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**

**Отчёт лабораторная №5.1**

**Дисциплина: Обработка больших данных**

Работу выполнил \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Гаранина Л.В.

Направление подготовки 02.03.02 Фундаментальная информатика и

информационные технологии

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Яхонтов А.А.

Краснодар

2025

**Тема:** Задачи классификация и кластеризация.

**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.

**Задание:**

1. Выполнить дескриптивный анализ данных.
2. Оценить оптимальное число кластеров, для этого построить диаграммы "Метод силуэта", "Метод локтя", "Статистика разрыва" и "Алгоритм консенсуса".
3. Выполнить иерархическую кластеризацию набора данных, построив дендрограмму. Подробно обосновать выбор числа групп.
4. Построить диаграмму со столбчатыми диаграммами и боксплотами групп. Провести сравнительный анализ полученных групп.
5. Выполнить кластеризацию датасета по k-means.
6. Выполнить построение scatterplot с помощью функций plot или pairs.
7. Построить трехмерную кластеризацию по scatterplot3d.

Вариант 10: Covid\_Russia.

Фрагмент датасета показан на рисунке 1.



Рисунок 1 - Датасет

**Ход работы**:

Выполним дескриптивный анализ данных о Covid в России. Для этого найдем среднее, медиану, моду, дисперсию, среднее квадратическое отклонение, минимальные и максимальные значения, изображенные на рисунке 2.

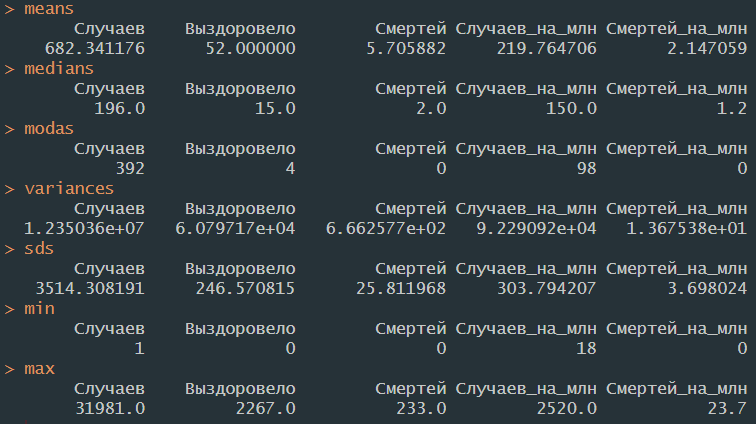


Рисунок 2 – Дескриптивный анализ данных

По каждому критерию датасета были построены боксплоты и гистограммы, построенные на рисунках 3-4.

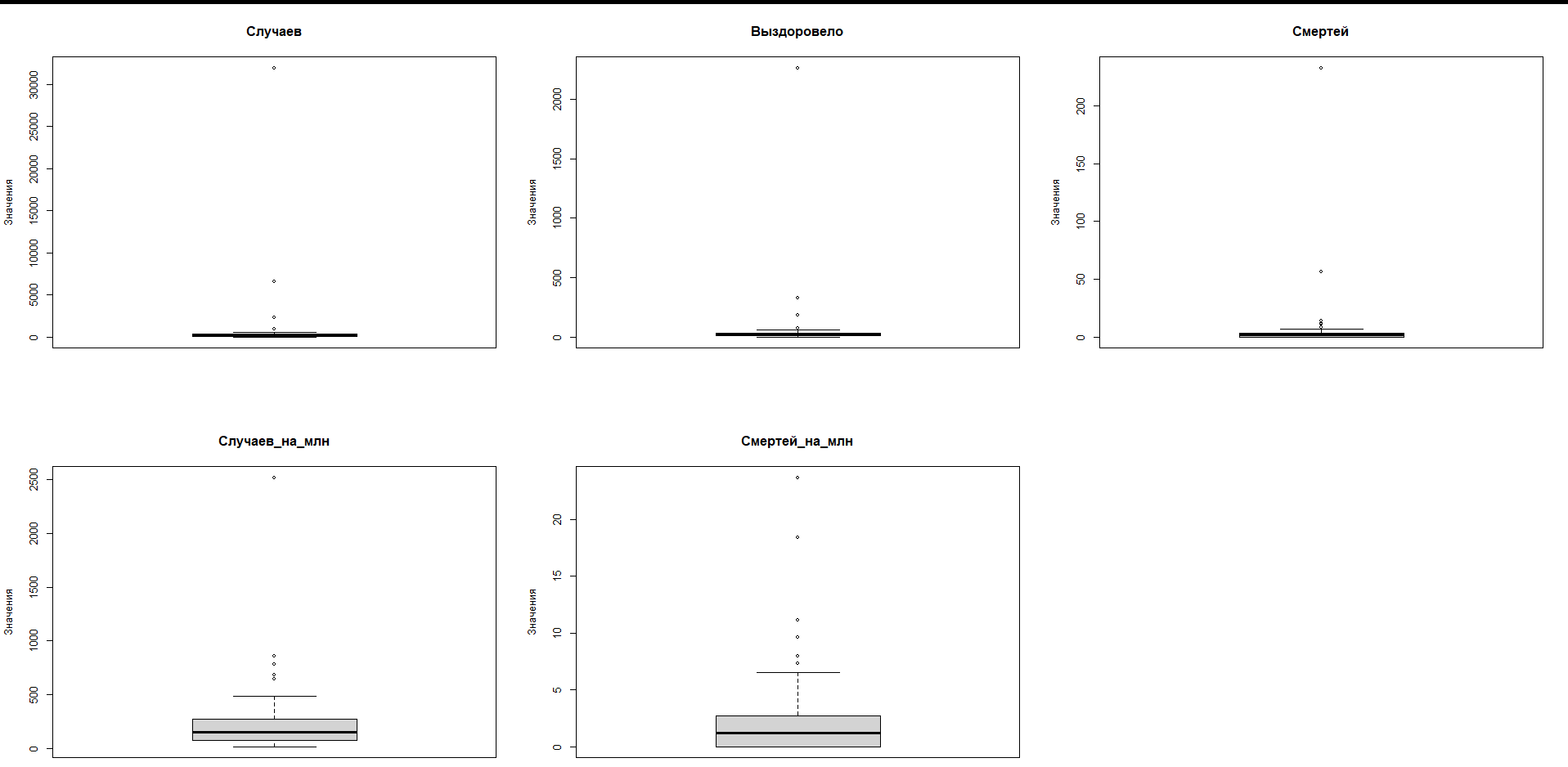


Рисунок 3 – Боксплоты

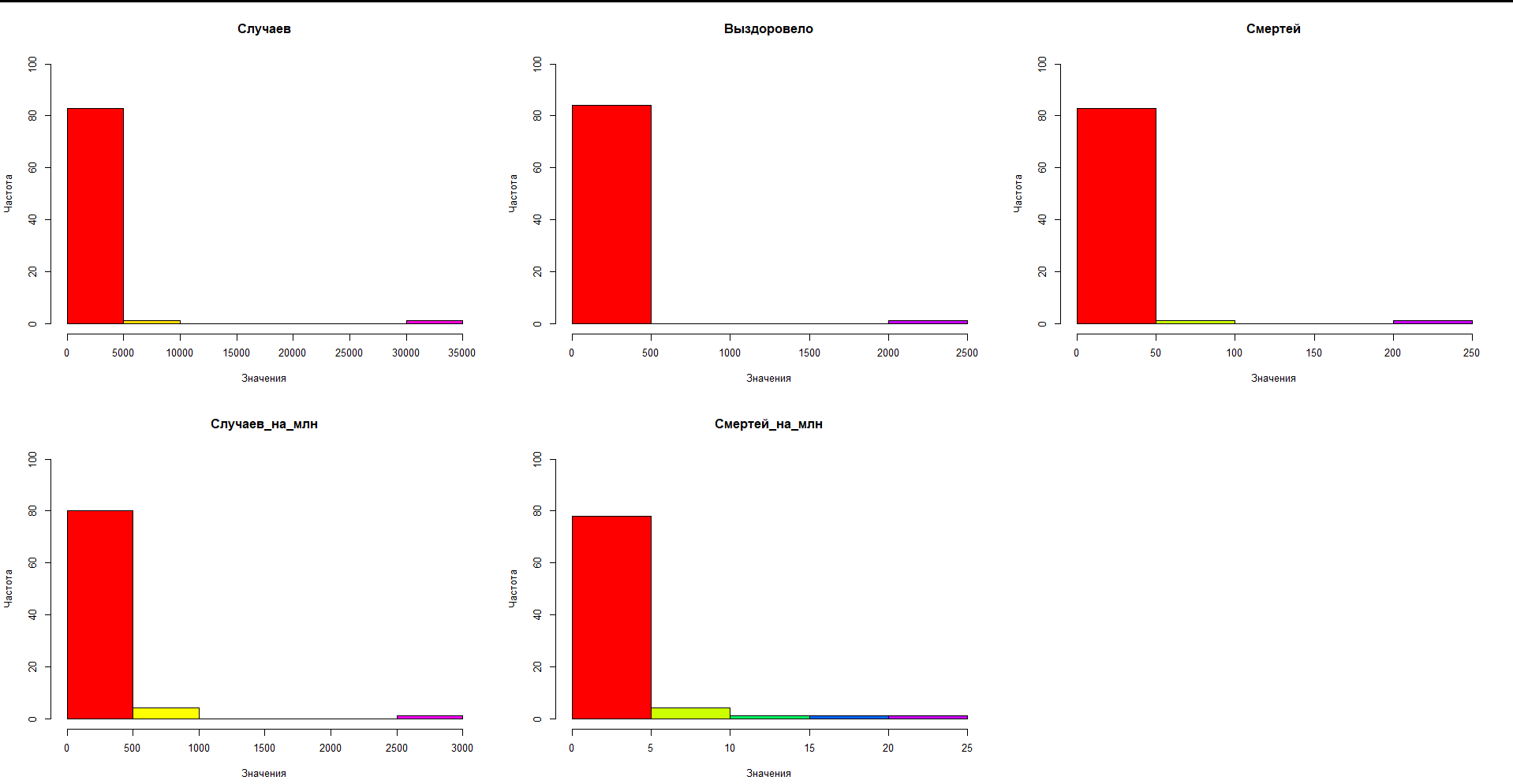


Рисунок 4 - Гистограммы

Все данные в датафрейме необходимо стандартизировать и нормировать. Это нужно для устранения различий в масштабе признаков и обеспечения корректной интерпретации результатов анализа. Для этого используем библиотеку BBmisc. Для стандартизации используем scale(), которая в каждом столбце делает среднее значение 0, а стандартное отклонение 1. Для нормализации используется функция normalize() с методом range, который приводит данные в заданный диапазон от 0 до 1.

Далее построим диаграммы: "Метод локтя", "Метод силуэта", "Статистика разрыва" и "Алгоритм консенсуса", изображенные на рисунках 5-8.

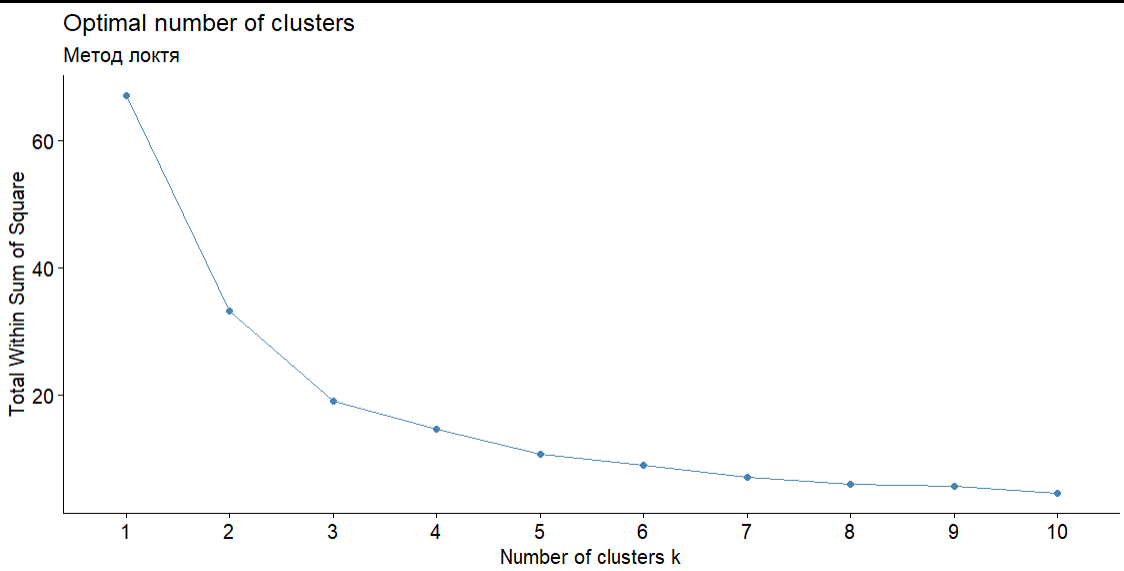


Рисунок 5 – Метод локтя

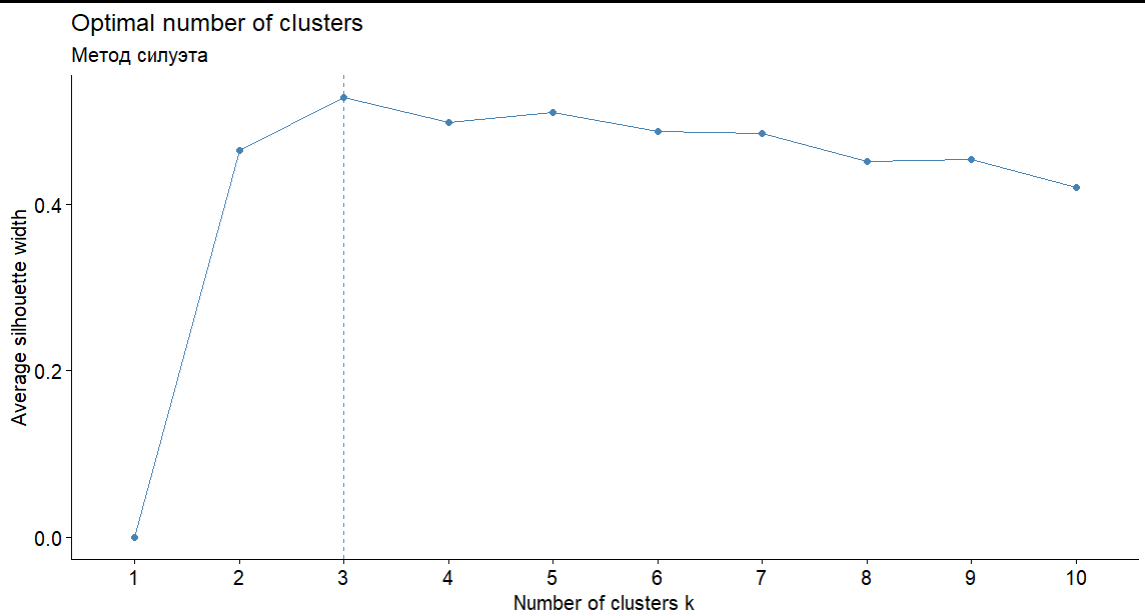


Рисунок 6 – Метод силуэта

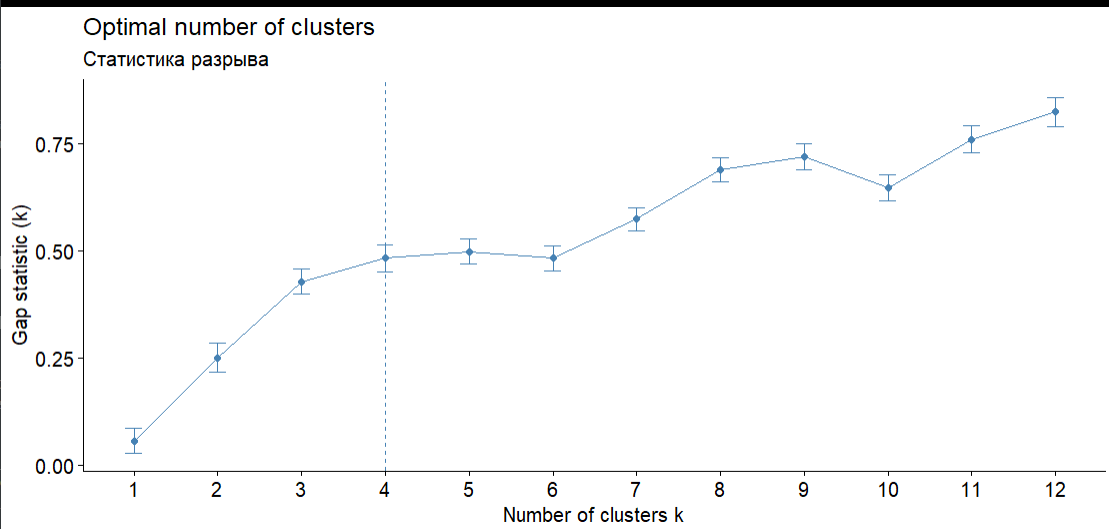


Рисунок 7 – Статистика разрыва

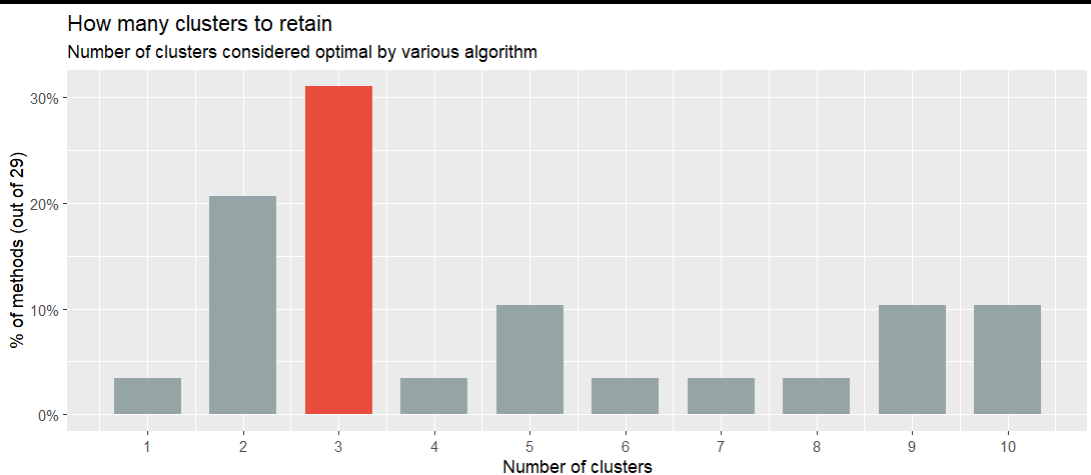


Рисунок 8 – Алгоритм консенсуса

По методу локтя оптимальное число кластеров около 3-4 штук. По методу силуэта и алгоритму консенсуса равно 3, а по статистике разрыва 4. Поэтому будем рассматривать и сравнивать данные по 3 и 4 кластерам.

Выполним иерархическую кластеризацию набора данных, построив две дендрограммы. Результаты построение представлены в виде деревьев на рисунках 9-10.



Рисунок 9 – Дендрограмма (3 группы)



Рисунок 10 – Дендрограмма (4 группы)

В 1 дендрограмме первый кластер включает регионы с относительно схожими умеренными значениями всех показателей: случаи заражения и количество выздоровевших стремятся к 1. Второй кластер включает разнообразные регионы, от северных до центральных, имеют разброс по конкретным показателям, но достаточно близки. Критерий смертей на миллион равен 1, низкая смертность и мало заболевших. В третий кластер входят регионы с наибольшими значениями заражений или смертности, учитывая плотность населения. Здесь показатель случаев на миллион почти везде равен 1.

Как можем видеть, при разбиении на 4 группы разбивается на 2 кластера самый большой из кластеров первой тройки: один из них содержит сильно сгруппированные регионы с близкими значениями по случаям заражения, числу выздоровевших и смертности, другой регионы со схожим средне-низким уровнем показателей, где случаи на миллион равны 0.

Разбиение на 3 кластера дает широкую картину: низкий, средний и высокий уровень заболеваемости/смертности; разбиение на 4 кластера позволяет выделить более тонкие особенности.

Кластеры помогают выявить регионы, нуждающиеся в разном подходе к борьбе с пандемией.

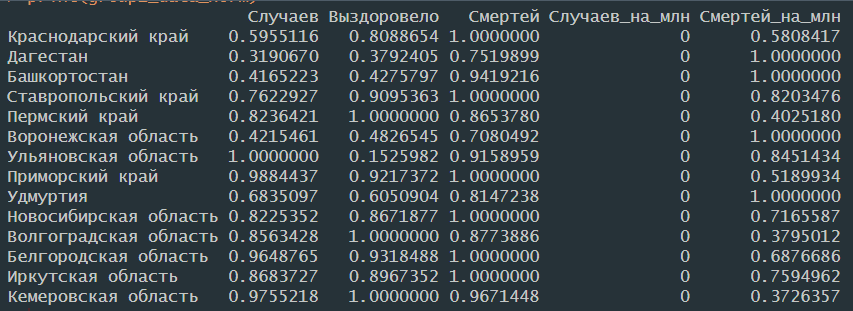


Рисунок 11 – Пример кластера при разбиении на 4 группы

Построим диаграмму со столбчатыми диаграммами для трех кластеров, показанную на рисунке 12, и боксплотами групп, изображенную на рисунке 13.

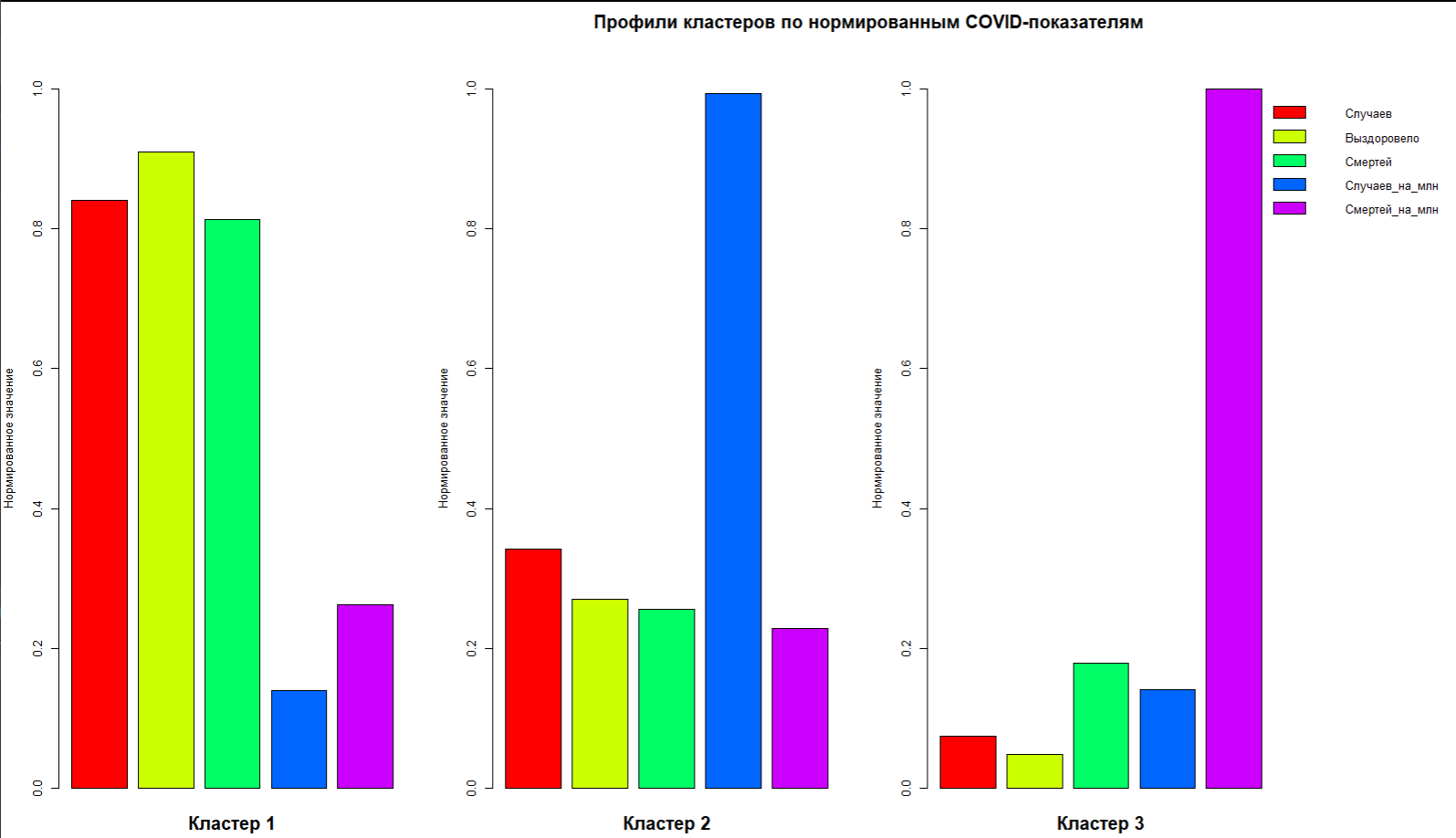


Рисунок 12 – Столбчатые диаграммы для 3 кластеров

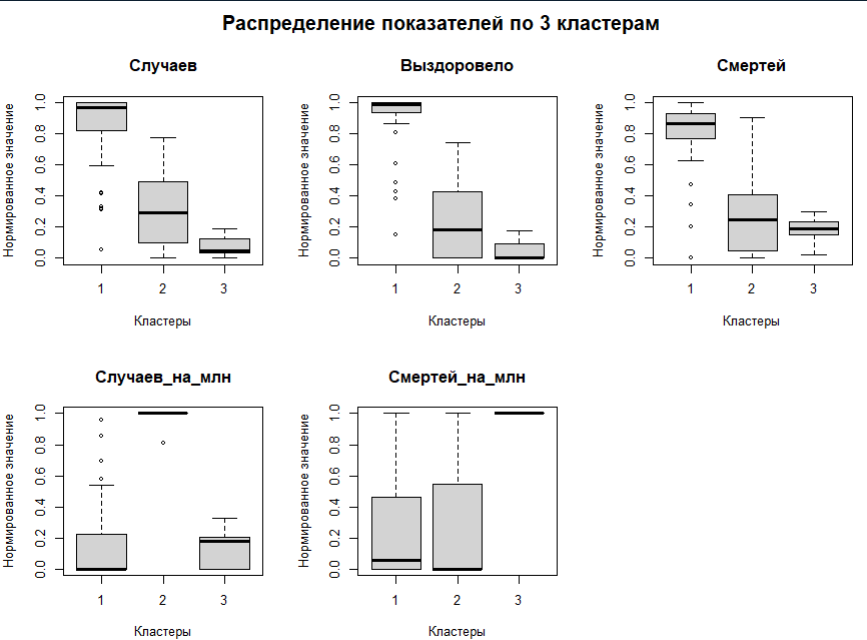


Рисунок 13 – Боксплоты для 3 кластеров

По столбчатой диаграмме видно, что почти все показатели отличаются друг от друга по сравнению с остальными кластерами. Так же и на боксплоте, где разбросы значений сильно отличаются.

Построим те же самые диаграммы, но для 4-х кластеров, показанные на рисунках 14-15.

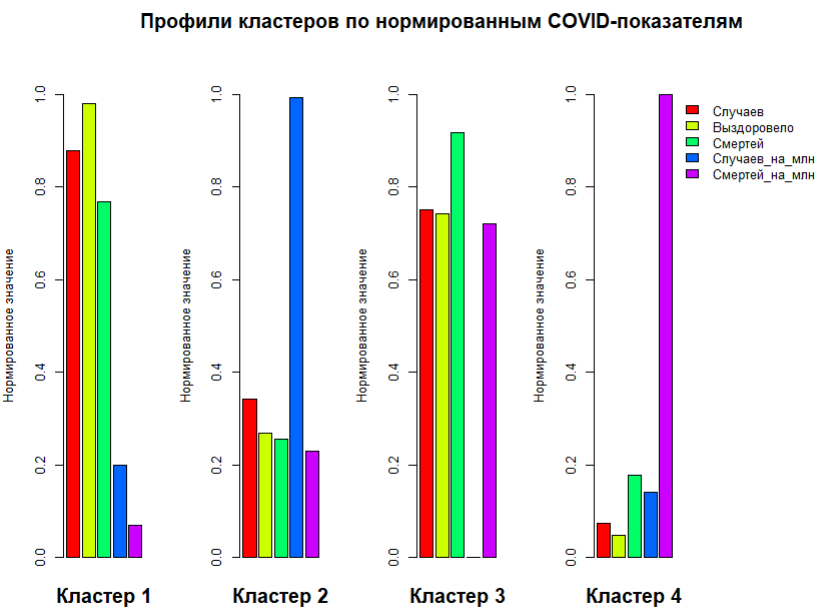


Рисунок 14 – Столбчатые диаграммы для 4 кластеров

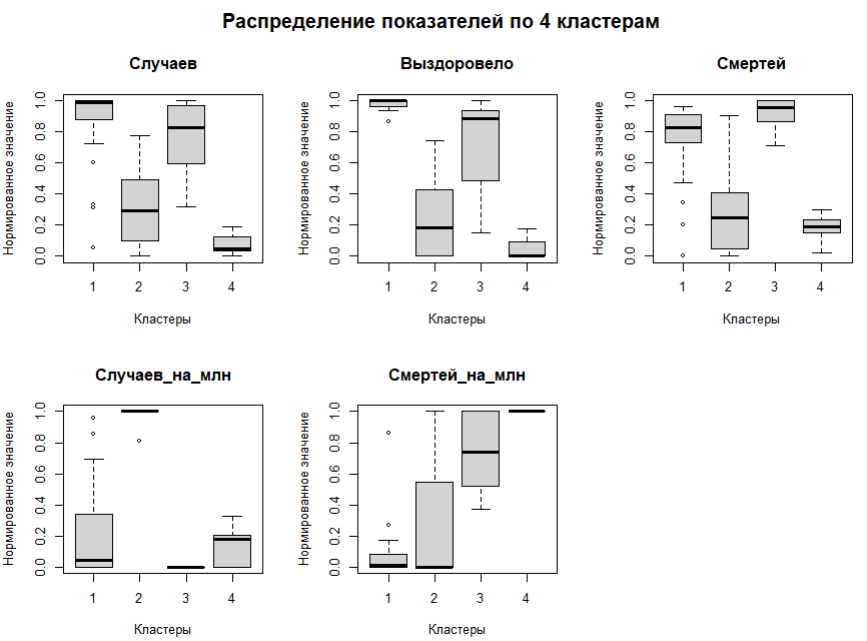


Рисунок 15 – Боксплоты для 4 кластеров

На диаграммах видно, что здесь похожая ситуация, как и на 3 кластерах – большинство параметров сильно отличаются друг от друга.

Далее выполним кластеризацию датасета по k-means.

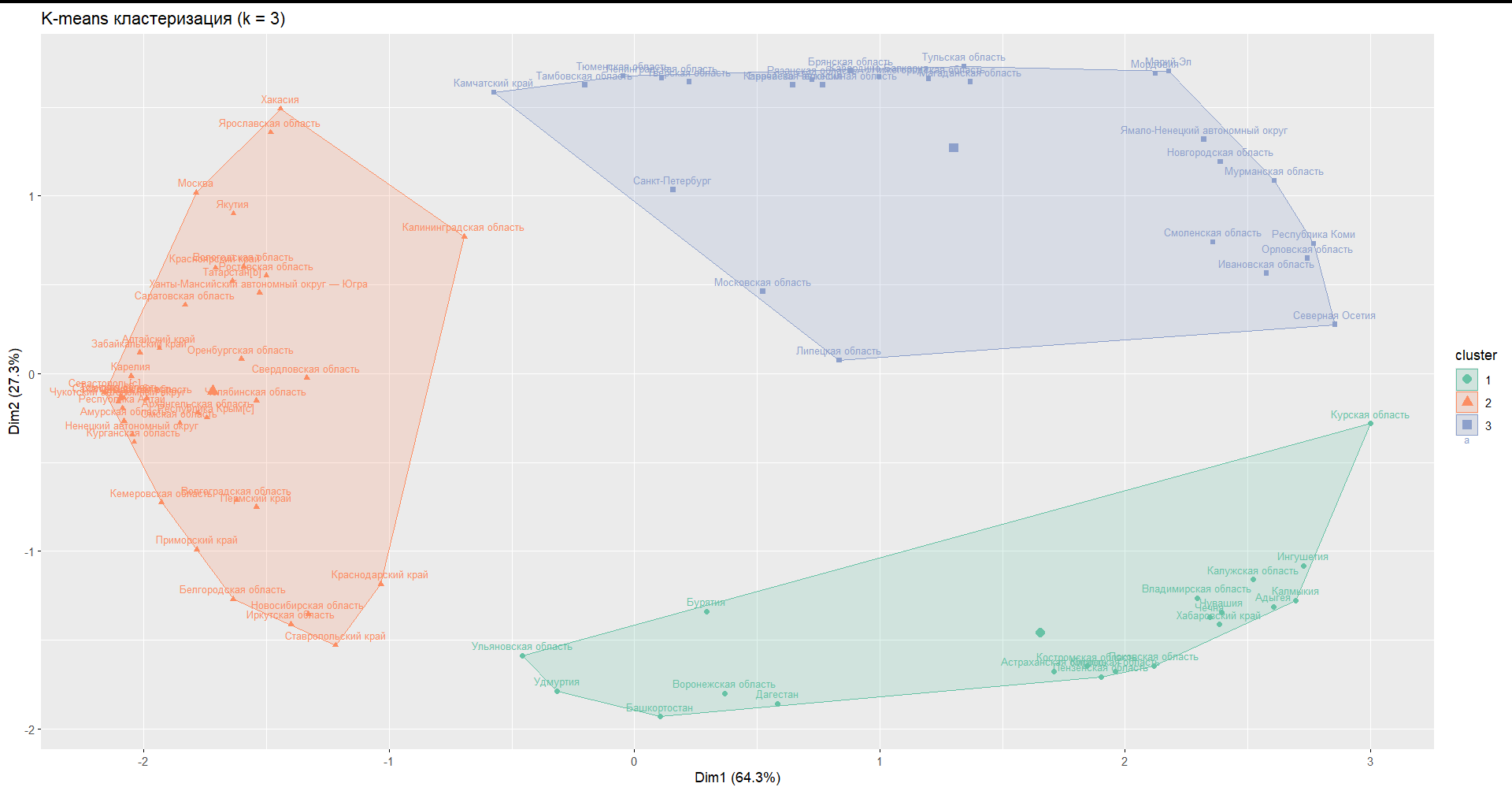


Рисунок 16 – Кластеризация на 3 кластерах

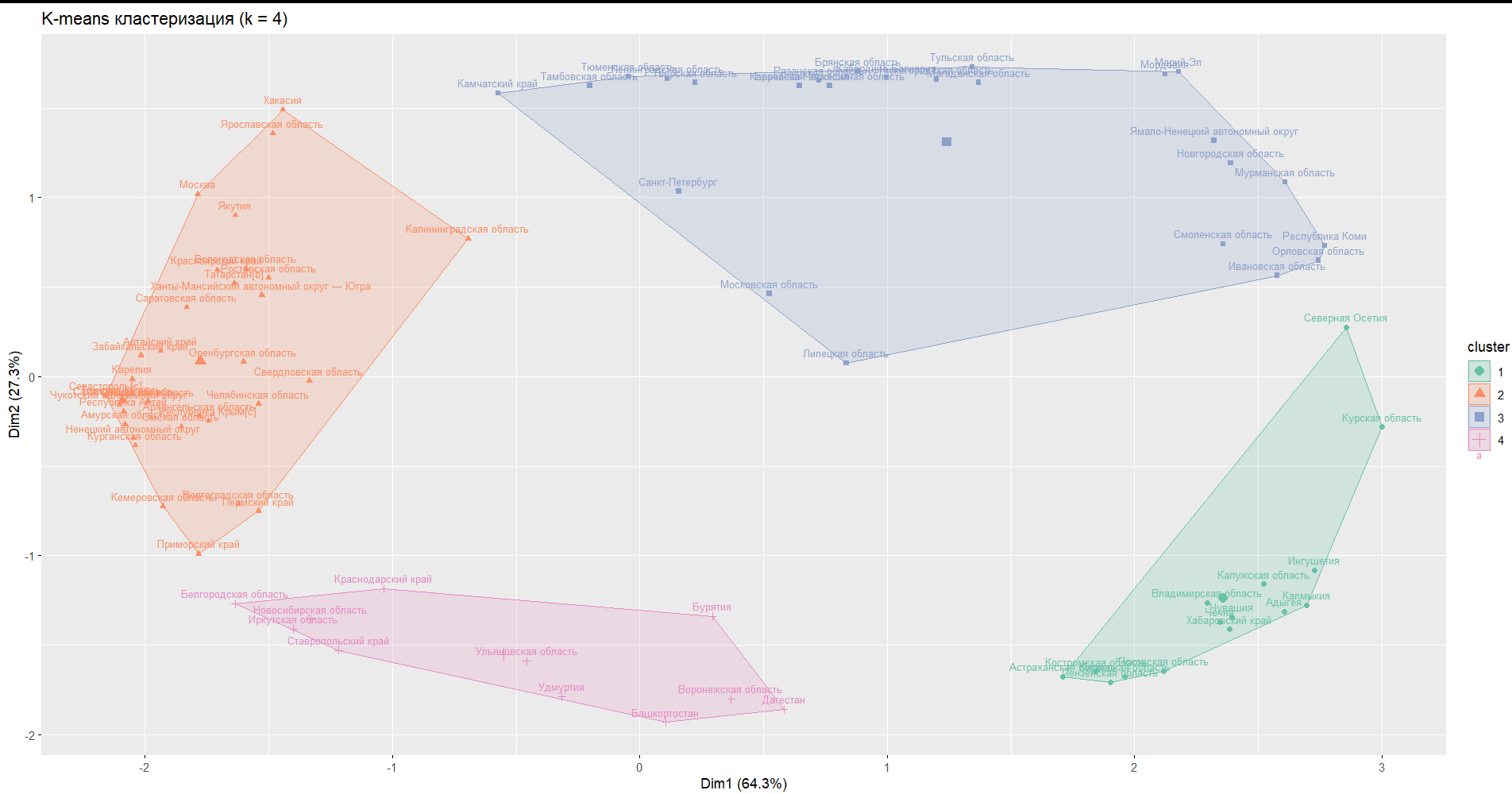


Рисунок 17 – Кластеризация на 4 кластерах

Можно увидеть, что на изображениях сохранилось 91,6% информации и при разбиениях нигде не прослеживается пересечение данных, каждый кластер четко видно. При разбиении на 4 кластера один их предыдущих разбился на 2 – значит, что внутри прежнего кластера были немного разные по характеристикам регионы.

Выполним построение scatterplot с помощью функции pairs, где каждая ячейка это отдельный график, сравнивающий 2 переменные. Сами переменные отображаются по главной диагонали. Графики с 3-4 кластерами изображены на рисунках 18-19.



Рисунок 18 – Диаграмма рассеивания для 3 кластеров

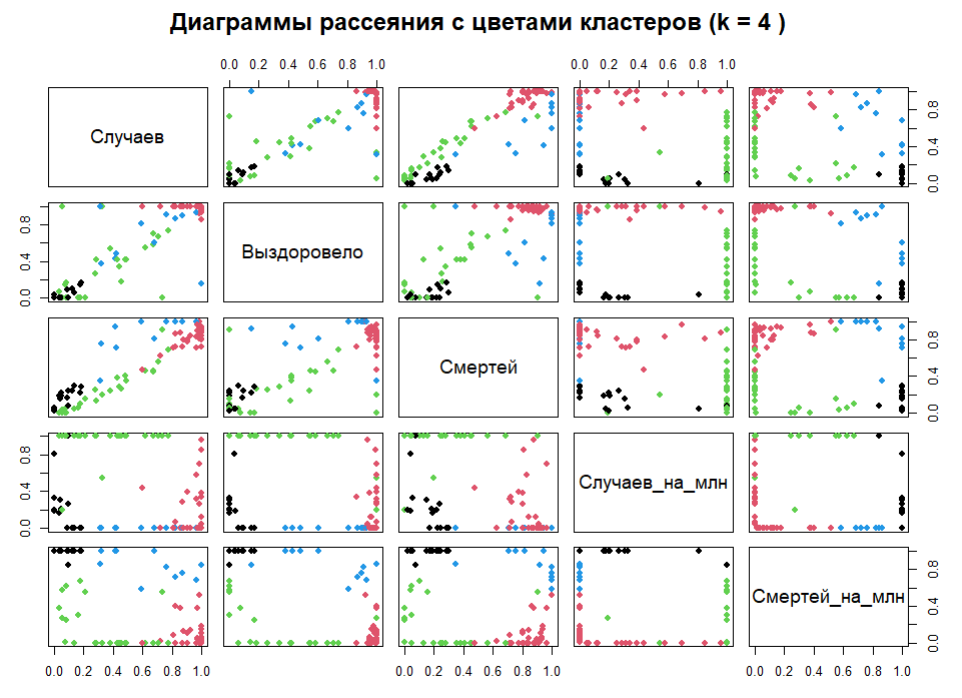


Рисунок 19 – Диаграмма рассеивания для 4 кластеров

Можно сказать, что по показателям случаев и выздоровления идет сильная положительная корреляция, линейная зависимость, то есть, чем больше люди заболевали, тем больше выздоравливали. График смертей и случаев также имеет положительную корреляцию, но более рассеянную. По остальным в основном идет разброс.

Далее выполним построения трехмерной кластеризации по scatterplot для 3 и 4 кластеров, изображенные на рисунках 20-21.

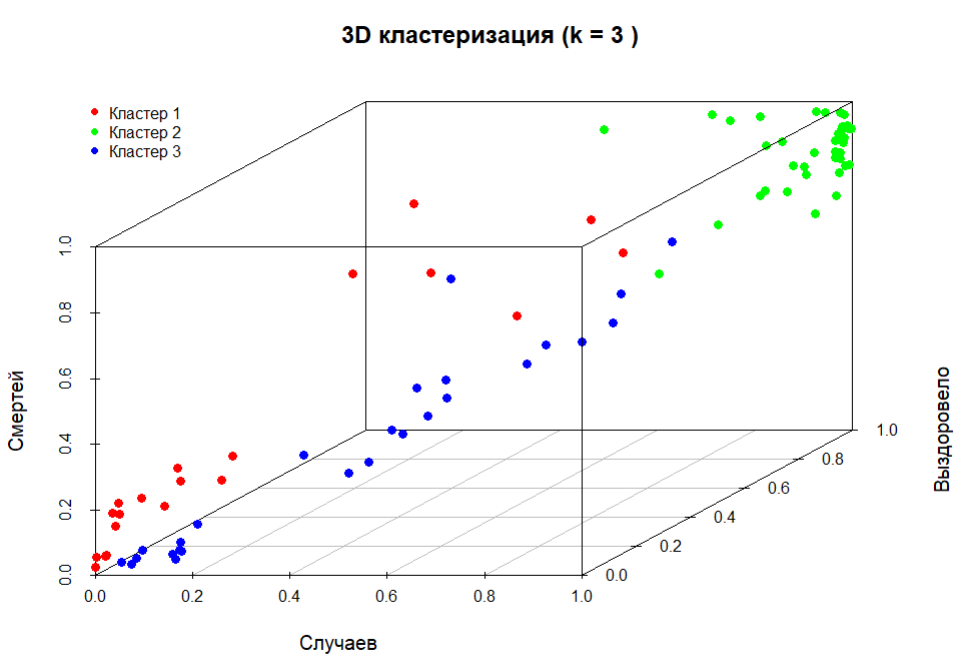


Рисунок 20 – Трехмерная кластеризация для 3 кластеров

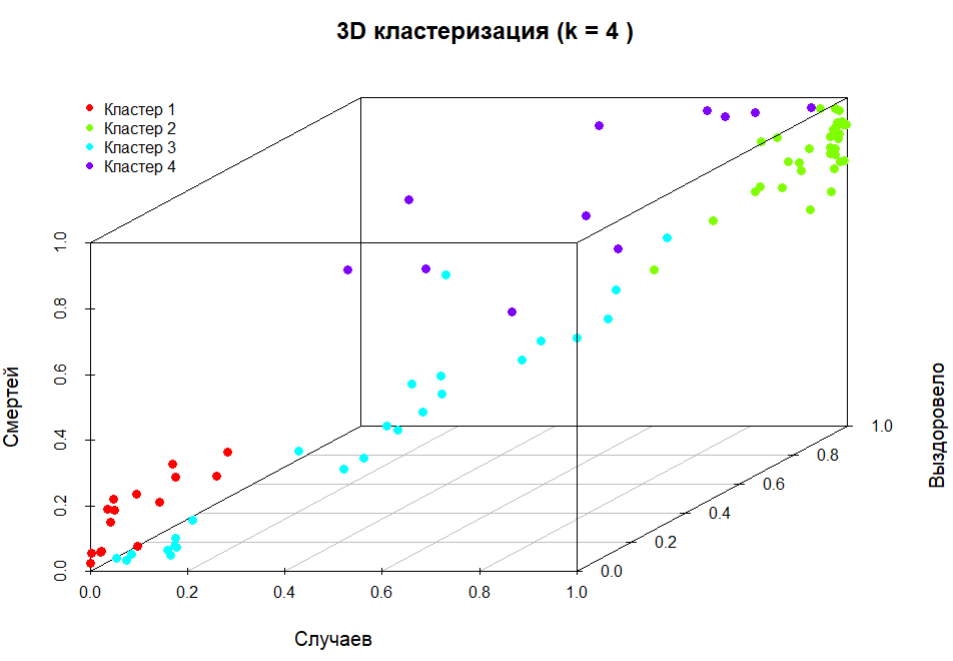


Рисунок 21 – Трехмерная кластеризация для 4 кластеров

Как можем видеть на рисунке 20, зеленый кластер располагается в правом верхнем углу, где все показатели высокие, красный кластер – имеет противоположные значения, синий расположен ближе к центру, значения умеренные.

На рисунке 21 красный и зеленый кластеры имеют такие же показатели, синий кластер имеет высокие показатели смертности, средне-высокие показатели случаев и разброс по "Выздоровело", голубой кластер имеет низко-средние значения смертей и "Выздоровело" и разное количество случаев заболевания.

На основании визуального анализа, структуры данных и интерпретируемости кластеров, кластеризация на 4 группы даёт более чёткое, детализированное и интерпретируемое разбиение. Она лучше отражает различия между группами стран/регионов по эпидемиологическим характеристикам.

**Вывод:** проделав лабораторную работу, были закреплены знания об алгоритмах классификации и кластеризации данных, освоены функции языка R для их реализации, а также приобретены навыки визуализации и интерпретации результатов анализа.

**Листинг программы:**

library(readxl)

setwd("C:/Users/Людмила/Desktop/big data/rstudioLAB")

df <- read\_excel("Covid\_Russia.xlsx", sheet = 1)

names <- df$Город

df <- df[-1]

row.names(df) <- names

# 2 Дескриптивный анализ

summary(df)

means = apply(df, 2, mean)

medians = apply(df, 2, median)

getmode <- function(v) {

uniqv <- unique(v)

uniqv[which.max(tabulate(match(v, uniqv)))]

}

modas = apply(df, 2, getmode)

#дисперсия, стандартное отклонения, мин и макс

variances = apply(df, 2, var)

sds = apply(df, 2, sd)

min = apply(df, 2, min)

max = apply(df, 2, max)

par(mfrow = c(2, 3))

for (i in 1:ncol(df)) {

boxplot(df[, i],

main = colnames(df)[i],

ylab = "Значения")

}

for (i in 1:ncol(df)) {

h <- hist(as.numeric(df[[i]]), plot = FALSE)

bar\_colors <- rainbow(length(h$counts))

plot(h,

main = colnames(df)[i],

xlab = "Значения",

ylab = "Частота",

col = bar\_colors,

ylim = c(0, nrow(df)+15))

}

# 3 Оценить оптимальное число кластеров, для этого построить диаграмму "Метод

# силуэта", “Метод локтя”, "Статистику разрыва" и Алгоритм консенсуса

library(BBmisc)

df\_scaled = scale(df) # Стандартизация данных

df\_sc\_norm = normalize(df\_scaled, method = "range", range = c(0, 1)) # Нормализация в диапазоне [0, 1]

library (factoextra)

library (cluster)

library(parameters)

# Метод силуэта

fviz\_nbclust(df\_sc\_norm, kmeans, method = "silhouette") + labs(subtitle = "Метод силуэта")

# Метод локтя

fviz\_nbclust(df\_sc\_norm, kmeans, method = "wss") + labs(subtitle = "Метод локтя")

# Статистика разрыва

gap\_stat <- clusGap(df\_sc\_norm, FUN = kmeans, K.max = 12)

fviz\_gap\_stat(gap\_stat) + labs(subtitle = "Статистика разрыва")

# Алгоритм консенсуса

n\_clust <- n\_clusters(data.frame(df\_sc\_norm),

package = c("easystats", "NbClust", "mclust"),

standardize = FALSE)

plot(n\_clust)

# 4. Выполнить иерархическую кластеризацию вашего набора данных, построив дендрограмму

dist.datas=dist(df\_sc\_norm)

clust.datas=hclust(dist.datas,'ward.D')

plot(clust.datas,cex=0.5, ann = FALSE)

title(main = "Дендрограмма кластеризации (3 группы)")

rect.hclust(clust.datas,k=3,border="red")

plot(clust.datas,cex=0.5, ann = FALSE)

title(main = "Дендрограмма кластеризации (4 группы)")

rect.hclust(clust.datas,k=4,border="blue")

cluster\_cut3 <- cutree(clust.datas, k = 3)

group1\_indices <- which(cluster\_cut3 == 1)

group1\_data\_norm <- df\_sc\_norm[group1\_indices, ]

print(group1\_data\_norm)

# 5. Построить диаграмму со столбчатыми диаграммами и боксплотами групп.

plot\_cluster\_profiles <- function(clust\_obj, df\_norm, k) {

cluster\_cut <- cutree(clust\_obj, k = k)

cluster\_means <- t(sapply(1:k, function(i) {

colMeans(df\_norm[cluster\_cut == i, ])

}))

param\_colors <- rainbow(ncol(df\_norm))

par(mfrow = c(1, k + 1), mar = c(6, 4, 4, 2), oma = c(0, 0, 2, 0), xpd = NA)

for (i in 1:k) {

barplot(cluster\_means[i, ],

col = param\_colors,

ylab = "Нормированное значение",

names.arg = FALSE,

ylim = c(0, 1))

mtext(paste("Кластер", i), side = 1, line = 2, font = 2)

}

plot.new()

legend("topright",

legend = colnames(df\_norm),

fill = param\_colors,

bty = "n")

mtext("Профили кластеров по нормированным COVID-показателям", outer = TRUE, cex = 1, font = 2, adj = 0.5)

}

plot\_cluster\_profiles(clust.datas, df\_sc\_norm, k = 3)

plot\_cluster\_profiles(clust.datas, df\_sc\_norm, k = 4)

plot\_cluster\_boxplots <- function(clust\_obj, df\_norm, k) {

cluster\_cut <- cutree(clust\_obj, k = k)

par(mfrow = c(2, 3), mar = c(5, 4, 4, 2), oma = c(0, 0, 2, 0))

for (i in 1:ncol(df\_norm)) {

boxplot(df\_norm[, i] ~ cluster\_cut,

main = colnames(df\_norm)[i],

ylab = "Нормированное значение",

xlab = "Кластеры")

}

mtext(paste("Распределение показателей по", k, "кластерам"), outer = TRUE, cex = 1, font = 2)

}

plot\_cluster\_boxplots(clust.datas, df\_sc\_norm, k = 3)

plot\_cluster\_boxplots(clust.datas, df\_sc\_norm, k = 4)

# 6. Выполнить кластеризацию датасета по k-means

set.seed(123)

kmeans\_3 <- kmeans(df\_sc\_norm, 3)

fviz\_cluster(kmeans\_3, df\_sc\_norm, palette="Set2", labelsize = 8) +

ggtitle("K-means кластеризация (k = 3)")

split(row.names(df), kmeans\_3$cluster)

# Для 4 кластеров

set.seed(123)

kmeans\_4 <- kmeans(df\_sc\_norm, 4)

fviz\_cluster(kmeans\_4, df\_sc\_norm, palette="Set2", labelsize = 8) +

ggtitle("K-means кластеризация (k = 4)")

# 7. Выполнить построение scatterplot с помощью функций plot или pairs.

plot\_cluster\_pairs <- function(kmeans, k) {

cluster\_cut <- kmeans$cluster

pairs(df\_sc\_norm,

col = cluster\_cut,

pch = 19,

main = paste("Диаграммы рассеяния с цветами кластеров (k =", k, ")"))

}

plot\_cluster\_pairs(kmeans\_3, 3)

plot\_cluster\_pairs(kmeans\_4, 4)

# 8. Построить трехмерную кластеризацию по scatterplot3d.

library(scatterplot3d)

plot\_3d\_clusters <- function(df\_norm, kmeans\_result, k) {

x <- df\_norm[, 1]

y <- df\_norm[, 2]

z <- df\_norm[, 3]

cluster\_palette <- rainbow(k)

point\_colors <- cluster\_palette[kmeans\_result$cluster]

scatterplot3d(x, y, z,

color = point\_colors,

pch = 19,

main = paste("3D кластеризация (k =", k, ")"),

xlab = colnames(df\_norm)[1],

ylab = colnames(df\_norm)[2],

zlab = colnames(df\_norm)[3])

legend("topleft",

legend = paste("Кластер", 1:k),

col = cluster\_palette,

pch = 19,

cex = 0.8,

inset = 0.02,

box.lty = 0)

}

plot\_3d\_clusters(df\_sc\_norm, kmeans\_3, k = 3)

plot\_3d\_clusters(df\_sc\_norm, kmeans\_4, k = 4)