ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

Τμήμα Ηλεκτρολόγων Μηχ. Και Μηχ. Υπολογιστών

Στατιστική Μοντελοποίηση και Αναγνώριση Προτύπων (ΤΗΛ311)

Φυλλάδιο Ασκήσεων 2

Οδηγίες:

1. **1.** Σας παρακαλώ να σεβαστείτε τον παρακάτω κώδικα τιμής τον οποίον θα θεωρηθεί ότι προσυπογράφετε μαζί με τη συμμετοχή σας στο μάθημα και τις εργασίες του:
   a) *Οι απαντήσεις στις εργασίες, τα quiz και τις εξετάσεις, ο κώδικας και γενικά οτιδήποτε αφορά τις εργασίες θα είναι προϊόν δικής μου δουλειάς.*
   b) *Δεν θα διαθέσω κώδικα, απαντήσεις και εργασίες μου σε κανέναν άλλο.*
   c) *Δεν θα εμπλακώ σε άλλες ενέργειες με τις οποίες ανέντιμα θα βελτιώνω τα αποτελέσματα μου ή ανέντιμα θα αλλάζω τα αποτελέσματα άλλων.*
2. Η εργασία είναι ατομική
3. Ημερομηνία παράδοσης: **Κυριακή, 26/4/2020 στις 23:00**
4. **Παραδοτέα:** α) Κώδικας και β) Αναφορά με τις απαντήσεις, παρατηρήσεις, πειράματα, αποτελέσματα και οδηγίες χρήσης του κώδικα.

Θέμα 1: Λογιστική Παλινδρόμηση: Αναλυτική εύρεση κλίσης (Gradient)

Υποθέτουμε ότι έχουμε ένα σύνολο m δεδομένων $\{(\boldsymbol{x}^{(1)}, y^{(1)}), (\boldsymbol{x}^{(2)}, y^{(2)}), \dots, (\boldsymbol{x}^{(m)}, y^{(m)})\}$, όπου $\boldsymbol{x}^{(i)} \in \mathbb{R}^{n\times 1}$ είναι τα διανύσματα χαρακτηριστικών και $y^{(i)} \in \{0,1\}$ ορίζουν την κλάση κάθε δείγματος (labels). Θέλουμε να προβλέψουμε τις τιμές των $y^{(i)}$ από τις αντίστοιχες τιμές $\boldsymbol{x}^{(i)}$, $i \in \{1,2,\dots,m\}$, χρησιμοποιώντας την συνάρτηση της λογιστικής παλινδρόμησης, η οποία ορίζεται ως εξής

$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = f(\boldsymbol{\theta}^T \boldsymbol{x})$$

όπου $\boldsymbol{\theta} = [\theta_1, \theta_2, \cdots, \theta_n]^T$ είναι οι παράμετροι του γραμμικού μοντέλου και $f()$ είναι η λογιστική συνάρτηση που ορίζεται ως:

$$f(z) = \frac{1}{1+e^{-z}}$$

Έστω $\hat{y}^{(i)} = h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})$ η εκτίμηση της λογιστικής συνάρτησης για το $y^{(i)}$. Σύμφωνα με τη θεωρία, στην περίπτωση της λογιστικής παλινδρόμησης, μπορούμε να υπολογίσουμε το σφάλμα με βάση τη συνάρτηση κόστους/σφάλματος (loss function), που ονομάζεται cross-entropy, και ορίζεται ως εξής:

$$J(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i=1}^{m}\left(-y^{(i)}\ln\left(\hat{y}^{(i)}\right) - (1-y^{(i)})\ln\left(1-\hat{y}^{(i)}\right)\right)$$

Αντικαθιστώντας το $\hat{y}^{(i)}$ έχουμε:

$$J(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i=1}^{m}\left(-y^{(i)}\ln\left(h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})\right) - (1-y^{(i)})\ln\left(1-h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})\right)\right)$$

Για τη βελτιστοποίηση του σφάλματος χρειάζεται να υπολογίσουμε την κλίση (gradient) του σφάλματος $J(\boldsymbol{\theta})$ η οποία θα είναι ένα διάνυσμα ίσης διάστασης με το $\boldsymbol{\theta}$.

a) Αν θ_j και $x_j^{(i)}$ είναι η j συνιστώσα των διανυσμάτων $\boldsymbol{\theta} = [\theta_1, \theta_2, \dots, \theta_n]^T$ και $\boldsymbol{x}^{(i)} = \left[x_1^{(i)}, x_2^{(i)}, \dots x_n^{(i)}\right]^T$ αντίστοιχα, να δείξετε ότι το j-στοιχείο της κλίσης του σφάλματος είναι:

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \theta_j} = \frac{1}{m}\sum_{i=1}^{m}\left(h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)}\right)x_j^{(i)}$$

b) Θα χρησιμοποιήσουμε την λογιστική παλινδρόμηση για να προβλέψουμε αν ένας φοιτητής θα γίνει δεκτός σε ένα πανεπιστήμιο με βάση τους βαθμούς του σε δύο εξετάσεις. Στο αρχείο «exam_scores_data1.txt» υπάρχουν δεδομένα από παλαιότερες αιτήσεις φοιτητών στη μορφή ***«Exam1Score, Exam2Score, [0: απόρριψη, 1: αποδοχή]»***, που θα χρησιμοποιηθούν ως δεδομένα εκμάθησης της λογιστικής παλινδρόμησης. Για να διαβάσετε τα δεδομένα τρέξτε την εντολή:

```
data = load('exam_scores_data1.txt');
```

Θα χρειαστεί να συμπληρώσετε κώδικα ώστε να τρέξει το κύριο αρχείο της άσκησης: ***My_logisticRegression.m***

i) Αρχικά δείτε τα δεδομένα με την συνάρτηση `plotData.m`

ii) Υλοποιήστε την σιγμοειδή συνάρτηση $f(z)$ στο αρχείο `sigmoid.m`. Αν η είσοδος z είναι πίνακας, η sigmoid θα πρέπει να εφαρμόζει την $f(z)$ σε κάθε στοιχείο του πίνακα.

iii) Υλοποιήστε την συνάρτηση *costFunction* στο αρχείο *costFunction.m* έτσι ώστε να επιστρέφει το κόστος $J(\boldsymbol{\theta})$ (J) και την κλίση $\nabla J(\boldsymbol{\theta})$ (grad) όπως ορίζονται στο ερώτημα (α) (***Σημείωση***: *Χρησιμοποιώντας ως αρχικές τιμές του* $\boldsymbol{\theta} = \mathbf{0}$*, το κόστος θα πρέπει να βγαίνει περίπου J=0.693 και η κλίση περίπου [-0.1, -12.009217, -11.262842]*).

iv) Η βελτιστοποίηση των παραμέτρων γίνεται με κώδικα που υπάρχει έτοιμος στην άσκηση `My_logisticRegression.m`. Εσείς απλά τρέξτε τον κώδικα και βρείτε το σύνορο απόφασης με την συνάρτηση `plotDecisionBoundary.m`. Επίσης τρέξτε την συνάρτηση `predict.m` για να προβλέψετε αν ο φοιτητής θα γίνει δεκτός με βάση διάφορες τιμές βαθμών στις δύο εξετάσεις.

Θέμα 2: Λογιστική Παλινδρόμηση με Ομαλοποίηση

Σε αυτή την άσκηση, θα εφαρμόσουμε ομαλοποιημένη λογιστική παλινδρόμηση για να προβλέψουμε αν τα μικροτσίπ από μια μονάδα κατασκευής περνούν τον έλεγχο ποιότητα (QA). Κατά τη διάρκεια της QA, κάθε μικροτσίπ περνάει από διάφορες δοκιμές για να εξασφαλιστεί ότι λειτουργεί σωστά. Υποθέτουμε ότι υπάρχουν τα αποτελέσματα δύο διαφορετικών δοκιμών για ορισμένα μικροτσίπ. Από αυτά τα αποτελέσματα θα πρέπει να καθορίσετε αν τα μικροτσίπ θα γίνουν αποδεκτά ή θα απορριφθούν. Θα χρησιμοποιήσουμε δεδομένα προηγούμενων δοκιμών για να δημιουργήσουμε ένα μοντέλο λογιστικής παλινδρόμησης. Θα πρέπει να συμπληρώσετε τον απαραίτητο κώδικα στα αρχεία matlab που βρίσκονται στον φάκελο *exercise2_2* ώστε να τρέξει η άσκηση από το κύριο αρχείο: *ex2_2_regularizedLogisticRegression.m*. Συγκεκριμένα:

a) Δείτε τα δεδομένα με την συνάρτηση `plotData.m`

b) Απεικονίστε τα δεδομένα σε χώρο μεγαλύτερης διάστασης όπου μπορούν να διαχωριστούν ευκολότερα με την λογιστική παλινδρόμηση. Συμπληρώστε την συνάρτηση `mapFeature.m` που απεικονίζει τα χαρακτηριστικά σε όλους τους όρους πολυωνύμων x_1 και x_2 βαθμού μέχρι και 6.

$$P(x_1, x_2) = \sum_{i=0}^{6} \sum_{j=0}^{i} x_1^j x_2^{i-j}$$

$$\mathrm{mapFeature}(x) = \left[1, x_1, x_2, x_1^2, x_1 x_2, x_2^2, x_1^3, \dots, x_1 x_2^5, x_2^6\right]^T$$

c) Η ομαλοποιημένη συνάρτηση κόστους δίνεται από την σχέση:

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} \left(-y^{(i)} \ln\left(h_\theta(x^{(i)})\right) - (1 - y^{(i)}) \ln\left(1 - h_\theta(x^{(i)})\right) + \frac{\lambda}{2m} \sum_{j=1}^{n} \theta_j^2\right.$$

d) Αν θ_j και $x_j^{(i)}$ είναι η j συνιστώσα των διανυσμάτων $\theta = [\theta_1, \theta_2, \dots, \theta_n]^T$ και

$x^{(i)} = \left[x_1^{(i)}, x_2^{(i)}, \dots x_n^{(i)}\right]^T$ αντίστοιχα, να δείξετε ότι το j-στοιχείο της κλίσης του σφάλματος είναι:

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)x_j^{(i)} + \frac{\lambda}{m}\theta_j$$

e) Συμπληρώστε την συνάρτηση `costFunctionReg.m`

f) Ο κώδικας για την βελτιστοποίηση των παραμέτρων υπάρχει στην άσκηση. Να βρείτε τα σύνορα απόφασης για διάφορες τιμές της παραμέτρου λ (π.χ. για λ =0, 1, 10, 100)

Θέμα 3: Εκτίμηση Παραμέτρων (Maximum Likelihood)

Υποθέστε ότι n δείγματα $D = \{x_1, \dots, x_n\}$ παράγονται ανεξάρτητα από μια κατανομή Poisson με παράμετρο λ:

$$p(x|\lambda) = \frac{\lambda^x e^{-\lambda}}{x!}, \quad x = 0,1,2, \dots \ , \quad \lambda > 0$$

Βρείτε τον εκτιμητή μεγίστης πιθανοφάνειας της παραμέτρου λ.

Θέμα 4: Εκτίμηση Παραμέτρων και Ταξινόμηση (ML - Naïve Bayes Classifier)

Σε αυτή την άσκηση θα υλοποιήσετε τον naive Bayes ταξινομητή για αναγνώριση ψηφίων. Τα δεδομένα της άσκησης υπάρχουν στο αρχείο `digits.mat`. Υπάρχουν 10 κλάσεις που αντιστοιχούν στα ψηφία από το 0 έως το 9. Κάθε πίνακας χαρακτηριστικών είναι μια ασπρόμαυρη εικόνα με διαστάσεις 28×28 ο οποίος μπορεί να αναπαρασταθεί σαν ένα διάνυσμα 784×1 με στοιχεία τις τιμές 0 ή 1.

Τα δείγματα εκπαίδευσης είναι στους πίνακες train0, …, train9. Αντίστοιχα τα δείγματα δοκιμής είναι στους πίνακες test0, …, test9. Σε κάθε πίνακα, κάθε γραμμή είναι ένα διάνυσμα το οποίο μπορεί να γίνει μία 28×28 ασπρόμαυρη εικόνα για οπτικοποίηση.

```
load('digits');
A = reshape(train3(43, :), 28, 28)';
imagesc(A);
Colorbar
```

Στον naive Bayes ταξινομητή, υποθέτουμε ότι τα χαρακτηριστικά, τα οποία στην περίπτωσή μας, είναι εικονοστοιχία (pixel) των εικόνων, είναι ανεξάρτητα. Θα χρησιμοποιήσουμε την κατανομή Bernoulli για να μοντελοποιήσουμε το κάθε χαρακτηριστικό (pixel) κάθε κλάσης.

Έστω x^{y_i} ότι είναι η τυχαία μεταβλητή που παριστάνει το i-στο χαρακτηριστικό του ψηφίου y και p^{y_i} η αντίστοιχη παράμετρος της κατανομής Bernoulli του i-στού χαρακτηριστικού του ψηφίου y.

1. Βρείτε τον εκτιμητή μεγίστης πιθανοφάνειας της παραμέτρου p^{y_i} δοσμένων n δειγμάτων $\{x_1^{y_i}, x_2^{y_i}, \dots, x_n^{y_i}\}$
2. Χρησιμοποιήστε τους εκτιμητές μεγίστης πιθανοφάνειας για να εκπαιδεύσετε μοντέλα για κάθε ψηφίο χρησιμοποιώντας τα δοσμένα δείγματα εκπαίδευσης. Οπτικοποιήστε τα εκπαιδευμένα μοντέλα για κάθε ψηφίο σχηματίζοντας για κάθε ψηφίο 28×28 εικόνες των οποίων η φωτεινότητα κάθε εικονοστοιχείου αναπαριστά την παράμετρο p^{y_i} της κατανομής Bernoulli που αντιστοιχεί στο i-στο εικονοστοιχείου του ψηφίου y.
3. Χρησιμοποιήστε τα εκπαιδευμένα μοντέλα για να ταξινομήσετε τα ψηφία στο σύνολο δοκιμής (test set). Βρείτε την ακρίβεια της ταξινόμησης (classification accuracy) ως τον λόγο των ψηφίων που έχουν ταξινομηθεί σωστά ως προς το σύνολο των δειγμάτων ελέγχου. Υπολογίστε τον πίνακα confusion matrix που το στοιχείο του (i, j) παριστάνει πόσο συχνά η εικόνα του ψηφίου i ταξινομείται ως ψηφίο j.

Θέμα 5: Bayesian εκτίμηση παραμέτρου (Bayesian Inference)

Θεωρήστε ένα πείραμα όπου κάνουμε n μετρήσεις στις χρονικές στιγμές $\kappa = 1,2,\dots,n$. Στο αρχείο `data_2_5.txt` δίνεται ένα σύνολο $H(n) = \{x_k,\ k = 1,\dots,n\}$, n μετρήσεων, όπου $n = 25$. Υποθέτουμε ότι η συνάρτηση πυκνότητας πιθανότητας των μετρήσεων είναι Γκαουσιανή $p(x) \sim \mathcal{N}(\mu, \sigma^2)$, με γνωστή τυπική απόκλιση $\sigma = 1.25$. Επίσης, υποθέτουμε ότι έχουμε κάποια εκ των προτέρων πληροφορία για την κατανομή της μέσης τιμής μ: $p(\mu) \sim \mathcal{N}(\mu_0, \sigma_0^2)$.

Σε αυτή την άσκηση, μελετούμε την Bayesian εκτίμηση του μ.

α. Υποθέτοντας ότι $\mu_0 = 0$ και $\sigma_0^2 = 10\sigma^2$, ζωγραφήστε στο ίδιο σχήμα την δεσμευμένη πυκνότητα $p(\mu|H(n))$, καθώς το n μεταβάλλεται από 1 έως 25.

β. Σχεδιάστε τις δεσμευμένες πυκνότητες $p(x|H(n))$ που προκύπτουν από την Bayesian εκτίμηση όταν $\sigma_0^2 = 10\sigma^2$, $\sigma_0^2 = \sigma^2$, $\sigma_0^2 = 0.1\sigma^2$ και $\sigma_0^2 = 0.01\sigma^2$. Τι παρατηρείτε;

Θέμα 6: Support Vector Machines (Αναλυτική βελτιστοποίηση με KKT)

Έστω ότι θέλουμε να χρησιμοποιήσουμε Support Vector Machines (SVM) σε ένα πρόβλημα ταξινόμησης σε δύο κατηγορίες ω_1και ω_2.

A. Έχουμε τα ακόλουθα 2D δείγματα εκπαίδευσης $\boldsymbol{x} = [x_1, x_2]$ για τις δύο κλάσεις:

$$\omega_1 : \boldsymbol{x}^+ = \{[3,1]^T, [3,-1]^T, [5,1]^T, [5,-1]^T\}$$

$$\omega_2 : \boldsymbol{x}^- = \{[1,0]^T, [0,1]^T, [0,-1]^T, [-1,0]^T\}$$

1. Σχεδιάστε στο χώρο τα δείγματα. Ποια είναι τα support vectors; Μπορείτε να προτείνετε διαισθητικά με βάση το διάγραμμα ποιο θα είναι το βέλτιστο γραμμικό υπερεπίπεδο διαχωρισμού με βάση τον αλγόριθμο SVM; Αιτιολογήστε την απάντηση σας
2. Χρησιμοποιώντας πολλαπλασιαστές Langrange και τις συνθήκες Karush-Khun-Tucker (KKT) βρείτε αναλυτικά την βέλτιστη γραμμή διαχωρισμού. Ελέγξτε το αποτέλεσμα με αυτό που προτείνατε διαισθητικά στο ερώτημα 1.

B. Έχουμε τα ακόλουθα δισδιάστατα δείγματα εκπαίδευσης $\boldsymbol{x} = [x_1, x_2]$ για τις δύο κλάσεις:

$$\omega_1 : \boldsymbol{x}^+ = \{[2,2]^T, [2,-2]^T, [-2,-2]^T, [-2,2]^T\}$$

$$\omega_2 : \boldsymbol{x}^- = \{[1,1]^T, [1,-1]^T, [-1,-1]^T, [-1,1]^T\}$$

1. Σχεδιάστε στο χώρο τα δείγματα. Ποια θα είναι τα support vectors σ’ αυτή την περίπτωση και ποιο το βέλτιστο γραμμικό υπερεπίπεδο διαχωρισμού μπορείτε να προτείνετε με βάση τον αλγόριθμο SVM; Εξηγήστε την απάντηση σας.
2. Έστω ότι εφαρμόζετε στα δεδομένα σας τον παρακάτω μετασχηματισμό:
$$\Phi(\boldsymbol{x}) = \boldsymbol{x} - 4 \cdot \left||\boldsymbol{x}|\right|_2 - 4, \ \text{όπου } \|\boldsymbol{x}\|_2^2 = x_1^2 + x_2^2$$
Υπολογίστε και σχεδιάστε τα νέα διανύσματα χαρακτηριστικών που προκύπτουν. Μπορείτε να προτείνετε γραμμικό υπερεπίπεδο διαχωρισμού;
3. Χρησιμοποιώντας πολλαπλασιαστές Langrange και τις συνθήκες Karush-Khun-Tucker (KKT), βρείτε αναλυτικά το βέλτιστο υπερεπίπεδο διαχωρισμού των μετασχηματισμένων δειγμάτων με βάση τον αλγόριθμο SVM.

Σημείωση*: Στην άσκηση αυτή μπορείτε να χρησιμοποιήσετε το Matlab για να σχεδιάσετε στο χώρο τα δείγματα.*

Θέμα 7: Support Vector Machines (Εφαρμογή σε τεχνητό σύνολο δεδομένων)

Υποθέστε ότι έχουμε ένα σύνολο n παραδειγμάτων $\{(\boldsymbol{x}^{(1)}, y^{(1)}), (\boldsymbol{x}^{(2)}, y^{(2)}), \dots, (\boldsymbol{x}^{(n)}, y^{(n)})\}$, όπου $\boldsymbol{x}^{(i)} \in \mathbb{R}^{1\times 2}$ και $y^{(i)} \in \{-1,1\}$. Συνήθως τα $\boldsymbol{x}^{(i)}$ ορίζονται σαν διανύσματα στήλης, όμως στο συγκεκριμένο πρόβλημα αποθηκεύουμε όλα τα δεδομένα σε έναν πίνακα X όπου οι γραμμές είναι τα δείγματα και οι στήλες τα χαρακτηριστικά. Θέλουμε να προβλέψουμε τις τιμές των $y^{(i)}$ από τις αντίστοιχες τιμές $\boldsymbol{x}^{(i)}$, $i \in \{1,2,\dots,n\}$, χρησιμοποιώντας Support Vector Machines (SVM).

$$\hat{y} = \begin{cases} 1, \text{ εάν } \boldsymbol{w} \cdot \boldsymbol{x}^T + w_0 > 0 \\ -1, \text{ εάν } \boldsymbol{w} \cdot \boldsymbol{x}^T + w_0 < 0 \end{cases}$$

Οι παράμετροι του SVM προσδιορίζονται λύνοντας ένα πρόβλημα βελτιστοποίησης με περιορισμούς.

Μέρος 1ο: Γραμμικά διαχωρίσιμα δείγματα.
Η Λαγκρανζιανή του δυικού προβλήματος είναι

$$\tilde{L}(\lambda) = \sum_{i=1}^{n} \lambda^{(i)} - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda^{(i)} \lambda^{(j)} y^{(i)} y^{(j)} K(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})$$

όπου $K\left(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}\right) = \boldsymbol{x}^{(i)} \cdot \left(\boldsymbol{x}^{(j)}\right)^T$
την οποία **μεγιστοποιούμε** κάτω από τους περιορισμούς

$$\lambda^{(i)} \geq 0, \quad i = 1, \dots, n$$

$$\sum_{i=1}^{n} \lambda^{(i)} y^{(i)} = 0$$

Να συμπληρώσετε τον κώδικα matlab, `exercise1.m`

α. **Ελαχιστοποιώντας** κατάλληλη συνάρτηση να υπολογίσετε τους πολλαπλασιαστές του Lagrange $\lambda^{(i)}, \quad i = 1, \dots, n$
β. Να υπολογίσετε τα βάρη w και το bias w_0 του SVM και το πλάτος της λωρίδας μεταξύ των support vectors.
γ. Να βάλετε στην αναφορά σας την γραφική παράσταση της γραμμής διαχωρισμού και των παράλληλων γραμμών που περνούν από τα support vectors. Χρησιμοποιήστε μόνο τα 50 από τα 51 παραδείγματα του αρχείου `twofeature1.txt`

Μέρος 2ο: Μεταβλητές περιθωρίου (slack variables).
Όταν υπάρχει θόρυβος στα δεδομένα χρησιμοποιούμε μεταβλητές περιθωρίου για να γίνονται ανεκτά κάποια παραδείγματα στην λάθος πλευρά της επιφάνειας απόφασης. Η παράμετρος που καθορίζει το επίπεδο ανοχής συνήθως συμβολίζεται με C και παίρνει τιμές στο διάστημα $(0, +\infty)$, με 0^+ να αντιστοιχεί σε μέγιστη ανοχή και $+\infty$ σε μηδενική ανοχή.
Η Λαγκρανζιανή του δυικού προβλήματος είναι ίδια με αυτή του πρώτου μέρους αλλά την **μεγιστοποιούμε** κάτω από τους παρακάτω περιορισμούς:

$$0 \leq \lambda^{(i)} \leq C, \quad i = 1, \dots, n$$

Αντιγράψετε το αρχείο του κώδικα `exercise1.m` σε `exercise2.m` και επαναλάβετε τα ερωτήματα α), β) και γ) του 1ου μέρους κάνοντας τις απαραίτητες τροποποιήσεις και χρησιμοποιώντας όλα τα 51 παραδείγματα του αρχείου `twofeature1.txt`.

Μέρος 3ο: Μη γραμμικά διαχωρίσιμα παραδείγματα.
Χρησιμοποιώντας την Λαγκρανζιανή του παραπάνω ερωτήματος συμπληρώστε τον κώδικα του αρχείου `exercise3.m` και χρησιμοποιήστε τα δεδομένα του αρχείου `twofeature2.txt`. Τα παραδείγματα αυτού του αρχείου δεν είναι γραμμικά διαχωρίσιμα. Για αυτό στο διάνυσμα $\boldsymbol{x} = (x_1, x_2)$ κάθε παραδείγματος προσθέσουμε σαν τρίτο χαρακτηριστικό το μέτρο του x στο τετράγωνο $\boldsymbol{x} = (x_1, x_2) \rightarrow (x_1, x_2, \|x\|_2^2)$, όπου $\|\boldsymbol{x}\|_2^2 = x_1^2 + x_2^2$.
Επαναλάβετε τα ερωτήματα α), β) και γ) λαμβάνοντας υπ' όψη ότι στο 3ο μέρος δεν χρειάζεται να υπολογίσετε το πλάτος της λωρίδας μεταξύ των support vectors.