

Statistical Learning Workshop - DBSCAN

Abdollah Safari

26/02/2021

Load data

Use the `penguins` dataset that was wrangled previously in “Penguins.R.”

```
# Set the seed to reproduce results
set.seed(1)

# LOAD AND PREPARE DATA #####

# Save the `penguins` dataset to `df`
df <- import("~/Downloads/Ex_Files_Data_Science_R/Exercise Files/data/penguins.RDS")

# Take a look at the data
df
```

```
## # A tibble: 342 x 5
##   y          bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
##   <fct>          <dbl>          <dbl>          <int>          <int>
## 1 Adelie          39.1            18.7            181            3750
## 2 Adelie          39.5            17.4            186            3800
## 3 Adelie          40.3             18             195            3250
## 4 Adelie          36.7            19.3            193            3450
## 5 Adelie          39.3            20.6            190            3650
## 6 Adelie          38.9            17.8            181            3625
## 7 Adelie          39.2            19.6            195            4675
## 8 Adelie          34.1            18.1            193            3475
## 9 Adelie          42             20.2            190            4250
## 10 Adelie         37.8            17.1            186            3300
## # ... with 332 more rows
```

Prepare data

```
# Summarize the data
df %>% summary()
```

```
##           y      bill_length_mm  bill_depth_mm  flipper_length_mm
## Adelie   :151    Min.      :32.10    Min.      :13.10    Min.      :172.0
## Chinstrap: 68    1st Qu.:39.23    1st Qu.:15.60    1st Qu.:190.0
## Gentoo   :123    Median   :44.45    Median   :17.30    Median   :197.0
##           Mean     :43.92    Mean     :17.15    Mean     :200.9
##           3rd Qu.:48.50    3rd Qu.:18.70    3rd Qu.:213.0
##           Max.     :59.60    Max.     :21.50    Max.     :231.0
##  body_mass_g
## Min.      :2700
## 1st Qu.:3550
## Median   :4050
## Mean     :4202
## 3rd Qu.:4750
## Max.     :6300
```

```
# Separate the class labels
species <- df %>% # Rename `y` back to `species`
  pull(y)         # Select just `y` as a vector

df %<>%
  select(-y) %>% # Select everything except `y`
  scale()        # Standardize variables
```

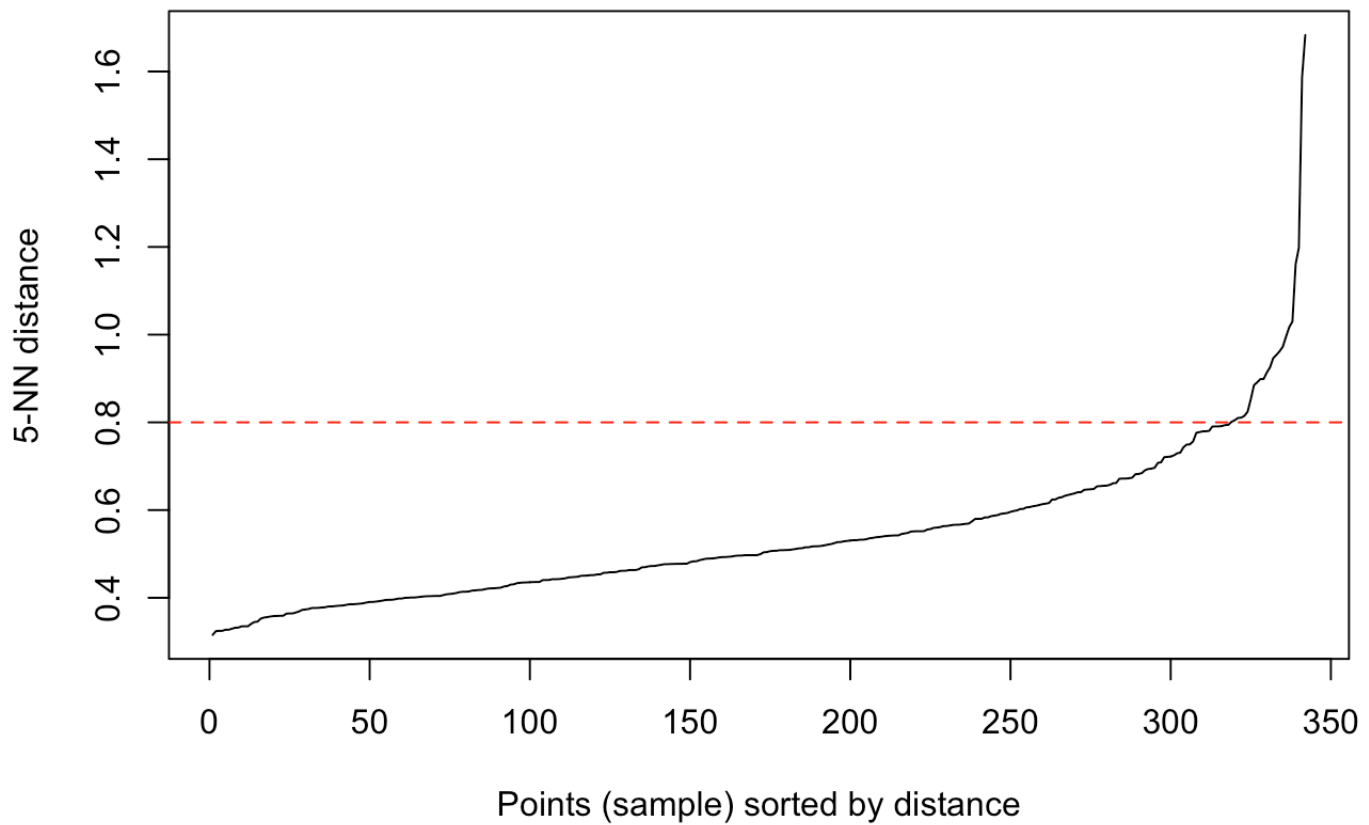
DBSCAN clustering

Choose a value for “minPts” or “k,” which is the minimum number of neighboring points for clustering. minPts should be odd and have a value of at least 3, with higher values for larger datasets. We’ll use 5 in this example.

Using the chosen value of minPts (k = 5 in our case), find the optimal value of “eps” (epsilon neighborhood radius) by graphing distances and looking for a pronounced “knee” or bend.

```
df %>%
  scale() %>%
  kNNdistplot(k = 5)

# Draw a horizontal line at the optimal value
abline(h = 0.8, lty = 2, col = "red")
```



```
# Run DBSCAN with the parameter values for minPts (k = 5)
# and eps (0.8)
db <- df %>%
  dbscan(
    eps = 0.8,
    minPts = 5
  )

# Print the DBSCAN object
db %>% print()
```

```
## DBSCAN clustering for 342 objects.  
## Parameters: eps = 0.8, minPts = 5  
## The clustering contains 2 cluster(s) and 5 noise points.  
##  
##    0    1    2  
##    5 216 121  
##  
## Available fields: cluster, eps, minPts
```

```
# Visualize the clusters according to species  
db %>%  
  fviz_cluster(  
    df,  
    geom = "point"  
  ) +  
  geom_text(  
    vjust=1.5, #add label below  
    aes(  
      #label points excluding noise  
      color = as.factor(db$cluster[db$cluster!=0]),  
      label = species[db$cluster!=0]  
    )  
  )
```

Cluster plot

