|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство науки и высшего образования Российской Федерации**  Калужский филиал  федерального государственного бюджетного  образовательного учреждения высшего образования  ***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»***  ***(КФ МГТУ им. Н.Э. Баумана)*** |

**ФАКУЛЬТЕТ** ***ИУ-КФ «Информатика и управление»***

**КАФЕДРА** ***ИУ4-КФ «Программное обеспечение ЭВМ, информационные технологии»***

**ЛАБОРАТОРНАЯ РАБОТА №6**

**«Методы классификации многомерных объектов пересекающихся классов с использованием карт Кохонена»**

**ДИСЦИПЛИНА: «Методы машинного обучения»**

|  |  |  |
| --- | --- | --- |
| Выполнил: студент гр. ИУК4-62Б | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ( Борисов Н.С. )  (Подпись) (Ф.И.О.) |
| Проверил: | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ( Кручинин И.И. )  (Подпись) (Ф.И.О.) |
| Дата сдачи (защиты):  Результаты сдачи (защиты): | | |
|  | - Балльная оценка:  - Оценка: | |

Калуга, 2022

**Цель работы**: приобрести практические навыки использования линейных классификации многомерных объектов.

**Задачи:**

1. Изучить линейные классификаторы;

2. Понять принципы методов классификации многомерных объектов пересекающихся классов;

3. Научится работать с линейными классификаторами.

**Задание:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Наименование рыбы** | **Калории** | **Жиры** | **Холестерин** | **Натрий** | **Калий** | **Белок** | **Коэффициент скорости** |
| Рыба мечь | 172 | 8 | 78 | 97 | 499 | 23 | 69 |
| Тунец | 184 | 6 | 49 | 50 | 323 | 30 | 68 |
| Акула | 130 | 4.5 | 51 | 79 | 160 | 21 | 70 |
| лосось | 208 | 13 | 55 | 59 | 363 | 20 | 58 |
| скумбрия | 262 | 18 | 75 | 83 | 401 | 24 | 59 |
| треска | 69 | 6.1 | 40 | 55 | 340 | 16 | 29 |
| кефаль | 88 | 2.2 | 53 | 72 | 468 | 23 | 28 |
| пикша | 90 | 0.6 | 66 | 261 | 351 | 20 | 37 |
| палтус | 186 | 14 | 46 | 80 | 268 | 14 | 55 |
| камбала | 70 | 1.9 | 45 | 296 | 160 | 12 | 44 |

**Классификация рыб по скорости движения:**

1. Очень быстроплавающие – коэффициент скорости от 61 до 70;

2. быстроплавающие– коэффициент скорости от 31 до 60;

3. Умеренно быстрые - – коэффициент скорости от 21 до 30

**Классификация рыб по пищевой ценности:** нежирные, умеренно жирные и жирные (характеристические показатели: калории, жиры, холестерин, натрий, калий, белок)

**Вариант 2**

Разработать классификатор для идентификации сортов рыбы с точки зрения пищевой ценности и скорости движения в воде, основанный на обучении без учителя.

Визуализировать результаты с помощью карт Кохонена для оценки распределения сортов рыбы в рамках выбранного водного пространства.

Использовать средства языка R – функции SOM иSOMGRID. Параметры для функции SOM GRID fct=” gaussian”, topo=” hexagonal”. Зададим число кластеров k=6 и выполним иерархическую кластеризацию (по умолчанию используются method = "complete" и distance = "euclidean"). Построить карты «mapping", "quality", "property", "count". Проверить результаты классификации с помощью функций RBF и rbfDDA.

Рыбы: акула, рыба - мечь, кефаль, треска, пикша, палтус, скумбрия необходимо разделить по категориям: умеренно жирные и нежирные, очень быстроплавающие, быстроплавающие и умеренно быстрые.

**Листинг программы:**

library(kohonen)

library(RSNNS)

ramFo <- read.csv2("file.csv")

alg <- matrix(0:0, nrow=150, ncol=4)

for (i in 1:50) {

for (j in 1:4) {

alg[i,j] = sample(5:50,1)

}

}

for (i in 51:100) {

for (j in 1:4) {

alg[i,j] = sample(51:95,1)

}

}

for (i in 101:150) {

for (j in 1:4) {

alg[i,j] = sample(1:4,1)

}

}

write.table(alg, file="GMB1.txt")

ramFo2 = data.frame(read.table("GMB1.txt", header = TRUE, sep = ""))

inputs <- normalizeData(ramFo2[,1:4], "norm")

model <- som(inputs, mapX=26, mapY=26, maxit=400,calculateActMaps=TRUE, targets=ramFo2[,4])

par(mfrow=c(3,3))

for(i in 1:ncol(inputs))

plotActMap(model$componentMaps[[i]],col=rev(topo.colors(12)))

plotActMap(model$map, col=rev(heat.colors(12)))

plotActMap(log(model$map+1), col=rev(heat.colors(12)))

persp(1:model$archParams$mapX, 1:model$archParams$mapY, log(model$map+1),theta = 30, phi = 30, expand = 0.5, col = "lightblue")

plotActMap(model$labeledMap)

model$componentMaps

model$labeledUnits

model$map

names(model)

j=3

for (i in 1:20000000) {

j=j+1

}

ramFo2 <- ramFo2[sample(1:nrow(ramFo2),length(1:nrow(ramFo2))),1:ncol(ramFo2)]

ramFo2Values <- ramFo2[,1:4]

ramFo2Targets <- decodeClassLabels(ramFo2[,4])

ramFo2 <- splitForTrainingAndTest(ramFo2Values, ramFo2Targets, ratio=0.15)

ramFo2 <- normTrainingAndTestSet(ramFo2)

model <- rbfDDA(ramFo2$inputsTrain, ramFo2$targetsTrain)

summary(model)

par(mfrow=c(2,2))

weightMatrix(model)

extractNetInfo(model)

par(mfrow=c(2,2))

plotIterativeError(model)

predictions <- predict(model,ramFo2$inputsTest)

plotRegressionError(predictions[,2], ramFo2$targetsTest[,2])

confusionMatrix(ramFo2$targetsTrain,fitted.values(model))

confusionMatrix(ramFo2$targetsTest,predictions)

plotROC(fitted.values(model)[,2], ramFo2$targetsTrain[,2])

plotROC(predictions[,2], ramFo2$targetsTest[,2])

confusionMatrix(ramFo2$targetsTrain, encodeClassLabels(fitted.values(model),method="402040", l=0.4, h=0.6))

ramFo2<-ramFo2[sample(1:nrow(ramFo2),length(1:nrow(ramFo2))),1:ncol(ramFo2)]

ramFo2Values <- ramFo2[,1:4]

ramFo2Targets <- decodeClassLabels(ramFo2[,4])

ramFo2 <- splitForTrainingAndTest(ramFo2Values, ramFo2Targets, ratio=0.15)

ramFo2 <- normTrainingAndTestSet(ramFo2)

model <- rbf(ramFo2$inputsTrain, ramFo2$targetsTrain, size=40, maxit=1000,

initFuncParams=c(0, 1, 0, 0.01, 0.01),

learnFuncParams=c(1e-8, 0, 1e-8, 0.1, 0.8), linOut=TRUE)

summary(model)

par(mfrow=c(2,2))

weightMatrix(model)

extractNetInfo(model)

par(mfrow=c(2,2))

plotIterativeError(model)

predictions <- predict(model,ramFo2$inputsTest)

plotRegressionError(predictions[,2], ramFo2$targetsTest[,2])

confusionMatrix(ramFo2$targetsTrain,fitted.values(model))

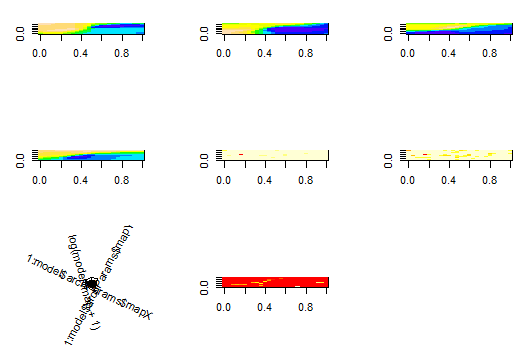
confusionMatrix(ramFo2$targetsTest,predictions)

plotROC(fitted.values(model)[,2], ramFo2$targetsTrain[,2])

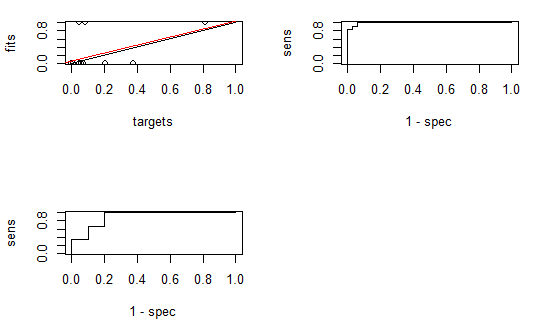
plotROC(predictions[,2], ramFo2$targetsTest[,2])

confusionMatrix(ramFo2$targetsTrain, encodeClassLabels(fitted.values(model),method="402040", l=0.4, h=0.6))

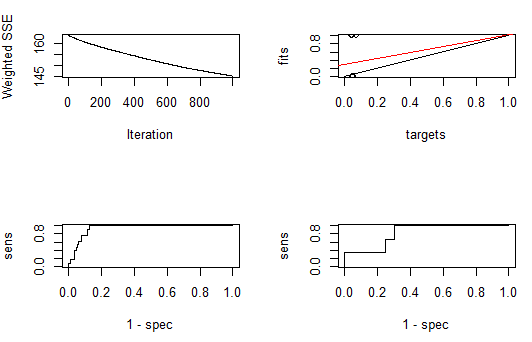
**Результаты работы:**



**Рис.1**



**Рис.2**



**Рис.3**

**Вывод:** в ходе лабораторной работы были приобретены практические навыки использования линейных классификации многомерных объектов.