



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

ФАКУЛЬТЕТ ИУК «Информатика и управление»

КАФЕДРА ИУК4 «Программная инженерия»

Лабораторная работа №4

Дискриминантный анализ

ДИСЦИПЛИНА: «Методы машинного обучения»

Выполнил: студент гр. ИУК5-72Б

(Подпись)

Ли Р. В.

(Ф.И.О.)

Проверил:

(Подпись)

(Ф.И.О.)

Дата сдачи (защиты):

Результаты сдачи (защиты):

- Балльная оценка:

- Оценка:

Цель работы: изучение основных процедур дискриминантного анализа: дискриминации и классификации, построение и определение количества дискриминантных функций и их разделительной способности, нахождение классифицирующих функций с использованием функций Фишера и расстояния Махalanобиса.

Вариант 14

1 4	Пневмоторакс	5 – открытый, закрытый, напряженный, первичный, вторичный	3 - эритроциты, тромбоциты, глюкоза	49
--------	--------------	---	-------------------------------------	----

С помощью средств языка R исследовать выбранное множество пациентов с помощью критерия Шапиро-Уилка, М-критерия Бокса

generateData.R

```

req_pkgs <- c("MASS", "mvnormtest", "biotools")
for(pkg in req_pkgs){
  if(!suppressWarnings(require(pkg, character.only = TRUE))){
    install.packages(pkg, repos = "https://cloud.r-project.org")
    library(pkg, character.only = TRUE)
  }
}

data <- data.frame(
  Kl = factor(c(
    1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
    2,2,2,2,
    3,3,3
  )),
  y6 =
  c(14.4,20.1,24.1,11.1,16.3,40.5,52.7,20.8,14.0,27.0,44.3,47.5,54.0,16.1,57.5,
  37.8,
      55.8,75.0,72.0,70.6,
      24.1,33.2,30.4),
  y9 =
  c(25.1,40.1,32.1,16.9,32.1,64.4,50.0,22.3,3.1,41.7,63.8,50.1,57.0,20.6,74.5,6
  3.0,
      48.0,60.0,65.0,45.0,

```

```

        45.0, 55.0, 44.6),
y10 =
c(0.20, 0.11, 0.17, 0.12, 0.36, 0.21, 0.53, 0.13, 0.18, 0.19, 0.22, 0.29, 0.19, 0.22, 0.49,
0.32,
        2.74, 1.37, 0.70, 1.40,
        0.22, 0.01, 0.09)
)

colnames(data) <- c("group", "y6", "y9", "y10")
data$group <- factor(data$group)

cat("== данные ==\n")
print(head(data, 10))
cat("\nРазмер выборки:", nrow(data), "наблюдений, переменных:", ncol(data)-1,
"\n")

library(mvnormtest)
library(biotools)

cat("\nПроверка многомерной нормальности (mshapiro.test):\n")
print(mshapiro.test(t(as.matrix(data[,2:4]))))

cat("\nПроверка однородности ковариаций (Box's M test):\n")
print(boxM(as.matrix(data[,2:4]), data$group))

library(MASS)

lda.model <- lda(group ~ y6 + y9 + y10, data = data)
cat("\n== Модель LDA ==\n")
print(lda.model)

pred <- predict(lda.model)
table_train <- table(Факт = data$group, Прогноз = pred$class)
cat("\nТаблица классификации (обучающая выборка):\n")
print(table_train)
cat("\nТочность на обучающей выборке:", mean(pred$class == data$group))

lda.cv <- lda(group ~ y6 + y9 + y10, data = data, CV = TRUE)
table_cv <- table(Факт = data$group, Прогноз = lda.cv$class)
cat("\nТаблица классификации (LOOCV):\n")
print(table_cv)
cat("\nОшибка LOOCV =", mean(lda.cv$class != data$group))

Sigma <- cov(data[,2:4])
Sigma.inv <- solve(Sigma)
group.means <- aggregate(data[,2:4], by = list(data$group), mean)
rownames(group.means) <- paste("Group", group.means[,1])
group.means <- group.means[, -1]
cat("\nСредние по группам:\n"); print(group.means)

mahalanobis_dist <- sapply(1:3, function(k){
  apply(data[,2:4], 1, function(x){
    diff <- x - as.numeric(group.means[k,])
    as.numeric(t(diff) %*% Sigma.inv %*% diff)
  })
})

```

```

colnames(mahalanobis_dist) <- paste0("D2_G", 1:3)
cat("\nПервые 6 строк расстояний Махalanобиса:\n")
print(round(head(mahalanobis_dist), 3))

eig <- lda.model$svd^2
lambda <- eig
can_corr <- sqrt(lambda / (1 + lambda))
Lambda <- prod(1 / (1 + lambda))
chi2 <- -(nrow(data) - ((ncol(data)-1 + 3)/2) - 1) * log(Lambda)
df <- (ncol(data)-1) * (3 - 1)
pval <- pchisq(chi2, df = df, lower.tail = FALSE)

cat("\nWilks'  $\Lambda$  =", round(Lambda, 6),
    "\n $\chi^2$  =", round(chi2, 3),
    "\nСтепени свободы =", df,
    "\nр-value =", round(pval, 6),
    "\nКанонические корреляции =", round(can_corr, 3), "\n")

plot(lda.model, col = c("blue", "red", "green"), main = "LDA - Discriminant
Functions")

saveRDS(list(
  lda = lda.model,
  table_train = table_train,
  table_cv = table_cv,
  group_means = group.means,
  Sigma = Sigma,
  mahalanobis = mahalanobis_dist
), "lda_results_var14.rds")

cat("\nРезультаты сохранены в файл lda_results_var14.rds\n")

```

