線形判別分析

1216574c　竹内　美穂

＊判別分析とは

- パターン認識の計算に使用される最も古典的な手法

  例）郵便番号による手紙の自動分類、指紋・顔の機械的照合

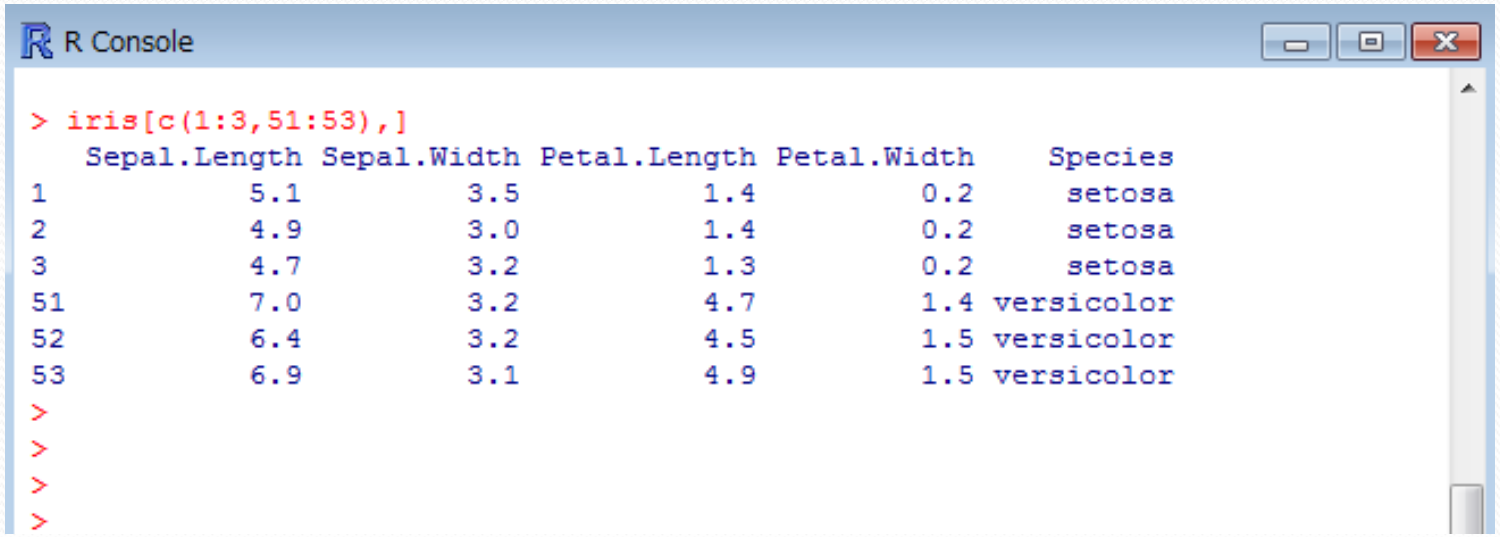
- 線形判別分析、非線形判別分析に分けられる－図9．1
  - 2群判別分析、多群判別分析

ある個体の所属グループの学習データ→判別モデル構築

そのモデルを使用

所属不明の個体の所属グループを判別

- 判別分析に適したデータ－irisデータ(アヤメ)
  最後のspecies列→各個体のグループ情報(外的基準)



  Setal＝ガク　Petal＝花弁

- 判別分析の外的基準－質的データ
- Fisher(1930)が提案した線形判別関数を使用
  (グループ分けの境界が直線か超平面の場合)

＊ケーススタディ

3群判別分析（Irisデータを用いる）

①練習用データとテスト用データの作成

cとrepはベクトルを作成する関数。
Sを50個、Cを50個、Vを50個繰り返したものをベクトルとして使用。

```
> iris.lab<-c(rep("S",50),rep("C",50),rep("V",50))
> iris1<-data.frame(iris[,1:4],species=iris.lab)
> even.n<-2*(1:75)-1
> iris.train<-iris1[even.n,]
> iris.test<-iris1[-even.n,]
>
```

偶数行と奇数行に分ける

※iris.train=練習用データ 、 iris.test=テスト用データ

②関数ldaを使用

－Rのパッケージ（MASS）の中に入っている関数

```
> library(MASS)
> (Z.lda<-lda(Species~.,data=iris.train))
Call:
lda(Species ~ ., data = iris.train)

Prior probabilities of groups:
        C         S         V
0.3333333 0.3333333 0.3333333

Group means:
  Sepal.Length Sepal.Width Petal.Length Petal.Width
C        5.992       2.776        4.308       1.352
S        5.024       3.480        1.456       0.228
V        6.504       2.936        5.564       2.076

Coefficients of linear discriminants:
                    LD1         LD2
Sepal.Length -0.5917846 -0.1971830
Sepal.Width  -1.8415262  2.2903417
Petal.Length  1.6530521 -0.7406709
Petal.Width   3.5634683  2.6365924

Proportion of trace:
   LD1    LD2
0.9913 0.0087
> |
```

第2判別関数の係数

第1判別関数の係数

グループ間の分散の比率 ←― LD1がグループ間の分散の99．13％を説明できる

③判別得点を求める

判別得点とは・・・判別係数で得られた値

＊第1判別関数を使用した場合の判別得点

```
> apply(Z.lda$means%*%Z.lda$scaling,2,mean)
     LD1      LD2
1.486146 6.282412
> Z.lda$scaling[,1]%*%t(iris.train[,1:4])-1.486146
             1         3         5         7         9        11        13        15
[1,] -7.922623 -7.298756 -8.047597 -7.086231 -6.403458 -8.303158 -7.180671 -9.588245
            17        19        21        23        25        27        29        31
[1,] -8.28938 -7.977889 -7.42009 -8.472104 -6.734408 -6.635987 -7.797648 -6.677866
            33        35        37        39        41        43        45        47
[1,] -9.277758 -6.90235 -8.324642 -6.752915 -7.672402 -7.121221 -6.935861 -8.14447
            49        51        53        55        57        59        61        63
[1,] -8.243979 1.236679 2.166967 2.460223 2.179469 1.504198 1.221029 1.087466
            65        67        69        71        73        75        77        79
[1,] 0.4429307 2.459219 3.577369 3.478334 3.626953 1.12664 2.256951 2.406657
            81        83        85        87        89        91        93        95
[1,] 1.040789 0.8324478 2.577576 1.954713 1.085304 2.020662 1.181906 1.803067
            97        99       101       103       105       107       109       111
[1,] 1.375583 -0.2290915 7.535558 6.023896 6.570008 4.506925 5.947027 4.331873
           113       115       117       119       121       123       125       127       129
[1,] 5.54021 6.90812 4.648705 8.771181 6.156031 7.003225 5.377541 4.037409 6.310534
           131       133       135       137       139       141       143       145
[1,] 5.832582 6.666881 4.361947 6.333838 3.787461 6.649582 5.310538 6.802929
           147       149
[1,] 5.217646 5.706059
>
```

④学習データの判別結果

・関数predictを使用－$class, $posterior, $xという結果を返す

＊$class・・・各個体が判別されたグループのラベル

＊$posterior・・・各個体がどのグループに判別されているかに関する確率

＊$x・・・各個体の判別関数得点

```
> table(iris.train[,5],predict(Z.lda)$class)

    C  S  V
  C 24  0  1
  S  0 25  0
  V  1  0 24
> plot(Z.lda,dimen=1)
> plot(Z.lda,dimen=2)
>
```

→ 図9．2

→ 図9．3

誤判別率は2/75≒0．0267（2．67％）

図9．2－第1判別関数得点の分布

※重なっている部分があると誤判別が起きやすい

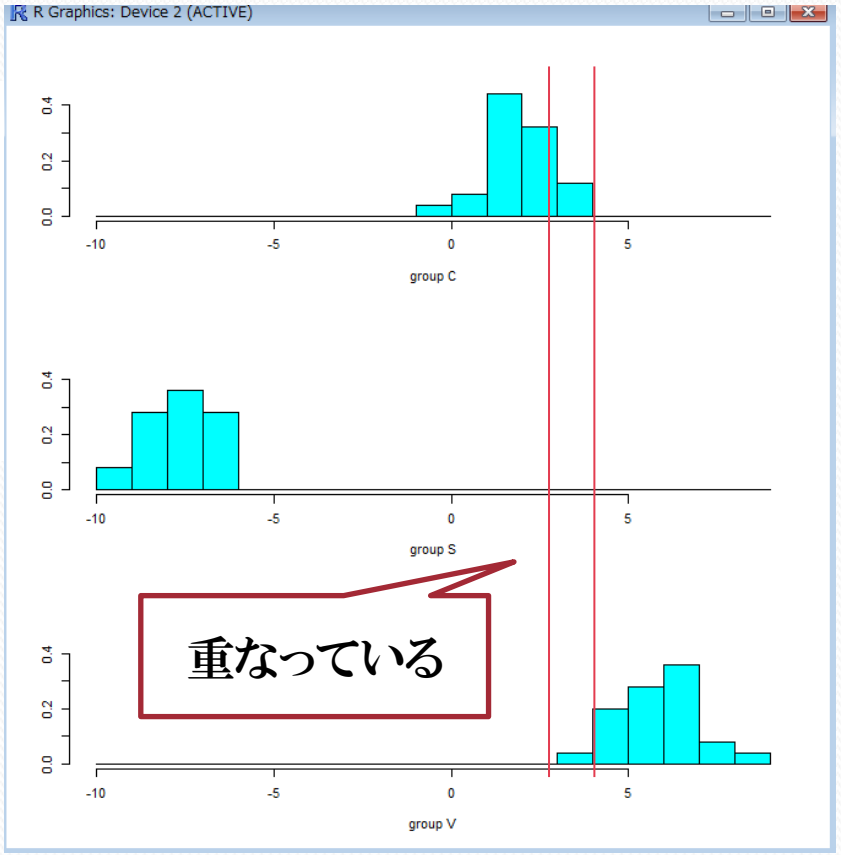
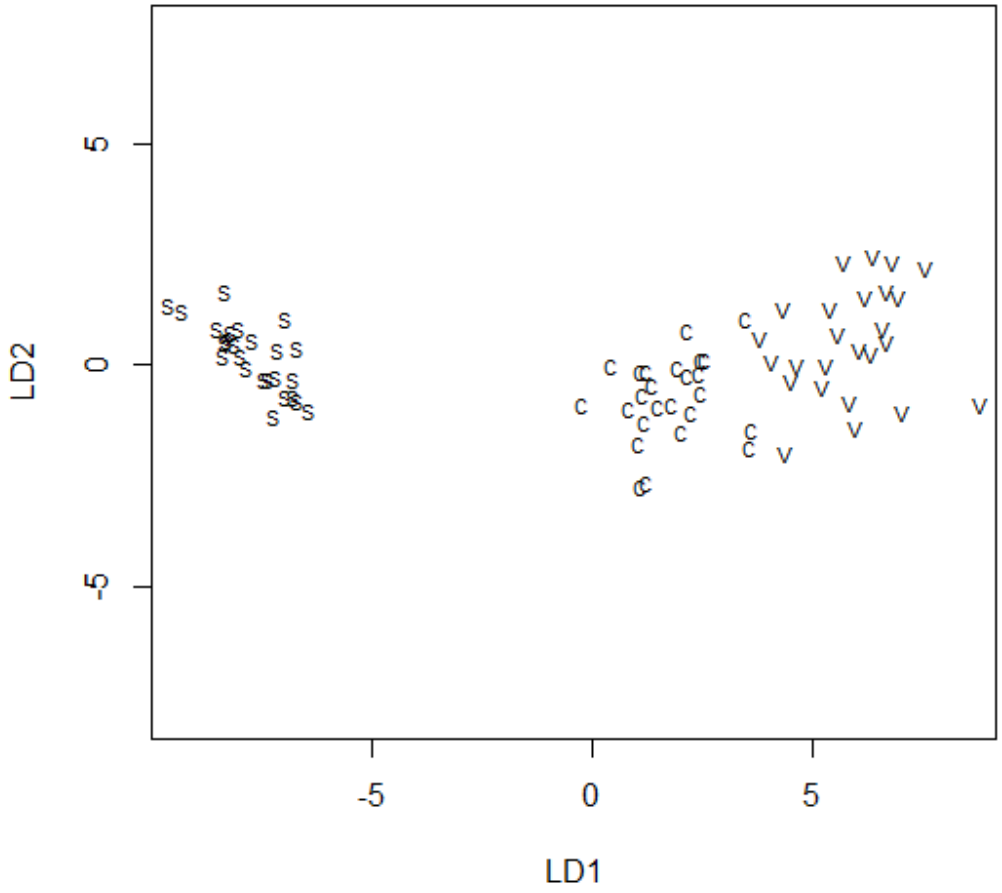


図9．2

図9．3－第1判別関数得点の散布図



※判別関数が3つ以上の場合
dimenを3以上にすれば、対散布図が作成される
各グループの母分散が等しいとの仮定に基づく

⑤テストデータの判別

```
> Y<-predict(Z.lda,iris.test[,-5])
> table(iris.test[,5],Y$class)

    C  S  V
  C 24  0  1
  S  0 25  0
  V  2  0 23
> plot(Y$x,type="n")
> text(Y$x,labels=iris.test$Species)
>
```

関数predictを利用し判別分析を行う。
その後テストデータのグループラベルと判別結果のグループラベルのクロス表を作成。

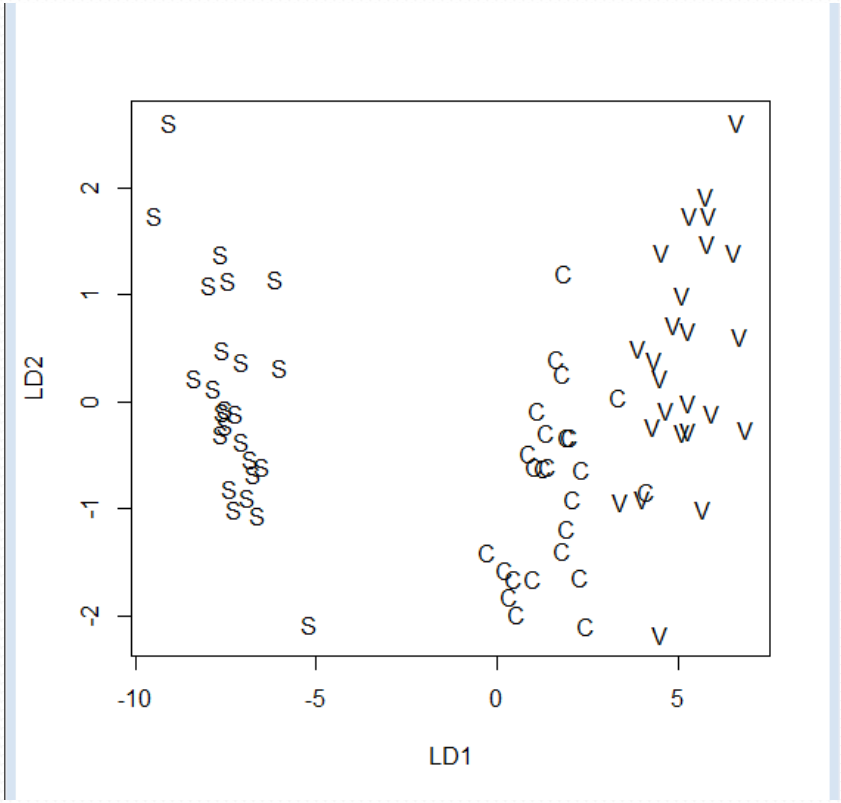


図9.4テストデータの判別の散布図

⑥交差確認

－データセットから学習用データ、テスト用データに分けてモデルの構築・テストをおこなう際に使う方法

データ全体＝n個　1等分＝1　学習用データ＝n-1

＊n重交差確認

重複しない組み合わせでn回のモデル構築とテストを行う

→n回のテスト結果の平均の平均を全体の評価に用いる

☆交差確認の関数＝CV

CV＝TRUEの場合、1つを除いた交差確認による結果が出る

```
> iris.CV<-lda(Species~.,data=iris,CV=TRUE)
> (lda.tab<-table(iris[,5],iris.CV$class))

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

```
> sum(lda.tab[row(lda.tab)==col(lda.tab)])/sum(lda.tab)
[1] 0.98
> sum(lda.tab[row(lda.tab)!=col(lda.tab)])/sum(lda.tab)
[1] 0.02
>
```