Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**

**Кафедра вычислительных технологий**

**ЛАБОРАТОРНАЯ РАБОТА №6**

**Дисциплина: Обработка больших данных**

Работу выполнила: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д. Н. Баева

Направление подготовки: 02.03.02 Фундаментальная информатика и информационные технологии

Преподаватель: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В. И. Шиян

**Цель:** закрепить знания об алгоритмах классификации и кластеризации данных, ознакомиться с некоторыми функциями языка R, осуществляющими этот вид анализа, принципами их работы. Научиться визуализировать результаты работы функций кластерного анализа и классификаторов, интерпретировать полученные результаты. Научиться выполнять классификацию на основе формулы Байеса и деревьев решений.

**Вариант:** 6

**Набор данных:** Филиалы.

**Часть 1.** **Задачи классификации и кластеризации**

**Описание данных.**

**Название:** Условный код филиала;

**Площадь:** Площадь торгового зала, кв.м.

**Проходим:** Проходимость, ср. кол-во входящих в магазин за 10 минут (с 14–00 до 18-00);

**Ассортим**: Ассортимент (текстовая переменная!);

**Конкурен**: Как выглядит филиал по совокупности показателей по отношению к ближайшему конкуренту (текстовая переменная!);

**Метро:** За сколько минут можно дойти до ближайшей станции метро;

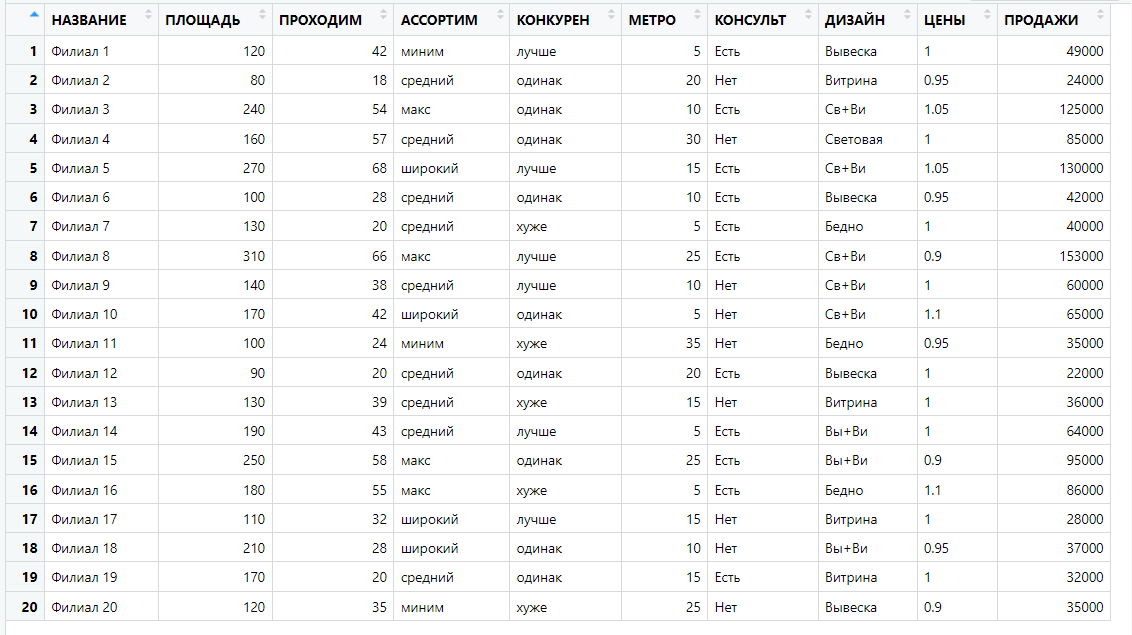
**Консульт:** Наличие в торговом зале консультантов (текстовая переменная!);

**Дизайн:** Наличие или отсутствие следующих компонентов: вывеска,витрина, световая вывеска (текстовая переменная!);

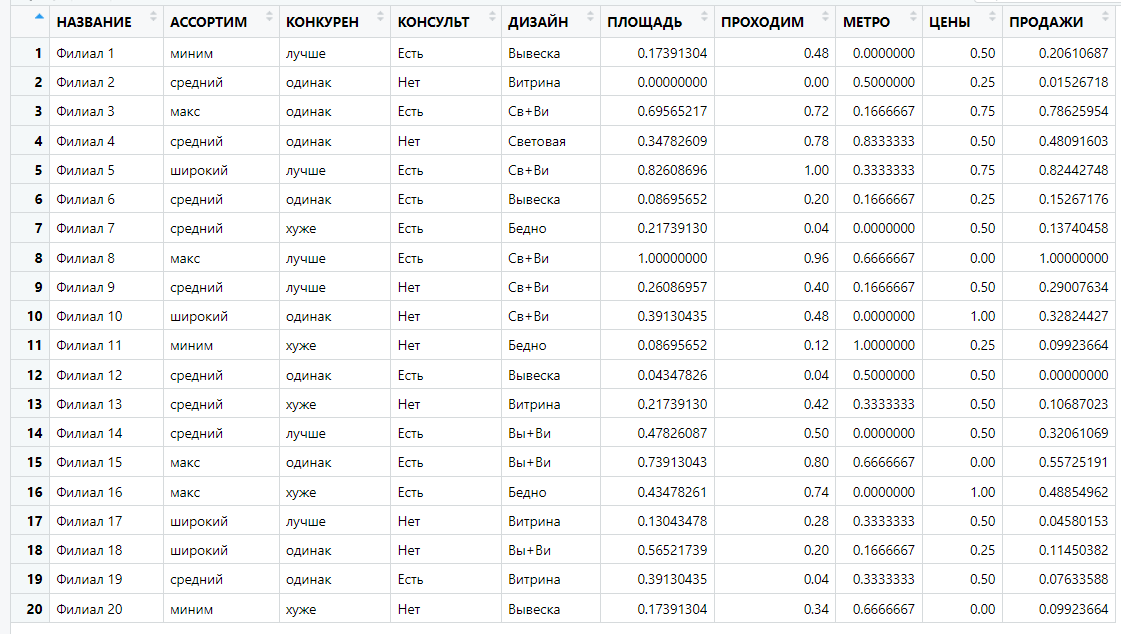
**Цены:** Индекс цен по отношению к базовым;

**Продажи:** Совокупные продажи за 2 последних месяца.

Из исходного .dat-файла были импортированы данные в RStudio. Таблица представлена на рисунке 1.

Рисунок 1 – Исходная таблица с данными

В данной задаче переменные существенно различны, поэтому необходимо выполнить нормализацию данных. На рисунке 2 изображена таблица после нормализации.

Рисунок 2 – Нормализированная таблица с данными

Затем была выполнена иерархическая кластеризация.

По полученным данным была построена дендрограмма. Она изображена на рисунке 3.

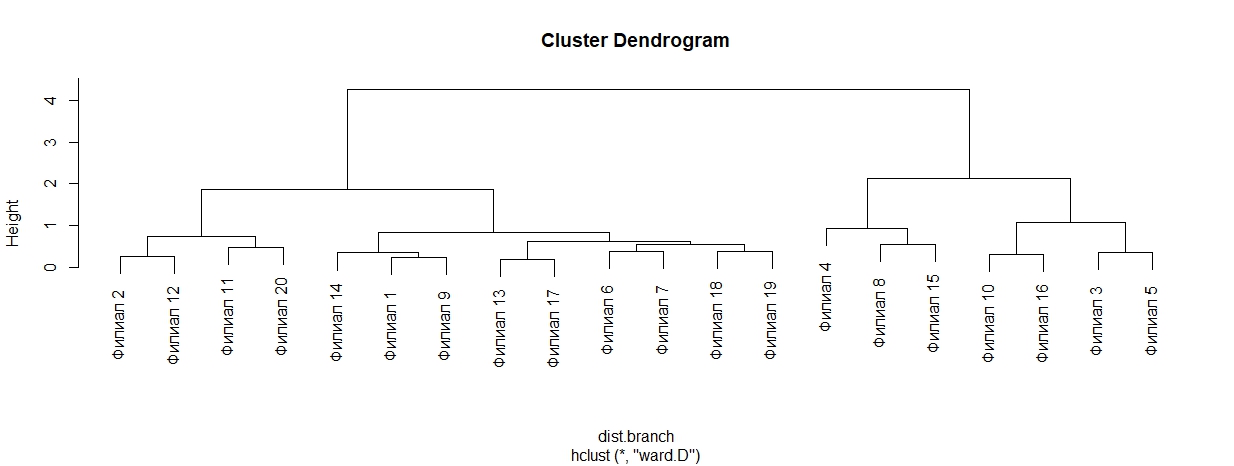


Рисунок 3 – Дендограмма, построенная по полученным данным

Увидев дендрограмму, очевидно, что целесообразнее выделить 4 кластера. Они представлены на рисунке 4.1, а на рисунке 4.2. отображены список филиалов, разбитых по кластерам.

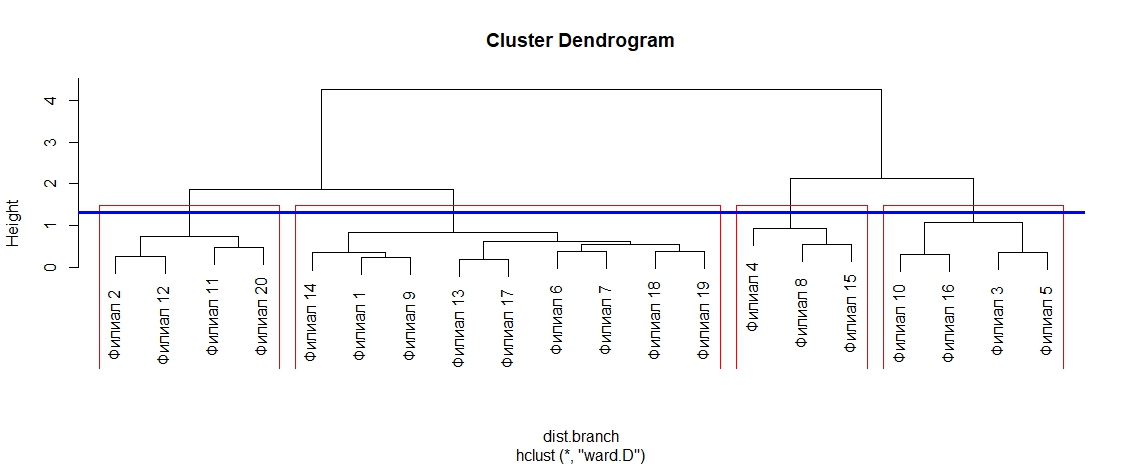


Рисунок 4.1 – Дендрограмма с кластерами

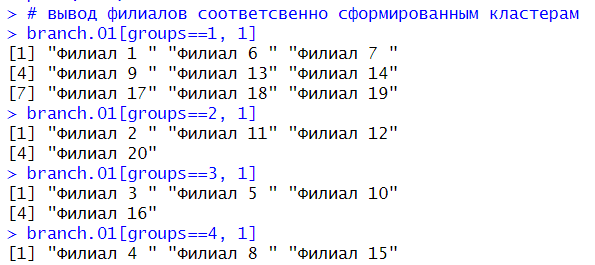


Рисунок 4.2 – Филиалы, разбитые по кластерам

Далее для каждого кластера определяем средние значения характеристик. На рисунке 5 изображен график распределения всех характеристик соответственно.

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 5 – Распределение характеристик

Кластер №4 выделяется большими значениями почти в каждой из характеристик. Это прибыльные, большие филиалы, которые точно не стоит закрывать. Кластер №2 выглядит невыгодным: довольно маленькая площадь, низкая проходимость, самые низкие цены и продажи, хотя он расположен ближе к метро. Вполне вероятно, близкое нахождение к метро не используется в маркетинговом продвижении, из-за чего покупатели просто не замечают эти филиалы. Остальные два кластера находятся на уровне между описанными выше и ничем особо не выделяются.

Затем, для анализа были построены графики изучения по каждой из характеристик на рисунках 6, 8, 9, 10, 11.Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 6 – Распределение характеристики ПЛОЩАДЬ

Исходя из графика, можно сделать вывод, что существует всего 2–3 типа площади для помещения филиалов.Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 7 – Распределение характеристики ПРОХОДИМОСТЬ

Данный график схож с предыдущим графиков. Аналогично можно сказать, что для всех филиалов выделяются 2–3 типа среднего количества входящих в магазин.

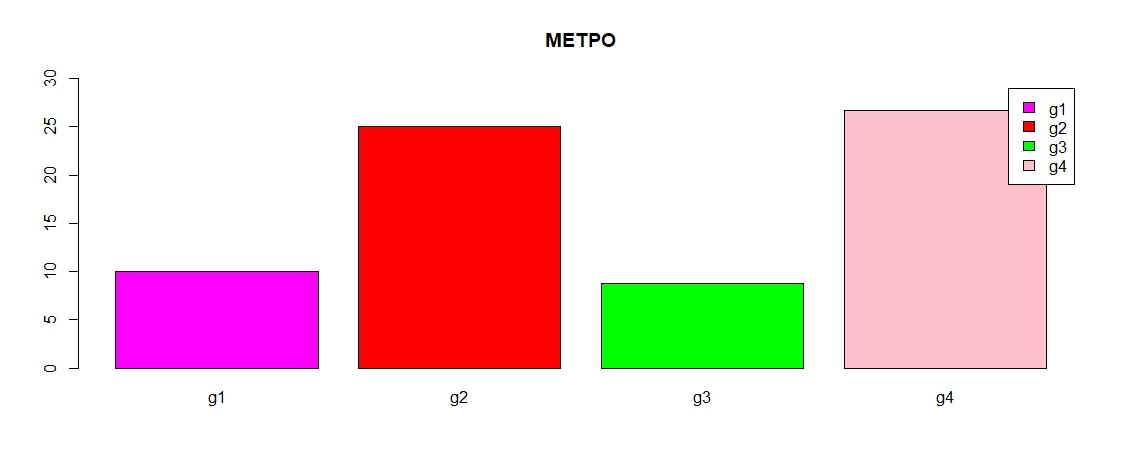


Рисунок 8 – Распределение характеристики МЕТРО

Анализ данной характеристики показывает, что филиалы помимо площади отличаются расположением у метро.

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 9 – Распределение характеристики ЦЕНЫ

Характеристика ЦЕНЫ показывает, что в филиалах все цены примерно на одном уровне.

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 10 – Распределение характеристики ПРОДАЖИ

Анализ продаж в филиалах показывает, что они делятся на 2 типа: до 45000 и выше 45000.

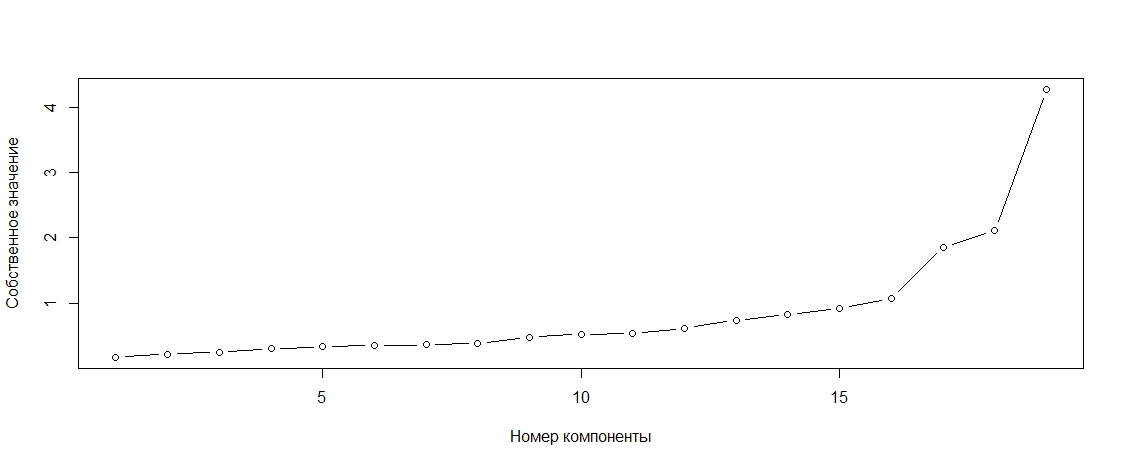
Следующим заданием было построение диаграмма “Каменная осыпь”. Она представлена на рисунке 11.

Рисунок 11 – Диаграмма “Каменная осыпь”

После этого была работа с построением scatterplot, используя ggplot2.

На рисунке 12 изображена диаграмма рассеивания, выражающая зависимость между продажами и проходимостью.

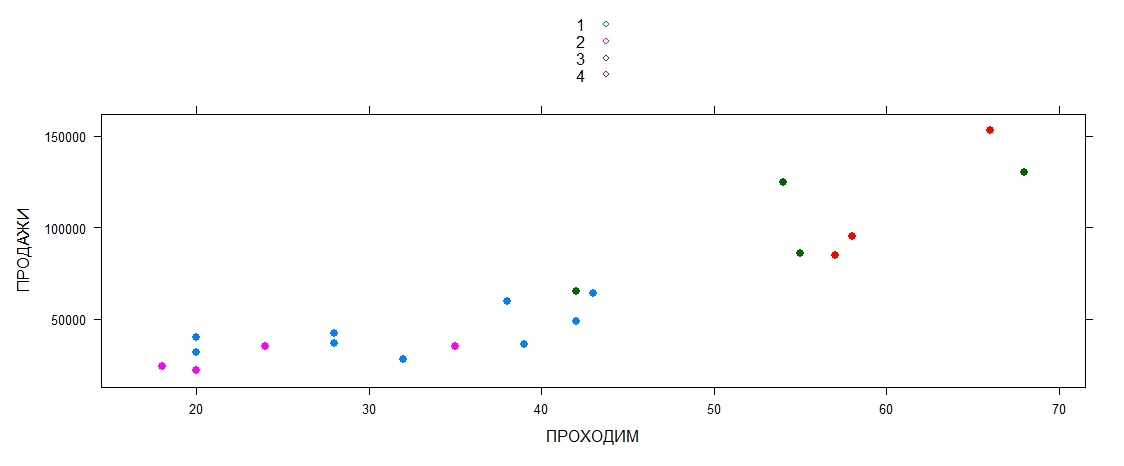


Рисунок 12 – Диаграмма рассеивания для ПРОХОДИМОСТИ и ПРОДАЖ

Заметна тенденция: чем выше проходимость, тем выше продажи.

После были построены «ящики с усами», показанные на рисунке 13.

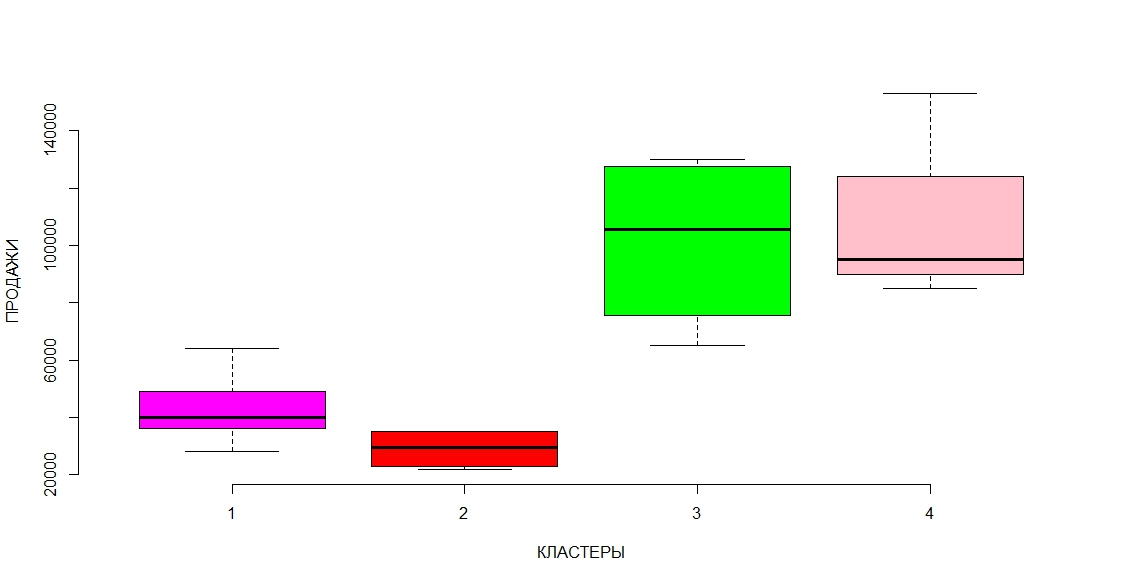


Рисунок 13 – Полученный Boxplot

На нем мы можем видеть, что в кластере №4 присутствуют высокие продажи, но не на постоянной основе. В кластере №2 выбросы отсутствуют, а в кластере №3 размах продаж невысок.

В завершении части 1 лабораторной работы был построен трехмерный график, классифицирующий филиалы по следующим характеристикам: продажи, метро и цены. Он изображён на рисунке 14.

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 14 – Трехмерный график

**Часть 2.** **Байесовская классификация и деревья принятия решений на R**

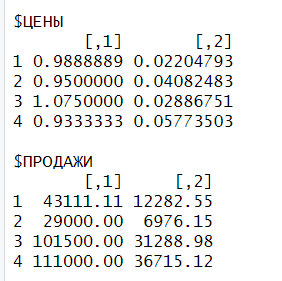
Сначала построим таблицы вероятностей по всем характеристикам. В первом столбце – средние значения параметров, во втором столбце – их стандартные отклонения. Фрагмент таблицы можно увидеть на рисунке 15.

Рисунок 15 – Фрагмент таблицы вероятностей

Затем строятся функции плотности для характеристик на рисунке 16. Распределение на графике подтверждает правильность распределения на кластеры.

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 16 – Функции плотности

Далее проведем классификацию Decision Tree. Результаты можно увидеть на рисунке 17 и 18.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 17 – Распределение филиалов

Данных в таблице мало, всего 20 записей. На обучение уходит 85%, то есть 17 записей. Их не хватит, чтобы обучить алгоритм.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 18 – Таблица с классификацией

Изображение выглядит как диаграмма

Автоматически созданное описание

Обучение происходит, но с довольно высоким количеством ошибок.

Попробуем использовать более точный алгоритм Random Forest. На рисунке 18 изображена информация о модели, а именно об ошибках.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 17 – Результат работы алгоритма Random Forest

Алгоритм выделил 3 ошибки, процент ошибки 11.76%. При нескольких запусках процент ошибки вырастал до 42%. Следовательно, предсказывать на таком количестве данных лучше не стоит.

**Вывод:** В данной работе были получены знания об алгоритмах классификации и кластеризации данных, некоторых функциях языка R, осуществляющих этот вид анализа. Помимо этого, были получены навыки о визуализации результатов работы функций кластерного анализа и классификаторов, интерпретирования полученных результатов, а также выполнена классификацию на основе формулы Байеса и деревьев решений.

**ПРИЛОЖЕНИЕ А**

**Листинг программы**

#-------------------------------------------------------

# Часть 1.

# чтение данных

branch.01 <- read.table("C:/Users/Дианочка/Desktop/Обработка больших данных/6LW/Филиалы.dat", header = TRUE,";", dec=',', fileEncoding="CP1251")

# нормализация данных

branch.02 <-branch.01[,-c(1,4,5,7,8)]

branch.02[,1:4]<-sapply(branch.02[,1:4],as.numeric)

branch.01[,c(2,3,6,9,10)]<-sapply(branch.01[,c(2,3,6,9,10)],as.numeric)

typeof(branch.02[2,4])

branch.02 <- scale(branch.02[,1:5], center = TRUE, scale = TRUE)

maxs <- apply(branch.02[,1:5], 2, max)

mins <- apply(branch.02[,1:5], 2, min)

branch.02 <- scale(branch.02, center = mins, scale = maxs - mins)

branch.03<-data.frame(branch.01[,c(1,4,5,7,8)],branch.02)

# создание матрицы попарных расстояний (по умолчанию - Евклидово расстояние)

dist.branch <- dist(branch.03 [,6:10])

# проводим кластерный анализ,

# результаты записываются в список clust.branch

# hclust ожидает матрицу расстояния, а не исходные данные.

clust.branch<- hclust(dist.branch, "ward.D")

# просмотр краткой сводки результатов анализа

clust.branch

# построение дендрограммы

plot(clust.branch, labels = branch.01$НАЗВАНИЕ)

rect.hclust(clust.branch, k = 4, border="red")

abline(h = 1.3, col = "blue", lwd='3') # h - horizontal line, col - color

k = 4

# построение диаграммы "Каменная осыпь"

plot(1:19, clust.branch$height, type='b',xlab="Номер компоненты",ylab = "Собственное значение")

# разделим Филиалы на 4 кластера

# вектор groups содержит номер кластера, в который попал классифицируемый объект

groups <- cutree(clust.branch, k)

# Разрезает дерево, например, полученное в результате hclust, на несколько групп

# путем указания желаемого количества групп или высоты среза.

groups

dend <- as.dendrogram(clust.branch)

if(!require(dendextend)) install.packages("dendextend");

library(dendextend)

dend <- color\_branches(dend, k)

plot(dend)

# вывод филиалов соответсвенно сформированным кластерам

branch.01[groups==1, 1]

branch.01[groups==2, 1]

branch.01[groups==3, 1]

branch.01[groups==4, 1]

# для каждой группы определяем средние значения характеристик и строим датафрейм

# в 1-ом кластере

g1<-colMeans(branch.01[groups==1, c(2,3,6,9,10)])

# во 2-ом кластере

g2<-colMeans(branch.01[groups==2, c(2,3,6,9,10)])

# в 3-ем кластере

g3<-colMeans(branch.01[groups==3, c(2,3,6,9,10)])

# во 4-ом кластере

g4<-colMeans(branch.01[groups==4, c(2,3,6,9,10)])

# для общего графика

# в 1-ом кластере

g11<-colMeans(branch.03[groups==1,6:10])

# во 2-ом кластере

g12<-colMeans(branch.03[groups==2, 6:10])

# в 3-ем кластере

g13<-colMeans(branch.03[groups==3, 6:10])

# в 4-ом кластере

g14<-colMeans(branch.03[groups==4, 6:10])

# делаем дата фрейм из векторов групп кластеров

df<-data.frame(g1,g2,g3,g4); df

df1<-t(df); df1

df11<-data.frame(g11,g12,g13,g14); df11

df12<-t(df11); df12

barplot(as.matrix(df12), col=c("magenta","red","green","pink"))

legend("topright",

legend = rownames(df12),

fill = c("magenta","red","green","pink"),

border = "black",

cex=0.8)

barplot(df1[,1], ylim=range(pretty(c(0,max(df1[,1])))),

main="Площадь",

col=c("magenta","red","green","pink"))

legend("topright",

legend = rownames(df1),

fill = c("magenta","red","green","pink"),

border = "black",

cex=0.8)

barplot(df1[,2], ylim=range(pretty(c(0,max(df1[,2])))),

main="ПРОХОДИМОСТЬ",

col=c("magenta","red","green","pink"),legend=rownames(df1))

barplot(df1[,3], ylim=range(pretty(c(0,max(df1[,3])))),

main="МЕТРО",

col=c("magenta","red","green","pink"),legend=rownames(df1))

barplot(df1[,4], ylim=range(pretty(c(0,max(df1[,4])))),

main="ЦЕНЫ",

col=c("magenta","red","green","pink"),legend=rownames(df1))

barplot(df1[,5], ylim=range(pretty(c(0,max(df1[,5])))),

main="ПРОДАЖИ",

col=c("magenta","red","green","pink"),legend=rownames(df1))

library (lattice)

library("scatterplot3d")

df <-read.table("C:/Users/Дианочка/Desktop/Обработка больших данных/6LW/Филиалы.dat", header = TRUE,";", dec=',', fileEncoding="CP1251")

df[,c(2,3,6,9,10)]<-sapply(df[,c(2,3,6,9,10)],as.numeric)

my\_data <- df

my\_data["Group"]<-groups

# вывод графика рассеяния с минимальным количеством параметров с выделением имени

xyplot(ПРОДАЖИ ~ ПРОХОДИМ,group = Group, data = my\_data,auto.key = TRUE,pch = 20,cex = 1.5)

# боксплот, отражающий характеристики классов типов

boxplot(ПРОДАЖИ ~ Group, data =my\_data, xlab="КЛАСТЕРЫ",ylab = "ПРОДАЖИ", frame = FALSE, col=c("magenta","red","green","pink"))

# построение трехмерного графика наших классов

cloud(ПРОДАЖИ~ЦЕНЫ\*МЕТРО, group = Group, data = my\_data, auto.key = TRUE,pch = 20,cex = 1.5)

#-------------------------------------------------------

# Часть 2

install.packages("vctrs")

library(klaR)

my\_data$Group<- c(as.factor(groups))

naive\_df <- NaiveBayes(my\_data$Group ~ ., data = my\_data)

naive\_df$tables

naive\_df$tables$ПРОДАЖИ

naive\_df

# построение графиков по Бпйесу

opar=par()

opar

layout(matrix(c(1,2),2,2, byrow = TRUE))

plot(naive\_df,lwd = 2, legendplot=TRUE,cex=0.1)

#восстановление

par=opar

# классификация Decision Tree

set.seed(1234)

ind <- sample(2, nrow(my\_data), replace=TRUE, prob=c(0.7, 0.3))

trainData <- my\_data[ind==1,]

testData <- my\_data[ind==2,]

nrow(trainData)

nrow(testData)

nrow(my\_data)

library(party)

myFormula <- Group ~ ПРОХОДИМ + ЦЕНЫ + ПРОДАЖИ

df\_ctree <- ctree(myFormula, data=trainData)

df\_ctree

table(predict(df\_ctree),trainData$Group)

predict(df\_ctree)

plot(predict(df\_ctree))

plot(df\_ctree)

# алгоритм Random Forest

library(randomForest)

rf <- randomForest(Group ~ .,data=trainData, ntree=100, proximity=TRUE)

table(predict(rf), trainData$Group)

print(rf)