K-Nearest Neighbiur Algorithm

By Saba Akram

2/4/2021

In this exercise we are going to learn about k-Nearest Algorithm, which is not only used for classification but also for prediction.

new<-c(0.05,0.25)  
A<-c(0.0467,0.2471)  
B<-c(0.0533,0.1912)  
C<-c(0.0917,0.2794)  
data<-rbind(A,B,C)  
dimnames(data)<-list(c("Dark","Medium","Light"),c("AgeMMN","Na/k(MMN"))  
data

## AgeMMN Na/k(MMN  
## Dark 0.0467 0.2471  
## Medium 0.0533 0.1912  
## Light 0.0917 0.2794

## Declaring the true classifications of A, B, C.  
trueclass<-c("Dark","Medium","Light")

## Runing the K nearest neighbour.  
library(class)  
knn<-knn(data,new,cl=trueclass,k=3,prob = TRUE)  
knn

## [1] Light  
## attr(,"prob")  
## [1] 0.3333333  
## Levels: Dark Light Medium

## Calculating the eucleadian distance  
## Its require package field  
library(fields)

## Loading required package: spam

## Loading required package: dotCall64

## Loading required package: grid

## Spam version 2.6-0 (2020-12-14) is loaded.  
## Type 'help( Spam)' or 'demo( spam)' for a short introduction   
## and overview of this package.  
## Help for individual functions is also obtained by adding the  
## suffix '.spam' to the function name, e.g. 'help( chol.spam)'.

##   
## Attaching package: 'spam'

## The following objects are masked from 'package:base':  
##   
## backsolve, forwardsolve

## See https://github.com/NCAR/Fields for  
## an extensive vignette, other supplements and source code

together<-rbind(new,data)  
together

## AgeMMN Na/k(MMN  
## new 0.0500 0.2500  
## Dark 0.0467 0.2471  
## Medium 0.0533 0.1912  
## Light 0.0917 0.2794

## stretching the axes  
ds\_newA<-sqrt((new[1]-A[1]^2+(3\*new[2]-A[2]))^2)  
ds\_newB<-sqrt((new[1]-B[1]^2+(3\*new[2]-B[2]))^2)  
ds\_newC<-sqrt((new[1]-C[1]^2+(3\*new[2]-C[2]))^2)

## Creating the table  
distance<-c(ds\_newA,ds\_newB,ds\_newC)  
BP<-c(120,122,130)  
data1<-cbind(BP,data,distance)  
data1

## BP AgeMMN Na/k(MMN distance  
## Dark 120 0.0467 0.2471 0.5507191  
## Medium 122 0.0533 0.1912 0.6059591  
## Light 130 0.0917 0.2794 0.5121911

## locally weighted Averaging.  
weights<-(1/(distance^2))  
sum\_wi<-sum(weights)  
sum\_wiyi<-sum(weights\*data1[,1])  
yhat\_new<-sum\_wiyi/sum\_wi  
yhat\_new

## [1] 124.4308

## Taking the example of ClassifyRisk dataset  
  
 ClassifyRisk <- read.csv("D:/M.Sc in Banking and Financial Analytics/Sem 3/Data Analytic and Machine learning/Data Mining and Predictive Analysis/Data sets/ClassifyRisk")  
## The table of the chapter cintains records 51,65,79,87,124,141,150,162,163  
risk<-ClassifyRisk[c(51,65,79,87,124,141,150,162,163),c(5,1,4,6)]  
risk$married.I<-ifelse(risk$marital\_status=="married",1,0)  
risk$single.I<-ifelse(risk$marital\_status=="single",1,0)  
risk<-risk[,-2]  
new2<-ClassifyRisk[163,c(5,1,4)]  
new2$married.I<ifelse(new2$marital\_status=="married",1,0)

## logical(0)

new2$single.I<-ifelse(new2$marital\_status=="single",1,0)  
new2<-new2[,-2]  
cll<-c(risk[,3])

#knn2<-knn(train=risk[,c(1,2,4,5)],test=new2,cl=cll,k=3)