Αναγνώριση Προτύπων

2ο Σύνολο Αναλυτικών Ασκήσεων

9ο Εξάμηνο - Χειμερινό εξάμηνο 2019-20 - Ροή Σ

Αντωνιάδης Παναγιώτης (03115009 - `el15009@central.ntua.gr`)

"AI is akin to building a rocket ship. You need a huge engine and a lot of fuel. The rocket engine is the learning algorithms but the fuel is the huge amounts of data we can feed to these algorithms."

– Andrew Ng, *Computer scientist and Statistician*

Άσκηση 1 (Hidden Markov Models)

Ζητείται να χρησιμοποιηθεί ένα HMM για να αποκωδικοποιηθεί μια απλή ακολουθία DNA. Είναι γνωστό ότι μια ακολουθία DNA είναι μια σειρά από στοιχεία του συνόλου A, C, G, T. Ας υποθέσουμε ότι υπάρχει μία κρυμμένη κατάσταση S που ελέγχει τη δημιουργία της ακολουθίας DNA και έχει 2 πιθανές καταστάσεις S_1, S_2. Επίσης, δίνονται οι ακόλουθες πιθανότητες μετάβασης για το HMM λ:

$$P(S_1|S_1) = 0.8 \quad P(S_2|S_1) = 0.2 \quad P(S_1|S_2) = 0.2 \quad P(S_2|S_2) = 0.8$$

οι ακόλουθες πιθανότητες των παρατηρήσεων:

$$\begin{array}{llll} P(A|S_1) = 0.4 & P(C|S_1) = 0.1 & P(G|S_1) = 0.4 & P(T|S_1) = 0.1 \\ P(A|S_2) = 0.1 & P(C|S_2) = 0.4 & P(G|S_2) = 0.1 & P(T|S_2) = 0.4 \end{array}$$

και οι ακόλουθες a-priori πιθανότητες:

$$P(S_1) = 0.5 \quad P(S_2) = 0.5$$

Έστω ότι η παρατηρούμενη ακολουθία είναι x = CGTCAG. Υπολογίστε:

1. **Την πιθανότητα $P(x|\lambda)$ χρησιμοποιώντας τον forward αλγόριθμο.**

   Από την θεωρία έχουμε ότι $\alpha(z_n) = p(x_1, x_2, ..., x_n, z_n)$. Να σημειωθεί ότι τα διανύσματα z_i είναι 2 διαστάσεων όπου πάντα μία διάσταση είναι 1 και η άλλη 0 (αυτή που είναι 1 υποδηλώνει ότι βρισκόμαστε σε αυτήν την κατάσταση την χρονική στιγμή i).

   <u>Αρχικοποίηση:</u>

$$\begin{aligned} \alpha(z_1) = p(x_1, z_1) &= p(z_1)p(x_1|z_1) \\ &= \prod_{k=1}^{K} \{\pi_k p(x_1|\phi_k)\}^{z_{1k}} \\ &= \begin{cases} P(S_1)P(C|S_1), \ z_{11} = 1 \\ P(S_2)P(C|S_2), \ z_{12} = 1 \end{cases} \\ &= \begin{cases} 0.05, \ z_{11} = 1 \\ 0.2, \ z_{12} = 1 \end{cases} \end{aligned}$$

   <u>Για t = 2:</u>

$$
\begin{aligned}
\alpha(z_2) &= p(x_2|z_2)\sum_{z_1}\alpha(z_1)p(z_2|z_1) \\
&= \begin{cases} P(G|S_1)(0.05P(S_1|S_1)+0.2P(S_1|S_2)), & z_{21}=1 \\ P(G|S_2)(0.05P(S_2|S_1)+0.2P(S_2|S_2)), & z_{22}=1 \end{cases} \\
&= \begin{cases} 0.4(0.05*0.8+0.2*0.2), & z_{21}=1 \\ 0.1(0.05*0.2+0.2*0.8), & z_{22}=1 \end{cases} \\
&= \begin{cases} 0.4*0.08, & z_{21}=1 \\ 0.1*0.17 & z_{22}=1 \end{cases} \\
&= \begin{cases} 0.032, & z_{21}=1 \\ 0.017, & z_{22}=1 \end{cases}
\end{aligned}
$$

<u>Για t = 3:</u>

$$
\begin{aligned}
\alpha(z_3) &= p(x_3|z_3)\sum_{z_2}\alpha(z_2)p(z_3|z_2) \\
&= \begin{cases} P(T|S_1)(0.032P(S_1|S_1)+0.017P(S_1|S_2)), & z_{31}=1 \\ P(T|S_2)(0.032P(S_2|S_1)+0.017P(S_2|S_2)), & z_{32}=1 \end{cases} \\
&= \begin{cases} 0.1(0.032*0.8+0.017*0.2), & z_{31}=1 \\ 0.4(0.032*0.2+0.017*0.8), & z_{32}=1 \end{cases} \\
&= \begin{cases} 0.1*0.029, & z_{31}=1 \\ 0.4*0.02, & z_{32}=1 \end{cases} \\
&= \begin{cases} 0.0029, & z_{31}=1 \\ 0.008, & z_{32}=1 \end{cases}
\end{aligned}
$$

<u>Για t = 4:</u>

$$
\begin{aligned}
\alpha(z_4) &= p(x_4|z_4)\sum_{z_3}\alpha(z_3)p(z_4|z_3) \\
&= \begin{cases} P(C|S_1)(0.0029P(S_1|S_1)+0.008P(S_1|S_2)), & z_{41}=1 \\ P(C|S_2)(0.0029P(S_2|S_1)+0.008P(S_2|S_2)), & z_{42}=1 \end{cases} \\
&= \begin{cases} 0.1(0.0029*0.8+0.008*0.2), & z_{41}=1 \\ 0.4(0.0029*0.2+0.008*0.8), & z_{42}=1 \end{cases} \\
&= \begin{cases} 0.1*0.00392, & z_{41}=1 \\ 0.4*0.00698, & z_{42}=1 \end{cases} \\
&= \begin{cases} 0.000392, & z_{41}=1 \\ 0.002792 & z_{42}=1 \end{cases}
\end{aligned}
$$

Για t = 5:

$$\begin{aligned}
\alpha(z_5) &= p(x_5|z_5)\sum_{z_4}\alpha(z_4)p(z_5|z_4) \\
&= \begin{cases} P(A|S_1)(0.000392P(S_1|S_1)+0.002792P(S_1|S_2)), \ z_{51}=1 \\ P(A|S_2)(0.000392P(S_2|S_1)+0.002792P(S_2|S_2)), \ z_{52}=1 \end{cases} \\
&= \begin{cases} 0.4(0.000392*0.8+0.002792*0.2), \ z_{51}=1 \\ 0.1(0.000392*0.2+0.002792*0.8), \ z_{52}=1 \end{cases} \\
&= \begin{cases} 0.4*0.000872, \ z_{51}=1 \\ 0.1*0.002312, \ z_{52}=1 \end{cases} \\
&= \begin{cases} 0.0003488, \ z_{51}=1 \\ 0.0002312, \ z_{52}=1 \end{cases}
\end{aligned}$$

Για t = 6:

$$\begin{aligned}
\alpha(z_6) &= p(x_6|z_6)\sum_{z_5}\alpha(z_5)p(z_6|z_5) \\
&= \begin{cases} P(G|S_1)(0.0003488P(S_1|S_1)+0.0002312P(S_1|S_2)), \ z_{61}=1 \\ P(G|S_2)(0.0003488P(S_2|S_1)+0.0002312P(S_2|S_2)), \ z_{62}=1 \end{cases} \\
&= \begin{cases} 0.4(0.0003488*0.8+0.0002312*0.2), \ z_{61}=1 \\ 0.1(0.0003488*0.2+0.0002312*0.8), \ z_{62}=1 \end{cases} \\
&= \begin{cases} 0.4*0.00032528, \ z_{61}=1 \\ 0.1*0.00025472, \ z_{62}=1 \end{cases} \\
&= \begin{cases} 0.000130112, \ z_{61}=1 \\ 0.000025472, \ z_{62}=1 \end{cases}
\end{aligned}$$

Συνεπώς η ζητούμενη πιθανότητα είναι:

$$P(x|\lambda) = \sum_{z_6}\alpha_{z_6} = 0.000130112 + 0.000025472 = 0.000155584$$

2. **Τις εκ των υστέρων πιθανότητες** $P(\pi_i = S_1|x,\lambda)$ **για** $i = 1, ..., 6$.
   Αντίστοιχα, πρέπει να υπολογίσουμε τα $\beta(z_n)$, όπου $\beta(z_n) = p(x_{n+1}, ..., x_N|z_n)$:

   Αρχικοποίηση:

   $$\beta(z_6) = 1, \text{για όλες τις πιθανές τιμές τιμές του } z_6$$

   Για t = 5:

$$\beta(z_5) = \sum_{z_6} \beta(z_6) p(x_6|z_6) p(z_6|z_5)$$
$$= \begin{cases} p(G|S_1)p(S_1|S_1) + p(G|S_2)p(S_2|S_1), \;\; z_{51} = 1 \\ p(G|S_1)p(S_1|S_2) + p(G|S_2)p(S_2|S_2), \;\; z_{52} = 1 \end{cases}$$
$$= \begin{cases} 0.4 * 0.8 + 0.1 * 0.2, \;\; z_{51} = 1 \\ 0.4 * 0.2 + 0.1 * 0.8, \;\; z_{52} = 1 \end{cases}$$
$$= \begin{cases} 0.34, \;\; z_{51} = 1 \\ 0.16 \;\; z_{52} = 1 \end{cases}$$

<u>Για t = 4:</u>

$$\beta(z_4) = \sum_{z_5} \beta(z_5) p(x_5|z_5) p(z_5|z_4)$$
$$= \begin{cases} 0.34 * p(A|S_1)p(S_1|S_1) + 0.16 * p(A|S_2)p(S_2|S_1), \;\; z_{41} = 1 \\ 0.34 * p(A|S_1)p(S_1|S_2) + 0.16 * p(A|S_2)p(S_2|S_2), \;\; z_{42} = 1 \end{cases}$$
$$= \begin{cases} 0.34 * 0.4 * 0.8 + 0.16 * 0.1 * 0.2, \;\; z_{41} = 1 \\ 0.34 * 0.4 * 0.2 + 0.16 * 0.1 * 0.8, \;\; z_{42} = 1 \end{cases}$$
$$= \begin{cases} 0.112, \;\; z_{41} = 1 \\ 0.04 \;\; z_{42} = 1 \end{cases}$$

<u>Για t = 3:</u>

$$\beta(z_3) = \sum_{z_4} \beta(z_4) p(x_4|z_4) p(z_4|z_3)$$
$$= \begin{cases} 0.112 * p(C|S_1)p(S_1|S_1) + 0.04 * p(C|S_2)p(S_2|S_1), \;\; z_{31} = 1 \\ 0.112 * p(C|S_1)p(S_1|S_2) + 0.04 * p(C|S_2)p(S_2|S_2), \;\; z_{32} = 1 \end{cases}$$
$$= \begin{cases} 0.112 * 0.1 * 0.8 + 0.04 * 0.4 * 0.2, \;\; z_{31} = 1 \\ 0.112 * 0.1 * 0.2 + 0.04 * 0.4 * 0.8, \;\; z_{32} = 1 \end{cases}$$
$$= \begin{cases} 0.01216, \;\; z_{31} = 1 \\ 0.01504 \;\; z_{32} = 1 \end{cases}$$

<u>Για t = 2:</u>

$$
\begin{aligned}
\beta(z_2) &= \sum_{z_3} \beta(z_3) p(x_3|z_3) p(z_3|z_2) \\
&= \begin{cases} 0.01216 * p(T|S_1)p(S_1|S_1) + 0.01504 * p(T|S_2)p(S_2|S_1), \ z_{21} = 1 \\ 0.01216 * p(T|S_1)p(S_1|S_2) + 0.01504 * p(T|S_2)p(S_2|S_2), \ z_{22} = 1 \end{cases} \\
&= \begin{cases} 0.01216 * 0.1 * 0.8 + 0.01504 * 0.4 * 0.2, \ z_{21} = 1 \\ 0.01216 * 0.1 * 0.2 + 0.01504 * 0.4 * 0.8, \ z_{22} = 1 \end{cases} \\
&= \begin{cases} 0.002176, \ z_{21} = 1 \\ 0.005056 \ z_{22} = 1 \end{cases}
\end{aligned}
$$

<u>Για t = 1:</u>

$$
\begin{aligned}
\beta(z_1) &= \sum_{z_2} \beta(z_2) p(x_2|z_2) p(z_2|z_1) \\
&= \begin{cases} 0.002176 * p(G|S_1)p(S_1|S_1) + 0.005056 * p(G|S_2)p(S_2|S_1), \ z_{11} = 1 \\ 0.002176 * p(G|S_1)p(S_1|S_2) + 0.005056 * p(G|S_2)p(S_2|S_2), \ z_{12} = 1 \end{cases} \\
&= \begin{cases} 0.002176 * 0.4 * 0.8 + 0.005056 * 0.1 * 0.2, \ z_{11} = 1 \\ 0.002176 * 0.4 * 0.2 + 0.005056 * 0.1 * 0.8, \ z_{12} = 1 \end{cases} \\
&= \begin{cases} 0.00079744, \ z_{11} = 1 \\ 0.00057856 \ z_{12} = 1 \end{cases}
\end{aligned}
$$

Συνεπώς, έχουμε για τις ζητούμενες πιθανότητες:

$$P(\pi_1 = S_1|x, \lambda) = \gamma(z_1 = S_1) = \frac{\alpha(z_1 = S_1)\beta(z_1 = S_1)}{p(X)} = \frac{0.05 * 0.00079744}{0.000155584} = 0.2563$$

$$P(\pi_2 = S_1|x, \lambda) = \gamma(z_2 = S_1) = \frac{\alpha(z_2 = S_1)\beta(z_2 = S_1)}{p(X)} = \frac{0.032 * 0.002176}{0.000155584} = 0.4476$$

$$P(\pi_3 = S_1|x, \lambda) = \gamma(z_3 = S_1) = \frac{\alpha(z_3 = S_1)\beta(z_3 = S_1)}{p(X)} = \frac{0.0029 * 0.01216}{0.000155584} = 0.2267$$

$$P(\pi_4 = S_1|x, \lambda) = \gamma(z_4 = S_1) = \frac{\alpha(z_4 = S_1)\beta(z_4 = S_1)}{p(X)} = \frac{0.000392 * 0.112}{0.000155584} = 0.2822$$

$$P(\pi_5 = S_1|x, \lambda) = \gamma(z_5 = S_1) = \frac{\alpha(z_5 = S_1)\beta(z_5 = S_1)}{p(X)} = \frac{0.0003488 * 0.34}{0.000155584} = 0.7622$$

$$P(\pi_6 = S_1|x, \lambda) = \gamma(z_6 = S_1) = \frac{\alpha(z_6 = S_1)\beta(z_6 = S_1)}{p(X)} = \frac{0.000130112 * 1}{0.000155584} = 0.8363$$

3. **Το πιο πιθανό μονοπάτι κρυμμένων καταστάσεων χρησιμοποιώντας τον αλγόριθμο Viterbi.**

Αρχικοποίηση:

$$\begin{aligned}
\omega(z_1) = p(x_1, z_1) &= p(z_1)p(x_1|z_1) \\
&= \prod_{k=1}^{K} \{\pi_k p(x_1|\phi_k)\}^{z_{1k}} \\
&= \begin{cases} P(S_1)P(C|S_1), \ z_{11} = 1 \\ P(S_2)P(C|S_2), \ z_{12} = 1 \end{cases} \\
&= \begin{cases} 0.05, \ z_{11} = 1 \\ 0.2, \ z_{12} = 1 \end{cases}
\end{aligned}$$

Για t = 2:

$$\begin{aligned}
\omega(z_2) &= p(x_2|z_2) * \max_{z_1}\{\omega(z_1)p(z_2|z_1)\} \\
&= \begin{cases} P(G|S_1) * \max\{0.05P(S_1|S_1), 0.2P(S_1|S_2))\}, \ z_{21} = 1 \\ P(G|S_2) * \max\{0.05P(S_2|S_1), 0.2P(S_2|S_2))\}, \ z_{22} = 1 \end{cases} \\
&= \begin{cases} 0.4 * \max\{0.05 * 0.8, 0.2 * 0.2)\}, \ z_{21} = 1 \\ 0.1 * \max\{0.05 * 0.2, 0.2 * 0.8)\}, \ z_{22} = 1 \end{cases} \\
&= \begin{cases} 0.4 * 0.04, \ z_{21} = 1 \\ 0.1 * 0.16, \ z_{22} = 1 \end{cases} \\
&= \begin{cases} 0.016, \ z_{21} = 1 \\ 0.016, \ z_{22} = 1 \end{cases}
\end{aligned}$$

Για t = 3:

$$\begin{aligned}
\omega(z_3) &= p(x_3|z_3) * \max_{z_2}\{\omega(z_2)p(z_3|z_2)\} \\
&= \begin{cases} P(A|S_1) * \max\{0.016P(S_1|S_1), 0.016P(S_1|S_2))\}, \ z_{31} = 1 \\ P(A|S_2) * \max\{0.016P(S_2|S_1), 0.016P(S_2|S_2))\}, \ z_{32} = 1 \end{cases} \\
&= \begin{cases} 0.1 * \max\{0.016 * 0.8, 0.016 * 0.2)\}, \ z_{31} = 1 \\ 0.4 * \max\{0.016 * 0.2, 0.016 * 0.8)\}, \ z_{32} = 1 \end{cases} \\
&= \begin{cases} 0.1 * 0.0128, \ z_{31} = 1 \\ 0.4 * 0.0128, \ z_{32} = 1 \end{cases} \\
&= \begin{cases} 0.00128, \ z_{31} = 1 \\ 0.00512, \ z_{32} = 1 \end{cases}
\end{aligned}$$

<u>Για t = 4:</u>

$$\omega(z_4) = p(x_4|z_4) * \max_{z_3}\{\omega(z_3)p(z_4|z_3)\}$$
$$= \begin{cases} P(C|S_1) * \max\{0.00128P(S_1|S_1), 0.00512P(S_1|S_2))\},\ z_{41} = 1 \\ P(C|S_2) * \max\{0.00128P(S_2|S_1), 0.00512P(S_2|S_2))\},\ z_{42} = 1 \end{cases}$$
$$= \begin{cases} 0.1 * \max\{0.00128 * 0.8, 0.00512 * 0.2)\},\ z_{41} = 1 \\ 0.4 * \max\{0.00128 * 0.2, 0.00512 * 0.8)\},\ z_{42} = 1 \end{cases}$$
$$= \begin{cases} 0.1 * 0.001024,\ z_{41} = 1 \\ 0.4 * 0.004096,\ z_{42} = 1 \end{cases}$$
$$= \begin{cases} 0.0001024,\ z_{41} = 1 \\ 0.0016384,\ z_{42} = 1 \end{cases}$$

<u>Για t = 5:</u>

$$\omega(z_5) = p(x_5|z_5) * \max_{z_4}\{\omega(z_4)p(z_5|z_4)\}$$
$$= \begin{cases} P(T|S_1) * \max\{0.0001024 * 0.8, 0.0016384 * 0.2)\},\ z_{51} = 1 \\ P(T|S_2) * \max\{0.0001024 * 0.2, 0.0016384 * 0.8)\},\ z_{52} = 1 \end{cases}$$
$$= \begin{cases} 0.1 * 0.00032768,\ z_{51} = 1 \\ 0.4 * 0.00131072,\ z_{52} = 1 \end{cases}$$
$$= \begin{cases} 0.000032768,\ z_{51} = 1 \\ 0.000524288,\ z_{52} = 1 \end{cases}$$

<u>Για t = 6:</u>

$$\omega(z_6) = p(x_6|z_6) * \max_{z_5}\{\omega(z_5)p(z_6|z_5)\}$$
$$= \begin{cases} P(G|S_1) * \max\{0.000032768 * 0.8, 0.000524288 * 0.2)\},\ z_{61} = 1 \\ P(G|S_2) * \max\{0.000032768 * 0.2, 0.000524288 * 0.8)\},\ z_{62} = 1 \end{cases}$$
$$= \begin{cases} 0.4 * 0.000104857,\ z_{61} = 1 \\ 0.1 * 0.00041943,\ z_{62} = 1 \end{cases}$$
$$= \begin{cases} 0.000041942,\ z_{61} = 1 \\ 0.000041943,\ z_{62} = 1 \end{cases}$$

Η βέλτιστη ακολουθία προκύπτει από την κατάσταση στην οποία προκύπτει το max σε κάθε χρονική στιγμή. Έχουμε την εξής βέλτιστη ακολουθία: $S_2S_2S_2S_2S_2S_2$.

Άσκηση 2 (Principal Component Analysis)

Ζητείται η εφαρμογή της Principal Component Analysis (PCA) πάνω στο ευρέως διαδεδομένο σύνολο δεδομένων κρίνων του Fisher, προκειμένου να μετασχηματιστούν τα δεδομένα σε ένα χώρο χαμηλότερων διαστάσεων. Τα δεδομένα αποτελούνται από 3 κλάσεις (για τους 3 διαφορετικούς τύπους κρίνου), καθεμιά από τις οποίες περιλαμβάνει 50 δείγματα. Τα δεδομένα περιγράφονται από 4 διαφορετικά χαρακτηριστικά:

- μήκος σεπάλων σε εκ.
- πλάτος σεπάλων σε εκ.
- μήκος πετάλων σε εκ.
- πλάτος πετάλων σε εκ.
- τύπος κρίνου (Iris Setosa/Iris Versicolour/Iris Virginica)

Αρχικά, ακολουθεί ο κώδικας σε Python που υλοποιεί όλα τα βήματα της άσκηση.

Listing 1:

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

if __name__ == "__main__":
    #################
    #    1st part    #
    #################
    # Read data from PCA.data and keep them in numpy arrays.

    n = 150
    m = 4
    data_path = '../data/PCA.data'
    X = np.zeros((n, m))
    n_classes = 3
    y = np.zeros(n, dtype= int)
    class_idx = {'Iris-setosa': 0,
                 'Iris-versicolor': 1, 'Iris-virginica': 2}
    features_names = ['length of the sepals', 'width of the sepals',
                      'length of the petals', 'width of the petals']

    with  open(data_path, 'r') as f:
        for i, line in  enumerate(f):
            tokens = line.split(',')
            X[i, :4] = tokens[:4]
            idx = class_idx[tokens[4].strip('\n')]
            y[i] = idx

    #################
```

```
    #    2nd part    #
    ##################

    # Compute the mean value and the standard deviation of each feature.
    means = np.mean(X, axis=0)
    stds = np.std(X, axis=0)

    print("Mean values of the features: ")
    print(means)

    print("Standard deviation of the features:")
    print(stds)

    # Subtract from each value the mean value and divide with the standard
    # deviation of the corresponding feature.
    X_processed = (X - means) / stds
    # Verify that the processed data have mean value equal to 0 and
    # standard deviation equal to 1.
    print("Mean values of the features after preprocessing: ")
    print(np.mean(X_processed, axis=0))
    print("Standard deviation of the features after preprocessing: ")
    print(np.std(X_processed, axis=0))

    ##################
    #    3rd part    #
    ##################

    # Compute the sampling covariance matrix.
    X_cov = np.cov(X_processed, rowvar=False)
    print("Sampling covariance matrix: ")
    print(X_cov)

    ##################
    #    4th part    #
    ##################
    # Factorize the covariance matrix using SVD.
    u, s, vh = np.linalg.svd(X_cov)

    print("Eigenvalues in descending order: ")
    print(s)

    print("Eigenvectors:")
    print(u)

    ##################
    #    5th part    #
```

```python
    ##################
    # Keep the first two principal components and project our data in them.
    X_2d = np.dot(X_processed, u[:, :2])

    # Plot our projected data.
    sel_colors = ['red', 'blue', 'green']
    color = [sel_colors[yi] for yi in y]
    plt.scatter(X_2d[:, 0], X_2d[:, 1], color=color)

    plt.savefig("../photos/ex2_plot.png")

    ##################
    #    6th part    #
    ##################

    # Compute the percentage of the total variation that is explained.
    perc_var = s**2 /  sum(s**2) * 100
    print("Percentage of the total variation that is explained: ")
    print(perc_var)
```

1. **Κατεβάστε το σύνολο δεδομένων που έχει ανεβεί στο mycourses (αρχείο PCA.data).**

2. **Προεπεξεργαστείτε τα δεδομένα αφαιρώντας τη μέση τιμή και διαιρώντας με την τυπική απόκλιση του κάθε χαρακτηριστικού ξεχωριστά. Τα προκύπτοντα δεδομένα θα πρέπει να έχουν μέση τιμή 0 και διασπορά 1.**

```
Mean values of the features:
[5.84333333 3.054      3.75866667 1.19866667]
Standard deviation of the features:
[0.82530129 0.43214658 1.75852918 0.76061262]
Mean values of the features after preprocessing:
[-1.69031455e-15 -1.63702385e-15 -1.48251781e-15 -1.62314606e-15]
Standard deviation of the features after preprocessing:
[1. 1. 1. 1.]
```

3. **Υπολογίστε τον δειγματικό πίνακα συνδιασπορών $\Sigma = \frac{1}{m-1}\sum_{i=1}^{m} x_i x_i^T$, όπου x_i είναι το i-στό δείγμα και m το πλήθος των δειγμάτων.**

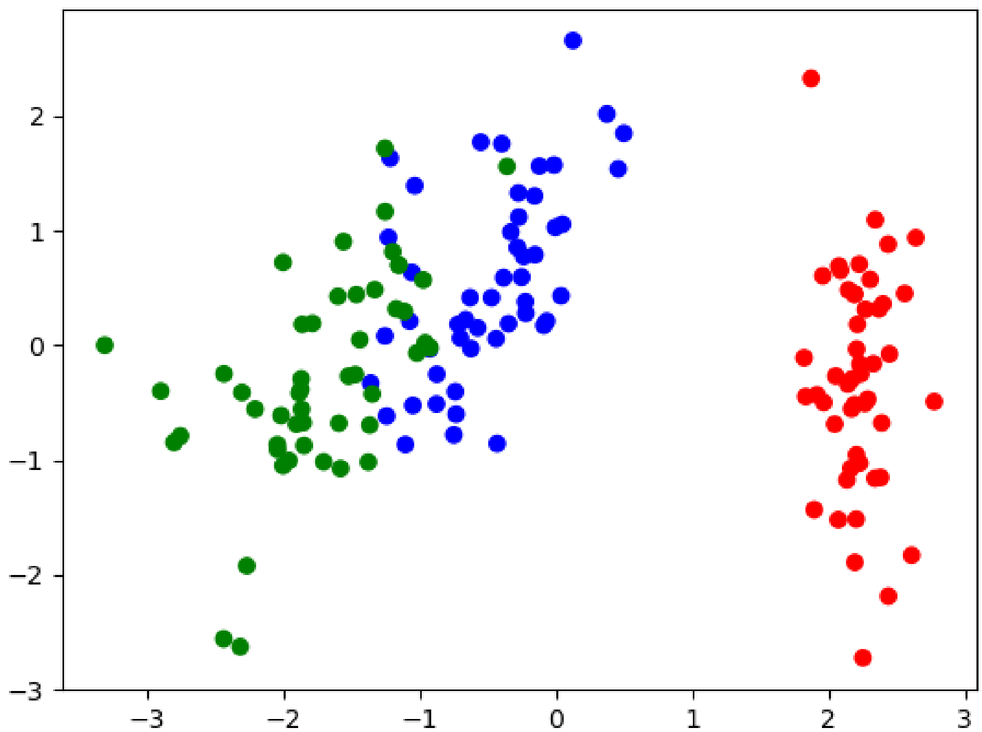
```
Sampling covariance matrix:
[[ 1.00671141 -0.11010327  0.87760486  0.82344326]
 [-0.11010327  1.00671141 -0.42333835 -0.358937  ]
 [ 0.87760486 -0.42333835  1.00671141  0.96921855]
 [ 0.82344326 -0.358937    0.96921855  1.00671141]]
```

4. **Παραγοντοποιήστε τον πίνακα συνδιασπορών κάνοντας χρήση του Singular Value Decomposition (SVD) και βρείτε τις αντίστοιχες ιδιοτιμές και ιδιοδιανύσματα.**

Προσέξτε εάν η συγκεκριμένη υλοποίηση του SVD δίνει τα αποτελέσματα με φθίνουσα ή αύξουσα σειρά ιδιοτιμών. Ο μετασχηματισμός SVD ενός πίνακα Σ είναι μια παραγοντοποίηση της μορφής $\Sigma = UDV^T$. Για έναν συμμετρικό, θετικά ορισμένο πίνακα Σ, οι U = V περιέχουν τα ιδιοδιανύσματα και D είναι ένας διαγώνιος πίνακας με τις αντίστοιχες ιδιοτιμές.

```
Eigenvalues in descending order:
[2.93035378 0.92740362 0.14834223 0.02074601]
Eigenvectors:
[[-0.52237162 -0.37231836  0.72101681  0.26199559]
 [ 0.26335492 -0.92555649 -0.24203288 -0.12413481]
 [-0.58125401 -0.02109478 -0.14089226 -0.80115427]
 [-0.56561105 -0.06541577 -0.6338014   0.52354627]]
```

5. **Προβάλετε τα δεδομένα πάνω στις δύο πρώτες κύριες συνιστώσες και σχεδιάστε τα αποτελέσματα που προκύπτουν.**



6. **Ποιος είναι ο ελάχιστος απαιτούμενος αριθμός από κύριες συνιστώσες ώστε να "ερμηνεύεται" το 95% της διασποράς των τιμών;**

```
Percentage of the total variation that is explained:
[9.06804526e+01 9.08261999e+00 2.32382358e-01 4.54509434e-03]
```

Παρατηρούμε ότι η 1η συνιστώσα ερμηνεύει 90% της διασποράς των τιμών και η 2η ερμηνεύει το 9%. Συνεπώς, ο ελάχιστος απαιτούμενος αριθμός συνιστωσών για να ερμηνεύεται το 95% της διασποράς των τιμών είναι 2.

Άσκηση 3 (Linear Discriminant Analysis)

Στο μάθημα είδαμε ότι η Linear Discriminant Analysis (LDA) βασίζεται στην ανάστροφη σχέση των μητρών (πινάκων) S_W και S_B:

$$S_W = \sum_{i=1}^{|Classes|} \mathbb{E}_{x|x\in\omega_i}[(\vec{x}-\vec{m})(\vec{x}-\vec{m})^T]$$

$$S_B = \sum_{i=1}^{|Classes|} P(\omega_i)(\vec{m}_i-\vec{m})(\vec{m}_i-\vec{m})^T$$

όπου το ω_i αναπαριστά μια κλάση με μέση τιμή $\vec{m}_i$, |Classes| είναι το πλήθος των κλάσεων και $\vec{m}$ είναι η μέση τιμή όλων των δειγμάτων.

1. **Δείξτε ότι στην περίπτωση διαχωρισμού δύο κλάσεων ω_1 και ω_2, ο πίνακας S_B μπορεί να γραφτεί στη μορφή** $S_B = P(\omega_1)P(\omega_2)(\vec{m}_2-\vec{m}_1)(\vec{m}_2-\vec{m}_1)^T$

Έχουμε ότι για $|Classes| = 2$:

$$S_B = P(\omega_1)(\vec{m}_1-\vec{m})(\vec{m}_1-\vec{m})^T + P(\omega_2)(\vec{m}_2-\vec{m})(\vec{m}_2-\vec{m})^T$$

Ακόμη, γνωρίζουμε ότι $\vec{m} = P(\omega_1)\vec{m}_1 + P(\omega_2)\vec{m}_2$ και $P(\omega_1)+P(\omega_2) = 1$. Συνδυάζοντας όλες αυτές τις σχέσεις:

$$\begin{aligned}
S_B &= P(\omega_1)(\vec{m}_1 - P(\omega_1)\vec{m}_1 - P(\omega_2)\vec{m}_2)(\vec{m}_1 - P(\omega_1)\vec{m}_1 - P(\omega_2)\vec{m}_2)^T + \\
&\quad P(\omega_2)(\vec{m}_2 - P(\omega_1)\vec{m}_1 - P(\omega_2)\vec{m}_2)(\vec{m}_2 - P(\omega_1)\vec{m}_1 - P(\omega_2)\vec{m}_2)^T \\
&= P(\omega_1)(P(\omega_2)\vec{m}_1 - P(\omega_2)\vec{m}_2)(P(\omega_2)\vec{m}_1 - P(\omega_2)\vec{m}_2)^T + \\
&\quad P(\omega_2)(P(\omega_1)\vec{m}_2 - P(\omega_1)\vec{m}_1)(P(\omega_1)\vec{m}_2 - P(\omega_1)\vec{m}_1)^T \\
&= P(\omega_1)P(\omega_2)(\vec{m}_1-\vec{m}_2)(\vec{m}_1-\vec{m}_2)^T + P(\omega_2)P(\omega_1)(\vec{m}_2-\vec{m}_1)(\vec{m}_2-\vec{m}_1)^T \\
&= 2P(\omega_1)P(\omega_2)(\vec{m}_2-\vec{m}_1)(\vec{m}_2-\vec{m}_1)^T
\end{aligned}$$

2. **Βασιζόμενοι στο υποερώτημα (α), να βρείτε το ιδιοδιάνυσμα του πίνακα $S_W^{-1} S_B$ και την ιδιοτιμή του.**

   Έχουμε την εξίσωση $S_W^{-1} S_B v = \lambda v$. Παρατηρούμε ότι:

$$
\begin{aligned}
S_B v &= P(\omega_1)P(\omega_2)(\vec{m_2} - \vec{m_1})(\vec{m_2} - \vec{m_1})^T v \\
&= (\vec{m_2} - \vec{m_1})(P(\omega_1)P(\omega_2)(\vec{m_2} - \vec{m_1})^T v) \\
&= \lambda(\vec{m_2} - \vec{m_1})
\end{aligned}
$$

   Βλέπουμε ότι το $S_B v$ έχει την ίδια διεύθυνση με το διάνυσμα $\vec{m_2} - \vec{m_1}$. Άρα, το πρόβλημα μπορεί να λυθεί ως εξής:

$$
v = S_W^{-1}(\vec{m_2} - \vec{m_1})
$$

   γιατί αν το αντικαταστήσουμε στην αρχική σχέση:

$$
\begin{aligned}
S_W^{-1} S_B v &= S_W^{-1} S_B (S_W^{-1}(\vec{m_2} - \vec{m_1})) \\
&= S_W^{-1}(\lambda(\vec{m_2} - \vec{m_1})) \\
&= \lambda(S_W^{-1}(\vec{m_2} - \vec{m_1})) \\
&= \lambda v
\end{aligned}
$$

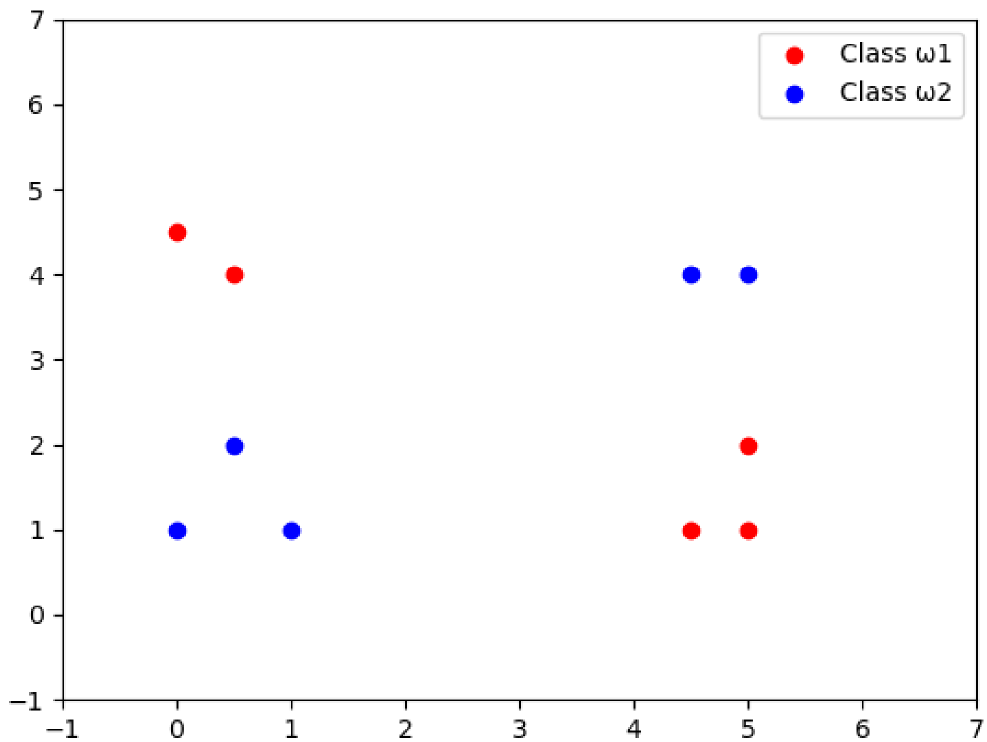
Άσκηση 4 (Multilayer Perceptron)

Μας δίνονται 10 διανύσματα χαρακτηριστικών που προέρχονται από δύο κλάσεις ω_1 και ω_2:

$$
\begin{aligned}
\omega_1 &: [4.5, 1]^T, [5, 2]^T, [5, 1]^T, [0, 4.5]^T, [0.5, 4]^T \\
\omega_2 &: [0, 1]^T, [0.5, 2]^T, [5, 4]^T, [4.5, 4]^T, [1, 1]^T
\end{aligned}
$$

Ελέγξτε αν οι δύο κλάσεις είναι γραμμικά διαχωρίσιμες, και αν όχι, σχεδιάστε ένα κατάλληλο multilayer perceptron με τους κόμβους να έχουν βηματική συνάρτηση ενεργοποίησης (step) για να ταξινομήσετε τα διανύσματα στις δύο κλάσεις.

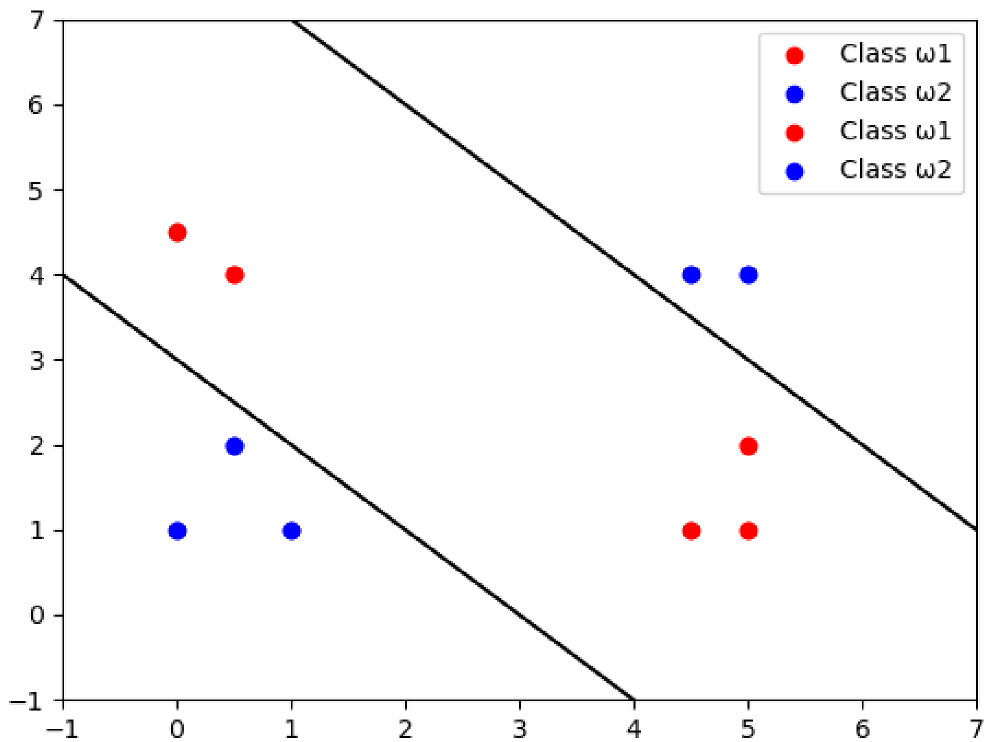
Αρχικά, ελέγχουμε αν οι κλάσεις είναι γραμμικά διαχωρίσιμες.



Παρατηρούμε ότι οι κλάσεις δεν είναι γραμμικά διαχωρίσιμες και ότι έχουν την ίδια μορφή με το πρόβλημα του XOR. Συνεπώς, μπορούμε να τις διαχωρίσουμε χρησιμοποιώντας ένα 2-layer perceptron. Ουσιαστικά σε πρώτη φάση θα διαχωρίσουμε με δύο γραμμές τις κλάσεις. Στην περίπτωσή μας οι γραμμές αυτές είναι:

$$g_1(x) = x_1 + x_2 - 3 = 0, \quad g_2(x) = x_1 + x_2 - 8 = 0$$

Προφανώς, οι εξισώσεις προκύψανε εμπειρικά από το διάγραμμα, αλλά θα μπορούσε να εφαρμοστεί και ένας αλγόριθμος εκπαίδευσης.



Το δεύτερο layer θα είναι ίδιο με το xor καθώς μεταφέραμε τα δεδομένα μας σε ένα χώρο y που είναι ίδιος με αυτόν του xor. Άρα:

$$g_3(y) = y_1 - y_2 - \frac{1}{2}$$

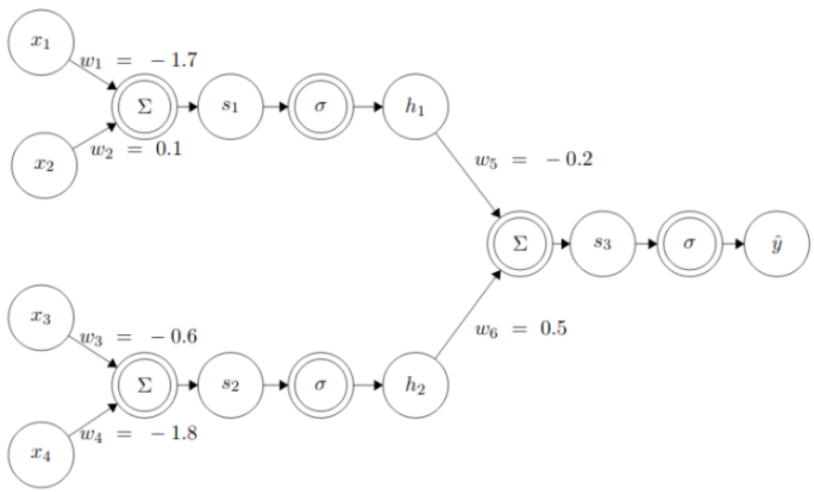
Αν κάνω τώρα την ταξινόμηση των δεδομένων μας έχουμε:

x1	x2	f(g_1(x))	f(g_2(x))	f(g_3(x))
4.5	1	1	0	1
5	2	1	0	1
5	1	1	0	1
0	4.5	1	0	1
0.5	4	1	0	1
0	1	0	0	0
0.5	2	0	0	0
5	4	1	1	0
4.5	4	1	1	0
1	1	0	0	0

Παρατηρούμε ότι όλα τα δεδομένα ταξινομήθηκαν σωστά.

Άσκηση 5 (Backpropagation)

Υποθέστε ότι έχουμε το ακόλουθο νευρωνικό δίκτυο. Οι κόμβοι που βρίσκονται μέσα σε μονό κύκλο υποδηλώνουν μεταβλητές (για παράδειγμα η x_1 είναι μια μεταβλητή εισόδου, h_1 είναι μια ενδιάμεση μεταβλητή, και $\hat{y}$ είναι μια μεταβλητή εξόδου). Οι κόμβοι που βρίσκονται μέσα σε διπλό κύκλο υποδηλώνουν συναρτήσεις (για παράδειγμα το $\sum$ υπολογίζει το άθροισμα των εισόδων του και η σ αναπαριστά τη συνάρτηση logistic $\sigma(x) = \frac{1}{1+e^{-x}}$.



Θεωρήστε ότι η συνάρτηση για το L2 loss δίνεται από τη σχέση $L(y, \hat{y}) = ||y - \hat{y}||_2^2$. Επίσης, μας δίνονται τα δεδομένα ενός δείγματος $(x_1, x_2, x_3, x_4) = (-0.7, 1.2, 1.1, -2)$ με τιμή για το πραγματικό του label ίση με 0.5. Χρησιμοποιήστε τον αλγόριθμο backpropagation για να υπολογίσετε τη μερική παράγωγο $\frac{\partial L}{\partial w_1}$. Σημείωση: Το gradient για μια συνάρτηση L2 loss είναι ίσο με $2||y - \hat{y}||$.

Αρχικά, εφαρμόζουμε την forward διαδικασία ώστε να υπολογίσουμε την πρόβλεψη $\hat{y}$ αλλά και όλες τις ενδιάμεσες μεταβλητές $(s_1, s_2, h_1, h_2, s_3)$. Έχουμε:

$$s_1 = w_1 x_1 + w_2 x_2 = -1.7 * (-0.7) + 0.1 * 1.2 = 1.19 + 0.12 = 1.31$$

$$h_1 = \sigma(s_1) = \frac{1}{1 + e^{-s_1}} = 0.788$$

$$s_2 = w_3 x_3 + w_4 x_4 = -0.6 * 1.1 - 1.8 * (-2) = -0.66 + 3.6 = 2.94$$

$$h_2 = \sigma(s_2) = \frac{1}{1 + e^{-s_2}} = 0.95$$

$$s_3 = w_5 h_1 + w_6 h_2 = -0.2 * 0.788 + 0.5 * 0.95 = -0.158 + 0.475 = 0.317$$

$$\hat{y} = \sigma(s_3) = \frac{1}{1 + e^{-s_3}} = 0.421$$

Εφαρμόζουμε στη συνέχεια τον κανόνα της αλυσίδας ώστε να υπολογίσουμε το $\frac{\partial L}{\partial w_1}$.

$$\begin{aligned}\frac{\partial L}{\partial w_1} &= \frac{\partial L}{\partial \hat{y}}\frac{\partial \hat{y}}{\partial w_1} \\ &= \frac{\partial L}{\partial \hat{y}}\frac{\partial \hat{y}}{\partial s_3}\frac{\partial s_3}{\partial w_1} \\ &= \frac{\partial L}{\partial \hat{y}}\frac{\partial \hat{y}}{\partial s_3}\frac{\partial s_3}{\partial h_1}\frac{\partial h_1}{\partial w_1} \\ &= \frac{\partial L}{\partial \hat{y}}\frac{\partial \hat{y}}{\partial s_3}\frac{\partial s_3}{\partial h_1}\frac{\partial h_1}{\partial s_1}\frac{\partial s_1}{\partial w_1}\end{aligned}$$

όπου

$$\begin{aligned}&\frac{\partial L}{\partial \hat{y}} = 2||y-\hat{y}|| = 2|0.5-0.421| = 2*0.079 = 0.158 \\ &\frac{\partial \hat{y}}{\partial s_3} = \sigma^{'}(s_3) = \sigma(s_3)(1-\sigma(s_3)) = \hat{y}(1-\hat{y}) = 0.421(1-0.421) = 0.244 \\ &\frac{\partial s_3}{\partial h_1} = w_5 = -0.2 \\ &\frac{\partial h_1}{\partial s_1} = \sigma^{'}(s_1) = \sigma(s_1)(1-\sigma(s_1)) = h_1(1-h_1) = 0.788(1-0.788) = 0.167 \\ &\frac{\partial s_1}{\partial w_1} = x_1 = -0.7\end{aligned}$$

Συνεπώς,

$$\begin{aligned}\frac{\partial L}{\partial w_1} &= \frac{\partial L}{\partial \hat{y}}\frac{\partial \hat{y}}{\partial s_3}\frac{\partial s_3}{\partial h_1}\frac{\partial h_1}{\partial s_1}\frac{\partial s_1}{\partial w_1} \\ &= 0.158*0.244*(-0.2)*0.167*(-0.7) \\ &= 0.0009\end{aligned}$$

Άσκηση 6 (Support Vector Machine)

Θεωρήστε το πρόβλημα του διαχωρισμού ενός συνόλου από διανύσματα εκπαίδευσης δύο κλάσεων. Τα δεδομένα εκπαίδευσης είναι της μορφής $\{(x_i, y_i)\}$, όπου τα διανύσματα χαρακτηριστικών $x_i \in R^m$ και τα labels των κλάσεων $y_i \in \{-1, 1\}$. Οπως είναι γνωστό, στην περίπτωση όπου τα δεδομένα εκπαίδευσης δεν είναι γραμμικώς διαχωρίσιμα (για παράδειγμα μέσω ενός κανόνα απόφασης $sign(w^T x + b)$ για κάποια w, b), τότε το πρόβλημα χρειάζεται να διατυπωθεί με χρήση slack variables $\{\xi_i\}$, $1 \leq i \leq n$. ΄Ετσι, ο ταξινομητής SVM με το μεγαλύτερο περιθώριο αποκτάται μέσω της επίλυσης του δυϊκού προβλήματος:

$$L(w,b,a,\xi,\beta) = \frac{1}{2}w^T w + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}a_i[(wx_i+b)y_i - (1-\xi_i)] - \sum_{i=1}^{n}\beta_i\xi_i$$

όπου το C είναι μια σταθερά, $a_i, \beta_i \geq 0, \forall\ i$ είναι οι πολλαπλασιαστές Lagrange, και $\xi_i \geq 0$ είναι οι slack variables.

1. Υποθέστε ότι n = 4 και ότι το x είναι δύο διαστάσεων $< x_i^1, x_i^2 >:< 2,2 >, < 2.5, 2.5 >, < 5,5 >, < 7,7 >$. Τώρα το SVM εκπαιδεύεται με βάση την παραπάνω εξίσωση. Να δείξετε ότι οποιαδήποτε labels y και αν έχουν τα τέσσερα δείγματα εκπαίδευσης, το βέλτιστο διάνυσμα παραμέτρων $\hat{w} = (\hat{w}^1, \hat{w}^2)$ έχει την ιδιότητα ότι $\hat{w}^1 = \hat{w}^2$.

   Σύμφωνα με τις συνθήκες Karush-Kuhn-Tucker, $\frac{\partial L}{\partial w} = 0 \implies \hat{w} = \sum_{i=1}^{n} \lambda_i y_i x_i$. Άρα στην δική μας περίπτωση:

$$\begin{aligned}
\hat{w} = \sum_{i=1}^{4} \lambda_i y_i x_i &= \lambda_1 y_1 \begin{bmatrix} x_1^1 \\ x_1^2 \end{bmatrix} + \lambda_2 y_2 \begin{bmatrix} x_2^1 \\ x_2^2 \end{bmatrix} + \lambda_3 y_3 \begin{bmatrix} x_3^1 \\ x_3^2 \end{bmatrix} + \lambda_4 y_4 \begin{bmatrix} x_4^1 \\ x_4^2 \end{bmatrix} \\
&= \lambda_1 y_1 \begin{bmatrix} 2 \\ 2 \end{bmatrix} + \lambda_2 y_2 \begin{bmatrix} 2.5 \\ 2.5 \end{bmatrix} + \lambda_3 y_3 \begin{bmatrix} 5 \\ 5 \end{bmatrix} + \lambda_4 y_4 \begin{bmatrix} 7 \\ 7 \end{bmatrix} \\
&= \begin{bmatrix} 2\lambda_1 y_1 + 2.5\lambda_2 y_2 + 5\lambda_3 y_3 + 7\lambda_4 y_4 \\ 2\lambda_1 y_1 + 2.5\lambda_2 y_2 + 5\lambda_3 y_3 + 7\lambda_4 y_4 \end{bmatrix} \\
&= \begin{bmatrix} \hat{w}^1 \\ \hat{w}^2 \end{bmatrix}
\end{aligned}$$

   Συνεπώς, για το βέλτιστο διάνυσμα παραμέτρων $\hat{w} = (\hat{w}^1, \hat{w}^2)$ έχουμε ότι $\hat{w}^1 = \hat{w}^2$.

2. Θεωρήστε την εκπαίδευση ενός SVM με slack variables, αλλά δίχως την ύπαρξη του bias όρου (b = 0). Θα χρησιμοποιήσουμε έναν kernel K(u, v) με την ιδιότητα ότι για δύο οποιαδήποτε σημεία u και v που ανήκουν στο σύνολο εκπαίδευσης ισχύει ότι $-1 < K(u, v) < 1$. Επιπρόσθετα, $K(u, u) < 1$. Να δείξετε ότι αν υπάρχουν n δείγματα στο σύνολο εκπαίδευσης και η σταθερά C επιλέγεται ούτως ώστε $C < \frac{1}{n-1}$, τότε όλες οι δυϊκές μεταβλητές a_i είναι μη μηδενικές (και άρα όλα τα δείγματα του συνόλου εκπαίδευσης αποτελούν support vectors).

   Το πρόβλημα ανάγεται στη μεγιστοποίηση της παρακάτω συνάρτησης:

$$\begin{aligned}
&L(\alpha) = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \\
&\text{subject to} \quad 0 \le \alpha_i \le C
\end{aligned}$$

   Για να αποδείξουμε ότι όλες οι δυϊκές μεταβλητές a_i είναι μη μηδενικές, αρκεί να αποδείξουμε ότι για κάθε διάνυσμα α με κάποια μηδενική συνιστώσα α_k, υπάρχει ένα άλλο διάνυσμα $\alpha^{'}$ χωρίς μηδενική συνιστώσα για το οποίο ισχύει $L(\alpha^{'}) > L(\alpha)$. Υποθέτουμε ότι $\alpha_k^{'} = \epsilon$ και αναπτύσσοντας την $L(\alpha^{'}) > L(\alpha)$ έχουμε:

$$\begin{aligned}
\sum_{i \ne k} \alpha_i + \epsilon - \frac{1}{2}(\epsilon^2 K(x_k, x_k) + 2\epsilon \sum_{i \ne k} \alpha_i y_i y_k K(x_i, x_k) + \sum_{i \ne k} \sum_{j \ne k} \alpha_i \alpha_j y_i y_j K(x_i, x_j) > \\
\sum_{i \ne k} \alpha_i - \frac{1}{2}(\sum_{i \ne k} \sum_{j \ne k} \alpha_i \alpha_j y_i y_j K(x_i, x_j)) \implies
\end{aligned}$$

$$\epsilon - \frac{1}{2}\epsilon^2 K(x_k, x_k) - \epsilon \sum_{i \neq k} a_i y_i y_k K(x_i, x_k) > 0 \implies$$
$$\frac{1}{2}\epsilon K(x_k, x_k) + \sum_{i \neq k} a_i y_i y_k K(x_i, x_k) < 1$$

Ακόμη, γνωρίζουμε ότι $\alpha_i \leq C \leq \frac{1}{n-1}$:

$$\begin{aligned}
\frac{1}{2}\epsilon K(x_k, x_k) + \sum_{i \neq k} a_i y_i y_k K(x_i, x_k) &\leq \left| \frac{1}{2}\epsilon K(x_k, x_k) + \sum_{i \neq k} a_i y_i y_k K(x_i, x_k) \right| \\
&\leq \frac{1}{2}\epsilon \left| K(x_k, x_k) \right| + \sum_{i \neq k} a_i \left| y_i y_k K(x_i, x_k) \right| \\
&\leq \frac{1}{2}\epsilon + C(n-1)
\end{aligned}$$

Παίρνουμε λοιπόν κάποιο ϵ τέτοιο ώστε $\frac{1}{2}\epsilon + C(n-1) < 1 \implies \epsilon < 2 - 2C(n-1)$ και το ζητούμενο αποδεικνύεται.

3. Θεωρήστε τον εξής kernel:

$$K(u, v) = uv + 4(uv)^2$$

όπου τα διανύσματα u και v είναι δύο διαστάσεων. Ο kernel αυτός είναι ίσος με το εσωτερικό γινόμενο $\phi(u)\phi(v)$ για κάποιο ορισμό της συνάρτησης ϕ. Ποια είναι η συνάρτηση αυτή ϕ;

Έστω $u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$ και $v = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$ και η εξίσωση γίνεται:

$$\begin{aligned}
K(u, v) &= (u_1 v_1 + u_2 v_2) + 4(u_1 v_1 + u_2 v_2)^2 \\
&= u_1 v_1 + u_2 v_2 + 4u_1^2 v_1^2 + 8 u_1 v_1 u_2 v_2 + 4 u_2^2 v_2^2 \\
&= (u_1, u_2, 2u_1^2, 2\sqrt{2} u_1 u_2, 2u_2^2)^T (v_1, v_2, 2v_1^2, 2\sqrt{2} v_1 v_2, 2v_2^2) \\
&= \phi(u)^T \phi(v)
\end{aligned}$$

όπου

$$\phi(u_1, u_2) = (u_1, u_2, 2u_1^2, 2\sqrt{2} u_1 u_2, 2u_2^2)$$

Άσκηση 7 (Logistic Regression)

Θεωρήστε το πρόβλημα logistic regression για ένα σύνολο δεδομένων $\{\phi_n, t_n\}$, όπου $t_n \in \{0, 1\}$ και $\phi_n = \phi(x_n)$ είναι οι κατηγορίες και οι συναρτήσεις βάσης, αντίστοιχα, για δείγματα $n = \{1, 2, .., N\}$. Η συνάρτηση σφάλματος E(w), η οποία αναφέρεται συνήθως και ως crossentropy, ορίζεται ως:

$$E(w) = -\sum_{n=1}^{N}\{t_n \ln y_n + (1-t_n)\ln(1-y_n)\}$$

όπου w είναι το διάνυσμα βαρών, $y_n = \sigma(w^T\phi_n)$ η έξοδος του μοντέλου logistic regression στο διάνυσμα εισόδου x_n, και $\sigma(a) = \frac{1}{1+exp(-a)}$ η logistic sigmoid συνάρτηση.

1. Να δείξετε ότι για ένα γραμμικώς διαχωρίσιμο σύνολο δεδομένων, η λύση μέγιστης πιθανοφάνειας για το μοντέλο logistic regression αντιστοιχεί στην εύρεση ενός διανύσματος w, για το οποίο η επιφάνεια απόφασης $w^T\phi(x) = 0$ διαχωρίζει τις κλάσεις, απειρίζοντας ταυτόχρονα το μέτρο του διανύσματος w.

   Παρατηρούμε ότι η λύση μέγιστης πιθανοφάνειας αντιστοιχεί στην περίπτωση όπου $\sigma = 0.5 \implies \frac{1}{1+exp(-w^T\phi)} = 0.5 \implies w^T\phi = 0$. Αφού είναι διαχωρίσιμα με επιφάνεια απόφασης την $w^T\phi(x) = 0$, για τα θετικά δείγματα $w^T\phi(x) > 0$ και για αρνητικά $w^T\phi(x) < 0$. Όσο και να αυθήσουμε το μέτρου του διανύσματος w, οι εξισώσεις αυτές θα συνεχίσουν να ισχύουν. Γι' αυτό το μέτρο του διανύσματος w απειρίζεται.

2. Η Hessian μήτρα για το logistic regression δίνεται από τη σχέση:

   $$H = [\Phi^T R \Phi]$$

   όπου Φ είναι ο πίνακας των χαρακτηριστικών και R είναι ένας διαγώνιος πίνακας με στοιχεία $y_n(1-y_n)$. Να δείξετε ότι η Hessian μήτρα H είναι θετικώς ορισμένη. Ως εκ τούτου, δείξτε ότι η συνάρτηση σφάλματος είναι κυρτή συνάρτηση του w και ότι έχει μοναδικό ελάχιστο.

   Έστω διάνυσμα $u \neq \vec{0}$. Πρέπει να δείξουμε ότι πάντα $u^T H u > 0$. Έχουμε:

   $$u^T H u = u^T \Phi^T R \Phi u$$

   Για τον πίνακα R:

   $$0 < y_n < 1 \implies 0 < y_n(1-y_n) < 1$$

   Άρα αφού όλα τα στοιχεία του πίνακα R είναι θετικό $u^T H u > 0$ και η Hessian μήτρα είναι θετικά ορισμένη. Αυτό σημαίνει ότι η συνάρτηση σφάλματος είναι κυρτή.

3. Να γράψετε κώδικα που θα υλοποιεί τον iterative reweighted least squares (IWLS) αλγόριθμο για logistic regression. Χρησιμοποιώντας τον αλγόριθμο αυτό, να υπολογίσετε και να σχεδιάσετε τα διαχωριστικά επίπεδα απόφασης που αντιστοιχούν στο σύνολο δεδομένων του προβλήματος τριών κλάσεων που έχει ανεβεί στο mycourses (αρχείο MLR.data). Οι δύο πρώτες στήλες περιλαμβάνουν τα διανύσματα χαρακτηριστικών, ενώ η τρίτη την κλάση. Συγκρίνετε τα αποτελέσματα με εκείνα που θα προέκυπταν εάν εφαρμοζόταν ταξινόμηση με βάση τα ελάχιστα τετράγωνα, σχεδιάζοντας τα αντίστοιχα διαχωριστικά επίπεδα απόφασης.

References

[1] G. Karagiannis and G. Steinhauer, *Pattern Recognition and Machine Learning.* NTUA, 2001.

[2] P. H. R. O. Duda and D. Stork, *Pattern Classification.* Wiley, 2001.

[3] C. M. Bishop, *Pattern Recognition and Machine Learning.* Springer, 2006.

[4] S. Theodoridis and K. Koutroumbas, *Pattern Recognition.* Academic Press, 2009.