Белорусский Государственный Университет

Информатики и Радиоэлектроники

Факультет компьютерных систем и сетей

Кафедра ЭВМ

Лабораторная работа №4

Тема «Классификация»

Выполнил: Проверил:

Студент группы 7М2431 Марченко В.В.

Зайцев Ю.В.

Минск, 2017

Задание:

Требуется исследовать работу алгоритма классификации объектов по ближайшему соседу. Для каждого набора данных требуется выполнить следующие задания:

1. Случайным образом разделить имеющуюся выборку примерно пополам на обучающую выборку и контрольную выборку.
2. Произвести классификацию объектов контрольной выборки, используя данные о классах объектов из обучающей выборки, с помощью алгоритма классификации по ближайшему соседу.
3. Изобразить объекты графически в трёхмерном пространстве. Для объектов разных классов и разных выборок следует использовать разные обозначения. Отдельно представить графики, на одном из которых объекты из контрольной выборки имеют свои настоящие классы, а на другом – классы, к которым их отнес классификатор.
4. Оценить вероятность ошибочной классификации.

Исходные данные:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| n1 | a1 | R1 | n2 | a1 | R2 |
| 100 |  |  | 50 |  |  |

Где независимые случайные векторы (X, Y), n1 из которых относятся к первому классу, а n2 – ко второму классу. Векторы, относящиеся к первому классу, распределены по гауссовскому закону с математическим ожиданием а1 и корреляционной матрицей R1, а векторы, относящиеся ко второму классу – по гауссовскому закону с математическим ожиданием а2 и корреляционной матрицей. R2.

Название файла: wine.csv

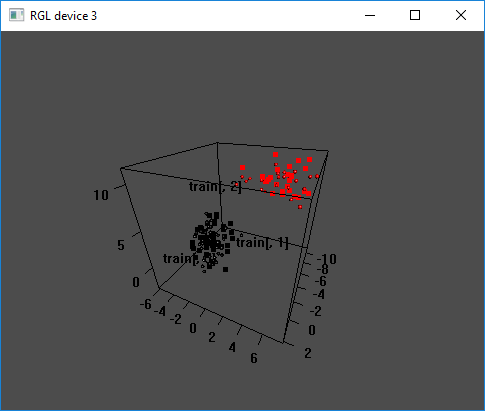
Ссылка: <http://archive.ics.uci.edu/ml/datasets/Wine>

Первый признак: alcohol (столбец № 2)

Второй признак: color-intensity (столбец № 11)

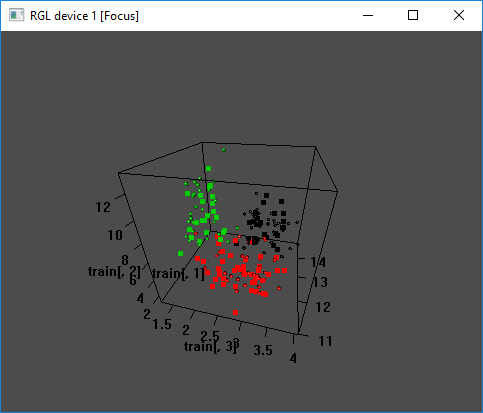
Результаты:

1. Смоделированные данные:



Вероятность ошибочной классификации: 0.

1. Реальные данные:



Вероятность ошибочной классификации: 0.061.

Листинг программы:

install.packages("rgl")

require(class)

require(MASS)

require(rgl)

plot\_points <- function(train, test, clazz.train,

clazz.test) {

rgl.open()

plot3d(train[, 1], train[, 2], train[, 3],

col=clazz.train, type='p', size=5, add=FALSE)

plot3d(test[, 1], test[, 2], test[, 3],

col=clazz.test, type='s', size=1, add=TRUE)

}

analyse\_knn <- function(dat, clazz) {

n <- nrow(dat)

rnd.num <- sample(1 : n)

train.num <- rnd.num[1 : (n %/% 2)]

test.num <- rnd.num[(n %/% 2 + 1) : n]

train <- dat[train.num,]

test <- dat[test.num,]

clazz.train <- clazz[train.num]

clazz.test <- clazz[test.num]

clazz.knn <- knn(train, test, clazz.train)

print(sum(clazz.test != clazz.knn) / n)

plot\_points(train, test, clazz.train, clazz.test)

plot\_points(train, test, clazz.train, clazz.knn)

}

dat <- read.table("wine.csv", sep=",")

analyse\_knn(cbind(dat$V2, dat$V11, dat$V13), unclass(dat$V1))

n1 <- 100

a1 <- c(-2, 2, -2)

r1 <- cbind(c(2, 1, 1), c(1, 4, 1.4), c(1, 1.4, 2))

n2 <- 50

a2 <- c(4, 8, -8)

r2 <- cbind(c(4, 0.2, 0.3), c(0.2, 2, 0.4), c(0.3, 0.4, 2))

dat <- rbind(mvrnorm(n1, a1, r1), mvrnorm(n2, a2, r2))

analyse\_knn(dat, c(rep(1, n1), rep(2, n2)))