

Дискриминантный анализ

Математические методы в зоологии с использованием R

Марина Варфоломеева

Дискриминантный анализ

Вы сможете

- провести линейный и квадратичный дискриминантный анализ с использованием обучающей выборки и проверить качество классификации на тестовых данных или с использованием кроссвалидации
- проверить условия применимости дискриминантного анализа

Дискриминантный анализ

Пример: Морфометрия ирисов

Сверхзадача — научиться классифицировать ирисы по нескольким измерениям цветка

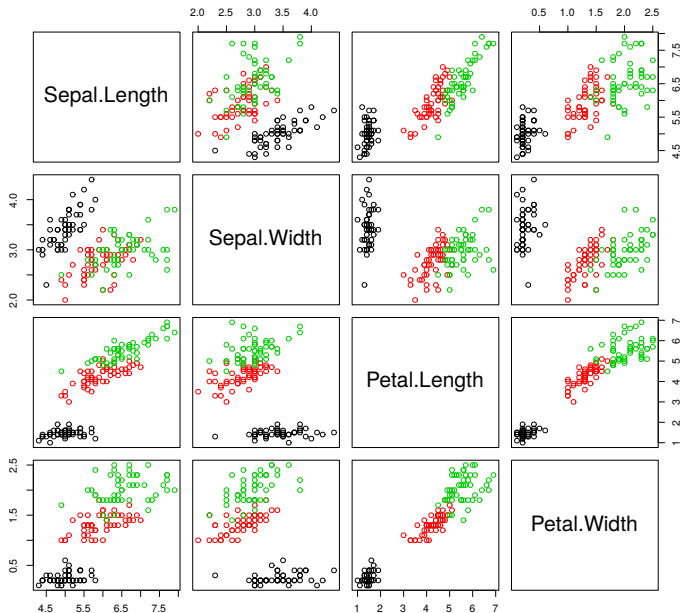
```
data(iris)
head(iris, 10)
```

#	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
# 1	5.1	3.5	1.4	0.2	setosa
# 2	4.9	3.0	1.4	0.2	setosa
# 3	4.7	3.2	1.3	0.2	setosa
# 4	4.6	3.1	1.5	0.2	setosa
# 5	5.0	3.6	1.4	0.2	setosa
# 6	5.4	3.9	1.7	0.4	setosa
# 7	4.6	3.4	1.4	0.3	setosa
# 8	5.0	3.4	1.5	0.2	setosa
# 9	4.4	2.9	1.4	0.2	setosa
# 10	4.9	3.1	1.5	0.1	setosa

По каким переменным легче всего различить группы?

Чтобы это узнать, построим графики всех пар переменных при помощи функции `pairs()` из базового пакета

```
pairs(iris[, -5], col = iris$Species)
```



Дискриминантный анализ

Дискриминантный анализ

- метод классификации объектов с учителем (**supervised learning**), т.е. применяется, когда принадлежность объектов к группе заранее известна.

Задачи дискриминантного анализа:

- выяснить, какие признаки лучше всего позволяют классифицировать объекты
- выяснить правило классификации существующих объектов
- классификация новых объектов неизвестной принадлежности по этому правилу

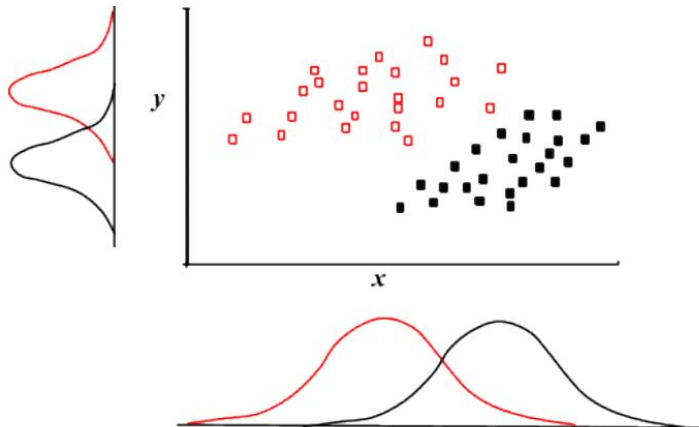
Требования к данным для дискриминантного анализа

- групп 2 или больше
- в каждой группе 2 и больше признаков
- число объектов должно быть больше, чем число признаков, лучше в несколько раз (в 4, например).
- признаки измерены в интервальной шкале

Дискриминантный анализ

Нужно найти такую ось, вдоль которой группы различаются лучше всего, с минимальным перекрытием.

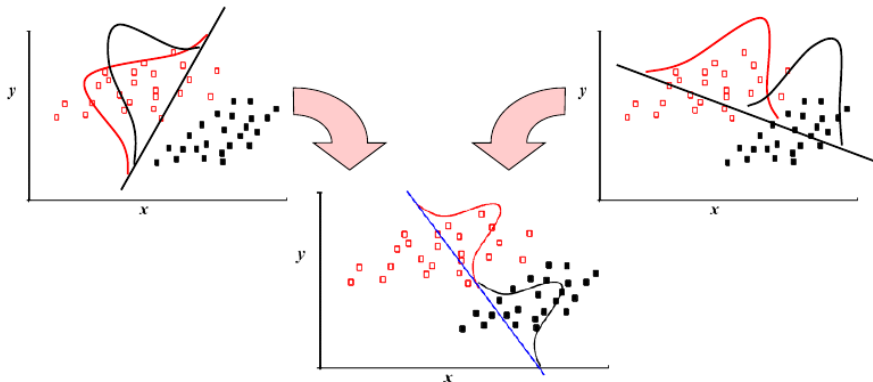
Как она может проходить?



Дискриминантные оси

Дискриминантные оси

- задаются дискриминантными функциями
- вдоль них минимальное перекрытие групп
- дискриминантных осей всего на одну меньше чем групп (или столько же, сколько признаков, если признаков меньше, чем групп)



Дискриминантные функции

Дискриминантные функции

- описывают положение дискриминантных осей

$$LD_j = d_{1j}X_1 + d_{2j}X_2 + \dots + d_{pj}X_p$$

- LD — линейная дискриминантная функция
- d — коэффициенты линейной дискриминантной функции
- X — переменные-признаки
- j = 1, ... min(k-1, p) — число дискриминантных функций
- p — число признаков
- k — число классов

Дискриминантные функции

Дискриминантные функции

- описывают положение дискриминантных осей

$$LD_j = d_{1j}X_1 + d_{2j}X_2 + \dots + d_{pj}X_p$$

- LD — линейная дискриминантная функция
- d — коэффициенты линейной дискриминантной функции
- X — переменные-признаки
- j = 1, ... min(k-1, p) — число дискриминантных функций
- p — число признаков
- k — число классов

Стандартизованные коэффициенты дискриминантной функции

- используются для сравнения переменных, измеренных в разных шкалах
используются стандартизованные коэффициенты дискриминантной функции
- большое абсолютное значение — большая дискриминирующая способность

Классификация объектов

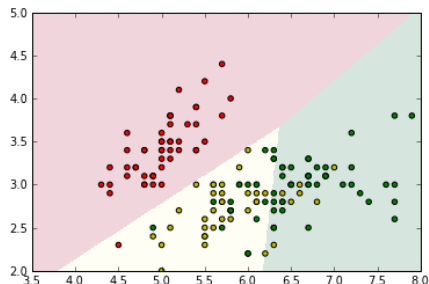
Функции классификации

- Описывают **правдоподобие** того, что объект с заданными свойствами относится к данной группе при данных значениях признаков согласно построенной классификации.

$$C_j = c_{j0} + c_{j1}X_1 + \dots + c_{jp}X_p$$

- C — функция классификации
- c — коэффициенты функций классификации
- X — переменные-признаки
- $j = 1, \dots, k$ — число групп
- p — число признаков

Для каждого (в том числе, нового) объекта можно вычислить значение всех функций классификации. Какое значение больше — к такой группе и надо отнести объект.



Пример расположения областей принятия решений в линейном дискриминантном анализе с тремя группами

Оценка качества классификации

Таблица классификации

- число верно или неверно классифицированных объектов (**confusion matrix**)

Было / Стало	Класс А	Класс Б
Класс А	верно	неверно (Б вместо А)
Класс Б	неверно (А вместо Б)	верно

Проблема: как оценить качество классификации, чтобы можно было экстраполировать результаты?

Если оценить качество классификации на тех же данных, что были использованы для ее построения — оценка неадекватная для классификации новых данных из-за “**переобучения**” (overfitting).

Возможные решения проблемы переобучения

- ① Разделить данные на **тренировочное и тестовое подмножества**:
 - тренировочные данные — для подбора классификации (для обучения)
 - независимые тестовые данные — для определения качества классификации
- ② **Кроссвалидация** — разделение на тренировочное и тестовое подмножество повторяют многократно и усредняют оценки качества классификации между повторами.

I. Дискриминантный анализ на тренировочных и тестовых данных

1) Разделяем на тренировочные и тестовые данные

```
# доля от объема выборки, которая пойдет в тренировочный датасет  
smp_size <- floor(0.80 * nrow(iris))  
# устанавливаем зерно для воспроизводимости результатов  
set.seed(982)  
# индексы строк, которые пойдут в тренировочный датасет  
in_train <- sample(sample(1:nrow(iris), size = smp_size))
```

2) На тренировочных данных получаем стандартизованные коэффициенты дискриминантных функций

```
library(MASS)
lda_tr_scaled <- lda(scale(iris[in_train, -5]), iris$Species[in_train])
# коэффициенты дискриминантных функций
lda_tr_scaled$scaling
```

#	LD1	LD2
# Sepal.Length	0.6519385	-0.02737872
# Sepal.Width	0.7433480	0.82887139
# Petal.Length	-3.7654694	-2.28038588
# Petal.Width	-2.3034593	2.78626682

По ним можно оценить вклады разных признаков в изменчивость вдоль дискриминантных осей.

3) На тренировочных данных получаем функции классификации

```
lda_tr <- lda.class(iris[in_train, -5], iris$Species[in_train])  
# Коэф. функций классификации  
lda_tr$class.funs
```

```
#           setosa versicolor virginica  
# constant    -85.93751 -69.983829 -99.85668  
# Sepal.Length 21.68238 13.979471 10.80913  
# Sepal.Width  26.90529  7.832079  3.63867  
# Petal.Length -16.12914  6.595690 13.38425  
# Petal.Width  -20.77742  5.006140 22.04759
```

По ним можно классифицировать объекты.

4) Оцениваем качество классификации на тренировочных данных

```
lda_tr_pred <- predict(lda_tr)
table(lda_tr_pred$class, iris$Species[in_train])
```

```
#
#           setosa versicolor virginica
#  setosa         38             0         0
#  versicolor      0             39         0
#  virginica       0              2        41
```

- Какова доля неправильно классифицированных случаев?

5) Оценка качества классификации на тестовых данных

Самое важное, если мы хотим использовать классификацию для прогноза

```
lda_test_pred <- predict(lda_tr, iris[-in_train, -5])  
table(lda_test_pred$class, iris$Species[-in_train])
```

```
#  
#           setosa versicolor virginica  
# setosa      12           0           0  
# versicolor   0           9           1  
# virginica    0           0           8
```

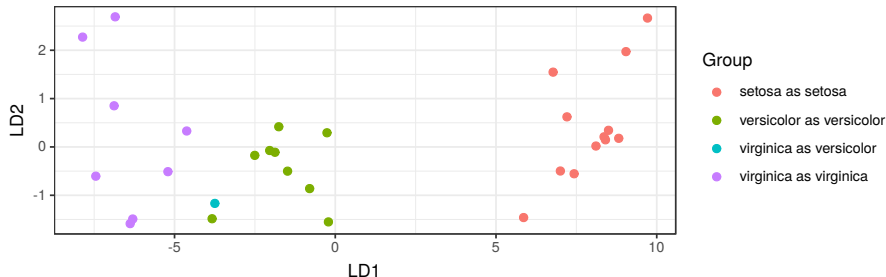
- Какова доля неправильно классифицированных случаев?

6) График классификации тестовых данных

Можно отметить неправильно классифицированные случаи своим цветом

```
library(ggplot2)
```

```
class_df <- data.frame(lda_test_pred$x,  
                        new = lda_test_pred$class,  
                        real = iris$Species[-in_train])  
class_df$Group <- factor(paste(class_df$real, class_df$new, sep = " as "))  
  
ggplot(data = class_df, aes(x = LD1, y = LD2)) +  
  geom_point(aes(colour = Group))
```



II. Дискриминантный анализ с кроссвалидацией

Кроссвалидация

```
lda_cv <- lda(iris[, -5], iris$Species, CV = TRUE)
names(lda_cv)
```

```
# [1] "class"      "posterior" "call"
```

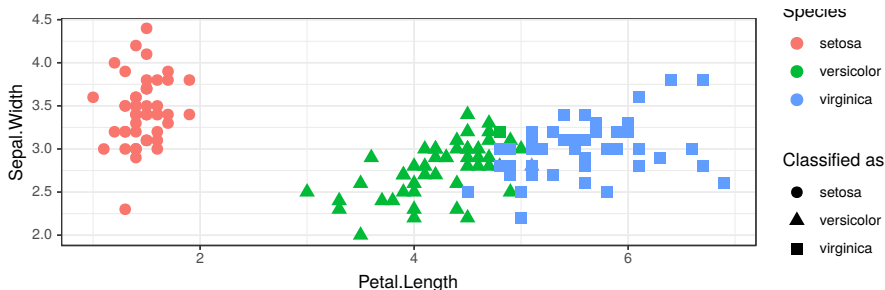
```
table(iris$Species, lda_cv$class)
```

```
#
#           setosa versicolor virginica
#  setosa         50           0         0
#  versicolor      0          48         2
#  virginica       0           1        49
```

lda_cv\$class — показывает, как классифицированы строки, если классификация обучена по остальным данным

График классификации

```
ggplot(data = iris, aes(x = Petal.Length,  
                        y = Sepal.Width,  
                        colour = Species,  
                        shape = lda_cv$class)) +  
geom_point(size = 3) +  
scale_shape_discrete("Classified as")
```



Условия применимости дискриминантного анализа

Условия применимости дискриминантного анализа

- **признаки независимы друг от друга** (чтобы не было избыточности, чтобы можно было инвертировать матрицы). Именно поэтому дискр. анализ часто применяется после анализа главных компонент.
- внутригрупповые ковариации приблизительно равны
- распределение признаков — многомерное нормальное

Условия применимости дискриминантного анализа

- **признаки независимы друг от друга** (чтобы не было избыточности, чтобы можно было инвертировать матрицы). Именно поэтому дискр. анализ часто применяется после анализа главных компонент.
- внутригрупповые ковариации приблизительно равны
- распределение признаков — многомерное нормальное

Если условия применимости нарушены:

- В некоторых случаях, дискриминантный анализ дает хорошо работающие классификации.
- Возможно, другие методы, с менее жесткими требованиями, дадут классификации лучшего качества (например, квадратичный дискриминантный анализ — quadratic discriminant analysis, дискриминантный анализ с использованием ядер — kernel discriminant analysis)

Проверка условий применимости

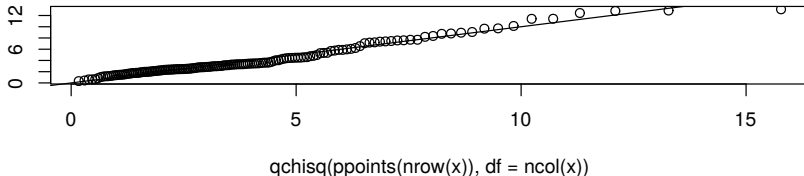
В данном случае, как и во многих других, они не выполняются, но мы уже убедились, что классификация работает...

Многомерная нормальность

```
x <- as.matrix(iris[, -5])  
d <- mahalanobis(x, colMeans(x), cov(x))  
qqplot(qchisq(ppoints(nrow(x))), df = ncol(x)), d,  
  main="QQ график для оценки многомерной нормальности",  
  ylab="Расстояние Махаланобиса")  
abline(a = 0, b = 1)
```

QQ график для оценки многомерной нормальности

Расстояние Махаланобиса



Гомогенность ковариационных матриц

```
source("BoxMTest.R")
BoxMTest(as.matrix(iris[, -5]), iris$Species)
```

```
# -----
# MBox Chi-sqr. df P
# -----
# 146.6632 140.9430 20 0.0000
# -----
# Covariance matrices are significantly different.

# $MBox
# setosa
# 146.6632
#
# $ChiSq
# setosa
# 140.943
#
# $df
# [1] 20
#
# $pValue
# setosa
# 3.352034e-20
```

Задание: Поссумы

- При помощи дискриминантного анализа классифицируйте популяции поссумов
- Хорошо ли работает классификация?
- Выполняются ли условия применимости?



```
library(DAAG)
data(possum)
possum <-
  possum[complete.cases(possum), ]
```


Take home messages

- Дискриминантный анализ — метод классификации объектов по правилам, выработанным на выборке объектов с заранее известной принадлежностью
- Качество классификации можно оценить по числу неверно классифицированных объектов. Чтобы не было “переобучения” можно:
 - Подобрать классификацию на тренировочных данных и проверить на тестовых
 - Использовать кроссвалидацию — классификацию объектов по правилам полученным по остальным данным (без учета этих объектов)
- Для дискриминантного анализа нужно отбирать признаки, независимые друг от друга или создавать синтетические признаки при помощи анализа главных компонент.
- Если внутригрупповые ковариации признаков различаются, лучше применять квадратичный дискриминантный анализ.

Дополнительные ресурсы

- Quinn, Keough, 2002, pp. 435–441