Άσκηση 2

Στατιστική Μοντελοποίηση

Αναστάσιος Πατερίτσας , ΑΜ: 2016030065

Θέμα 1.

Ο σκοπός του συγκεκριμένου θέματος είναι να κάνουμε μια εκτίμηση μέσω λογιστικής παλινδρόμησης, όποτε ο στόχος μας είναι να βρούμε τις κατάλληλες παραμέτρους για να κάνουμε καλυτέρα την πρόβλεψη. Από την εκφώνηση της άσκησης γνωρίζουμε ότι η συνάρτηση λογιστικής παλινδρόμησης είναι $h_\theta(x) = f(\Theta^T x)$ οπου υλοποιηται στην συναρτηση sigmoid. Έπειτα υπολογίσαμε την cross entropy (loss function) με βάση τον τύπο :

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

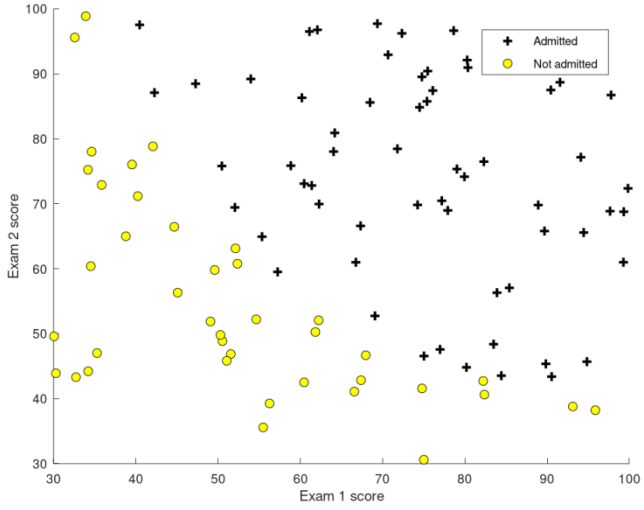
Και υπολογίζουμε την κλήση του σφάλματος με τον τύπο :

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

Απόδειξη τέλος pdf*.

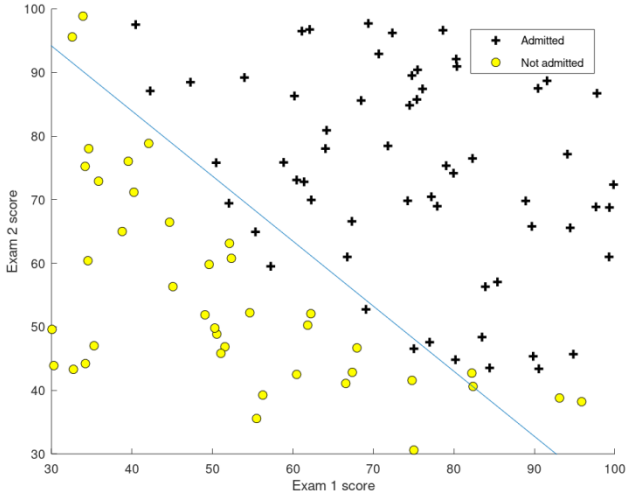
Και οι δυο τύποι υπολογίζονται στην συνάρτηση costFunction.

Αρχικά απεικονίσαμε τα δεδομένα με την plotData



Ενώ στην συνεχεία εκτελέσαμε τα βήματα που προαναφέρθηκαν για τον υπολογισμό του σφάλματος και της κλήσης. Τα αποτελέσματα της συνάρτησης του σφάλματος είναι πολύ λογικά και είναι φυσιολογικό η τιμή του J να είναι πολύ υψηλή (0.69) ενώ το gradient διάνυσμα είναι [-0.1 ,-12 , -11.26] .

Εφόσον εφαρμόσαμε την συνάρτηση fminunc η όποια βρίσκει το τοπικό ελάχιστο μιας συνάρτησης πολλών παραμέτρων . Τα αποτελέσματα που βρεθήκαν ήταν εμφανώς καλυτέρα από τα προηγούμενα (Κόστος: 0.2 , gradient:[-25.16,0.2,0.2]).

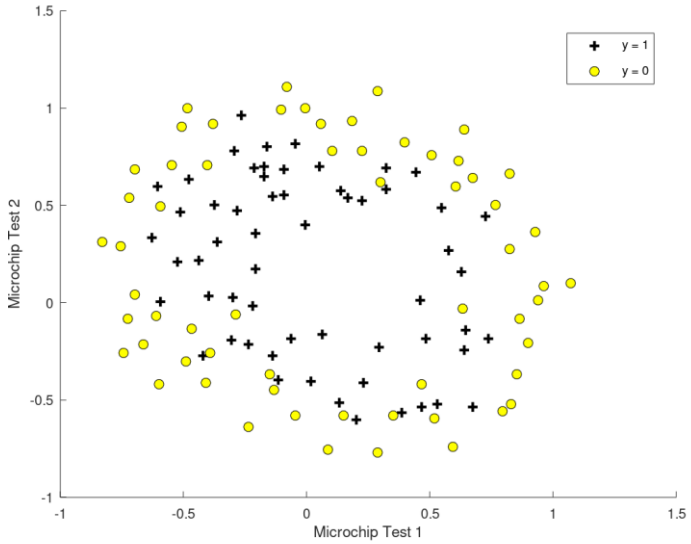


Έχοντας στην διάθεση μας τις συνιστώσες του dicision boundary μπορούμε να κάνουμε πρόβλεψη για τον μαθητή που έχει γράψει στο πρώτο διαγώνισμα 45 και στο δεύτερο 85. Η πιθανότητα να περάσει είναι 0.77 με ακρίβεια 0.89 .Αν δούμε και στην παραπάνω εικόνα , θα παρατηρήσουμε ότι 11 από τα 100 δείγματα είναι λάθος ταξινομημένα.

Για να πάρουμε την βέλτιστη λύση θα πρέπει η σιγμοειδούς συνάρτησης σε συνδυασμό με την cross entropy θα πρέπει να συνδυαστεί με την εύρεση κατάλληλου τοπικού ελάχιστου.

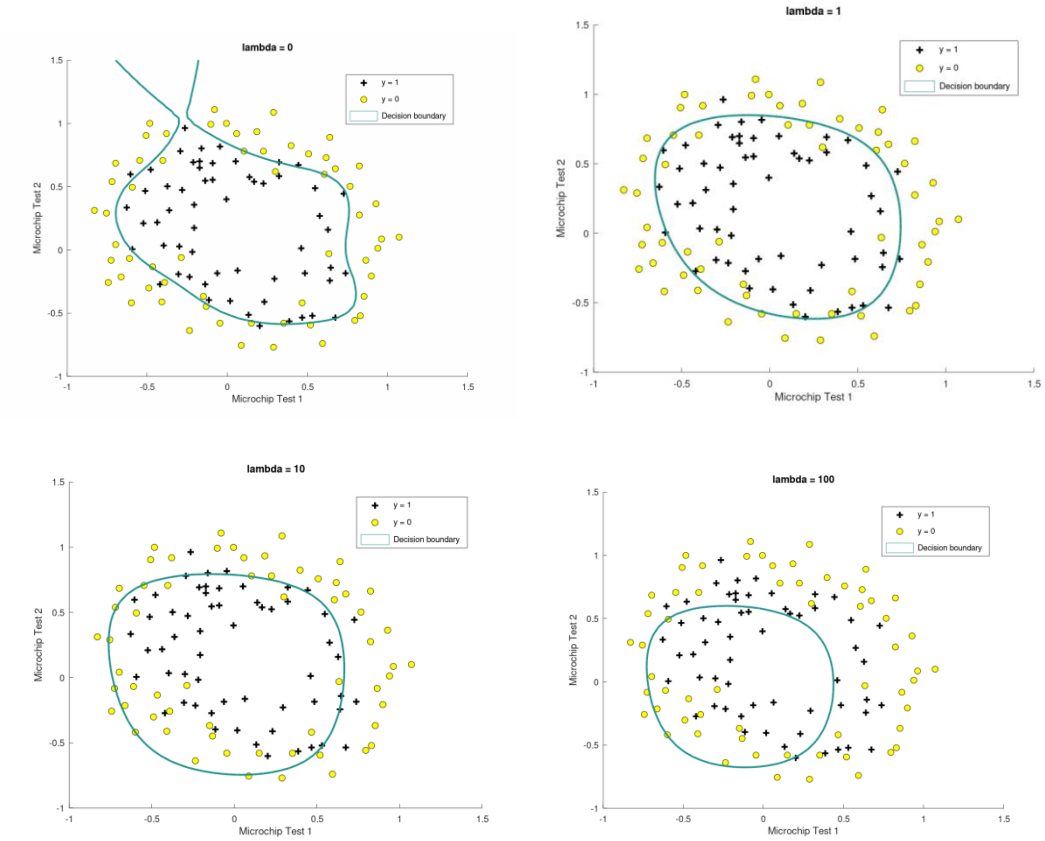
Θέμα 2.

Για το δεύτερο θέμα χρειάστηκε να εφαρμόσουμε ομαλοποιημένη λογιστική παλινδρόμηση για να προβλέψουμε αν περνούν τον έλεγχο ποιότητας χρησιμοποιώντας δυο από τα αποτελέσματα δοκιμών. Τα δεδομένα ήταν τα εξής:



Στην συνεχεία χρειάστηκε να απεικονίσουμε τα δεδομένα σε μεγαλύτερη διάσταση διότι με αυτό τον τρόπο θα μπορούμε να τα διαχωρίσουμε ευκολότερα με την λογιστική παλινδρόμηση. Ο τύπος που μετατρέπει τα δεδομένα σε μεγαλύτερη διάσταση είναι $\sum_{i=0}^{6}\sum_{j=0}^{i} x1^{i-j}\, x2^{j}$. Έπειτα χρησιμοποιήσαμε αυτά τα δεδομένα και με λ =1 βρήκαμε το κόστος με την τιμή του J να είναι υψηλή (0.69 το initial cost).

Εφόσον εφαρμόσαμε την συνάρτηση fminunc η όποια βρίσκει το τοπικό ελάχιστο μιας συνάρτησης πολλών παραμέτρων . Τα αποτελέσματα που βρεθήκαν ήταν εμφανώς καλυτέρα από τα προηγούμενα (Κόστος: 0.28, lamda = 0).



Όπως μπορούμε να παρατηρήσουμε , το πόσο καλά αποτελέσματα θα πάρουμε όσον αφορά τα σύνορα εξαρτάται κατά πολύ από το λ . Είναι εύκολο να παρατηρηθεί ότι για λ =0 έχουμε τα καλυτέρα αποτελέσματα με train accuracy = 86.44 και κόστος 0.28(overfitting λογο του κακο σχημα). Για λ = 1 έχουμε train accuracy 83.1 , για λ = 10 train accuracy 74.57 ενώ για λ = 100 train accuracy 61.01(underfitting) με το κόστος να γίνεται 0.68 δηλαδή λίγο καλυτερο από το initial cost.

Θέμα 3.

Για το τρίτο θέμα έπρεπε να βρούμε τον εκτιμητή μεγίστης πιθανοφανείς της παραμέτρου λ για n δείγματα που παράγονται ανεξάρτητα από μια κατανομή Poisson με παράμετρο λ.

Likelihood function:

$$L_D(\lambda) = \prod_{i=1}^{n} \lambda^{xi} \frac{e^{-\lambda}}{xi!}$$

Η log-likelihood είναι :

$$l_D(\lambda) = \log\big(L_D(\lambda)\big) = \sum_{i=1}^{n} \log\left(\lambda^{xi} \frac{e^{-\lambda}}{xi!}\right) =$$

$$= \sum_{i=1}^{n} (-\lambda + xilog(\lambda) - logxi!) =$$

$$= -n\lambda - \sum_{i=1}^{n} logxi! + log \sum_{i=1}^{n} xi$$

Η Maximum Likelihood Estimation μπορεί να βρεθεί με την επίλυση του κυρτού προβλήματος βελτιστοποίησης $\text{argmax}_\lambda(l_D(\lambda))$

$$\nabla_{\lambda_{MLE}} l_D(\lambda) = 0 \leftrightarrow -n + \frac{1}{\lambda_{MLE}} \sum_{i=0}^{n} xi = 0$$

$$\lambda_{MLE} = \frac{1}{n} \sum_{i=0}^{n} xi$$

Ωστόσο το λ_{MLE} είναι ο μέσος Όρος των n παρατηρήσεων στο δείγμα.

Θέμα 4.

Σε αυτό το θέμα κληθήκαμε να φτιάξουμε ένα Naïve Bayes classifier για αναγνώριση ψηφίων από το 0 έως το 9. Στο αρχείο digits.mat υπάρχουν 10 κλάσεις όπου σε κάθε μια κλάση 500 στοιχειά τα όποια είναι μια ασπρόμαυρη εικόνα με διαστάσεις 28x28 οι οποίες αναπαριστώνται σαν πινάκας 784x1 με τιμές 0 η 1. Για τον Naive Bayes ,υποθέσαμε ότι κάθε pixel είναι ανεξάρτητο από τα αλλά , και χρησιμοποιήσαμε την κατανομή Bernouli για να μοντελοποιήσουμε κάθε πιξελ.

Likelihood function:

$$L_{Dx^{yi}} = P\left(D_{x^{yi}} \middle| p^{yi}\right) = \prod_{i=1}^{n} P\left(X = x_j{}^{yi}\right) = \left(p^{yi}\right)^k (1 - p^{yi})^{k'}$$

Το k = $\sum_{j=1}^{n} x_j{}^{yi}$ και k' = 1- k.

Log likelihood

$$l_{Dx^{yi}}\left(p^{yi}\right) = \log\left(\left(p^{yi}\right)^k \left(1 - p^{yi}\right)^{k'}\right) = k\log\left(p^{yi}\right) + k'\log(1 - p^{yi})$$

$$\nabla_{p^{yi}{}_{MLE}}\, l_{Dx^{yi}}\left(p^{yi}\right) = 0 \leftrightarrow \frac{k}{p^{yi}{}_{MLE}} - \frac{k'}{1 - p^{yi}{}_{MLE}} = 0$$

$$p^{yi}{}_{MLE} = \frac{k}{k + k'}$$

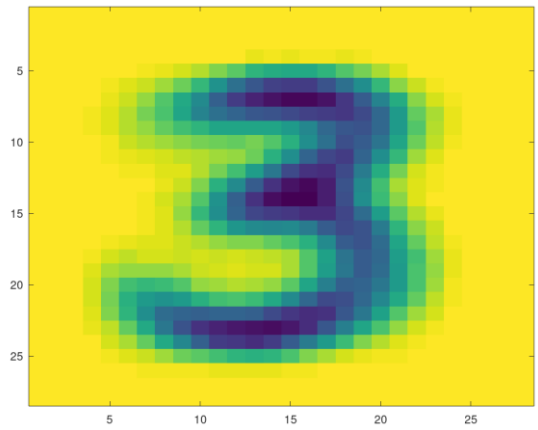
$p^{yi}{}_{MLE} = \frac{1}{n}\sum_{j=1}^{n} x_j{}^{yi}$

Όποτε η $p^{yi}{}_{MLE}$ δινεται από τον μεσο ορο του πιξελ της κλασης.

Χρησιμοποιώντας τον παραπάνω τύπο εκπαιδεύσαμε τα μοντέλα για κάθε ψηφίο αφού πρώτα υπολογίσαμε τους εκτιμητές μεγίστης πιθανοφανείς (άθροισμα όλων των γραμμών της κάθε κλάσης διαιρεμένο με τον αριθμό των εικόνων που έχει δηλαδή 500) Στην συνεχεία αφού

εκπαιδεύσαμε τα μοντέλα για κάθε αριθμό οπτικοποιησαμε το κάθε μοντέλο για το κάθε ψηφίο.

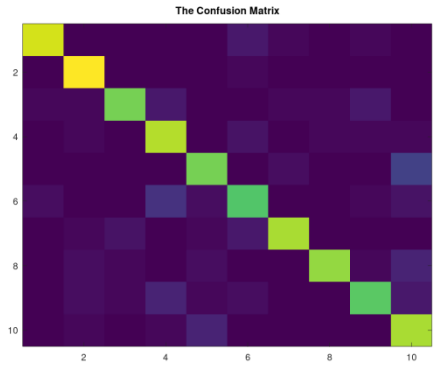
Παράδειγμα του αριθμού 3



Έπειτα εφαρμόσαμε την αντίστοιχη διαδικασία με τα εκπαιδευμένα μοντέλα για την πρόβλεψη των αριθμών στο τεστ σετ .Για κάθε εικόνα του σετ επιλέχτηκε το στοιχειό που είχε την μεγαλύτερη λογαριθμημενη πιθανότητα. Με τα εξής αποτελέσματα :

```
Accuracy for testing digits 0 : 0.882
Accuracy for testing digits 1 : 0.946
Accuracy for testing digits 2 : 0.748
Accuracy for testing digits 3 : 0.83
Accuracy for testing digits 4 : 0.752
Accuracy for testing digits 5 : 0.69
Accuracy for testing digits 6 : 0.824
Accuracy for testing digits 7 : 0.788
Accuracy for testing digits 8 : 0.696
Accuracy for testing digits 9 : 0.818
Accuracy: 0.7974
```

Ενώ ο confusion matrix είναι ο εξής :



Θέμα 5.

Στο συγκεκριμένο θέμα έπρεπε να πειραματιστούμε με n μετρήσεις τις χρονικές στιγμές κ=1,2…n . Ενώ υποθέτουμε ότι η συνάρτηση πυκνότητας πιθανότητας των μετρήσεων είναι Gauss.

Για το α ερώτημα έπρεπε να ζωγραφίσουμε στο ίδιο σχήμα την δεσμευμένη πιθανότητα p(μ|H(n)) καθώς το n μεταβάλλεται από 1 έως 25. Από θεωρία γνωρίζουμε ότι:

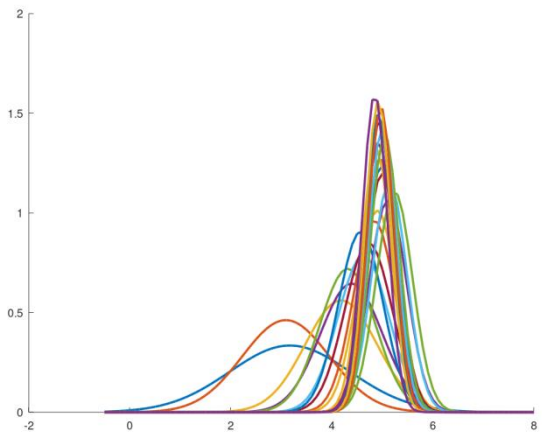
$$p(\mu|H(n)) = \frac{1}{\sqrt{2\pi}\,\sigma_n}\exp[-\frac{1}{2}(\frac{\mu-\mu_n}{\sigma_n})^2]$$

Με

$$\mu_n = \left(\frac{n\sigma_0^2}{n\sigma_0^2+\sigma^2}\right)\bar{x}_n + \frac{\sigma^2}{n\sigma_0^2+\sigma^2}\mu_0 \quad , \quad \sigma_n^2 = \frac{\sigma_0^2\sigma^2}{n\sigma_0^2+\sigma^2}. \quad ,$$

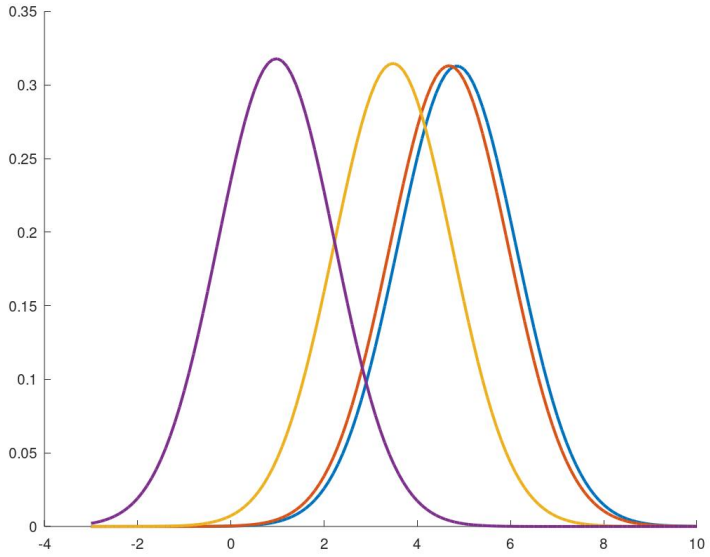
$$\bar{x}_n = \frac{1}{n}\sum_{k=1}^{n} x_k.$$

Όμως από την εκφώνηση γνωρίζουμε το μ0=0 όποτε ο δεύτερος Όρος της πρόσθεσης στο μ_n μηδενίζεται.



Για το δεύτερο ερώτημα κληθήκαμε να σχεδιάσουμε $p(\chi|H(n))$ που προκυπτουν από την Bayesian εκτιμηση όταν το $\sigma 0^2 = 10\ \sigma^2$, $\sigma 0^2 = \sigma^2$, $\sigma 0^2 = 0.1\sigma^2$ και $\sigma 0^2 = 0.01\ \sigma^2$

Από την θεωρία προκύπτει ότι η $p(\chi|H(n))$ είναι ομοιομορφα κατανεμημενη με mean μ_n και varience $\sigma^2 + \sigma^2_n$. Όποτε με τον ίδιο τρόπο όπως στο ερώτημα α αλλά για n =25 έχουμε :



Σε αυτό το σημείο παρατηρήσαμε ότι η πιθανότητες δε αλλάζουν για τις διαφορές τιμές του σ^2
(πλάτος σταθερό) αλλά μόνο μετατοπίζονται στον άξονα του x(ολισθαίνει αριστερά).

Θέμα 7.

Για το μέρος πρώτο έπρεπε να εφαρμόσουμε SVM για γραμμικά διαχωρίσιμα δείγματα. Για να το κάνουμε αυτό έπρεπε να υπολογίσουμε την Λαγκαρσιανη του δυικού προβλήματος. Για να γίνει αυτό χρησιμοποιούμε γραμμικό kernel που δίνετε από τον τύπο :

$$K(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}) = \boldsymbol{x}^{(i)} \cdot (\boldsymbol{x}^{(j)})^T$$

Ενώ η Λαγκαρσιανη

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

Την όποια την υπολογίσαμε με την συνάρτηση της matlab quadprog. Για τον υπολογισμό του διανύσματος w χρησιμοποιήσαμε τον τύπο :

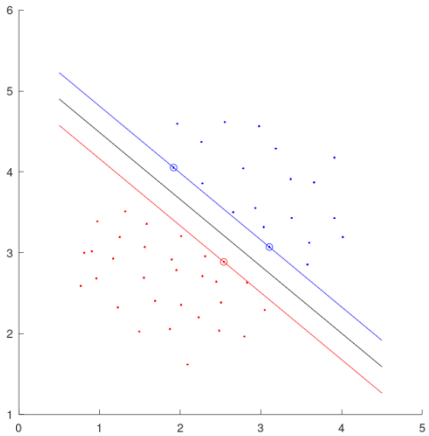
$$\mathbf{w} = \mathbf{X}^{\intercal} \boldsymbol{\alpha} \mathbf{y}$$

Όπου α είναι τα λ. Ενώ υπολογίσαμε το bias με τον τύπο :

$w0 = (y - Xw)_{\epsilon S}$ που S τα Support vector subset

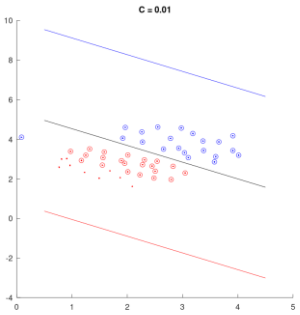
Ενώ το width = -w0/max(w)

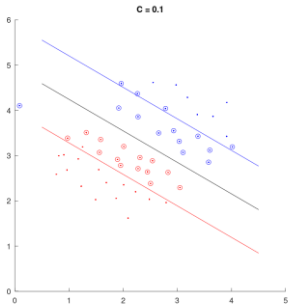
Η γραφική παράσταση διαχωρισμού είναι

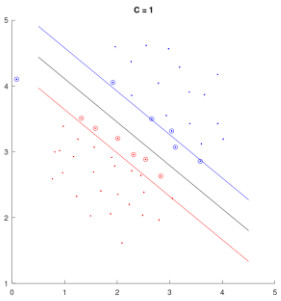


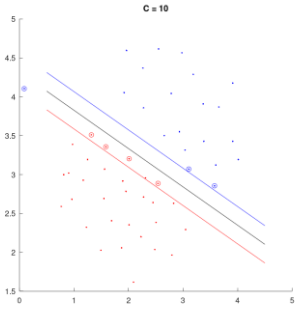
Εύκολα παρατηρούνται τα 3 Support Vector Machines 1 τα αρνητικά και 2 για τα θετικά. Με κόκκινο είναι η παράλληλη στην γραμμή διαχωρισμού που παρανοεί από το αρνητικό SVM ενώ η μπλε αντίστοιχα για τα θετικά SVM.

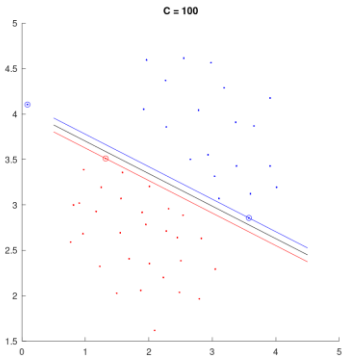
Για το β μέρος χρησιμοποιήσαμε τον ίδιο κώδικα αλλά για τα διάφορα C τα όποια περιορίζουν τα λ :

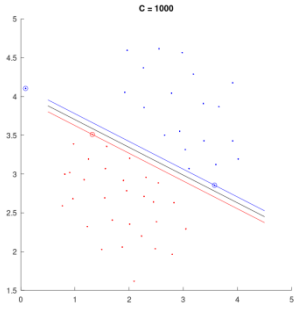


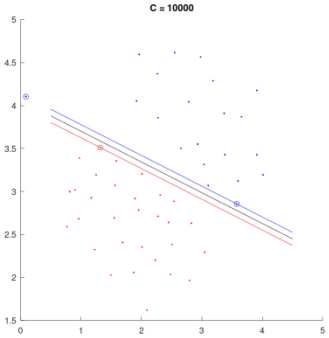






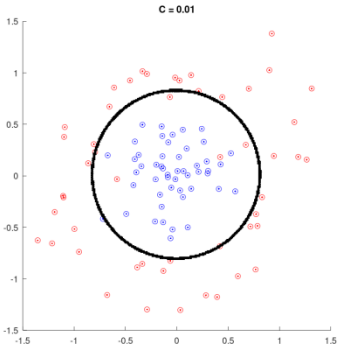


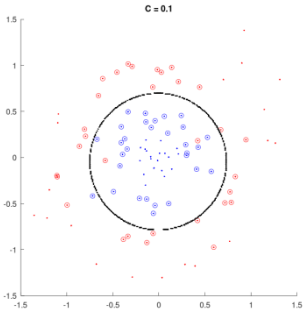


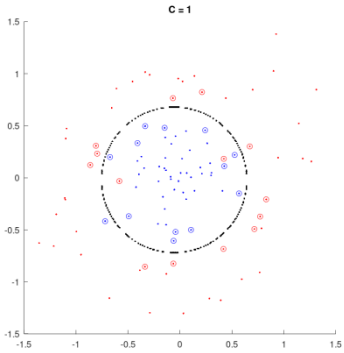


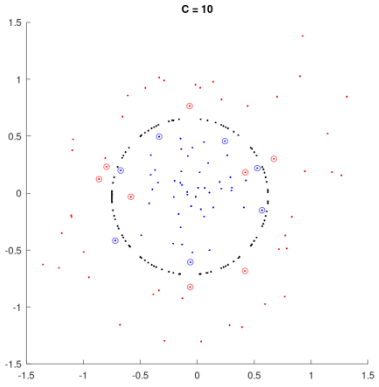
Όπως μπορεί να παρατηρηθεί όσο μικρότερο είναι το C τόσο λιγότερα λ έχουμε άρα τα αποτελέσματα είναι χειρότερα και έχουμε παρά πολλά SVM με αποτέλεσμα να έχουμε misclassifications.

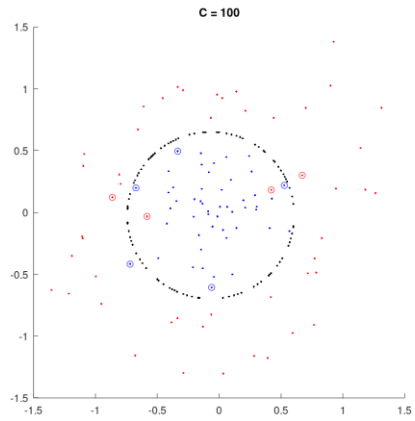
Για το γ ερώτημα τα δεδομένα μας δε είναι γραμμικά όποτε δε θα μπορούσαμε να χρησιμοποιήσουμε τον συγκεκριμένο kernel. Για τον λόγο αυτό προσθέτουμε ένα τρίτο χαρακτηριστικό στο διάνυσμα , και με αυτό το trick τα δεδομένα γίνονται γραμμικά χωρισμένα στον τρισδιάστατο χώρο.

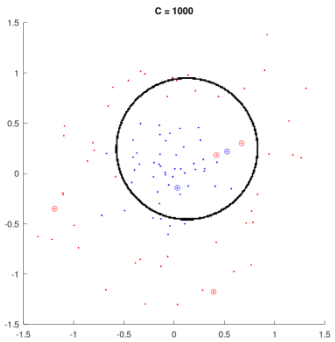


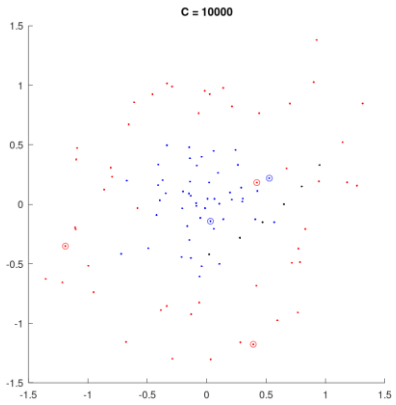












Όπως και πριν για μικρά C βρίσκουμε πολλά SVM όμως για πολύ μεγάλο C βρίσκουμε πολύ λίγα και δε μπορεί να βρει decision Boundary.

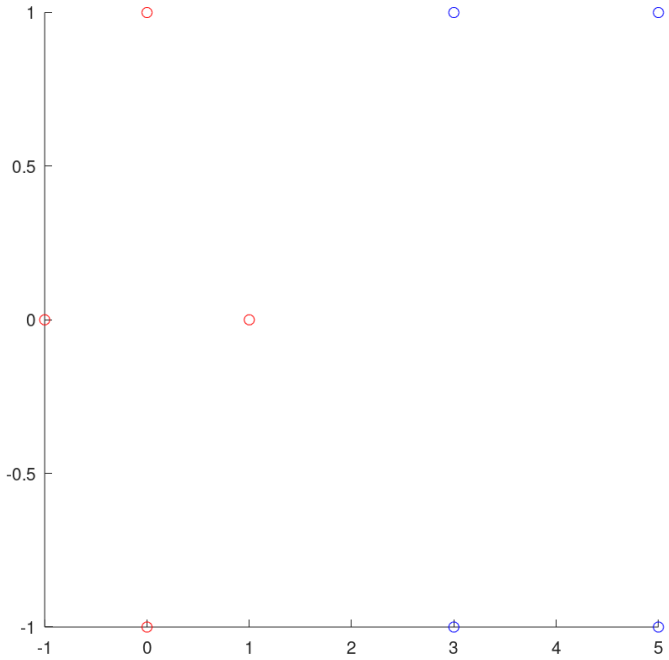
Συμπερασμα για β,γ :

Το C πρεπει να επιλεγεται στο χερι για την καθε διανομη.

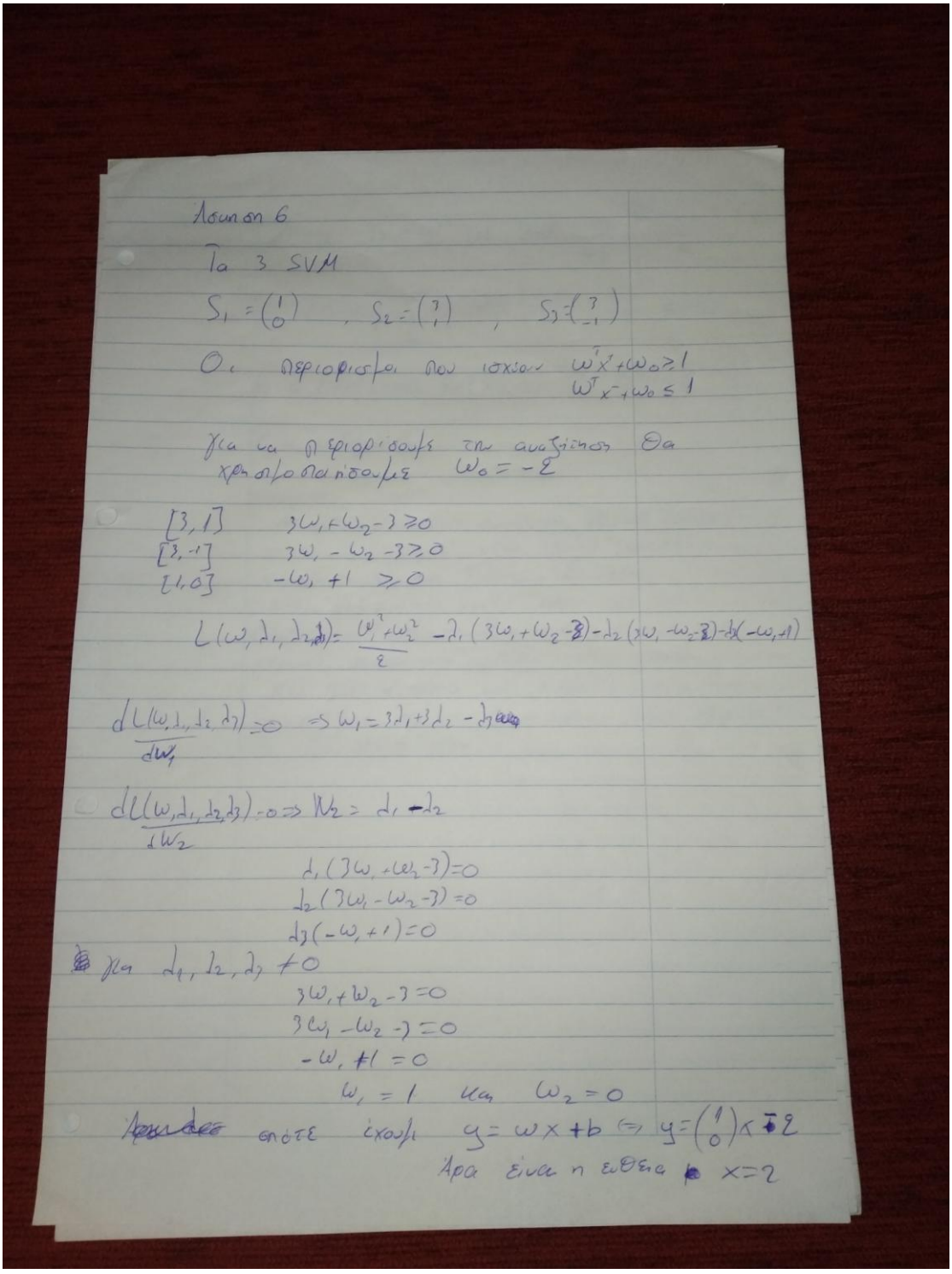
Θεμα 6.

Α)

Τα δειγματα στον χωρο είναι τα εξης:

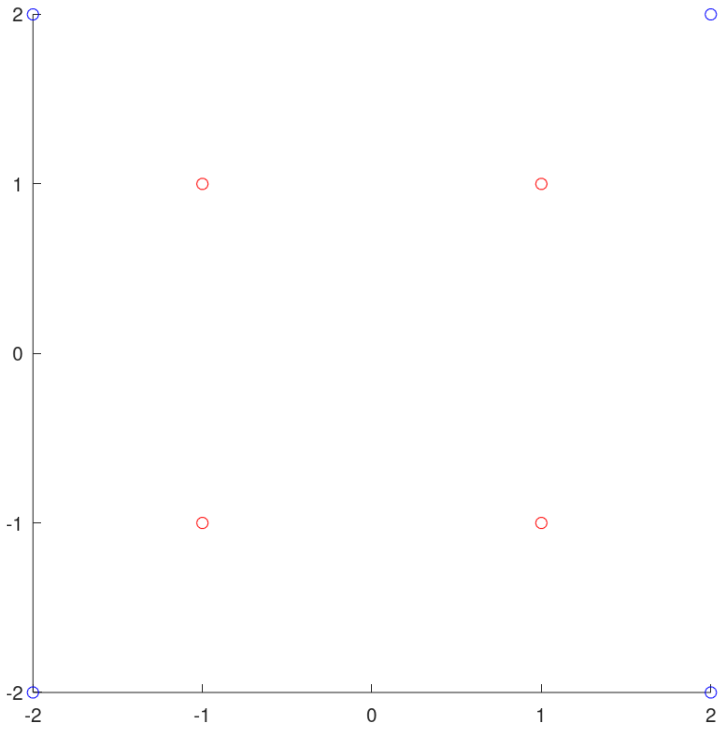


Τα 3 svm θα είναι το σημειο (1,0) για τα αρνητικα και (3,-1) , (3,1) για τα θετικα . Η γραμμη διαχωρισμου θα είναι η καθετη ευθεια που διερχεται από x = 2.



B)

Τα δείγματα στο Β ερώτημα δε είναι γραμμικά χωρισμένα οπότε θα πρέπει να προσθέσουμε ένα τρίτο χαρακτηριστικό για να τα κάνουμε γραμμικά διαχωρίσιμα.


2
1
0
-1
-2
-2
-1
0
1
2

Άσκηση 1

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

Η αναρρίχηση κόστους είναι η εξής:

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{\partial}{\partial \theta_j}\left(-\frac{1}{m}\sum_{i=1}^{n}\left[y^i \log(h_\theta(x^i)) + (1-y^i)\log(1-h_\theta(x^i))\right]\right) \overset{\text{παραδοχή } h_\theta(x) = \sigma(\theta^T x)}{=}$$

$$= -\frac{1}{m}\sum_{i=1}^{m}\left[y^i \frac{\frac{\partial}{\partial \theta_j}\sigma(\theta^T x^i)}{h_\theta(x^i)} + (1-y^i)\frac{\frac{\partial}{\partial \theta_j}(1-\sigma(\theta^T x^i))}{1-h_\theta(x^i)}\right] =$$

$$= -\frac{1}{m}\sum_{i=1}^{m}\left[y^i \frac{\sigma(\theta^T x^i)(1-\sigma(\theta^T x^i))\frac{\partial}{\partial \theta_j}\theta^T x^i}{h_\theta(x^i)} - (1-y^i)\frac{\sigma(\theta^T x^i)(1-\sigma(\theta^T x^i))\frac{\partial}{\partial \theta_j}\theta^T x^i}{1-h_\theta(x^i)}\right] =$$

$$\frac{\partial}{\partial \theta_j}\theta^T x^i = x_j^i$$

$$= -\frac{1}{m}\sum_{i=1}^{m}\left[y^i(1-h_\theta(x^i))x_j^i - (1-y^i)h_\theta(x^i)x_j^i\right] =$$

$$= -\frac{1}{m}\sum_{i=1}^{m}\left[y^i - y^i h_\theta(x^i) - h_\theta(x^i) + y^i h_\theta(x^i)\right]x_j^i =$$

$$= -\frac{1}{m}\sum_{i=1}^{m}\left[y^i - h_\theta(x^i)\right]x_j^i = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x^i) - y^i\right)x_j^i$$

Άσκηση 2

Θυμίζουμε πρώτα με τον κύριο διαφορά

$$\frac{\partial}{\partial \theta}\left(\frac{1}{2m}\sum_{j=1}^{n}\theta_j^2\right) = \frac{2}{2m}\sum_{j=1}^{n}\theta_j = \frac{1}{m}\theta_j$$