Εθνικό Μετσόβιο Πολυτεχνείο

Σχολή Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών

Εξάμηνο 7ο: Μηχανική Μάθηση

1η Ομάδα Ασκήσεων

Ημερομηνία Παράδοσης: 15/11/2024

Γιάννης Πολυχρονόπουλος, el21089

1η Σειρά Ασκήσεων Μηχανική Μάθηση Γιάννης Πολυχρονόπουλος 03121089

Άσκηση 1.1

(α)

Έχουμε $y_i = x_i^T w + \varepsilon_i \xrightarrow[\text{Διανυσματική}]{\text{Μορφή}} \underline{y} = X\underline{w} + \underline{\varepsilon}$, με $\varepsilon = [\varepsilon_1, \varepsilon_2, \dots, \varepsilon_N]^T$

και $\varepsilon_i \sim N(0, \sigma^2)$

Άρα $\underline{\varepsilon} \sim N(\underline{\mu}, \Sigma)$, όπου $\underline{\mu} = 0 = E[\underline{\varepsilon}]$ και $\Sigma = \sigma^2 I_N$ καθώς $Cov(\underline{\varepsilon}) = \Sigma$

και $Cov(\varepsilon_i, \varepsilon_j) = 0$ με $i \neq j$ καθώς τα σφάλματα μεταξύ τους είναι ανεξάρτητα, και $cov(\varepsilon_i, \varepsilon_i) = \sigma$ $\forall i$.

Έχουμε το $\underline{y}$ να είναι άθροισμα ενός σταθερού διανύσματος $X\underline{w}$ και του διανύσματος σφάλματος $\underline{\varepsilon}$ όπου $\underline{\varepsilon} \sim N(\underline{0}, \sigma^2 I_N)$. Συνεπώς, το $\underline{y}$ θα ακολουθεί πολυμεταβλητή Κανονική κατανομή.

Άρα $\underline{y} \sim N(\underline{\mu_y}, \Sigma_y)$, με $\underline{\mu_y} = E[\underline{y}] = E[X \cdot \underline{w} + \underline{\varepsilon}] = E[X \cdot \underline{w}] + E[\underline{\varepsilon}]$

$$\Rightarrow \underline{\mu_y} = X \cdot \underline{w} + \begin{bmatrix} E[\varepsilon_1] \\ \vdots \\ E[\varepsilon_N] \end{bmatrix} = X \cdot \underline{w} + \underline{0} = X \cdot \underline{w}$$

Και $\Sigma_y = Cov(\underline{y}) = Cov(\underbrace{X \cdot \underline{w}}_{\text{σταθερό}} + \varepsilon) = Cov(\varepsilon) = \sigma^2 I_N$

Άρα $\underline{y} \sim N(X \cdot \underline{w}, \sigma^2 I_N) \rightarrow P(\underline{y}) = \dfrac{1}{(2\pi)^{N/2} \cdot \sigma} \cdot \exp\left(-\frac{1}{2}(\underline{y} - X \cdot \underline{w})^T \sigma^2 (\underline{y} - X\underline{w}\right.$

(β) Ορίζουμε το κόστος ελαχίστων τετραγώνων:

$$J(w) = \|\underline{\varepsilon}\|^2 = \|\underline{y} - X\underline{w}\|^2 = (\underline{y} - X\underline{w})^T(\underline{y} - X\underline{w}) = \underline{y}^T\underline{y} \boxed{- \underline{w}^T X^T \underline{y} - \underline{y}^T X \underline{w}} + \underline{w}^T X^T X \cdot \underline{w}$$

Ιδιότητα $A^T B = B^T A$

$\Rightarrow J(w) = \underline{y}^T\underline{y} - 2\underline{w}^T X \underline{y} + \underline{w}^T X^T X \underline{w}$, Παραγωγίζουμε για να βρούμε το ελάχιστο

$$\rightarrow \nabla J(w) = 0 \Rightarrow -2X^T\underline{y} + 2X^T X \underline{w} = 0 \Rightarrow \hat{\underline{w}} = (X^T X)^{-1} X^T \cdot \underline{y}$$

$rank(X) = p+1$ άρα αντιστρέψιμος

Scanned with CamScanner

(γ) Θέλουμε να δείξουμε πως $E[\hat{\underline{w}}] = \underline{w}$

Έχουμε: $E[\hat{\underline{w}}] = E[(X^TX)^{-1} \cdot X^T \cdot \underline{y}] = E[(X^TX)^{-1} X^T (X\underline{w} + \underline{\varepsilon})]$,

$$\Rightarrow E[\hat{\underline{w}}] = E[\underbrace{(X^TX)^{-1} \cdot X^T X}_{I_{N\times(p+1)}} \cdot \underline{w} + (X^TX)^{-1} X^T \cdot \underline{\varepsilon}] = \underline{w} + (X^TX)^{-1}X^T \cdot \cancel{E[\underline{\varepsilon}]}^{0}$$

έχουμε $\hat{w} = w + (X^TX)^{-1}X^T\underline{\varepsilon}$ (1)

$$\Rightarrow E[\hat{\underline{w}}] = \underline{w}$$

~~• Υπολογισμός $Cov(\hat{\underline{w}})$:~~

~~Από ορισμό $Cov(\hat{w}) = E[(\hat{w} - E[\hat{w}])(\hat{w} - E[\hat{w}])^T]$~~

~~$\Rightarrow Cov(\hat{w}) = E[(\hat{w} - w)(\hat{w} - w)^T]$ (1)~~

• Υπολογισμός του $Cov(\hat{\underline{w}})$: $Cov(\hat{\underline{w}}) \overset{(1)}{=} Cov(\underline{w} + (X^TX)^{-1} X^T \cdot \underline{\varepsilon})$, Ιδιότητα: $Cov(AX + \underline{a}) = A\,Cov(X)A^T$ με A, a σταθερά

$$\Rightarrow Cov(\hat{\underline{w}}) = (X^TX)^{-1}X^T \cdot Cov(\underline{\varepsilon}) \cdot X(X^TX)^{-1}$$

$$\Rightarrow Cov(\hat{\underline{w}}) = (X^TX)^{-1}X^T \sigma^2 I_N \cdot X(X^TX)^{-1} = \sigma^2 \underbrace{(X^TX)^{-1} \cdot X^T X}_{I} \cdot (X^TX)^{-1}$$

$$\Rightarrow Cov(\hat{\underline{w}}) = \sigma^2 \cdot (X^T \cdot X)^{-1}$$

Άρα $\hat{w} \sim N(\underline{w}, \sigma^2 (X^T \cdot X)^{-1})$

(δ) Έχουμε την σχέση $\hat{\underline{y}}_i = x_i^T \cdot \hat{w}$ άρα $\hat{\underline{y}} = X \cdot \hat{\underline{w}} \Rightarrow$ το $\hat{\underline{y}}$ ακολουθεί κανονική κατανομή

$$E[\hat{\underline{y}}] = E[X \cdot \hat{\underline{w}}] = X \cdot E[\hat{\underline{w}}] = X \cdot \underline{w} \Rightarrow \boxed{E[\hat{\underline{y}}] = X \cdot \underline{w}}$$

$$Cov(\hat{\underline{y}}) = Cov(X \cdot \hat{\underline{w}}) = X\,Cov(\hat{\underline{w}}) \cdot X^T = X \sigma^2 (X^T X)^{-1} X^T \Rightarrow \boxed{Cov(\hat{\underline{y}}) = \sigma^2 X (X^T X)^{-1} X^T}$$

Άρα $\hat{\underline{y}} \sim N(X\underline{w}, \sigma^2 X (X^T X)^{-1} X^T)$. Σε συνέχεια θα δείξουμε πως το $\hat{y}$ είναι προβολή του $\underline{y}$ στο χώρο στηλών του X.

• Αρκεί να δείξουμε ότι: $\hat{\underline{y}} = P_X(\underline{y}) = X (X^T X)^{-1} X^T \cdot \underline{y}$

Πράγματι: $\hat{\underline{y}} = X \cdot \hat{\underline{w}} = X (X^T X)^{-1} X^T \cdot \underline{y} = P_X(\underline{y})$

Scanned with CamScanner

(ε) Έχουμε $\hat{\varepsilon}_i = y_i - \hat{y}_i \xrightarrow{\text{Δηλαδή}} \underline{\hat{\varepsilon}} = \underline{y} - \underline{\hat{y}}$, $y, \hat{y}$ ακολουθούν κανονική κατανομή άρα και $\hat{\varepsilon}$ ακολουθεί κανονική κατανομή

- $E[\underline{\hat{\varepsilon}}] = E[\underline{y} - \underline{\hat{y}}] = E[\underline{y}] - E[\underline{\hat{y}}] = X\underline{w} - X\underline{w} = \underline{0}$

- $Cov(\underline{\hat{\varepsilon}}) = Cov(\underline{y} - \underline{\hat{y}}) = Cov\left(X\underline{w} + \underline{\varepsilon} - X(X^T \cdot X)^{-1} \cdot X^T \cdot (X\underline{w} + \underline{\varepsilon})\right)$

$\Rightarrow Cov(\underline{\hat{\varepsilon}}) = Cov(X\underline{w} + \underline{\varepsilon} - X(X^T X)^{-1} \cdot X^T \cdot X \cdot \underline{w} - X(X^T \cdot X)^{-1} \cdot X^T \cdot \underline{\varepsilon}$

$\Rightarrow Cov(\underline{\hat{\varepsilon}}) = Cov(\cancel{X\underline{w}} + \underline{\varepsilon} - \cancel{X\underline{w}} - X(X^T X)^{-1} \cdot X^T \cdot \underline{\varepsilon})$ → θέτω $X(X^T \cdot X)^{-1} \cdot X^T = P$

$\Rightarrow \underbrace{Cov(\underline{\hat{\varepsilon}}) = Cov((I-P)\underline{\varepsilon})}_{\boxed{\underline{\hat{\varepsilon}} = (I-P) \cdot \underline{\varepsilon}}} = (I-P) \cdot Cov(\underline{\varepsilon}) \cdot (I-P)^T = \sigma^2 (I-P)(I-P)^T$

$\Rightarrow Cov(\hat{\varepsilon}) = \sigma^2 (I-P)(I^T - P^T) \underset{I^T = I}{\overset{P^T = P}{=}} \sigma^2 (I-P)(I-P)$

$\Rightarrow Cov(\underline{\hat{\varepsilon}}) = \sigma^2 (I \cdot I - IP - PI + P \cdot P) = \sigma^2 (I - P - P + P \cdot P)$

$(*)\; P \cdot P = X \cdot \underbrace{(X^T \cdot X)^{-1} \cdot X^T \cdot X}_{I} \cdot (X^T \cdot X)^{-1} \cdot X^T = X(X^T \cdot X)^{-1} \cdot X^T = P$

$\overset{(*)}{\Rightarrow} Cov(\underline{\hat{\varepsilon}}) = \sigma^2 (I - P)$ άρα $\hat{\varepsilon} \sim N(0, \sigma^2 (I-P))$

- Θα δείξουμε πως $\underline{\hat{\varepsilon}}^T \cdot \underline{\varepsilon} = \underline{y}^T (I-P) \underline{y}$

→ $\underline{\hat{\varepsilon}}^T \cdot \underline{\hat{\varepsilon}} = (\underline{y} - \underline{\hat{y}})^T (\underline{y} - \underline{\hat{y}}) = (\underline{y} - P\underline{y})^T (\underline{y} - P\underline{y})$

$\Rightarrow \underline{\hat{\varepsilon}}^T \cdot \underline{\hat{\varepsilon}} = (\underline{y}^T - \underline{y}^T P^T) \cdot (\underline{y} - P\underline{y}) = (\underline{y}^T (I - P^T)) \cdot ((I-P)\underline{y})$

$\overset{P^T = P}{\Rightarrow} \underline{\hat{\varepsilon}}^T \cdot \underline{\hat{\varepsilon}} = \underline{y}^T \underbrace{(I-P)(I-P)}_{I^2 - IP - PI + P^2 = (I-P)} \underline{y} = \underline{y}^T (I-P) \cdot \underline{y}$

, όπως δείξαμε και προηγουμένως

Ασκηση 1.2

Το διανυσμα $w=\begin{bmatrix} x \\ y \end{bmatrix}$ στον $R^2 (l=2)$ ακολουθει κανονικη κατανομη $N(\underline{\mu}, \Sigma)$

με $\underline{\mu} = \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}$, και $\Sigma = \begin{bmatrix} cov(x,x) & cov(x,y) \\ cov(y,x) & cov(y,y) \end{bmatrix} = \begin{bmatrix} \sigma_x^2 & \rho_{x,y}\cdot\sigma_x\cdot\sigma_y \\ \rho_{x,y}\cdot\sigma_x\sigma_y & \sigma_y^2 \end{bmatrix} = \begin{bmatrix} 9 & 7,2 \\ 7,2 & 16 \end{bmatrix}$

$\Rightarrow \underline{\mu} = \begin{bmatrix} 5 \\ 10 \end{bmatrix}$

Επειδη $\boxed{\rho_{x,y} = \dfrac{cov(x,y)}{\sigma_x \cdot \sigma_y}}$

Άρα $$p(\underline{w}) = \frac{1}{(2\pi)^{2/2}\ |\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T \cdot \Sigma^{-1} \cdot (x-\mu)\right)$$

Απο το wikipedia για 2-D η συναρτηση πυκνότητας πιθανότητας είναι:

$$p(x,y) = \frac{1}{2\pi\cdot\sigma_x\cdot\sigma_y\sqrt{1-\rho^2}} \exp\left(-\frac{1}{2[1-\rho^2]}\left[\left(\frac{x-\mu_x}{\sigma_x}\right)^2 - 2\rho\left(\frac{x-\mu_x}{\sigma_x}\right)\left(\frac{y-\mu_y}{\sigma_y}\right) + \left(\frac{y-\mu_y}{\sigma_y}\right)^2\right]\right)$$

Άρα $$p(x,y) = \frac{1}{19.2\pi} \cdot \exp\left(-\frac{1}{1.28}\left[\left(\frac{x-5}{3}\right)^2 - 1.2\left(\frac{x-5}{3}\right)\left(\frac{y-10}{5}\right) + \left(\frac{y-10}{5}\right)^2\right]\right)$$

(B) Απο διαφάνειες έχουμε πως $(\underline{x_1}, \underline{x_2} = \underline{a}) \sim N(\hat{\underline{\mu}}, \hat{\Sigma})$, με

$\hat{\mu} = \underline{\mu_1} + \Sigma_{12}\Sigma_{22}^{-1}(\underline{a} - \mu_2)$ και $\hat{\Sigma} = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}$

Για 2-D απο το wikepedia οι πράξεις απλοποιούνται και έχουμε:

$$y|x=a \sim N\left(\mu_y + \frac{\sigma_y}{\sigma_x}\cdot\rho(a-\mu_x),\ (1-\rho^2)\sigma_y^2\right)$$
$$\sim N\left(10 + \frac{4}{3}\cdot 0,6(6-5),\ (1-0,6^2)\cdot 4^2\right)$$
$$\sim N(10,8,\ 10,24)$$

Άρα $$p(y|x=6) = \frac{1}{\sqrt{10{,}24\cdot 2\pi}} \exp\left(-\frac{(y-10{,}8)^2}{20{,}48}\right)$$

Για $P(y>10|x=6)$ υπολογίζουμε το $P(y>10|x=6) = \int_{10}^{100} P(y>10|x=6)\,dy$

Με python script υπολογίσαμε $P(y>10|x=6) = 0.5987$

Scanned with CamScanner

(γ) Αντίστοιχα $X|y=9 \sim N\left(\mu_x + \frac{\sigma_x}{\sigma_y}\cdot\rho(a-\mu_y),\ (1-\rho^2)\sigma_x^2\right)$

$$\sim N\left(5+\frac{3}{4}\cdot 0{,}6(9-10),\ (1-0{,}6^2)\cdot 9\right)$$

$$\sim N(4{,}55,\ 5{,}76)$$

⊛ $\left(\frac{x-5}{3}\right)^2 - 1.2\left(\frac{x-5}{3}\right)\left(\frac{y-10}{5}\right) + \left(\frac{y-10}{5}\right)^2 = 0.3111$

Άρα $p(x|y=9) = \frac{1}{\sqrt{2\pi\cdot 5{,}76}}\exp\left(-\frac{(x-4{,}55)^2}{11{,}52}\right)$

Για $P(x>5|y=9) = \int_5^{100} p(x|y=9)\,dx = 0{,}4256$ με Python script

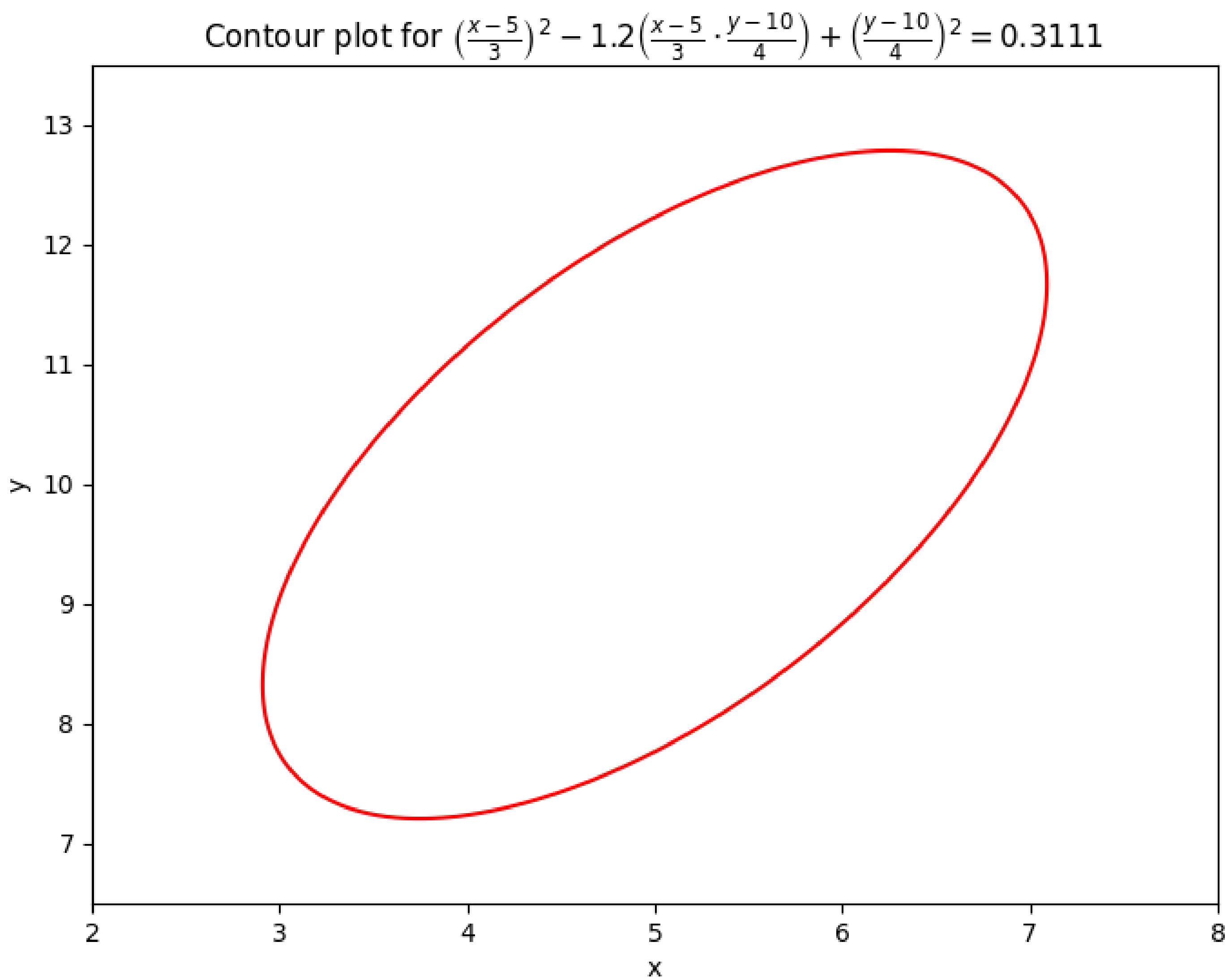
(δ) Χρησιμοποιήθηκε Python Script. Για $(x,y)=(6,12)$: $p(x,y) = 0.013$

⊛

Άρα ζωγραφίζουμε: $\frac{1}{19.2\pi}\cdot\exp\left(-\frac{1}{1.28}\left[\left(\frac{x-5}{3}\right)^2 - 1.2\left(\frac{x-5}{3}\right)\left(\frac{y-10}{4}\right) + \left(\frac{y-10}{4}\right)^2\right]\right) = 0.013$

Άσκηση 1.2, Ερώτημα Δ

Σχεδιασμός της ισοσταθμικής καμπύλης:



Ο κώδικας που χρησιμοποιήθηκε είναι ο ακόλουθος:

```python
import numpy as np
import matplotlib.pyplot as plt
# Define the function based on the given equation
def contour_function(x, y):
    term1 = ((x - 5) / 3) ** 2
    term2 = -1.2 * ((x - 5) / 3) * ((y - 10) / 5)
    term3 = ((y - 10) / 5) ** 2
    return term1 + term2 + term3

# Set up the grid of x and y values
x = np.linspace(-5, 15, 400)
y = np.linspace(0, 20, 400)
X, Y = np.meshgrid(x, y)

# Calculate Z values on the grid
Z = contour_function(X, Y)
# Define the contour level based on the given constant
contour_level = 0.111111111113
# Plotting
plt.figure(figsize=(8, 6))
contour = plt.contour(X, Y, Z, levels=[contour_level], colors='red')
plt.xlabel('x')
plt.ylabel('y')
plt.title(r'Contour Plot')
plt.xlim([2.5, 7.5])
plt.ylim([7.5,12.5])
plt.show()
```

Για τα προηγούμενα ερωτήματα οι υπολογισμοί έγιναν με τον ακόλουθο κώδικα

```python
from scipy.stats import norm
from math import sqrt
# (b)
y = 10 # y > 10
mean_y_given_x = 10.8
std_y_given_x = sqrt(10.24)
prob = norm.sf(y, loc=mean_y_given_x, scale=std_y_given_x)
print(f"Probability is : {prob:.4f}")
# (c)
x = 5 # x > 5
mean_y_given_x = 4.55
std_y_given_x = sqrt(5.76)
prob = norm.sf(x, loc=mean_y_given_x, scale=std_y_given_x)
print(f"Probability is : {prob:.4f}")
```

Άσκηση 1.3

(α) Θέλουμε να υπολογίσουμε το $P(w_i|\underline{x})$ με $i=1,2,3$ και έχουμε ~~τις~~ $P(w_i)$ και τις κατανομές $\underline{x}|w_i$

Από τον κανόνα του Bayes $P(w_i|\underline{x}) = \dfrac{P(\underline{x}|w_i)\cdot P(w_i)}{P(\underline{x})}$

Όπου $P(\underline{x}) = \sum_{i=1}^{3} P(\underline{x}|w_i)\cdot P(w_i)$, από τον τύπο ολικής Πιθανότητας.

Μέσω Python υπολογίζουμε τις Πιθανότητες $P(w_i|\underline{x})$.

Έχουμε: $P(w_1|\underline{x}) = 0{,}0618$, $P(w_2|\underline{x}) = 0{,}5864$, $P(w_3|\underline{x}) = 0{,}3518$

(β) Σύμφωνα με τον ταξινομητή Bayes το $x=(3,5)$ θα ταξινομηθεί στην ~~κλά~~ κλάση w_2, που αποτελεί την μεγαλύτερη posterior πιθανότητα για το $\underline{x}$

(γ) Για τις καμπύλες απόφασης πρέπει να δούμε πότε ισούνται οι πιθανότητες $P(w_i|\underline{x})$

- Για w_1, w_2: $P(w_1|\underline{x}) = P(w_2|\underline{x}) \Rightarrow P(\underline{x}|w_1)\cdot P(w_1) = P(\underline{x}|w_2)\cdot P(w_2)$ (1)
- Για w_2, w_3: $P(w_2|\underline{x}) = P(w_3|\underline{x}) \Rightarrow P(\underline{x}|w_2)\cdot P(w_2) = P(\underline{x}|w_3)\cdot P(w_3)$ (2)
- Για w_1, w_3: $P(w_1|\underline{x}) = P(w_3|\underline{x}) \Rightarrow P(\underline{x}|w_1)\cdot P(w_1) = P(\underline{x}|w_3)\cdot P(w_3)$ (3)

Scanned with CamScanner

Για $x=(x_1,x_2)$

- $$P(\underline{x}|\omega_1) = \frac{1}{2\pi\cdot 1\cdot 1\cdot\sqrt{1-0{,}5^2}} \exp\left(-\frac{1}{2[1-0{,}5^2]}\left[\left(\frac{x_1-3}{1}\right)^2 - 2\cdot 0{,}5\left(\frac{x_1-3}{1}\right)\left(\frac{x_2+2}{1}\right) + \left(\frac{x_2+2}{1}\right)^2\right]\right)$$

(χρησιμοποιούμε τον τύπο που είδαμε στην άσκηση 1.2 ερώτημα (α))

- $$P(\underline{x}|\omega_2) = \frac{1}{2\pi\cdot 1\cdot 1\sqrt{1-0{,}5^2}} \exp\left(-\frac{1}{2(1-0{,}5^2)}\left[\left(\frac{x_1-4}{1}\right)^2 - 2\cdot 0{,}5\cdot\left(\frac{x_1-4}{1}\right)\left(\frac{x_2-3}{1}\right) + \left(\frac{x_2-3}{1}\right)^2\right]\right)$$

- $$P(\underline{x}|\omega_3) = \frac{1}{2\pi\cdot 1\cdot 1\sqrt{1-0{,}5^2}} \exp\left(-\frac{1}{2(1-0{,}5^2)}\left[\left(\frac{x_1-6}{1}\right)^2 - 2\cdot 0{,}5\left(\frac{x_1-6}{1}\right)\left(\frac{x_2-7}{1}\right) + \left(\frac{x_2-7}{1}\right)^2\right]\right)$$

Άρα ① $\Rightarrow$

$$\exp\left(-\frac{1}{1{,}5}\left[(x_1-3)^2 - (x_1-3)(x_2+2) + (x_2+2)^2\right]\right)\cdot P(\omega_1) = \exp\left(-\frac{1}{1{,}5}\left[(x_1-4)^2 - (x_1-4)(x_2-3) + (x_2-3\right.\right.$$
$$\cdot P(\omega_2)$$

$$\overset{\ln}{\Rightarrow} -\frac{1}{1{,}5}\left[(x_1-3)^2 - (x_1-3)(x_2+2) + (x_2+2)^2 - (x_1-4)^2 + (x_1-4)(x_2-3) - (x_2-3)^2\right] = \ln\left(\frac{P(\omega_2)}{P(\omega_1)}\right)$$

$$\Rightarrow (\cancel{x_1^2} - 6x_1 + 9) - (\cancel{x_1x_2} + 2x_1 - 3x_2 - 6) + (\cancel{x_2^2} + \cancel{4x_2} + 4) - (\cancel{x_1^2} - \cancel{8x_1} + 16) + (\cancel{x_1x_2} - 3x_1 - \cancel{4x_2}$$
$$- (\cancel{x_2^2} - 6x_2 + 9) = \ln\left(\frac{0{,}5}{0{,}2}\right)\cdot(-$$

$$\Rightarrow \cancel{9} + 3x_2 + 6 + 4 - 16 - 3x_1 + 12 + 6x_2 - \cancel{9} = (-1{,}5)\cdot\ln\left(\frac{0{,}5}{0{,}2}\right)$$

$$\Rightarrow -3x_1 + 9x_2 + 6 = (-1{,}5)\ln(2{,}5)$$

$$\Rightarrow \boxed{3x_1 - 9x_2 = 1{,}5\ln(2{,}5) + 6}$$ για ω_1, ω_2

② $\overset{\text{Παρομοίως}}{\Rightarrow}$ $$\boxed{x_2 = 5 - \frac{1{,}5}{6}\cdot\ln\left(\frac{0{,}3}{0{,}5}\right)}$$ για ω_2, ω_3

③ $\Rightarrow$ για ω_1, ω_3

$$-3x_1 + 15x_2 = 24 - 1.5\ln\left(\frac{0{,}3}{0{,}2}\right)$$

Scanned with CamScanner

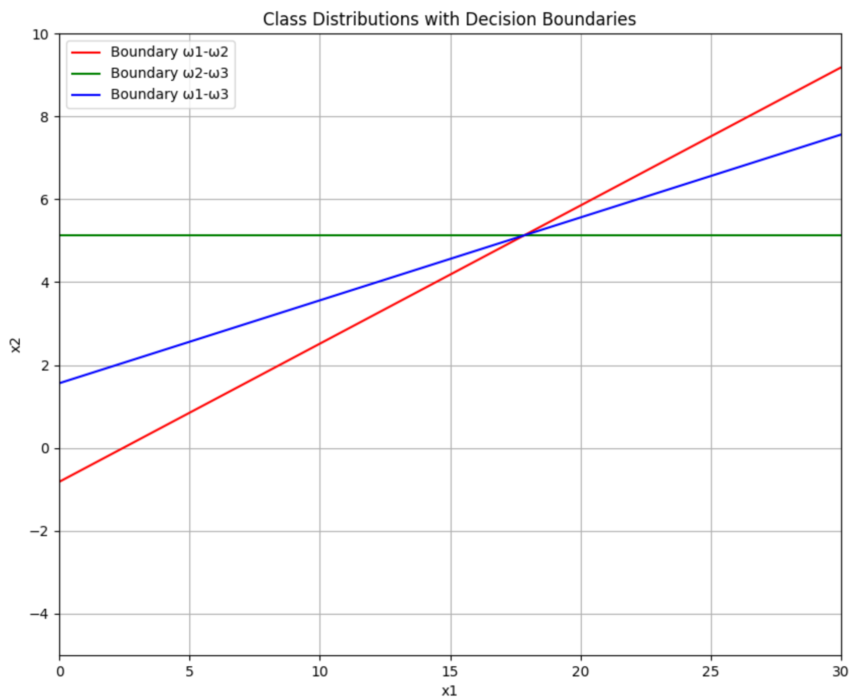
Άσκηση 1.3

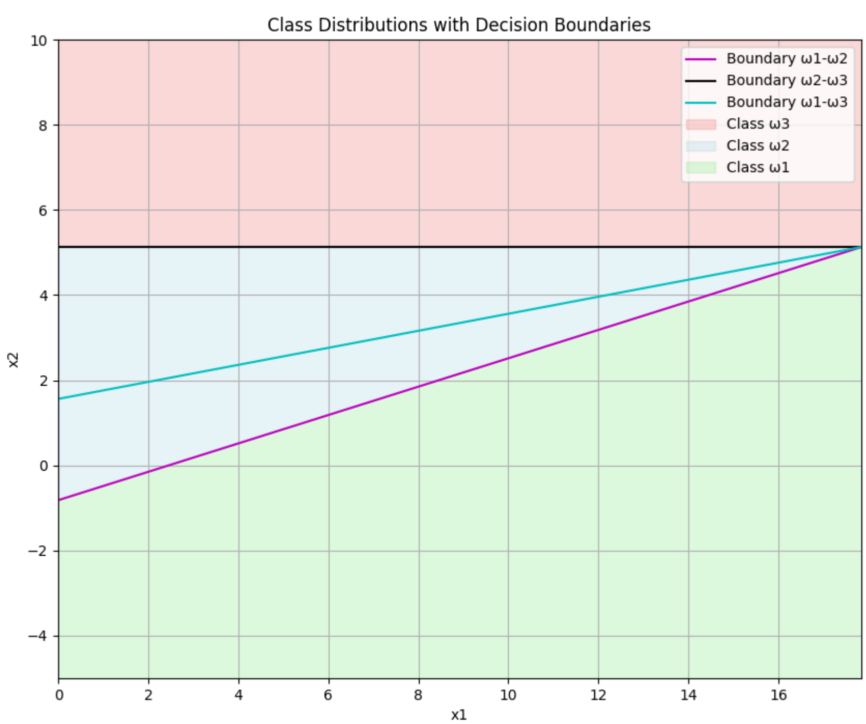
(α) Python Code για το υπολογισμό πιθανοτήτων

```python
import numpy as np

from scipy.stats import multivariate_normal

# Given data
mu1 = np.array([3, 2])
mu2 = np.array([4, 3])
mu3 = np.array([6, 7])
cov_matrix = np.array([[1, 0.5], [0.5, 1]])
priors = np.array([0.2, 0.5, 0.3])
x = np.array([3, 5])

# Calculate likelihoods
p_x_given_w1 = multivariate_normal.pdf(x, mean=mu1, cov=cov_matrix)
p_x_given_w2 = multivariate_normal.pdf(x, mean=mu2, cov=cov_matrix)
p_x_given_w3 = multivariate_normal.pdf(x, mean=mu3, cov=cov_matrix)

# Calculate posteriors using Bayes' rule
evidence = p_x_given_w1 * priors[0] + p_x_given_w2 * priors[1] +
p_x_given_w3 * priors[2]
p_w1_given_x = (p_x_given_w1 * priors[0]) / evidence
p_w2_given_x = (p_x_given_w2 * priors[1]) / evidence
p_w3_given_x = (p_x_given_w3 * priors[2]) / evidence

print("Posterior probabilities:")
print(f"P(ω1 | x) = {p_w1_given_x:.4f}")
print(f"P(ω2 | x) = {p_w2_given_x:.4f}")
print(f"P(ω3 | x) = {p_w3_given_x:.4f}")
```

```
Output:

Posterior probabilities:
P(ω1 | x) = 0.0618
P(ω2 | x) = 0.5864
P(ω3 | x) = 0.3518
```

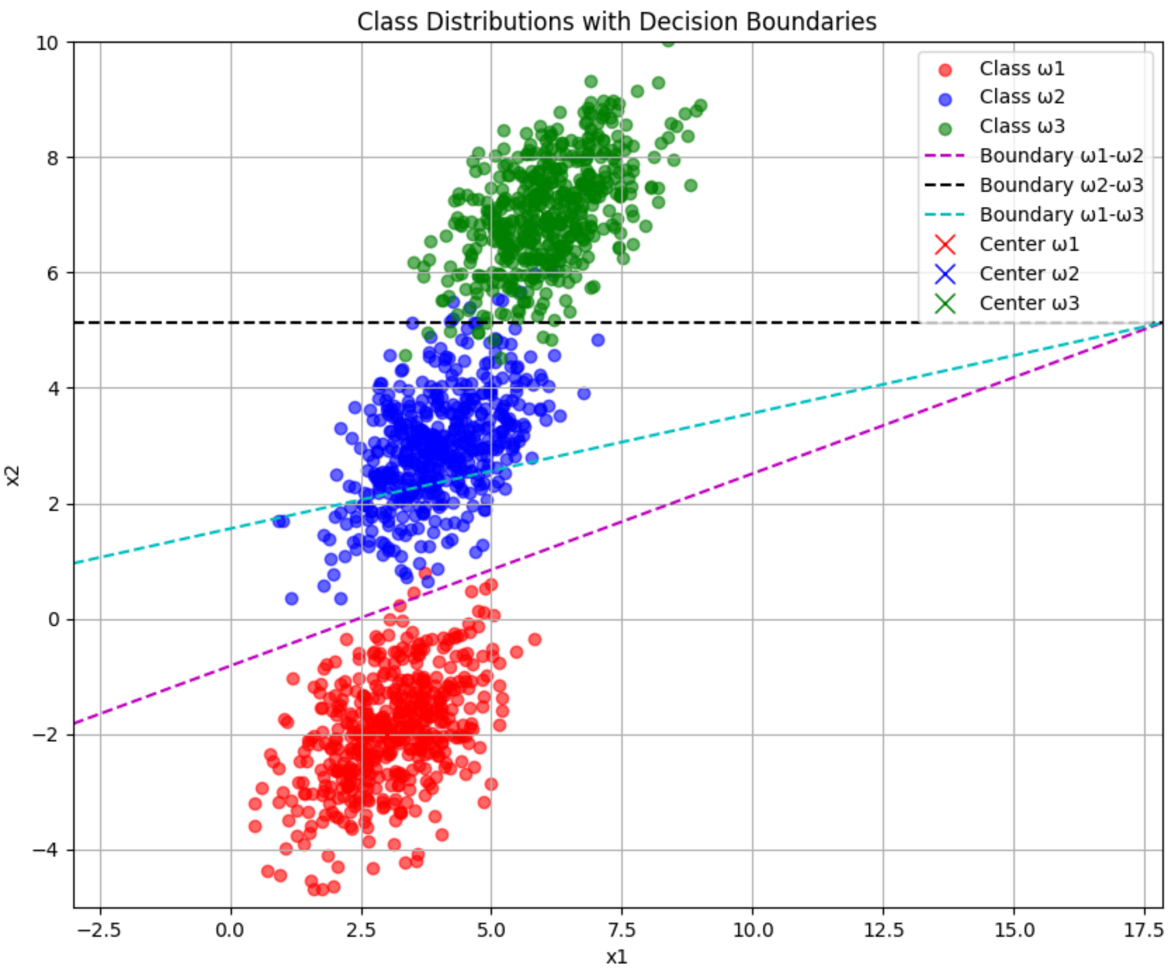
(γ) Παρακάτω φαίνονται οι καμπύλες απόφασης που βρήκαμε.





```python
import numpy as np
import matplotlib.pyplot as plt
x_vals = np.linspace(0, 30, 1000)
# Boundary line between ω1 and ω2: 3x1 - 9x2 = 7.3744
boundary12_y = (3 * x_vals - 7.3744) / 9
# Boundary line between ω2 and ω3: x2 = 5.1277 (horizontal line)
boundary23_y = np.full_like(x_vals, 5.1277)
# Boundary line between ω1 and ω3: -3x1 + 15x2 = 23.3918
boundary13_y = (3 * x_vals + 23.3918) / 15
# Plotting
plt.figure(figsize=(10, 8))
# Plot decision boundaries
plt.plot(x_vals, boundary12_y, 'r', label='Boundary ω1-ω2')
plt.plot(x_vals, boundary23_y, 'g', label='Boundary ω2-ω3')
plt.plot(x_vals, boundary13_y, 'b', label='Boundary ω1-ω3')
# Additional plot settings
plt.xlabel('x1')
plt.ylabel('x2')
plt.title('Class Distributions with Decision Boundaries')
plt.legend()
plt.grid(True)
plt.xlim(0, 30)
plt.ylim(-5, 10)
# Plotting
plt.figure(figsize=(10, 8))
# Plot decision boundaries
plt.plot(x_vals, boundary12_y, 'm', label='Boundary ω1-ω2')
plt.plot(x_vals, boundary23_y, 'k', label='Boundary ω2-ω3')
plt.plot(x_vals, boundary13_y, 'c', label='Boundary ω1-ω3')
# # Fill areas based on classification regions
plt.fill_between(x_vals, boundary23_y, 10, color='lightcoral', alpha=0.3,
label='Class ω3')
plt.fill_between(x_vals, boundary12_y, boundary23_y, where=(boundary12_y
<= boundary23_y),color='lightblue', alpha=0.3, label='Class ω2')
plt.fill_between(x_vals, -10, boundary12_y, color='lightgreen', alpha=0.3,
label='Class ω1'
# Additional plot settings
plt.xlabel('x1')
plt.ylabel('x2')
plt.title('Class Distributions with Decision Boundaries')
plt.legend()
plt.grid(True)
plt.xlim(0, 17.84126)
plt.ylim(-5, 10)
plt.show()
```

(δ) Δίνεται το ζητούμενο σχήμα:



Για το ζητούμενο σχήμα χρησιμοποιήθηκε ο εξής κώδικας:

```
import numpy as np
import matplotlib.pyplot as plt

# Define parameters for each class
mean1 = np.array([3, -2])
mean2 = np.array([4, 3])
mean3 = np.array([6, 7])
cov_matrix = np.array([[1, 0.5], [0.5, 1]])  # Shared covariance matrix
priors = np.array([0.2, 0.5, 0.3])
```

```python
# Generate 500 points for each class
np.random.seed(0)  # For reproducibility
p_x_given_w1 = np.random.multivariate_normal(mean1, cov_matrix, 500)
p_x_given_w2 = np.random.multivariate_normal(mean2, cov_matrix, 500)
p_x_given_w3 = np.random.multivariate_normal(mean3, cov_matrix, 500)

# Define decision boundaries as lines based on given equations
x_vals = np.linspace(-3, 20, 1000)
# Boundary line between ω1 and ω2: 3x1 - 9x2 = 7.3744
boundary12_y = (3 * x_vals - 7.3744) / 9
# Boundary line between ω2 and ω3: x2 = 5.1277 (horizontal line)
boundary23_y = np.full_like(x_vals, 5.1277)
# Boundary line between ω1 and ω3: -3x1 + 15x2 = 23.3918
boundary13_y = (3 * x_vals + 23.3918) / 15
# Plotting
plt.figure(figsize=(10, 8))
# Scatter plot for each class
plt.scatter(p_x_given_w1[:, 0], p_x_given_w1[:, 1], color='red',
label='Class ω1', alpha=0.6)
plt.scatter(p_x_given_w2[:, 0], p_x_given_w2[:, 1], color='blue',
label='Class ω2', alpha=0.6)
plt.scatter(p_x_given_w3[:, 0], p_x_given_w3[:, 1], color='green',
label='Class ω3', alpha=0.6)

# Plot decision boundaries
plt.plot(x_vals, boundary12_y, 'm--', label='Boundary ω1-ω2')
plt.plot(x_vals, boundary23_y, 'k--', label='Boundary ω2-ω3')
plt.plot(x_vals, boundary13_y, 'c--', label='Boundary ω1-ω3')
# Plot centers of each class
plt.plot(mean1[0], mean1[1], 'ro', marker='x', markersize=10,
label='Center ω1')
plt.plot(mean2[0], mean2[1], 'bo', marker='x', markersize=10,
label='Center ω2')
plt.plot(mean3[0], mean3[1], 'go', marker='x', markersize=10,
label='Center ω3')

# Additional plot settings
plt.xlabel('x1')
plt.ylabel('x2')
plt.title('Class Distributions with Decision Boundaries')
plt.legend()
plt.grid(True)
plt.xlim(-3, 17.84126)
plt.ylim(-5, 10)
plt.show()
```

(ε) Υπολογίστηκε με δύο τρόπους. Αρχικά, πήρα 10000 δείγματα από την κατανομή $x \mid \omega_2 \sim N(\boldsymbol{\mu_2}, \Sigma)$ και βλέπουμε το ποσοστό που αυτά σύμφωνα με τον ταξινομιτή Bayes γίνονται misclassified.

Αποτέλεσμα: Probability of misclassification of Class w2 is: 0.0191

Κώδικας:

```
import numpy as np
from scipy.stats import multivariate_normal
# Data
mu1 = np.array([3, -2])
mu2 = np.array([4, 3])
mu3 = np.array([6, 7])
sigma = np.array([[1, 0.5], [0.5, 1]])
priors = [0.2, 0.5, 0.3]

# Creating 10000 samples
num_samples = 10000
samples = multivariate_normal.rvs(mean=mu2, cov=sigma, size=num_samples)

# Miclassification initialization
misclassified_count = 0

# Compute posterior probability and classify them with Bayes
for x in samples:
    # Computation of PDF for every distribution
    p_x_given_w1 = multivariate_normal.pdf(x, mean=mu1, cov=sigma) *
priors[0]
    p_x_given_w2 = multivariate_normal.pdf(x, mean=mu2, cov=sigma) *
priors[1]
    p_x_given_w3 = multivariate_normal.pdf(x, mean=mu3, cov=sigma) *
priors[2]

    # Classification with Bayes classifier
    predicted_class = np.argmax([p_x_given_w1, p_x_given_w2,
p_x_given_w3])

    # If misclassification then increase variable
    if predicted_class != 1:
        misclassified_count += 1
# Final Output
error_probability = misclassified_count / num_samples
print("Probability of misclassification of Class w2 is:",
error_probability)
```

Ο δεύτερος τρόπος είναι ο υπολογισμός του ολοκληρώματος $p(x\,|\omega_2)$ στις περιοχές που δεν ανήκει στην κλάση ω_2.

Αποτέλεσμα: Probability of Misclaffication is: 0.01909033805483369

Κώδικας:
```
import numpy as np

from scipy.stats import multivariate_normal
from scipy.integrate import dblquad
# Given parameters
mean_w2 = [4, 3]
cov_matrix = [[1, 0.5], [0.5, 1]]
# Define the distribution for p(x|w2)
rv_w2 = multivariate_normal(mean=mean_w2, cov=cov_matrix)

# Define the boundaries
boundary12 = lambda x1: (3 * x1 - 7.3744) / 9  # Boundary ω1-ω2
boundary23 = 5.1277  # Boundary ω2-ω3

# Misclassification region for ω3 (x2 > boundary23)
def integrand_w3(x2, x1):
    return rv_w2.pdf([x1, x2])

# Misclassification region for ω1 (x2 < boundary12)
def integrand_w1(x2, x1):
    return rv_w2.pdf([x1, x2])

# Integration limits
x1_min, x1_max = -np.inf, 17.84126
# Compute the probability for ω3 region
P_error_w3, _ = dblquad(
    integrand_w3,
    x1_min, x1_max,
    lambda x1: boundary23,   # Lower bound for x2
    lambda x1: np.inf        # Upper bound for x2
)
# Compute the probability for ω1 region
P_error_w1, _ = dblquad(
    integrand_w1,
    x1_min, x1_max,
    lambda x1: -np.inf,             # Lower bound for x2
    lambda x1: boundary12(x1)       # Upper bound for x2
)
# Total misclassification probability
P_misclassification = P_error_w1 + P_error_w3
print(f"Probability of Misclassification is: {P_misclassification}")
```

Άσκηση 1.4

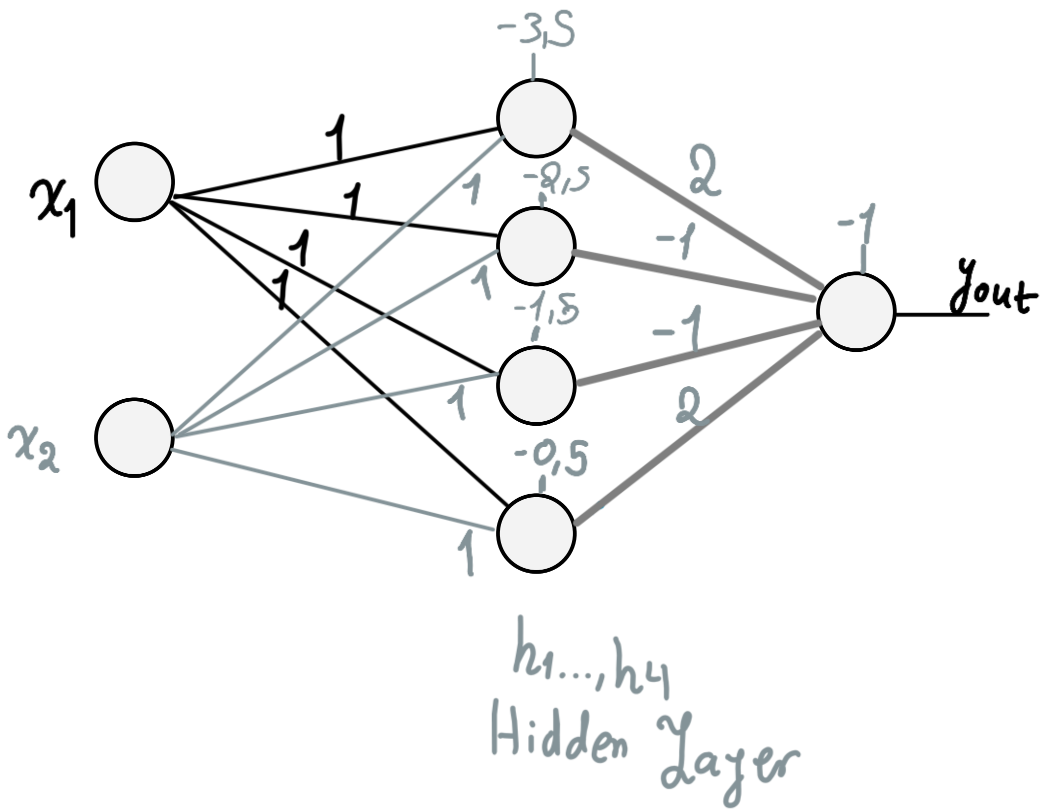
Ερώτημα Α

Πίνακας Input – Output.

$x1$	$x2$	$y = x_1 + x_2 \; mod \; 3$
0	0	0
0	1	1
0	2	2
1	0	1
1	1	2
1	2	0
2	0	2
2	1	0
2	2	1

Ορίζουμε ως συνάρτηση ενεργοποίησης την ακόλουθη:

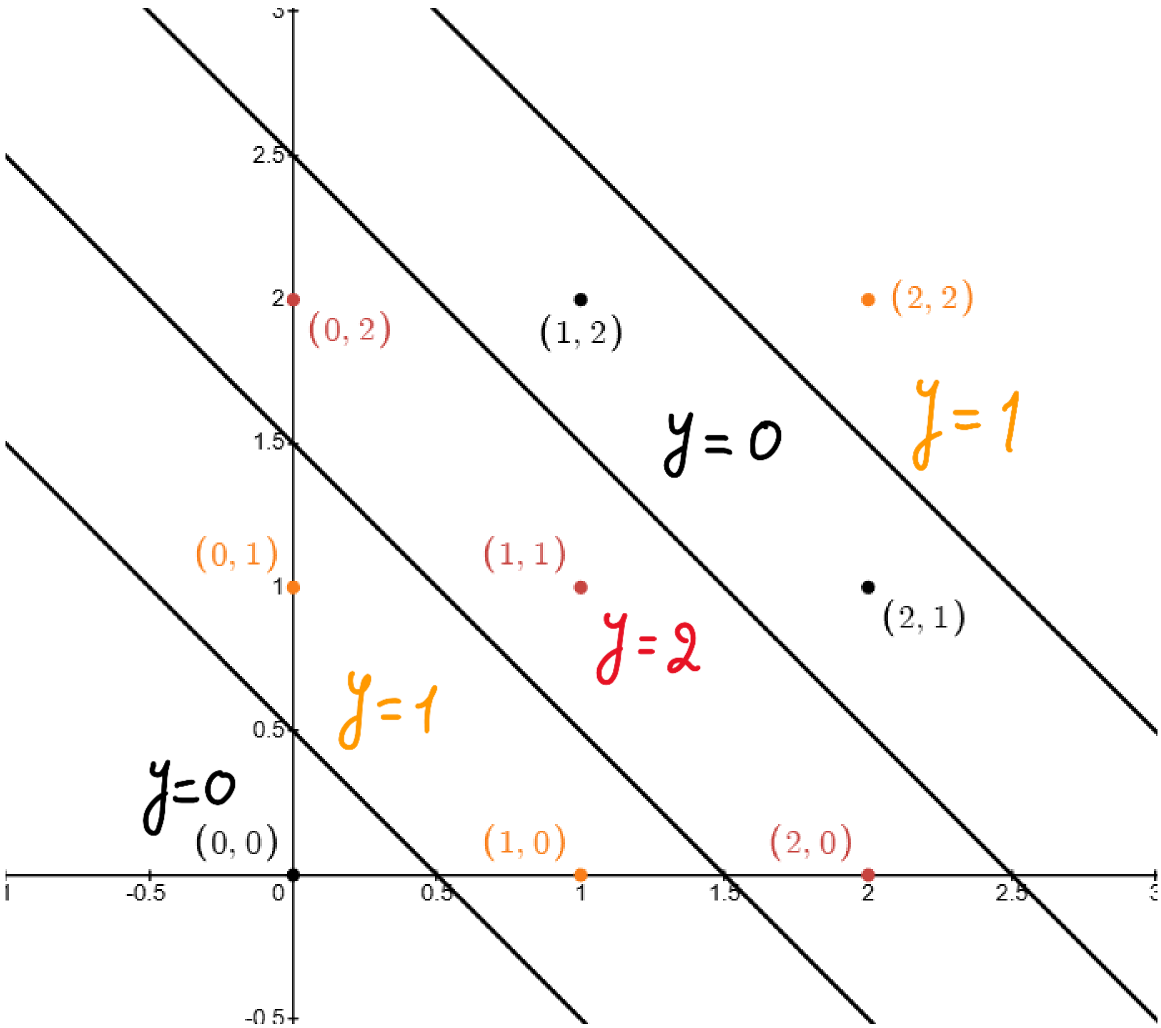
$$f(x) = \begin{cases} 0, & x < 0 \\ 1, & x > 0 \\ 2, & x = 0 \end{cases}$$



Άρα, έχουμε αναλυτικά:

- $w_{i,j} = 1\ \forall i, j\ \kappa\alpha\iota\ b_{h_1} = -3.5, b_{h_2} = -2.5, b_{h_3} = -1.5, b_{h_4} = -0.5$
- $a_1 = 2,\ \ a_2 = \ -1,\ \ a_3 = \ -1,\ \ a_4 = 2$

Ερώτημα Β



Ερώτημα Γ

Κατασκευάζουμε τους ακόλουθους πίνακες για να ελέγξουμε όλες τις τιμές των x_1, x_2

$x1$	$x2$	h_1	$f(h_1)$	h_2	$f(h_2)$	h_3	$f(h_3)$	h_4	$f(h_4)$
0	0	-3.5	0	-2.5	0	-1.5	0	-0.5	0
0	1	-2.5	0	-1.5	0	-0.5	0	0.5	1
0	2	-1.5	0	-0.5	0	0.5	1	1.5	1
1	0	-2.5	0	-1.5	0	-0.5	0	0.5	1
1	1	-1.5	0	-0.5	0	0.5	1	1.5	1
1	2	-0.5	0	0.5	1	1.5	1	2.5	1
2	0	-1.5	0	-0.5	0	0.5	1	1.5	1
2	1	-0.5	0	0.5	1	1.5	1	2.5	1
2	2	0.5	1	1.5	1	2.5	1	3.5	1

$f(h_1)$	$f(h_2)$	$f(h_3)$	$f(h_4)$	$y_{out} = 2f(h_1) - f(h_2) - f(h_3) + 2f(h_4) - 1$	$f(y_{out})$
0	0	0	0	-1	0
0	0	0	1	1	1
0	0	1	1	0	2
0	0	0	1	1	1
0	0	1	1	0	2
0	1	1	1	-1	0
0	0	1	1	0	2
0	1	1	1	-1	0
1	1	1	1	1	1

Συγκρίνουμε και παρατηρούμε πως το δίκτυό μας υπολογίζει σωστά τις εξόδους για κάθε δυνατό συνδυασμό εισόδων.

$f(y_{out})$	$y = x_1 + x_2 \ mod\ 3$
0	0
1	1
2	2
1	1
2	2
0	0
2	2
0	0
1	1

Ασκηση 1.5

(α)

α.1: Θέλουμε να δείξουμε πως $K(x_i,x_j)=K(x_j,x_i)$

Απο τον ορισμό $K(x_i,x_j)=\langle\phi(x_i),\phi(x_j)\rangle$, ένα εσωτερικό γινόμενο στον χώρο Q

Το εσωτερικο γινόμενο, όσο υψηλή και είναι η διάσταση, έχει την ιδιότητα της συμμετρικότητας

Δηλαδή, αν $\underline{u},\underline{v}$ διανύσματα στον Q τοτε: $\langle u,v\rangle=\langle v,u\rangle$

Άρα $K(x_i,x_j)=\langle\varphi(x_i),\varphi(x_j)\rangle=\langle\phi(x_j),\varphi(x_i)\rangle=K(x_j,x_i) \Rightarrow K(x_i,x_j)=K(x_j,x_i)$

α.2: Έχουμε $K(x_i,x_j)=\exp\left(-\frac{1}{2}\|\underline{x}_i-\underline{x}_j\|^2\right)=\langle\varphi(x_i),\varphi(x_j)\rangle$ ⓐ

$\to \|\varphi(x_i)-\varphi(x_j)\|^2 = \langle \varphi(x_i)-\varphi(x_j), \varphi(x_i)-\varphi(x_j)\rangle \overset{\text{Ιδιότητες}}{\underset{\text{Εσ. Γινομένου}}{=}}$

$= \langle\varphi(x_i),\varphi(x_i)\rangle - 2\langle\varphi(x_i),\varphi(x_j)\rangle+\langle\varphi(x_j),\varphi(x_j)\rangle$ ①

για $\langle\varphi(x_i),\varphi(x_i)\rangle \overset{ⓐ}{=} \exp\left(-\frac{1}{2}\|x_i-x_i\|^2\right)=\exp(0)=1$

$\langle\varphi(x_j),\varphi(x_j)\rangle \overset{ⓐ}{=} \exp\left(-\frac{1}{2}\|x_j-x_j\|^2\right)=\exp(0)=1$

Άρα ① $\Rightarrow \|\varphi(x_i)-\varphi(x_j)\|^2 = 2-2\langle\varphi(x_i),\varphi(x_j)\rangle = 2-2K(x_i,x_j)$ ①'

όμως $K(x_i,x_j)=\langle\varphi(x_i),\varphi(x_j)\rangle=\exp\left(-\frac{1}{2}\|x_i-x_j\|^2\right)>0 \to \in[0,1]$

Άρα $K(x_i,x_j)>0 \iff -2K(x_i,x_j)<0 \iff 2-2K(x_i,x_j)<2 \overset{①'}{\Rightarrow} \underline{\|\varphi(x_i)-\varphi(x_j)\|^2<2}$

Στην ακραία περίπτωση που η απόσταση μεταξύ x_i,x_j τείνει στο άπειρο ισχύει η ισότητα:

Άρα $\underline{\|\varphi(x_i)-\varphi(x_j)\|^2\le 2}$

Scanned with CamScanner

(β) Αν θεωρήσουμε x_{far} πολύ μακρινό από κάθε x_i έχουμε πως $\|x_{far}-x_i\|^2 \gg$

$$\Rightarrow \quad -\frac{1}{2}\|x_{far}-x_i\|^2 \ll 0 \Leftrightarrow 0 < \exp\left(-\frac{1}{2}\|x_{far}-x_i\|^2\right) \ll 1$$

$$\Rightarrow 0 < K(x_{far}, x_i) \ll 1 \rightarrow K(x_i, x_{far}) \simeq 0$$

Άρα
$$f(x_{far}; a, \hat{w}_0) = \sum_{i \in SV} y_i \cdot a_i \cancelto{\simeq 0}{K(x_i, x_{far})} + \hat{w}_0$$

$$\Rightarrow \boxed{f(x_{far}; a, \hat{w}_0) \simeq \hat{w}_0}$$