



**Departemen Statistika**

Fakultas Matematika dan Ilmu Pengetahuan Alam  
Institut Pertanian Bogor

# **PEMODELAN KLASIFIKASI**

## **PERTEMUAN #5**

### **ANALISIS DISKRIMINAN**

**Bagus Sartono**

[bagusco@ipb.ac.id](mailto:bagusco@ipb.ac.id)

[bagusco@gmail.com](mailto:bagusco@gmail.com)

# Konsep Dasar

Andaikan ada vektor peubah acak  $X$ , dan variabel kelas  $Y$  yang berisi  $m$  buah label kelas:  $1, \dots, m$

Analisis diskriminan bekerja dengan menentukan fungsi kepekatan dari masing-masing kelas, yang kemudian digabungkan dengan informasi prior untuk menghasilkan peluang bersyarat kelas terhadap  $X$  atau  $P(Y = k \mid X)$

Suatu amatan akan dikelaskan ke dalam kelas ke- $k$  jika nilai peluang bersyaratnya adalah yang terbesar diantara kelas-kelas lainnya

# Konsep Dasar

Andaikan  $\pi_k$  adalah peluang prior dari kelas  $k$ ,  $k = 1, \dots, m$  yang memenuhi kendala

$$\sum_{i=1}^m \pi_i = 1$$

Secara sederhana, nilai  $\pi_k$  dapat diduga sebagai

$$\pi_k = \frac{\text{banyaknya amatan dari kelas } k}{\text{banyaknya seluruh amatan}}$$

# Konsep Dasar

Andaikan  $f_k(\mathbf{x})$  adalah fungsi sebaran bersyarat  $\mathbf{X}$  pada kelas  $k$

Sedangkan kita tertarik pada peluang kelas  $k$  bersyarat terhadap vektor  $\mathbf{X}$ , yang menggunakan kaidah Bayes dapat dituliskan sebagai

$$P(Y = k | x) = \frac{f_k(x)\pi_k}{\sum_{i=1}^m f_i(x)\pi_i}$$

Berdasarkan konsep maximum a posteriori,

$$\hat{Y} = \arg \max P(Y = k | x) = \arg \max f_k(x)\pi_k$$

# Konsep Dasar

Berdasarkan konsep maximum a posteriori

$$\hat{Y} = \arg \max P(Y = k | x) = \arg \max f_k(x) \pi_k$$

Untuk kasus dua kelas, misalnya kelas 0 dan kelas satu, maka

$$\hat{Y} = 0 \text{ jika } P(Y = 0 | x) > P(Y = 1 | x)$$

dan

$$\hat{Y} = 1 \text{ jika } P(Y = 1 | x) > P(Y = 0 | x)$$

# Konsep Dasar

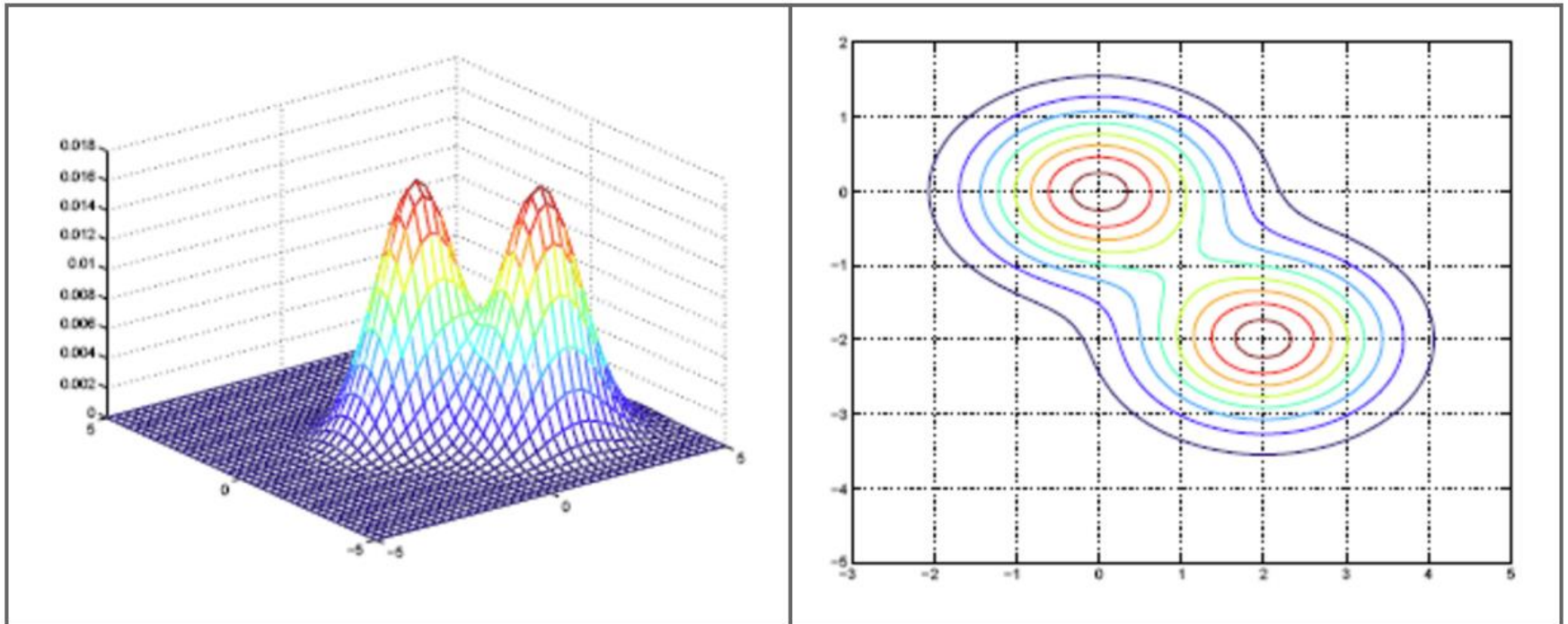
Pada teknik linear discriminant analysis,  $f_k(\mathbf{x})$  diasumsikan berupa sebaran multivariate normal


$$f_k(x) = \frac{1}{(2\pi)^{p/2} |\Sigma_k|^{1/2}} e^{-\frac{1}{2} (x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k)}$$

dengan  $p$  adalah dimensi (banyaknya peubah),  $\mu_k$  adalah vektor rata-rata, dan  $\Sigma_k$  adalah matriks ragam-peragam

serta  $\Sigma_k = \Sigma$  untuk semua  $k = 1, \dots, m$ , atau dengan kata lain antar kelas memiliki matriks ragam-peragam yang homogen

# Konsep Dasar





```
a <- seq(10, 30,length.out=50)
b <- seq(10, 30,length.out=50)
grid <- NULL
```

```
for (i in a) {
  for (j in b) {
    grid <- rbind(grid, c(i, j))
  }
}
```

```
mean1 <- c(15, 15)
mean2 <- c(22, 22)
sigma <- matrix(c(5, 3, 3, 5), 2, 2)
```

```
require(mvtnorm)
y = 0.5*dmvnorm(grid, mean1, sigma, log=FALSE) + 0.5*dmvnorm(grid, mean2, sigma, log=FALSE)
```

```
z <- matrix(y, length(a), length(b), byrow=TRUE)
```

```
persp(a, b, z, phi=45, theta=-10)
contour(a, b, z)
```



# Konsep Dasar

Kembali ke konsep maximum a posteriori

$$\hat{Y} = \arg \max P(Y = k | x) = \arg \max f_k(x) \pi_k$$

Karena logaritma adalah fungsi yang bersifat monoton naik maka dapat juga kita tuliskan

$$\hat{Y} = \arg \max \log(f_k(x) \pi_k)$$

Jika kita gunakan sebaran normal ganda yang dibicarakan sebelumnya akan diperoleh

$$\begin{aligned} &= \arg \max_k \left[ -\log((2\pi)^{p/2} |\Sigma|^{1/2}) - \frac{1}{2} (x - \mu_k)^T \Sigma^{-1} (x - \mu_k) + \log(\pi_k) \right] \\ &= \arg \max_k \left[ -\frac{1}{2} (x - \mu_k)^T \Sigma^{-1} (x - \mu_k) + \log(\pi_k) \right] \end{aligned}$$

# Konsep Dasar

$$-\frac{1}{2}(x - \mu_k)^T \Sigma^{-1} (x - \mu_k) = x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k - \frac{1}{2} x^T \Sigma^{-1} x$$

maka akan diperoleh

$$\hat{Y} = \arg \max_k \left[ x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + \log(\pi_k) \right]$$

Untuk kasus dua kelas, yaitu kelas  $l$  dan  $k$

The decision boundary between class  $k$  and  $l$  is:

$$\{x : \delta_k(x) = \delta_l(x)\}$$

Or equivalently the following holds

$$\log \frac{\pi_k}{\pi_l} - \frac{1}{2} (\mu_k + \mu_l)^T \Sigma^{-1} (\mu_k - \mu_l) + x^T \Sigma^{-1} (\mu_k - \mu_l) = 0$$

# Ilustrasi Garis Diskriminan

```
> 0.5 * t(c(37,37)) %*% solve(sigma) %*% c(-7,-7)
```

```
[,1]
```

```
[1,] -32.375
```

```
> solve(sigma) %*% c(-7, -7)
```

```
[,1]
```

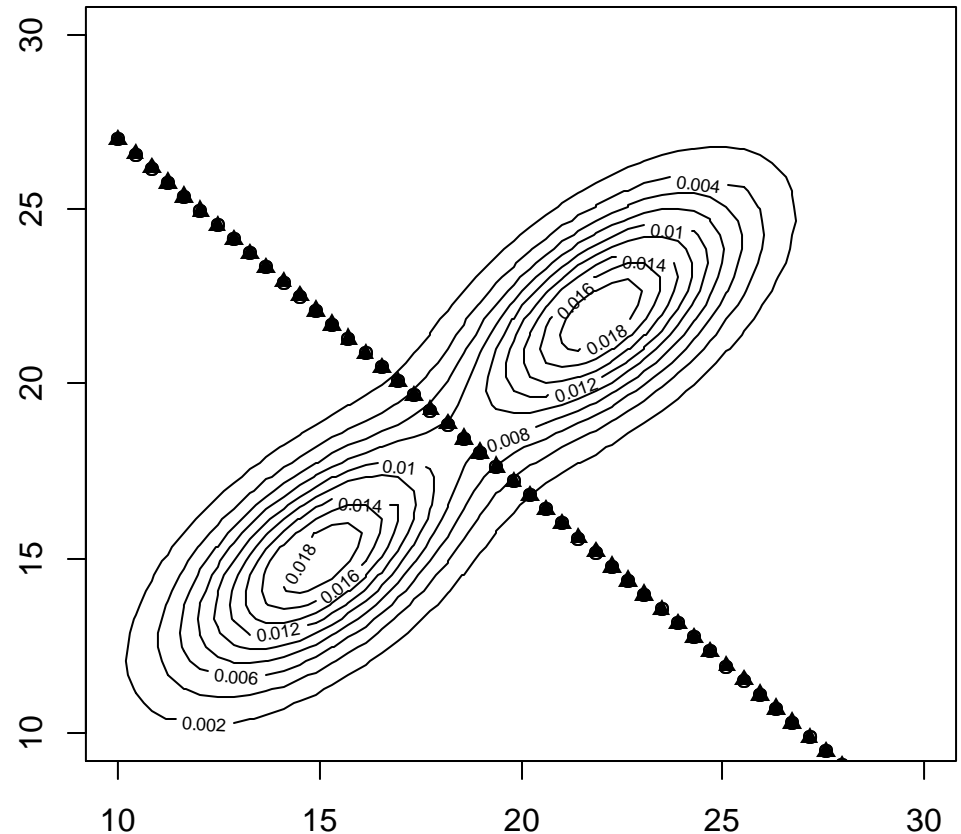
```
[1,] -0.875
```

```
[2,] -0.875
```

```
> garis = (32.375 - 0.875 * a) / (0.875)
```

```
> > contour(a, b, z)
```

```
> points(a, garis, pch=17)
```



# Menduga Parameter Sebaran

$$\hat{\pi}_k = N_k / N$$

$$\hat{\mu}_k = \sum_{g_i=k} x^{(i)} / N_k$$

$$\hat{\Sigma} = \sum_{k=1}^K \sum_{g_i=k} \left( x^{(i)} - \hat{\mu}_k \right) \left( x^{(i)} - \hat{\mu}_k \right)^T / (N - K)$$

# **Ilustrasi menggunakan Data White\_wine2.csv**

```
data.wine <- read.csv("D:/white_wine2.csv",header=TRUE)
```

```
#melihat nama-nama kolom pada data.wine  
names(data.wine)
```

```
#membuat kelas baru
```

```
#yang skor quality lebih dari 6 dikelaskan menjadi kelas=1
```

```
#yang skor quality tidak lebih dari 6 dikelaskan menjadi kelas=0
```

```
quality <- as.factor(ifelse(data.wine$quality>6,1,0))
```

```
alcohol <- data.wine$alcohol
```

```
density <- data.wine$density
```

```
plot (density, alcohol,
```

```
  cex=ifelse(quality == 1, 1, 0.3),
```

```
  col=ifelse(quality == 1, 3, 6), pch=ifelse(quality == 1, 6, 3))
```

```
library(MASS)
```

```
ldafit <- lda(quality ~ density + alcohol)
```

```
ldafit
```

```
plot(ldafit)
```