

Лабораторная работа №6. Моделирование. Дискриминантный анализ

Студент:

Dubovskij Jan

Вариант

N/A

Генерация случайных данных

```
dat_1 = cbind(rnorm(20, 0, 9 / 3), rnorm(20, 0, 9 / 3))
dat_2 = cbind(rnorm(10, 9, 9 / 3), rnorm(10, 9, 9 / 3))
dat = rbind(dat_1, dat_2)
dat
```

```
##           [,1]      [,2]
## [1,] -0.37441662  6.1910101
## [2,] -0.01921261  2.3523424
## [3,]  1.72975208 -1.5304585
## [4,]  0.69307479 -0.9715490
## [5,]  1.47198673 -3.0469955
## [6,] -3.54597902 -2.5453705
## [7,] -0.08157950  4.2980950
## [8,]  3.09559253  2.0465648
## [9,]  0.98714710  4.0429209
## [10,] -1.16948734 -1.9044277
## [11,] -0.10972765  1.1473335
## [12,] -3.36452946 -1.7935724
## [13,] -4.66719505 -2.9639322
## [14,] -1.11219357 -1.9937626
## [15,] -1.09473456  0.5597099
## [16,] -7.17019809  2.3991350
## [17,] -0.75583618 -0.2131011
## [18,] -1.15272482  1.0527132
## [19,] -0.93880780  5.6873926
## [20,]  0.28242917  2.2633558
## [21,] 11.78079046 10.7576275
## [22,]  9.19908306  9.3371496
## [23,] 10.51780921  9.3468266
## [24,] 10.67088975  6.1785641
```

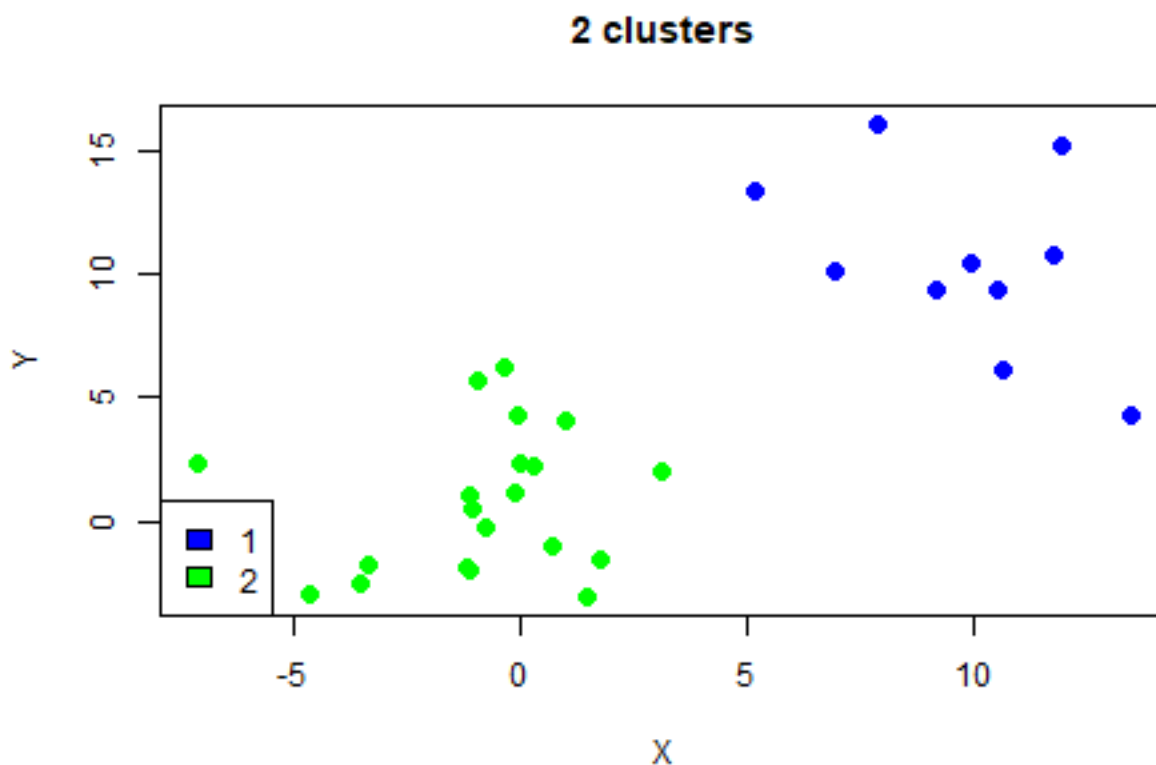
```
## [25,]  7.87300477 16.0994075
## [26,] 11.97391279 15.1441806
## [27,] 13.50385503  4.3089619
## [28,]  6.93593816 10.1460977
## [29,]  5.20414559 13.3176032
## [30,]  9.95912158 10.4819086
```

Построение истинной классификации

```
cl = kmeans(scale(dat), 2)
sapply(1:2, function(i) cl$centers[,i] * sd(dat[,i]) + mean(dat[,i]))
```

```
##      [,1]      [,2]
## 1  9.761855 10.5118327
## 2 -0.864832  0.7538702
```

```
plot(dat, col = c("blue", "green")[cl$cluster], pch = 20, cex = 2, xlab = "X", ylab = "Y", main = "2 clusters",
      legend = "bottomleft", legend = c("1", "2"), fill = c("blue", "green"))
```



Дискриминантный анализ с использованием верной классификации

```
library("MASS")
#Дискриминантный анализ № 1
n = length(dat[,1])

n.train = floor(n * 0.7)
n.test = n - n.train

idx.train = sample(1:n, n.train)
data.train = dat[idx.train, ]
idx.test = setdiff(1:n, idx.train)
data.test = dat[idx.test, ]

cl.train = factor(cl[["cluster"]][idx.train])
cl.test = factor(cl[["cluster"]][idx.test])

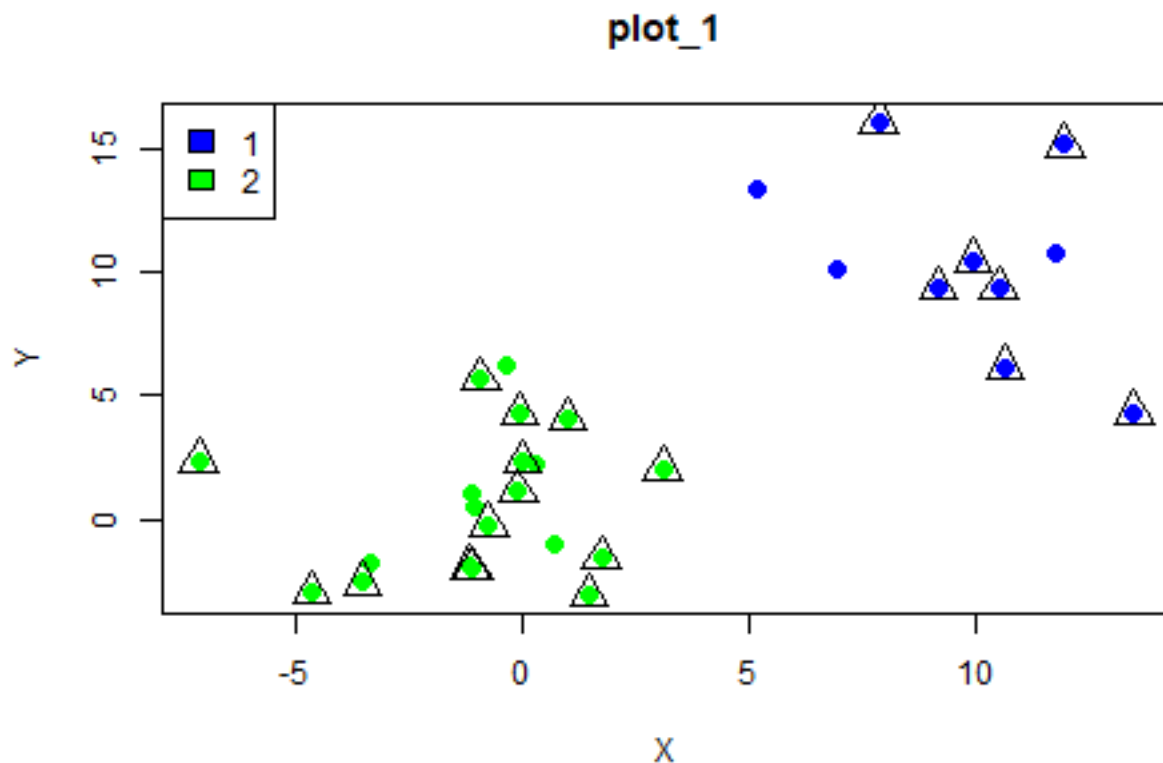
#Обучение полученной тестовой выборки
mod = qda(data.train, cl.train)
cl.test_est = predict(mod, data.test)$class

#Оценка ошибки классификации
sum(cl.test_est != cl.test) / n.test

## [1] 0

idx.wrong = idx.test[cl.test_est != cl.test]

#Строим первый график
plot(dat, col = c("blue", "green")[cl$cluster], pch = 20, cex = 2, xlab = "X", ylab = "Y", ma
legend("topleft", legend = c("1", "2"), fill = c("blue", "green"))
points(dat[idx.wrong, 1], dat[idx.wrong, 2], col = "red", pch = 4, cex = 2, lwd = 3)
points(dat[idx.train, ], pch = 2, cex = 2)
```



Дискриминантный анализ с использованием неверной классификации

```
#Меняем номер кластера в 20% выборки
idx.new = sample(1:n, n*0.2)
for (i in 1:length(idx.new)) {
  if(cl[["cluster"]][idx.new[i]] == 1){
    cl[["cluster"]][idx.new[i]] = 2}
  else{
    cl[["cluster"]][idx.new[i]] = 1
  }
}

#Дискриминантный анализ № 2
cl_train = factor(cl[["cluster"]][idx.train])
cl_test = factor(cl[["cluster"]][idx.test])

#Обучение полученной тестовой выборки
mod2 = qda(data.train, cl.train)
cl.test_est2 = predict(mod2, data.test)$class

#Оценка ошибки классификации
sum(cl.test_est2 != cl.test) / n.test
```

```
## [1] 0
```

```
idx.wrong2 = idx.test[cl.test_est2 != cl.test]
```

```
#Спроектировать второй график
```

```
plot(dat, col = c("blue", "green")[cl$cluster], pch = 20, cex = 2, xlab = "X", ylab = "Y", ma
```

```
legend("topleft", legend = c("1", "2"), fill = c("blue", "green"))
```

```
points(dat[idx.wrong2, 1], dat[idx.wrong2, 2], col = "red", pch = 4, cex = 2, lwd = 3)
```

```
points(dat[idx.new, ], pch = 3, cex = 2, lwd = 2)
```

