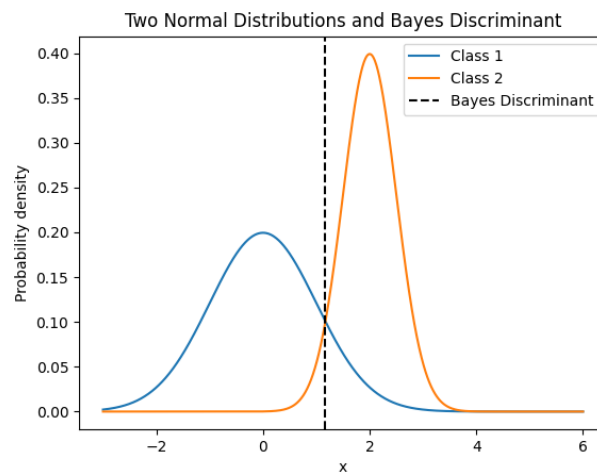


```

1 import numpy as np
2 import matplotlib.pyplot as plt
3 from scipy.stats import norm
4
5 def bayes_discriminant(mu1, sigma1, mu2, sigma2):
6     """
7     計算貝氏判別點
8
9     Args:
10         mu1, sigma1: 第一類常態分佈的均值和標準差
11         mu2, sigma2: 第二類常態分佈的均值和標準差
12
13     Returns:
14         float: 貝氏判別點
15     """
16     # 假設先驗概率相等，則貝氏判別點為使得兩個類別似然比等於1的點
17     def f(x):
18         return norm.pdf(x, mu1, sigma1) / norm.pdf(x, mu2, sigma2) - 1
19
20     # 使用scipy.optimize.fsolve求解方程f(x) = 0
21     from scipy.optimize import fsolve
22     return fsolve(f, (mu1 + mu2) / 2)
23
24 # 自定義參數
25 mu1, sigma1 = 0, 1 #第一類常態分佈的均值和標準差
26 mu2, sigma2 = 2, 0.5#第二類常態分佈的均值和標準差
27
28 # 生成數據（僅用於繪圖，實際分類時不需要生成大量數據）
29 x = np.linspace(-3, 6, 1000)
30
31 y1 = norm.pdf(x, mu1, sigma1)/2
32 y2 = norm.pdf(x, mu2, sigma2)/2
33
34 # 計算貝氏判別點
35 discriminant_point = bayes_discriminant(mu1, sigma1, mu2, sigma2)
36
37 # 繪圖
38
39 plt.plot(x, y1, label='Class 1')
40 plt.plot(x, y2, label='Class 2')
41 plt.axvline(discriminant_point, color='k', linestyle='--', label='Bayes Discriminant')
42 plt.legend()
43 plt.xlabel('x')
44 plt.ylabel('Probability density')
45 plt.title('Two Normal Distributions and Bayes Discriminant')
46 plt.show()

```



先建立副程式 `bayes_discriminant` 使用 `scipy.optimize.fsolve` 求解似然比等於 1 的方程，得到貝氏判別點。

再建立常態分布的平均值以及標準差，使用 `norm.pdf` 計算數值再/2，以便達成跟圖片一樣，最後使用 `Matplotlib` 繪製兩個常態分佈的曲線和貝氏判別點的垂直線。