k-means Clustering of the Iris Dataset

Jon Kinsey

Mon Dec 1 13:04:39 2014

```
#
# k-means is an unsupervised learning technique that deals with finding a
# structure (cluster) in a collection of unlabeled data. Clustering of the data
# involves organizing objects into k-groups whose members are similar in some way.
# Here, the similarity criterion is distance: two or more objects belong to the
# same cluster if they are "close" according to a given distance. The k-means
# approach utilizes an exclusive clustering algorithm.
#
# The algorithm of Hartigan and Wong (1979) is used by default. Note that some
# authors use k-means to refer to a specific algorithm rather than the general
# method: most commonly the algorithm given by MacQueen (1967) but sometimes that
# given by Lloyd (1957) and Forgy (1965). The Hartigan-Wong algorithm generally
# does a better job than either of those, but trying several random
# starts (nstart> 1) is often recommended. In rare cases, when some of the
# points (rows of x) are extremely close, the algorithm may not converge in the
# "Quick-Transfer" stage, signalling a warning (and returning ifault = 4).
# Slight rounding of the data may be advisable in that case.
#
# Hartigan, J. A. and Wong, M. A. (1979). A K-means clustering algorithm.
# Applied Statistics 28, 100-108.
#
# Prior to clustering data, you may want to remove or estimate missing data and
# rescale variables for comparability.
# mydata <- na.omit(mydata) # listwise deletion of missing
# mydata <- scale(mydata) # standardize variables
#
# some EDA first
dim(iris)
```

```
## [1] 150   5
```

```
names(iris) # variable names
```

```
## [1] "Sepal.Length" "Sepal.Width"  "Petal.Length" "Petal.Width"
## [5] "Species"
```

```
str(iris) # structure
```

```
## 'data.frame':    150 obs. of  5 variables:
##  $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
##  $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
##  $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
##  $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
##  $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

```r
attributes(iris) # attributes
```

```
## $names
## [1] "Sepal.Length" "Sepal.Width"  "Petal.Length" "Petal.Width" 
## [5] "Species"     
## 
## $row.names
##   [1]   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17
##  [18]  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34
##  [35]  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51
##  [52]  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68
##  [69]  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85
##  [86]  86  87  88  89  90  91  92  93  94  95  96  97  98  99 100 101 102
## [103] 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119
## [120] 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136
## [137] 137 138 139 140 141 142 143 144 145 146 147 148 149 150
## 
## $class
## [1] "data.frame"
```

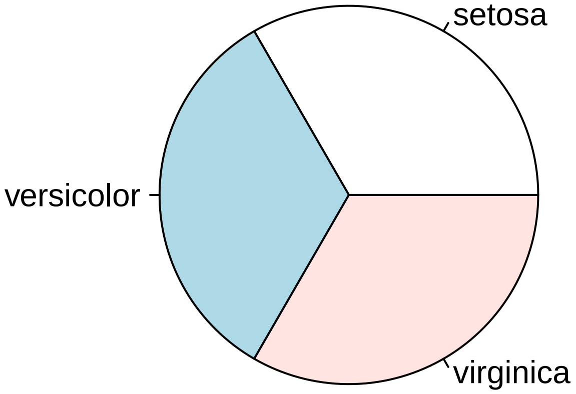
```r
summary(iris)
```

```
##   Sepal.Length    Sepal.Width     Petal.Length    Petal.Width   
##  Min.   :4.30   Min.   :2.00   Min.   :1.00   Min.   :0.1  
##  1st Qu.:5.10   1st Qu.:2.80   1st Qu.:1.60   1st Qu.:0.3  
##  Median :5.80   Median :3.00   Median :4.35   Median :1.3  
##  Mean   :5.84   Mean   :3.06   Mean   :3.76   Mean   :1.2  
##  3rd Qu.:6.40   3rd Qu.:3.30   3rd Qu.:5.10   3rd Qu.:1.8  
##  Max.   :7.90   Max.   :4.40   Max.   :6.90   Max.   :2.5  
##        Species  
##  setosa    :50  
##  versicolor:50  
##  virginica :50  
##                 
##                 
## 
```

```r
#
table(iris$Species)  # frequency
```

```
## 
##     setosa versicolor  virginica 
##         50         50         50
```

```r
pie(table(iris$Species))  # pie chart
```



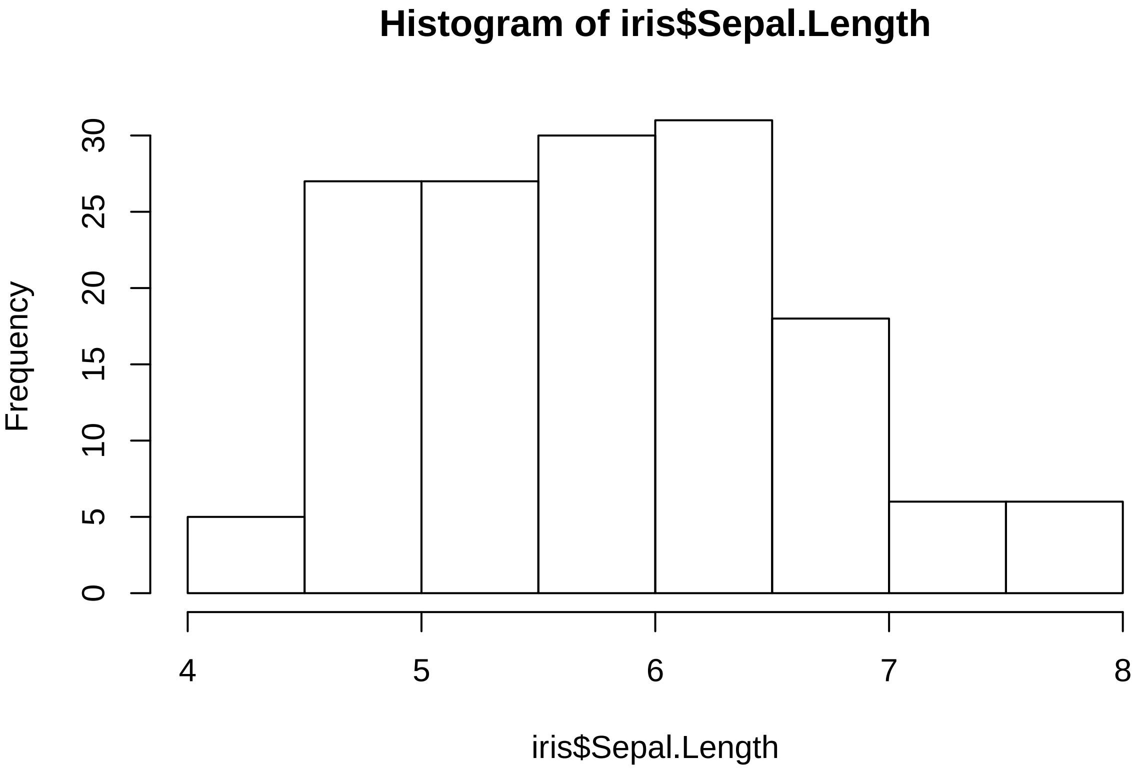
```
#
cor(iris$Sepal.Length, iris$Petal.Length) # correlation between sepal, petal
```

```
## [1] 0.8718
```

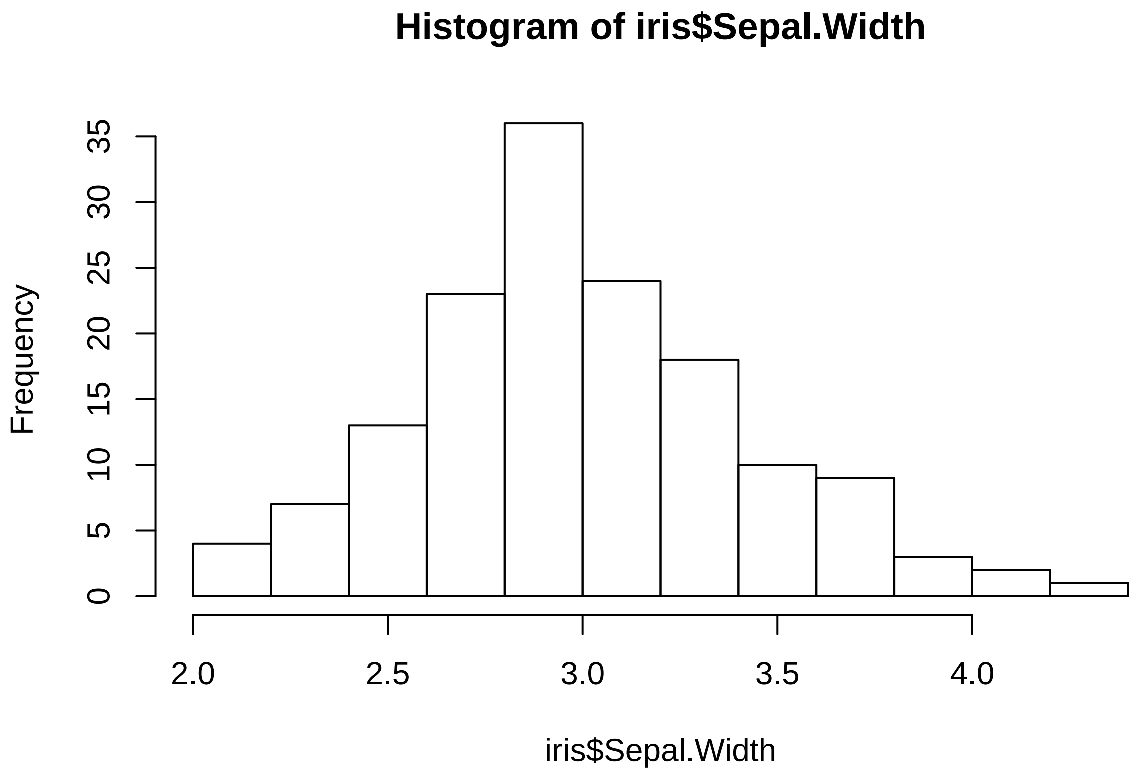
```
cor(iris$Sepal.Length, iris$Sepal.Width)
```

```
## [1] -0.1176
```

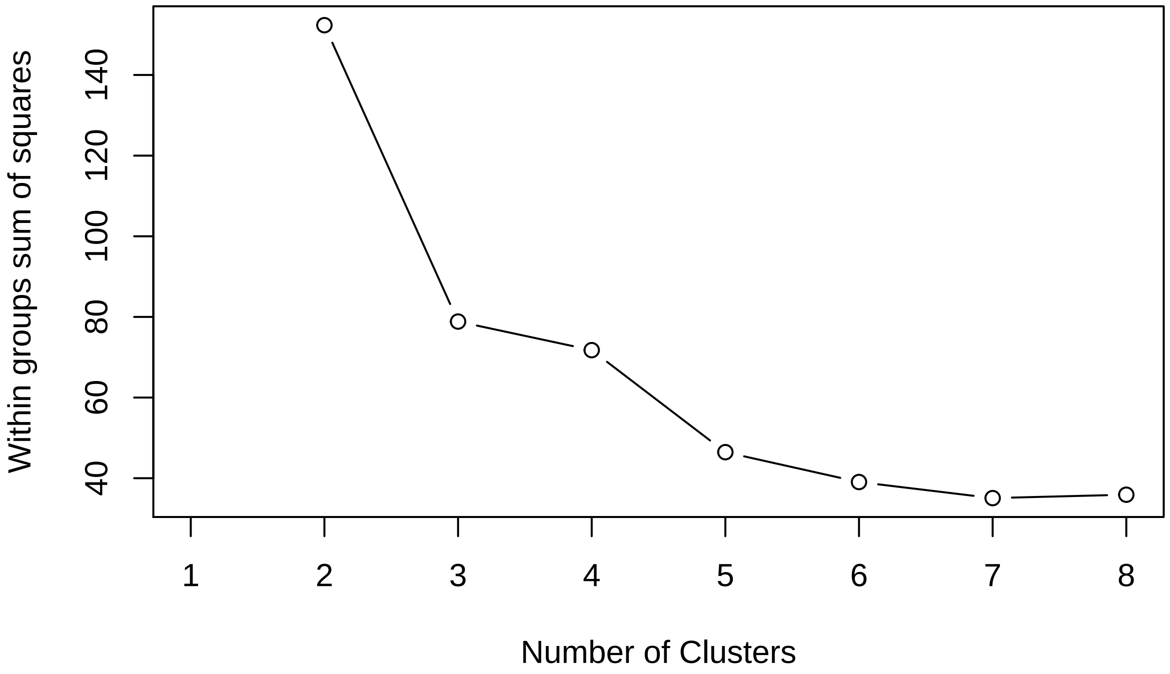
```
hist(iris$Sepal.Length)
```



```
hist(iris$Sepal.Width)
```



```
#
newiris <- iris
newiris$Species <- NULL # remove Species
#
# Below, we iterate through kmeans() with clusters argument varying from
# 1 to maxCluster and plot the within groups sum of squares for each iteration.
#
ssPlot <- function(data, maxCluster = 8) {
  # Initialize within sum of squares
  SSw <- (nrow(data) - 1) * sum(apply(data, 2, var))
  SSw <- vector()
  for (i in 2:maxCluster) {
    SSw[i] <- sum(kmeans(data, centers = i)$withinss)
  }
  plot(1:maxCluster, SSw, type = "b", xlab = "Number of Clusters", ylab = "Within groups sum of squares
}
ssPlot(newiris)
```



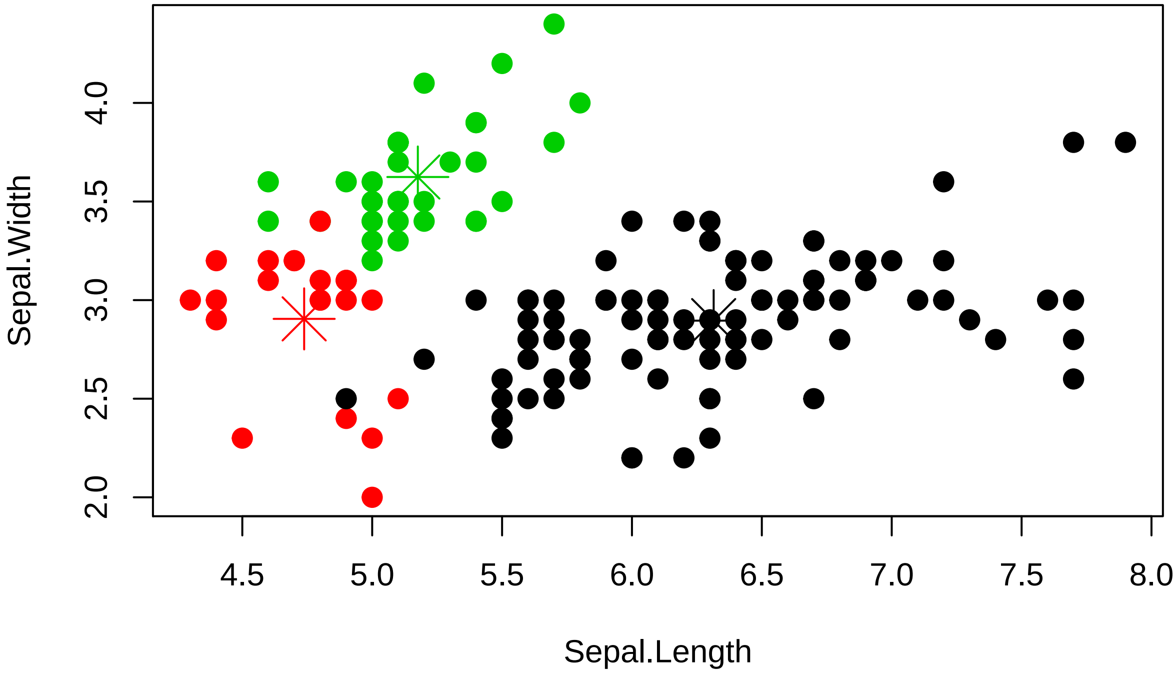
```
# largest decrease going from 2 to 3 clusters. So, lets just go with 3.
# K-means clustering with 3 clusters of sizes 38, 50, 62
(kc <- kmeans(newiris, 3))
```

```
## K-means clustering with 3 clusters of sizes 96, 21, 33
##
## Cluster means:
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1        6.315       2.896        4.974      1.7031
## 2        4.738       2.905        1.790      0.3524
## 3        5.176       3.624        1.473      0.2727
##
## Clustering vector:
##   [1] 3 2 2 2 3 3 3 3 2 2 3 3 2 2 3 3 3 3 3 3 3 3 3 3 2 2 3 3 3 2 2 3 3 3 2
##  [36] 3 3 3 2 3 3 2 2 3 3 2 3 2 3 3 1 1 1 1 1 1 1 2 1 1 2 1 1 1 1 1 1 1 1 1
##  [71] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 2 1 1 1 1 1 1
## [106] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## [141] 1 1 1 1 1 1 1 1 1 1
##
## Within cluster sum of squares by cluster:
## [1] 118.652  17.670   6.432
##  (between_SS / total_SS =  79.0 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

```
#
# Compare the Species label with the clustering result
table(iris$Species, kc$cluster)
```

```
##
##               1  2  3
##   setosa      0 17 33
##   versicolor 46  4  0
##   virginica  50  0  0
```

```
# This result shows that cluster "setosa" can be easily separated from the other
# clusters, and that clusters "versicolor" and "virginica" are to a small degree
# overlapped with each other.
#
# Plot the clusters and their centres. Note that there are four dimensions in the data
# and that only the first two dimensions are used to draw the plot below.
# Some black points close to the green centre (asterisk) are actually closer
# to the black centre in the four dimensional space.
par(mfrow=c(1,1)) # make sure only 1 plot per page
plot(newiris[c("Sepal.Length", "Sepal.Width")], col=kc$cluster, pch=20,cex=2)
points(kc$centers[,c("Sepal.Length", "Sepal.Width")], col=1:3, pch=8, cex=3)
```



```
#
# Here are the exact locations of the cluster centers are.
# Note: just pay attention to Length and Width
kc$centers
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1        6.315       2.896        4.974      1.7031
## 2        4.738       2.905        1.790      0.3524
## 3        5.176       3.624        1.473      0.2727
```

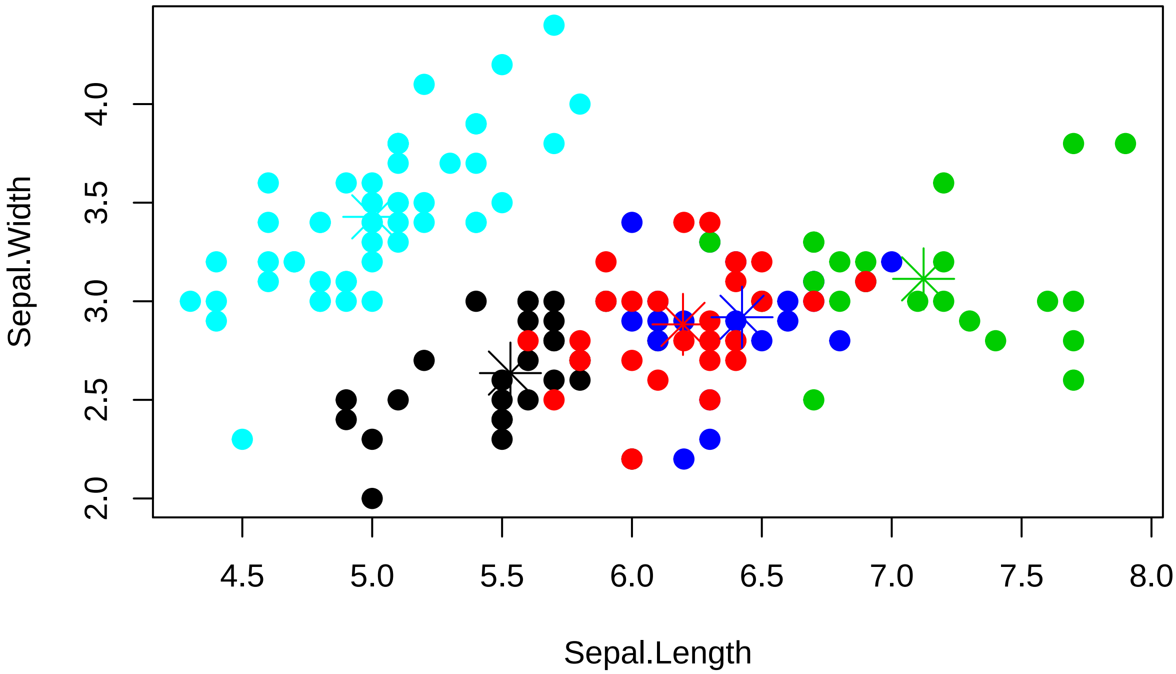
```
#
# Lets try 5 clusters just for fun
(kc <- kmeans(newiris, 5))
```

```
## K-means clustering with 5 clusters of sizes 28, 29, 22, 21, 50
##
## Cluster means:
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1        5.532       2.636        3.961       1.229
## 2        6.197       2.883        5.183       1.934
## 3        7.123       3.114        6.032       2.132
## 4        6.424       2.919        4.605       1.438
## 5        5.006       3.428        1.462       0.246
##
## Clustering vector:
##   [1] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
##  [36] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 4 4 4 1 4 1 4 1 4 1 1 1 1 4 1 4 1 1 4 1
##  [71] 2 1 4 4 4 4 4 4 4 1 1 1 1 2 1 4 4 4 1 1 1 4 1 1 1 1 1 4 1 1 3 2 3 2 3
## [106] 3 1 3 3 3 2 2 3 2 2 2 2 3 3 2 3 2 3 2 3 3 2 2 2 3 3 3 2 2 2 3 2 2 2 3
## [141] 3 2 2 3 3 2 2 2 2 2
##
## Within cluster sum of squares by cluster:
## [1]  9.749  8.738 11.540  4.650 15.151
##  (between_SS / total_SS =  92.7 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

```
#
# Compare the Species label with the clustering result
table(iris$Species, kc$cluster)
```

```
##
##               1  2  3  4  5
##   setosa      0  0  0  0 50
##   versicolor 27  2  0 21  0
##   virginica   1 27 22  0  0
```

```
# This result shows that cluster "setosa" can be easily separated from the other
# clusters, and that clusters "versicolor" and "virginica" are to a small degree
# overlapped with each other.
#
# Plot the clusters and their centres. Note that there are four dimensions in the data
# and that only the first two dimensions are used to draw the plot below.
# Some black points close to the green centre (asterisk) are actually closer
# to the black centre in the four dimensional space.
par(mfrow=c(1,1)) # make sure only 1 plot per page
plot(newiris[c("Sepal.Length", "Sepal.Width")], col=kc$cluster, pch=20,cex=2)
points(kc$centers[,c("Sepal.Length", "Sepal.Width")], col=1:5, pch=8, cex=3)
```


Sepal.Width
2.0
2.5
3.0
3.5
4.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
Sepal.Length

```
#
```