zoo-clustering-kMeans

This is a tutorial on using the kMeans algorithm in the RWeka package to cluster animal types in a zoo.

# load in data

zoo <- read.csv("/Users/byu/Desktop/Data/zoo.csv")  
str(zoo)

## 'data.frame': 101 obs. of 18 variables:  
## $ animal.name: Factor w/ 100 levels "aardvark","antelope",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ hair : int 1 1 0 1 1 1 1 0 0 1 ...  
## $ feathers : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ egg : int 0 0 1 0 0 0 0 1 1 0 ...  
## $ milk : int 1 1 0 1 1 1 1 0 0 1 ...  
## $ airborne : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ aquatic : int 0 0 1 0 0 0 0 1 1 0 ...  
## $ predator : int 1 0 1 1 1 0 0 0 1 0 ...  
## $ toothed : int 1 1 1 1 1 1 1 1 1 1 ...  
## $ backbone : int 1 1 1 1 1 1 1 1 1 1 ...  
## $ breathes : int 1 1 0 1 1 1 1 0 0 1 ...  
## $ venomous : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ fins : int 0 0 1 0 0 0 0 1 1 0 ...  
## $ legs : int 4 4 0 4 4 4 4 0 0 4 ...  
## $ tail : int 0 1 1 0 1 1 1 1 1 0 ...  
## $ domestic : int 0 0 0 0 0 0 1 1 0 1 ...  
## $ catsize : int 1 1 0 1 1 1 1 0 0 0 ...  
## $ type : int 1 1 4 1 1 1 1 4 4 1 ...

# remove columns "names" and "type"

zoo\_unlabel <- zoo[, c(2:17)]  
str(zoo\_unlabel)

## 'data.frame': 101 obs. of 16 variables:  
## $ hair : int 1 1 0 1 1 1 1 0 0 1 ...  
## $ feathers: int 0 0 0 0 0 0 0 0 0 0 ...  
## $ egg : int 0 0 1 0 0 0 0 1 1 0 ...  
## $ milk : int 1 1 0 1 1 1 1 0 0 1 ...  
## $ airborne: int 0 0 0 0 0 0 0 0 0 0 ...  
## $ aquatic : int 0 0 1 0 0 0 0 1 1 0 ...  
## $ predator: int 1 0 1 1 1 0 0 0 1 0 ...  
## $ toothed : int 1 1 1 1 1 1 1 1 1 1 ...  
## $ backbone: int 1 1 1 1 1 1 1 1 1 1 ...  
## $ breathes: int 1 1 0 1 1 1 1 0 0 1 ...  
## $ venomous: int 0 0 0 0 0 0 0 0 0 0 ...  
## $ fins : int 0 0 1 0 0 0 0 1 1 0 ...  
## $ legs : int 4 4 0 4 4 4 4 0 0 4 ...  
## $ tail : int 0 1 1 0 1 1 1 1 1 0 ...  
## $ domestic: int 0 0 0 0 0 0 1 1 0 1 ...  
## $ catsize : int 1 1 0 1 1 1 1 0 0 0 ...

# build kMeans model using RWeka

library(RWeka)  
model\_rweka <- SimpleKMeans (zoo\_unlabel, control = Weka\_control(N = 7, I=500, S=100))  
model\_rweka

##   
## kMeans  
## ======  
##   
## Number of iterations: 5  
## Within cluster sum of squared errors: 94.70379505535756  
##   
## Initial starting points (random):  
##   
## Cluster 0: 0,0,1,0,0,1,1,1,1,0,0,1,0,1,0,1  
## Cluster 1: 1,0,0,1,0,0,1,1,1,1,0,0,4,1,0,1  
## Cluster 2: 0,0,1,0,0,1,1,0,0,0,0,0,6,0,0,0  
## Cluster 3: 0,0,1,0,0,1,1,1,1,1,1,0,4,0,0,0  
## Cluster 4: 0,1,1,0,0,0,1,0,1,1,0,0,2,1,0,0  
## Cluster 5: 0,0,1,0,0,0,0,0,1,1,0,0,4,1,0,1  
## Cluster 6: 0,0,0,1,0,1,1,1,1,1,0,1,0,1,0,1  
##   
## Missing values globally replaced with mean/mode  
##   
## Final cluster centroids:  
## Cluster#  
## Attribute Full Data 0 1 2 3 4 5 6  
## (101.0) (13.0) (37.0) (18.0) (8.0) (14.0) (7.0) (4.0)  
## ===================================================================================================  
## hair 0.4257 0 1 0.2222 0 0 0 0.5  
## feathers 0.198 0 0 0 0 1 0.8571 0  
## egg 0.5842 1 0.027 0.9444 0.875 1 1 0  
## milk 0.4059 0 1 0 0 0 0 1  
## airborne 0.2376 0 0.0541 0.3333 0 0.9286 0.4286 0  
## aquatic 0.3564 1 0.0541 0.3333 0.625 0.2857 0.2857 1  
## predator 0.5545 0.6923 0.4865 0.5 0.875 0.4286 0.4286 1  
## toothed 0.604 1 0.973 0 1 0 0 1  
## backbone 0.8218 1 1 0 1 1 1 1  
## breathes 0.7921 0 1 0.6111 0.875 1 1 1  
## venomous 0.0792 0.0769 0 0.2222 0.375 0 0 0  
## fins 0.1683 1 0 0 0 0 0 1  
## legs 2.8416 0 3.6757 4.7222 2.5 2 2.2857 0.5  
## tail 0.7426 1 0.8649 0.0556 0.625 1 1 0.75  
## domestic 0.1287 0.0769 0.2162 0.0556 0 0.2143 0 0  
## catsize 0.4356 0.3077 0.7568 0.0556 0 0 1 1

# Since it is difficult to interpret and visualize the clustering results with RWeka, we will introduce some built-in R functions, as well as visualization packages

## use the kMeans algorith in R

model\_r <- kmeans(zoo\_unlabel, 7)  
model\_r

## K-means clustering with 7 clusters of sizes 17, 23, 8, 12, 15, 5, 21  
##   
## Cluster means:  
## hair feathers egg milk airborne aquatic predator  
## 1 1.00000000 0 0.05882353 1.0000000 0.0000000 0.11764706 1.0000000  
## 2 0.04347826 0 0.82608696 0.1304348 0.0000000 0.78260870 0.7391304  
## 3 0.00000000 0 1.00000000 0.0000000 0.0000000 0.75000000 0.7500000  
## 4 0.33333333 0 0.91666667 0.0000000 0.5000000 0.25000000 0.4166667  
## 5 0.00000000 1 1.00000000 0.0000000 1.0000000 0.33333333 0.3333333  
## 6 0.00000000 1 1.00000000 0.0000000 0.2000000 0.20000000 0.8000000  
## 7 1.00000000 0 0.00000000 1.0000000 0.0952381 0.04761905 0.0952381  
## toothed backbone breathes venomous fins legs tail  
## 1 0.9411765 1.000000 1.0000000 0.000000 0.00000000 4.000000 0.88235294  
## 2 0.8260870 0.826087 0.3043478 0.173913 0.69565217 0.000000 0.78260870  
## 3 0.6250000 0.750000 0.7500000 0.125000 0.00000000 4.125000 0.37500000  
## 4 0.0000000 0.000000 0.7500000 0.250000 0.00000000 6.333333 0.08333333  
## 5 0.0000000 1.000000 1.0000000 0.000000 0.00000000 2.000000 1.00000000  
## 6 0.0000000 1.000000 1.0000000 0.000000 0.00000000 2.000000 1.00000000  
## 7 1.0000000 1.000000 1.0000000 0.000000 0.04761905 3.333333 0.85714286  
## domestic catsize  
## 1 0.05882353 0.88235294  
## 2 0.04347826 0.30434783  
## 3 0.00000000 0.12500000  
## 4 0.08333333 0.08333333  
## 5 0.20000000 0.13333333  
## 6 0.00000000 0.80000000  
## 7 0.33333333 0.66666667  
##   
## Clustering vector:  
## [1] 1 7 2 1 1 7 7 2 2 7 1 5 2 2 3 4 5 7 2 2 5 5 7 5 4 3 3 7 7 7 4 7 7 5 2  
## [36] 7 7 5 2 4 4 6 4 5 1 1 4 1 1 1 1 4 3 4 1 7 6 5 6 5 2 2 2 1 1 7 2 1 1 1  
## [71] 7 6 4 2 2 7 2 2 5 5 2 2 2 5 7 3 2 5 4 3 3 3 2 7 7 6 7 4 1 2 5  
##   
## Within cluster sum of squares by cluster:  
## [1] 8.117647 44.608696 12.375000 25.666667 10.800000 3.200000 36.095238  
## (between\_SS / total\_SS = 79.9 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss"   
## [5] "tot.withinss" "betweenss" "size" "iter"   
## [9] "ifault"

## print the centroids

model\_r$centers

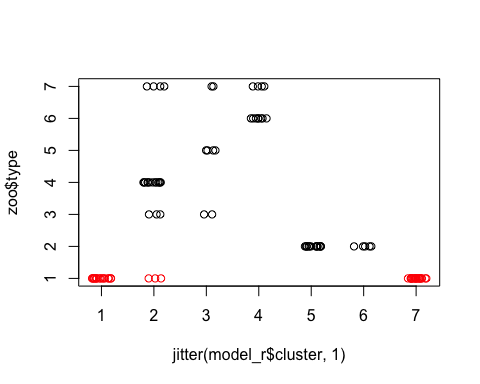
## hair feathers egg milk airborne aquatic predator  
## 1 1.00000000 0 0.05882353 1.0000000 0.0000000 0.11764706 1.0000000  
## 2 0.04347826 0 0.82608696 0.1304348 0.0000000 0.78260870 0.7391304  
## 3 0.00000000 0 1.00000000 0.0000000 0.0000000 0.75000000 0.7500000  
## 4 0.33333333 0 0.91666667 0.0000000 0.5000000 0.25000000 0.4166667  
## 5 0.00000000 1 1.00000000 0.0000000 1.0000000 0.33333333 0.3333333  
## 6 0.00000000 1 1.00000000 0.0000000 0.2000000 0.20000000 0.8000000  
## 7 1.00000000 0 0.00000000 1.0000000 0.0952381 0.04761905 0.0952381  
## toothed backbone breathes venomous fins legs tail  
## 1 0.9411765 1.000000 1.0000000 0.000000 0.00000000 4.000000 0.88235294  
## 2 0.8260870 0.826087 0.3043478 0.173913 0.69565217 0.000000 0.78260870  
## 3 0.6250000 0.750000 0.7500000 0.125000 0.00000000 4.125000 0.37500000  
## 4 0.0000000 0.000000 0.7500000 0.250000 0.00000000 6.333333 0.08333333  
## 5 0.0000000 1.000000 1.0000000 0.000000 0.00000000 2.000000 1.00000000  
## 6 0.0000000 1.000000 1.0000000 0.000000 0.00000000 2.000000 1.00000000  
## 7 1.0000000 1.000000 1.0000000 0.000000 0.04761905 3.333333 0.85714286  
## domestic catsize  
## 1 0.05882353 0.88235294  
## 2 0.04347826 0.30434783  
## 3 0.00000000 0.12500000  
## 4 0.08333333 0.08333333  
## 5 0.20000000 0.13333333  
## 6 0.00000000 0.80000000  
## 7 0.33333333 0.66666667

## get cluster assignment

cluster\_assignment <- data.frame(zoo,model\_r$cluster)  
#View(cluster\_assignment)

## visualize animal types and clusters by specific features, red = milk yes, black = milk no.

plot(zoo$type ~ jitter(model\_r$cluster, 1), pch=21,col=as.factor(zoo$milk))



## use PCA in visualization package "cluster" to visualize kMeans model. PCA is principal components analysis

#install.packages("cluster")  
library(cluster)  
clusplot(zoo\_unlabel,model\_r$cluster,color=TRUE,shade=TRUE, labels=2, lines=0) # plot clusters

