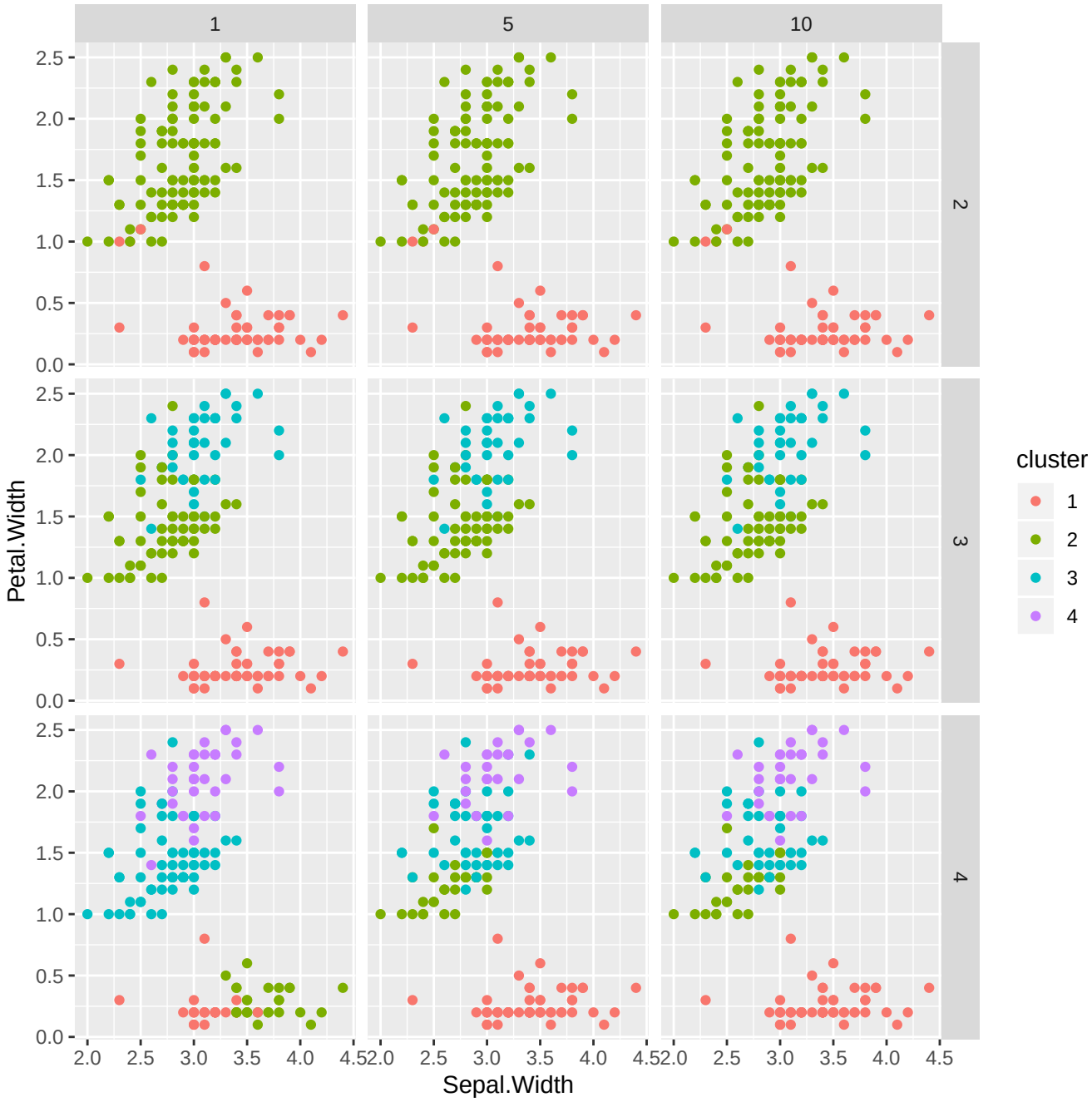
3.1.2 Slightly-Less-Magic Parameter Knowledge

```
db_iris_parm <- expand.grid(
  eps = c(0.4, 0.8, 1.0),
  mp  = c(3,5,7)
)
tic()
db_iris_list <- lapply(
  1:nrow(db_iris_parm),
  function(i) {
    dres <- dbscan(
      iris[,1:4],
      eps = db_iris_parm$eps[i],
      minPts = db_iris_parm$mp[i]
    )
    return(cbind(
      iris,
      data.frame(
        cluster = factor(dres$cluster + 1),
        eps = paste('epsilon =',db_iris_parm$eps[i]),
        minPts = paste('minPts =',db_iris_parm$mp[i])
      )
    ))
  }
)
toc()

## 0.032 sec elapsed

db_iris_df <- Reduce(f = rbind, x = db_iris_list)

(
  ggplot(
    db_iris_df,
    aes(x=Sepal.Width, y = Petal.Width, color = cluster)
  )
  + geom_point()
  + facet_grid(rows = vars(eps), cols = vars(minPts))
)
```


minPts = 3
minPts = 5
minPts = 7
epsilon = 0.4
epsilon = 0.8
epsilon = 1
Petal.Width
Sepal.Width
cluster
1
2
3
4
5
6
7

3.2 K-Means: Cake or Death?

```
km_iris_parm <- expand.grid(
  centers = c(2:4),
  nstart = c(1, 5, 10),
  iter.max = c(10, 100, 1000)
)
tic()
km_iris_list <- lapply(
  1:nrow(km_iris_parm),
  function(i) {
    dres <- kmeans(
      iris[,1:4],
      centers = km_iris_parm$centers[i],
      nstart = km_iris_parm$nstart[i],
      iter.max = km_iris_parm$iter.max[i]
    )
    # reorder clusters
    dres$cluster <- order(order(dres$centers[,4]))[dres$cluster]
    return(cbind(
      iris,
      data.frame(
        cluster = factor(dres$cluster + 0),
        centers = km_iris_parm$centers[i],
        nstart = km_iris_parm$nstart[i],
        iter.max = km_iris_parm$iter.max[i]
      )
    ))
  }
)
toc()

## 0.11 sec elapsed

km_iris_df <- Reduce(f = rbind, x = km_iris_list)

(
  ggplot(
    km_iris_df[km_iris_df$iter.max == 1000,],
    aes(x=Sepal.Width, y = Petal.Width, color = cluster)
  )
  + geom_point()
  + facet_grid(rows = vars(centers), cols = vars(nstart))
)
```


1
5
10
2
3
4
Petal.Width
Sepal.Width
0.0
0.5
1.0
1.5
2.0
2.5
2.0
2.5
3.0
3.5
4.0
4.5
cluster
1
2
3
4