#این کد احتمال تعلق نقطه با مختصات 2 به هر کلاس را برای دو حالت محاسبه می‌کند.  
  
import numpy as np  
from scipy.stats import norm  
  
# تعریف مقادیر  
mu1**,** sigma1 = **0, 0.8**mu2**,** sigma2 = **4, 0.6**x = **2**# تابع چگالی احتمال نرمال  
pdf1 = norm.pdf(x**,** mu1**,** sigma1)  
pdf2 = norm.pdf(x**,** mu2**,** sigma2)  
  
# حالت اول: احتمال پیشین دو کلاس برابر است  
prior1**,** prior2 = **0.5, 0.5**posterior1 = pdf1 \* prior1 / (pdf1 \* prior1 + pdf2 \* prior2)  
posterior2 = pdf2 \* prior2 / (pdf1 \* prior1 + pdf2 \* prior2)  
  
print("حالت اول: احتمال پیشین دو کلاس برابر است")  
print("احتمال تعلق به کلاس اول:"**,** posterior1)  
print("احتمال تعلق به کلاس دوم:"**,** posterior2)  
  
# حالت دوم: احتمال پیشین کلاس اول دو برابر کلاس دوم است  
prior1**,** prior2 = **0.67, 0.33**posterior1 = pdf1 \* prior1 / (pdf1 \* prior1 + pdf2 \* prior2)  
posterior2 = pdf2 \* prior2 / (pdf1 \* prior1 + pdf2 \* prior2)  
  
print("\nحالت دوم: احتمال پیشین کلاس اول دو برابر کلاس دوم است")  
print("احتمال تعلق به کلاس اول:"**,** posterior1)  
print("احتمال تعلق به کلاس دوم:"**,** posterior2)  
  
  
# برای پیدا کردن آستانه جداکننده و محاسبه 𝑃𝑒، می‌توانیم از روش‌های زیر استفاده کنیم:  
#این کد آستانه جداکننده و 𝑃𝑒 را برای هر دو حالت محاسبه می‌کند.  
  
from scipy.optimize import fsolve  
  
# تابع برای پیدا کردن آستانه جداکننده  
def threshold(x**,** mu1**,** sigma1**,** mu2**,** sigma2**,** prior1**,** prior2):  
 return norm.pdf(x**,** mu1**,** sigma1) \* prior1 - norm.pdf(x**,** mu2**,** sigma2) \* prior2  
  
# حالت اول: احتمال پیشین دو کلاس برابر است  
prior1**,** prior2 = **0.5, 0.5**thresh1 = fsolve(threshold**, 0,** args=(mu1**,** sigma1**,** mu2**,** sigma2**,** prior1**,** prior2))  
  
# حالت دوم: احتمال پیشین کلاس اول دو برابر کلاس دوم است  
prior1**,** prior2 = **0.67, 0.33**thresh2 = fsolve(threshold**, 0,** args=(mu1**,** sigma1**,** mu2**,** sigma2**,** prior1**,** prior2))  
  
print("آستانه جداکننده برای حالت اول:"**,** thresh1)  
print("آستانه جداکننده برای حالت دوم:"**,** thresh2)  
  
# محاسبه 𝑃𝑒  
Pe1 = prior1 \* (**1** - norm.cdf(thresh1**,** mu1**,** sigma1)) + prior2 \* norm.cdf(thresh1**,** mu2**,** sigma2)  
Pe2 = prior1 \* (**1** - norm.cdf(thresh2**,** mu1**,** sigma1)) + prior2 \* norm.cdf(thresh2**,** mu2**,** sigma2)  
  
print("𝑃𝑒 برای حالت اول:"**,** Pe1)  
print("𝑃𝑒 برای حالت دوم:"**,** Pe2)  
  
  
  
import numpy as np  
from scipy.stats import norm  
from scipy.optimize import fsolve  
import matplotlib.pyplot as plt  
  
# تعریف مقادیر  
mu1**,** sigma1 = **0, 0.8**mu2**,** sigma2 = **4, 0.6**x = **2**# تابع برای پیدا کردن آستانه جداکننده  
def threshold(x**,** mu1**,** sigma1**,** mu2**,** sigma2**,** prior1**,** prior2):  
 return norm.pdf(x**,** mu1**,** sigma1) \* prior1 - norm.pdf(x**,** mu2**,** sigma2) \* prior2  
  
# تعریف محدوده برای رسم نمودار  
x\_values = np.linspace(-**3, 7, 1000**)  
  
# حالت اول: احتمال پیشین دو کلاس برابر است  
prior1**,** prior2 = **0.5, 0.5**thresh1 = fsolve(threshold**,** (mu1 + mu2) / **2,** args=(mu1**,** sigma1**,** mu2**,** sigma2**,** prior1**,** prior2))  
plt.figure(figsize=(**10, 6**))  
plt.plot(x\_values**,** norm.pdf(x\_values**,** mu1**,** sigma1) \* prior1**,** label='Class 1')  
plt.plot(x\_values**,** norm.pdf(x\_values**,** mu2**,** sigma2) \* prior2**,** label='Class 2')  
plt.axvline(x=thresh1**,** color='r'**,** linestyle='--'**,** label='Threshold')  
plt.legend()  
plt.title('Class Distributions - Equal Priors')  
plt.show()  
  
# حالت دوم: احتمال پیشین کلاس اول دو برابر کلاس دوم است  
prior1**,** prior2 = **0.67, 0.33**thresh2 = fsolve(threshold**,** (mu1 + mu2) / **2,** args=(mu1**,** sigma1**,** mu2**,** sigma2**,** prior1**,** prior2))  
plt.figure(figsize=(**10, 6**))  
plt.plot(x\_values**,** norm.pdf(x\_values**,** mu1**,** sigma1) \* prior1**,** label='Class 1')  
plt.plot(x\_values**,** norm.pdf(x\_values**,** mu2**,** sigma2) \* prior2**,** label='Class 2')  
plt.axvline(x=thresh2**,** color='r'**,** linestyle='--'**,** label='Threshold')  
plt.legend()  
plt.title('Class Distributions - Unequal Priors')  
plt.show()

