

# 判別分析

## 分析の評価

村田 昇

## 判別分析の復習

### 判別分析

- 個体の特徴量からその個体の属するクラスを予測する関係式を構成
- **事前確率**:  $\pi_k = P(Y = k)$  (prior probability)
  - $X = \mathbf{x}$  が与えられる前に予測されるクラス
- **事後確率**:  $p_k(\mathbf{x})$  (posterior probability)
  - $X = \mathbf{x}$  が与えられた後に予測されるクラス

$$p_k(\mathbf{x}) = P(Y = k | X = \mathbf{x})$$

- 所属する確率が最も高いクラスに個体を分類

### 判別関数

- 判別の手続き
  - 説明変数  $X = \mathbf{x}$  の取得
  - 事後確率  $p_k(\mathbf{x})$  の計算
  - 事後確率最大のクラスにデータを分類
- **判別関数**:  $\delta_k(\mathbf{x})$  ( $k = 1, \dots, K$ )

$$p_k(\mathbf{x}) < p_l(\mathbf{x}) \Leftrightarrow \delta_k(\mathbf{x}) < \delta_l(\mathbf{x})$$

事後確率の順序を保存する計算しやすい関数

- 判別関数  $\delta_k(\mathbf{x})$  を最大化するようなクラス  $k$  に分類

### 線形判別

- $f_k(\mathbf{x})$  の仮定
  - $q$  変量正規分布の密度関数
  - 平均ベクトル  $\boldsymbol{\mu}_k$ : クラスごとに異なる
  - 共分散行列  $\Sigma$ : **すべてのクラスで共通**

$$f_k(\mathbf{x}) = \frac{1}{(2\pi)^{q/2} \sqrt{\det \Sigma}} \exp \left( -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_k)^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \right)$$

- 線形判別関数:  $\mathbf{x}$  の 1 次式

$$\delta_k(\mathbf{x}) = \mathbf{x}^T \Sigma^{-1} \boldsymbol{\mu}_k - \frac{1}{2} \boldsymbol{\mu}_k^T \Sigma^{-1} \boldsymbol{\mu}_k + \log \pi_k$$

## 2 次判別

- $f_k(\mathbf{x})$  の仮定
  - $q$  変量正規分布の密度関数
  - 平均ベクトル  $\boldsymbol{\mu}_k$ : クラスごとに異なる
  - 共分散行列  $\Sigma_k$ : **クラスごとに異なる**

$$f_k(\mathbf{x}) = \frac{1}{(2\pi)^{q/2} \sqrt{\det \Sigma_k}} \exp \left( -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_k)^T \Sigma_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \right)$$

- 2 次判別関数:  $\mathbf{x}$  の 2 次式

$$\delta_k(\mathbf{x}) = -\frac{1}{2} \det \Sigma_k - \frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_k)^T \Sigma_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) + \log \pi_k$$

## Fisher の線形判別

- 新しい特徴量  $Z = \boldsymbol{\alpha}^T X$  を考える
- 良い  $Z$  の基準
  - クラス内では集まっているほど良い ( $\boldsymbol{\alpha}^T W \boldsymbol{\alpha}$  は小)
  - クラス間では離れているほど良い ( $\boldsymbol{\alpha}^T B \boldsymbol{\alpha}$  は大)
- Fisher の基準

$$\text{maximize } \boldsymbol{\alpha}^T B \boldsymbol{\alpha} \quad \text{s.t.} \quad \boldsymbol{\alpha}^T W \boldsymbol{\alpha} = \text{const.}$$

- $\boldsymbol{\alpha}$  は  $W^{-1}B$  の第 1 から第  $K-1$  固有ベクトル
- 判別方法: 特徴量の距離を用いる
- $d_k = \sum_{l=1}^{K-1} (\alpha_l^T \mathbf{x} - \alpha_l^T \boldsymbol{\mu}_k)^2$  が最小のとなるクラス  $k$  に判別

## 2 値判別分析の評価

### 誤り率

- 単純な誤り

$$(\text{誤り率}) = \frac{(\text{誤って判別されたデータ数})}{(\text{全データ数})}$$

- 判別したいラベル: 陽性 (positive)
  - **真陽性**: 正しく陽性と判定 (true positive; TP)
  - **偽陽性**: 誤って陽性と判定 (false positive; FP) (**第 I 種過誤**)
  - **偽陰性**: 誤って陰性と判定 (false negative; FN) (**第 II 種過誤**)
  - **真陰性**: 正しく陰性と判定 (true negative; TN)

### 混同行列

	真値は陽性	真値は陰性
判別は陽性	真陽性 (True Positive)	偽陽性 (False Positive)
判別は陰性	偽陰性 (False Negative)	真陰性 (True Negative)

- **confusion matrix**
- 転置で書く流儀もあるので注意

## 混同行列

	判別は陽性	判別は陰性
真値は陽性	真陽性 (True Positive)	偽陰性 (False Negative)
真値は陰性	偽陽性 (False Positive)	真陰性 (True Negative)

- パターン認識や機械学習で多く見られた書き方
- 誤差行列 (error matrix) と呼ばれる

## 基本的な評価基準

- 定義

$$\begin{aligned}(\text{真陽性率}) &= \frac{TP}{TP + FN} && (\text{true positive rate}) \\(\text{真陰性率}) &= \frac{TN}{FP + TN} && (\text{true negative rate}) \\(\text{適合率}) &= \frac{TP}{TP + FP} && (\text{precision}) \\(\text{正答率}) &= \frac{TP + TN}{TP + FP + TN + FN} && (\text{accuracy})\end{aligned}$$

- 別名 (分野で異なるので注意)
  - 感度 (sensitivity) あるいは 再現率 (recall)

$$(\text{真陽性率}) = \frac{TP}{TP + FN}$$

- 特異度 (specificity)

$$(\text{真陰性率}) = \frac{TN}{FP + TN}$$

- 精度

$$(\text{正答率}) = \frac{TP + TN}{TP + FP + TN + FN}$$

## F-値

- F-measure, F-score
- 定義

$$\begin{aligned}F_1 &= \frac{2}{1/(\text{再現率}) + 1/(\text{適合率})} \\F_\beta &= \frac{\beta^2 + 1}{\beta^2/(\text{再現率}) + 1/(\text{適合率})}\end{aligned}$$

- 再現率 (真陽性率) と適合率の (重み付き) 調和平均

## Cohen の kappa 値

- Cohen's **kappa measure**
- 定義

$$p_o = \frac{TP + TN}{TP + FP + TN + FN} \quad (\text{accuracy})$$
$$p_e = \frac{TP + FP}{TP + FP + TN + FN} \cdot \frac{TP + FN}{TP + FP + TN + FN} + \frac{FN + TN}{TP + FP + TN + FN} \cdot \frac{FP + TN}{TP + FP + TN + FN}$$
$$\kappa = \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e}$$

- 観測された精度と偶然の精度の比較

## 解析の事例

### データについて

- 気象庁より取得した東京の気候データ
  - 気象庁 <https://www.data.jma.go.jp/gmd/risk/obsdl/index.php>
  - データ [https://noboru-murata.github.io/multivariate-analysis/data/tokyo\\_weather.csv](https://noboru-murata.github.io/multivariate-analysis/data/tokyo_weather.csv)

### 気温と湿度による月の判別

- 温度と湿度による 9/10 月の線形判別
- 温度と湿度による 9/10 月の 2 次判別

### 判別結果の比較

- 線形判別の混同行列

予測値\真値	9	10
9	27	13
10	3	18

- 2 次判別の混同行列

予測値\真値	9	10
9	27	14
10	3	17

- 線形判別の評価指標

Accuracy	Kappa
0.738	0.478

Sensitivity	Specificity	Precision	Recall	F1
0.9	0.581	0.675	0.9	0.771

- 2 次判別の評価指標

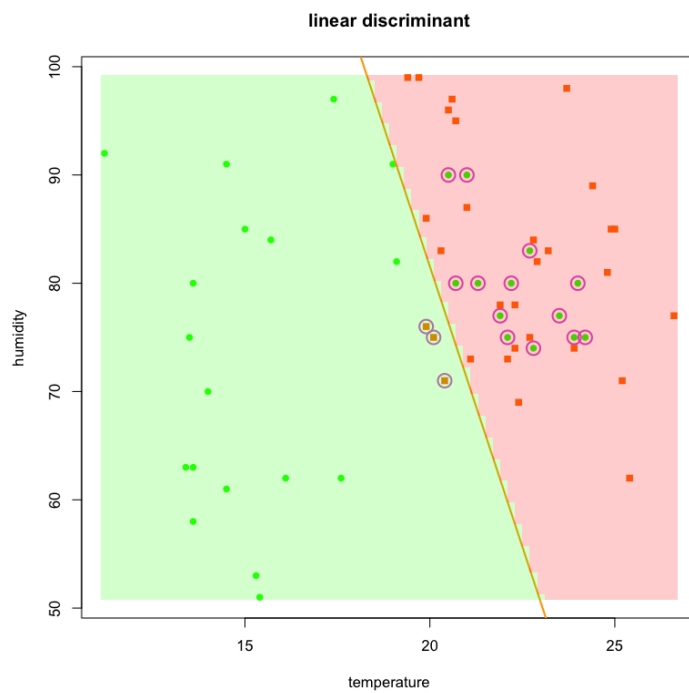


図 1: 線形判別

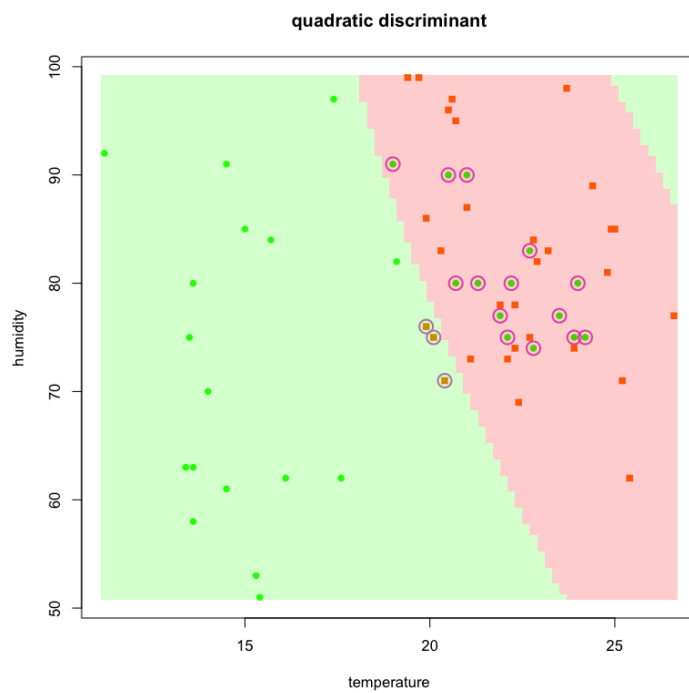


図 2: 2 次判別

	Accuracy	Kappa			
	0.721	0.446			
	Sensitivity	Specificity	Precision	Recall	F1
	0.9	0.548	0.659	0.9	0.761

- テスト

## 演習

### 問題

- 以下の問に答えなさい
  - F-値, 再現率, 適合率の大小関係はどのようなになるか
  - 2値判別 ( $P = 1, N = 0$  とする) において正解ラベル  $Y$  と予測ラベル  $\hat{Y}$  の相関係数を  $TP, FP, TN, FN$  およびデータ数  $N$  を用いて表せ

### 解答例

- 最大最小と平均の関係から以下が成り立つ

$$\min(\text{再現率}, \text{適合率}) \leq F_1 \leq \max(\text{再現率}, \text{適合率})$$

さらに相加・相乗平均の関係から

$$F_1 \leq (\text{相乗平均}) \leq (\text{相加平均})$$

も成り立つ

- 相関係数の定義に従って計算すればよい

$$\rho = \frac{\text{Cov}(Y, \hat{Y})}{\sqrt{\text{Var}(Y)\text{Var}(\hat{Y})}}$$

- 例えば分子の共分散は以下のように計算される

$$\begin{aligned}
 \text{Cov}(Y, \hat{Y}) &= \mathbb{E}[(Y - \mathbb{E}[Y])(\hat{Y} - \mathbb{E}[\hat{Y}])] \\
 &= \mathbb{E}[Y\hat{Y}] - \mathbb{E}[Y]\mathbb{E}[\hat{Y}] \\
 &= \frac{TP}{N} - \frac{TP + FN}{N} \frac{TP + FP}{N} \\
 &= \frac{TP(TP + FN + FP + TN)}{N^2} \\
 &\quad - \frac{(TP + FN)(TP + FP)}{N^2} \\
 &= \frac{TP \cdot TN - FP \cdot FN}{N^2}
 \end{aligned}$$

- 平均は標本平均で置き換えた

- 同様に分母の分散は以下ようになる

$$\begin{aligned}\text{Var}(Y) &= \mathbb{E}[Y^2] - \mathbb{E}[Y]^2 \\ &= \frac{(TP + FN)(TN + FP)}{N^2} \\ \text{Var}(\hat{Y}) &= \mathbb{E}[\hat{Y}^2] - \mathbb{E}[\hat{Y}]^2 \\ &= \frac{(TP + FP)(TN + FN)}{N^2}\end{aligned}$$

- したがって以下のようにまとめられる

$$\rho = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

- これは Matthews correlation coefficient (MCC) と呼ばれる評価指標の一つである

## 実習

### データセットの準備

- 以下のデータセットを使用します
  - winequality-red.csv  
UC Irvine Machine Learning Repository で公開されている Wine Quality Data Set の一部  
<https://archive.ics.uci.edu/ml/datasets/Wine+Quality>
  - 以下に download せずに読み込む方法を紹介します

```
wq_org <- read.csv("https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-red.csv",
  sep=";", ) # データの区切りが ";" となっている
wq_data <- transform(wq_org,
  quality=as.factor( # quality を再分類
    ifelse(quality %in% 7:10, "A",
    ifelse(quality %in% 5:6, "B", "C"))))
```

### R : 混同行列 caret::confusionMatrix()

- caret: 評価のためのパッケージ
- 判別結果の評価

```
install.packages("caret") # 右下ペインの package タブから install
library(caret) # または require(caret)
confusionMatrix(data, reference)
## data: 判別関数による予測ラベル (factor)
## reference: 真のラベル (上と同じ factor である必要がある)
```

### 練習問題

- 前回と同様に東京の気候データの線形判別を行い、以下を確認しなさい
  - 10月と11月の気温と湿度のデータを抽出する

```
tw_data <- read.csv("data/tokyo_weather.csv")
tw_subset <- transform(subset(tw_data,
  subset= month %in% c("10", "11"),
  select=c(temp, humid, month)),
  month=as.factor(month)) # 因子にする
```

- 全てデータを用いて線形判別関数を構成する
- 構成した判別関数の評価を行う

## 練習問題

- Wine Quality Data Set を用いて以下を確認しなさい
  - 全てデータを用いて線形判別関数を構成し、評価を行う
  - 全てデータを用いて 2 次判別関数を構成し、評価を行う

## 予測誤差

### 訓練誤差と予測誤差

- 訓練誤差: 既知データに対する誤り (training error)
- 予測誤差: 未知データに対する誤り (predictive error)
- 訓練誤差は予測誤差より良くなることが多い
- 既知データの判別に特化している可能性がある
  - 過適応 (over-fitting)
  - 過学習 (over-training)

## 交叉検証

- データを訓練データと試験データに分割して用いる
  - 訓練データ: 判別関数を構成する (training data)
  - 試験データ: 予測精度を評価する (test data)
- データの分割に依存して予測誤差の評価が偏る
- 偏りを避けるために複数回分割を行ない評価する
- “交差” と書く場合もある

## 交叉検証法

- cross-validation (CV)
- $k$ -重交叉検証法 ( $k$ -fold cross-validation;  $k$ -fold CV)
  - $n$  個のデータを  $k$  ブロックにランダムに分割
  - 第  $i$  ブロックを除いた  $k-1$  ブロックで判別関数を推定
  - 除いておいた第  $i$  ブロックで予測誤差を評価
  - $i = 1, \dots, k$  で繰り返し  $k$  個の予測誤差で評価 (平均や分散)
- leave-one-out 法 (leave-one-out CV; LOO-CV)
  - $k = n$  として上記を実行

## 実習

### R : LOO 交叉検証法

- 関数 `lda()` と `qda()` はオプションで LOO 交叉検証を行うことができる
- オプションの指定方法

```
est <- lda(formula, data, CV=TRUE)
est$class # LOO CV による予測結果
## 特定のデータを除いて判別関数を構成し、そのデータの予測を行っている
est <- qda(formula, data, CV=TRUE)
est$class # LOO CV による予測結果
```



```
## 2次判別についても同様
```

## R : k-重交叉検証法 `caret::train()`

- `caret` パッケージの関数 `train()` で実行可能

```
train(formula, data,
      method,
      trControl=trainControl(method="cv", number))
## formula: R の式
## data: データフレーム
## method: 推定を行う関数 method="lda"/"qda"などを指定
## trControl: 学習方法の指定
## trainControl のオプション
## method: 評価方法など指定 method="cv"/"LOOCV"
## number: k-重交叉検証のブロック数 (k)
```

## 練習問題

- Wine Quality Data Set を用いた線形判別と 2 次判別に関して、以下を行いなさい
  - LOO 交叉検証法を用いて予測誤差の評価を行いなさい
  - k-重交叉検証法を用いて予測誤差の評価を行いなさい

## 解析の事例

### データについて

- UC Irvine Machine Learning Repository の公開データ
  - <https://archive.ics.uci.edu/ml/datasets/Wine+Quality>  
Wine Quality Data Set  
P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. Modeling wine preferences by data mining from physicochemical properties. In Decision Support Systems, Elsevier, 47(4):547-553. ISSN: 0167-9236.

以下では `winequality-red.csv` を利用

- データ概要
  - データ数 1599
  - 説明変数 (based on physicochemical tests)
    - 1 - fixed acidity
    - 2 - volatile acidity
    - 3 - citric acid
    - 4 - residual sugar
    - 5 - chlorides
    - 6 - free sulfur dioxide
    - 7 - total sulfur dioxide
    - 8 - density
    - 9 - pH
    - 10 - sulphates
    - 11 - alcohol
  - 目的変数 (based on sensory data)
    - 12 - quality (score between 0 and 10)

\* ただし解析では A,B,C の 3 値に集計

- 実際のデータの一部

7.4	0.7	0	1.9
7.8	0.88	0	2.6
7.8	0.76	0.04	2.3
11.2	0.28	0.56	1.9
7.4	0.7	0	1.9
7.4	0.66	0	1.8
7.9	0.6	0.06	1.6
7.3	0.65	0	1.2
7.8	0.58	0.02	2
7.5	0.5	0.36	6.1
6.7	0.58	0.08	1.8
7.5	0.5	0.36	6.1
5.6	0.615	0	1.6
7.8	0.61	0.29	1.6
8.9	0.62	0.18	3.8
8.9	0.62	0.19	3.9

- 実際のデータの一部 (続き)

0.076	11	34	0.9978
0.098	25	67	0.9968
0.092	15	54	0.997
0.075	17	60	0.998
0.076	11	34	0.9978
0.075	13	40	0.9978
0.069	15	59	0.9964
0.065	15	21	0.9946
0.073	9	18	0.9968
0.071	17	102	0.9978
0.097	15	65	0.9959
0.071	17	102	0.9978
0.089	16	59	0.9943
0.114	9	29	0.9974
0.176	52	145	0.9986
0.17	51	148	0.9986

- 実際のデータの一部 (続き)

3.51	0.56	9.4	B
3.2	0.68	9.8	B
3.26	0.65	9.8	B
3.16	0.58	9.8	B
3.51	0.56	9.4	B
3.51	0.56	9.4	B
3.3	0.46	9.4	B
3.39	0.47	10	A
3.36	0.57	9.5	A
3.35	0.8	10.5	B
3.28	0.54	9.2	B
3.35	0.8	10.5	B
3.58	0.52	9.9	B
3.26	1.56	9.1	B
3.16	0.88	9.2	B
3.17	0.93	9.2	B

## LOO 交叉検証による予測誤差の評価

- 線形判別 (訓練誤差/予測誤差)

83	65	2
134	1239	55
0	15	6

- 線形判別の過学習は微小

- 2 次判別 (訓練誤差/予測誤差)

120	141	4
96	1133	50
1	45	9

- 2 次判別は若干過学習している

- 予測誤差の比較 (線形/2 次)

0.382488479262673	0.951519536903039	0.553333333333333	0.382488479262673	0.452316076294278
0.939347990902199	0.325	0.867647058823529	0.939347990902199	0.902074990899163
0.0952380952380952	0.990234375	0.285714285714286	0.0952380952380952	0.142857142857143

0.552995391705069	0.895079594790159	0.452830188679245	0.552995391705069	0.49792531120332
0.858984078847612	0.478571428571429	0.885848318999218	0.858984078847612	0.872209391839877
0.142857142857143	0.970052083333333	0.163636363636364	0.142857142857143	0.152542372881356

- 予測誤差の観点からは線形判別の方が良さそう

## 次週の予定

- 第 1 日 : クラスタ分析の考え方と階層的方法
- 第 2 日 : 非階層的方法と分析の評価