

# ΑΝΑΓΝΩΡΙΣΗ ΠΡΟΤΥΠΩΝ

## PROJECT

Βασιλόπουλος Βασίλειος 2024

Ζιόγας Δημήτριος 2214

### ΜΕΡΟΣ Α: Γέννηση Δεδομένων

Έστω 2 κλάσεις,  $\omega_1$  και  $\omega_2$ , με διανύσματα χαρακτηριστικών σε δύο διαστάσεις ( $x \in R^2$ ), και έστω ότι έχουμε διαθέσιμα 500 συνολικά δείγματα,  $N_1 = 400$  από την κλάση  $\omega_1$  και  $N_2 = 100$  από την κλάση  $\omega_2$ . Τα δείγματα αυτά είναι ομοιόμορφα κατανεμημένα σε παραλληλόγραμμα περιοχές του  $R^2$ , συγκεκριμένα στο παραλληλόγραμμο εντός διαστημάτων  $[2, 8] \times [1, 2]$  για την  $\omega_1$  και εντός διαστημάτων  $[6, 8] \times [2.5, 5.5]$  για την  $\omega_2$  (δηλ., για παράδειγμα, η κλάση  $\omega_1$  έχει σημεία μόνο εντός του παραλληλογράμμου με κάτω αριστερή γωνία το σημείο  $[2, 1]$  και πάνω δεξιά γωνία το σημείο  $[8, 2]$ ). Χρησιμοποιώντας γεννήτρια τυχαίων αριθμών δημιουργήστε τα ζητούμενα 500 δείγματα, και σχεδιάστε τα στο δυσδιάστατο χώρο (χρησιμοποιείστε διαφορετικές « ετικέτες » (labels) και χρώματα στο διάγραμμα για κάθε κλάση)

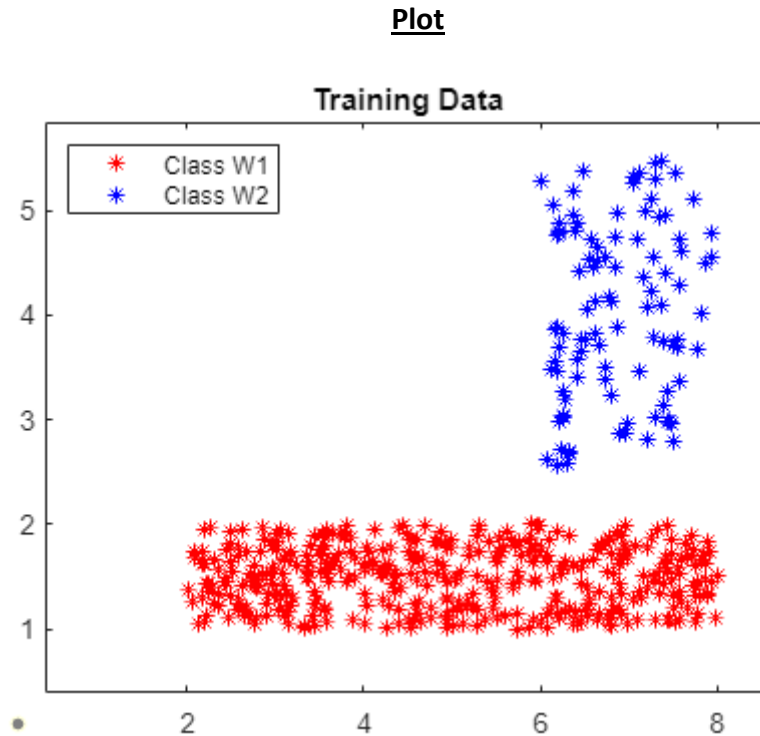
### Απάντηση

#### Κώδικας

```
rng('default') %% seed, so we get the same results every time we run the
simulation
W1x = 2 + (8-2)*rand(1,400); %% o typos einai xmin + (xmax -xmin)*rand(1,size)
%,giati to rand epistrefei timh sto [0,1]
W1y = 1 + (2-1)*rand(1,400);
W1 = [W1x ; W1y]; %%first row consists of the x variable, second row consists of
the y variable
W2x = 6 + (8-6)*rand(1,100);
W2y = 2.5 + (5.5-2.5)*rand(1,100);
W2 = [W2x ; W2y]; %% same as above

plot(W1x,W1y, 'r*');
hold on
plot(W2x,W2y, 'b*');
```

```
hold on
axis([0 9 0 6]);
legend({'Class W1', 'Class W2'}, 'Location', 'northwest')
title('Training Data');
```



Ουσιαστικά σε αυτό το μέρος σχεδιάζουμε τα σημεία , με βάση την περιγραφή που δίνεται στην εκφώνηση, οπότε έχουμε τυχαία ομοιόμορφα κατανομημένα δεδομένα. Επομένως, τα δεδομένα μας είναι έτοιμα για επεξεργασία στα επόμενα ερωτήματα. Επίσης, μία παρατήρηση είναι ότι αντί να χρησιμοποιηθούν τα variables  $x_1$  και  $x_2$  , χρησιμοποιήσαμε  $x$  και  $y$  ώστε να μην μπερδευτούμε στα ονόματα των μεταβλητών.

## ΜΕΡΟΣ Β: Bayesian Ταξινόμηση στον 2-Δ Χώρο

### B.1

Από τα δεδομένα του συνόλου εκπαίδευσης που ανήκουν στην κάθε κλάση, εκτιμήστε τις μέσες τιμές και μητρώα συνδιασποράς των δύο υπό συνθήκη συναρτήσεων πυκνότητας πιθανότητας, χρησιμοποιώντας εκτίμηση μέγιστης πιθανοφάνειας (maximum likelihood estimation), θεωρώντας τα ότι ακολουθούν δυσδυάστατες κανονικές κατανομές (multivariate Gaussian distributions).

## Απάντηση

### Κώδικας

```
%% Meros B1 %%
Ew1 = [sum(W1x)/length(W1x) ; sum(W1y)/length(W1y)]; %%mean value of class w1
Ew2 = [sum(W2x)/length(W2x) ; sum(W2y)/length(W2y)]; %%mean value of class w2

Sw1 = [sum((W1x - Ew1(1))*(W1x - Ew1(1)))/length(W1x) , sum((W1x - Ew1(1))*(W1y -
Ew1(2)))/length(W1x)
      ; sum((W1x - Ew1(1))*(W1y - Ew1(2)))/length(W1x) , sum((W1y - Ew1(2))*(W1y -
Ew1(2)))/length(W1x) ] %%cov matrix of class1

Sw2 = [sum((W2x - Ew2(1))*(W2x - Ew2(1)))/length(W2x) , sum((W2x - Ew2(1))*(W2y -
Ew2(2)))/length(W2x)
      ; sum((W2x - Ew2(1))*(W2y - Ew2(2)))/length(W2x) , sum((W2y - Ew2(2))*(W2y -
Ew2(2)))/length(W2x) ] %%cov matrix of class2

fprintf('Ew1: [%d \n      %d]\n', [Ew1(1);Ew1(2)])
fprintf('Ew2: [%d \n      %d]\n', [Ew2(1);Ew2(2)])
fprintf('Sw1: [%d , %d\n      %d , %d]\n', [Sw1(1,1) Sw1(1,2); Sw1(2,1)
Sw1(2,2)])
fprintf('Sw2: [%d , %d\n      %d , %d]\n', [Sw2(1,1) Sw2(1,2); Sw2(2,1)
Sw2(2,2)])
```

```
Ew1: [4.967300e+00
      1.491769e+00]
Ew2: [6.861904e+00
      4.036298e+00]
Sw1: [2.980883e+00 , -1.338331e-02
      -1.338331e-02 , 7.962752e-02]
Sw2: [2.884929e-01 , 7.636442e-02
      7.636442e-02 , 6.800259e-01]
```

Στο ερώτημα αυτό , μας έχει ζητηθεί η μέση τιμή και ο πίνακας συνδιασποράς της κάθε κλάσης , οπότε απλά τα βρίσκουμε , ώστε να τα χρησιμοποιήσουμε στην υπόλοιπη άσκηση Β.

## B.2

Ταξινομήστε τα δεδομένα του συνόλου εκπαίδευσης στις δύο κλάσεις με βάση τον ταξινομητή Ευκλείδειας απόστασης, χρησιμοποιώντας τις εκτιμήσεις των μέσων τιμών από το Βήμα Β.1. Στη συνέχεια, υπολογίστε το λάθος ταξινόμησης (%), συγκρίνοντας τις αποφάσεις του ταξινομητή με τις ετικέτες των δεδομένων (από το μέρος Α). Τέλος, σχεδιάστε τα δείγματα που έχουν ταξινομηθεί λανθασμένα στο διάγραμμα του μέρους Α, αν φυσικά υπάρχουν τέτοια.

## Απάντηση

### Κώδικας

```
%% Meros B2 %%
Class1 = 0;
missClass2 = 0; %% elements that are in Class1 but should be classified in Class 2
Class2 = 0;
missClass1 = 0; %% elements that are classified in Class2 but should be in Class 1
missMat1 = []; %% matrix containing elements are now classified into Class 1 but
belonged to Class 2
missMat2 = []; %% matrix containing elements are now classified into Class 2 but
belonged to Class 1
for i=1:400
    if norm(W1(:,i) - Ew1) < norm(W1(:,i) - Ew2)
        Class1 = Class1 + 1;
    else
        missClass2 = missClass2 + 1;
        missMat2 = [missMat2 , W1(:,i)]; %% appending the missclassified elements
matrix
    end
end

fprintf('The number of elements in Class1 is %d and the number of elements that
should be in Class2 is %d , according to Euclidean distance\n', Class1, missClass2)

Class2 = 0;
missClass1 = 0;

for i=1:100
    if norm(W2(:,i) - Ew1) < norm(W2(:,i) - Ew2)
        missClass1 = missClass1 + 1;
        missMat1 = [missMat1 , W2(:,i)];
    else
        Class2 = Class2 + 1;
    end
end

fprintf('The number of elements in Class2 is %d and the number of elements that
should be in Class1 is %d, according to Euclidean distance\n', Class2, missClass1)
```

```
fprintf('The classification error is %d, according to Euclidean distance \n\n',
(missClass2+missClass1)/(length(W1)+length(W2))) %% error is No of missclassified
divided by the total No of samples
```

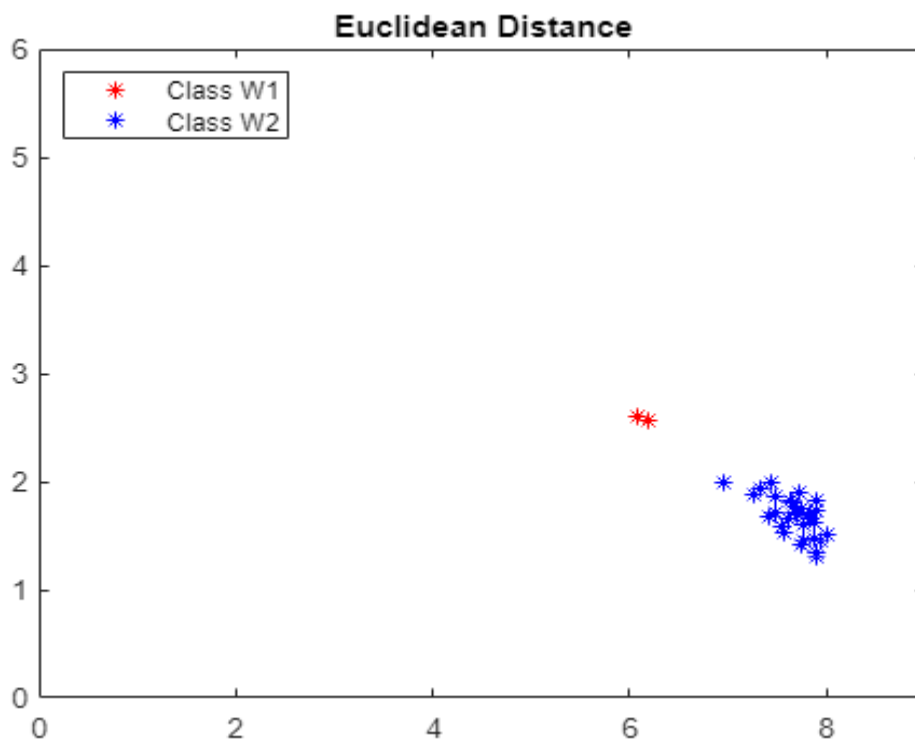
```
hold off;
```

```
plot(missMat1(1,:),missMat1(2,:),'ro');
hold on
plot(missMat2(1,:),missMat2(2,:),'bo');
hold on
axis([0 9 0 6]);
legend({'Class W1', 'Class W2'}, 'Location', 'northwest')
title('Euclidean Distance');
```

Ο κώδικάς μας εκτυπώνει τα εξής:

```
The number of elements in Class1 is 370 and the number of elements that should be in Class2 is 30 , according to Euclidean distance
The number of elements in Class2 is 98 and the number of elements that should be in Class1 is 2, according to Euclidean distance
The classification error is 6.400000e-02, according to Euclidean distance
```

Άρα, έχουμε ένα classification error της τάξεως του 6.4% . Επίσης, το διάγραμμα το οποίο εμφανίζει ο κώδικας είναι :



Όπου κάνει classify τα μπλε ως κλάση w2 και τα κόκκινα ως κλάση w1.

Ο κώδικας σε αυτό το ερώτημα τρέχει επαναλήψεις , όπου κοιτάει για κάθε σημείο εάν βρίσκεται πιο κοντά στο E της κλάσης w1 ή στο E της κλάσης w2 και αναλόγως το διατηρεί στην κλάση που βρισκόταν ή του αλλάζει την κλάση.

### B.3

Επαναλάβετε το Βήμα B.2, αλλά αυτήν τη φορά χρησιμοποιώντας την απόσταση Mahalanobis , με κοινό πίνακα συνδιασποράς των δύο κλάσεων που προκύπτει ως ο σταθμισμένος μέσος των μητρώων συνδιασποράς των δύο κλάσεων (υπολογισμένων στο Βήμα B.1).

### Απάντηση

#### Κώδικας

```
%% Meros B3 %%

S = (Sw1 + Sw2)/2; %% mean of the 2 covariance matrices

Class1 = 0;
missClass2 = 0;
Class2 = 0;
missClass1 = 0;
missMat1 = [];
missMat2 = [];

for i=1:400
    if sqrt((W1(:,i)-Ew1)' * inv(S) * (W1(:,i)-Ew1)) < sqrt((W1(:,i)-Ew2)' * inv(S)
* (W1(:,i)-Ew2))
        %% if Mahalanobis distance of W1 from Ew1 less than Mahalanobis distance of
W1 from Ew2
        Class1 = Class1 + 1;
    else
        missClass2 = missClass2 + 1;
        missMat2 = [missMat2 , W1(:,i)]; %% appending the misclassified elements
    end
end

fprintf('The number of elements in Class1 is %d and the number of elements that
should be in Class2 is %d , according to Mahalanobis distance\n', Class1,
missClass2)
```

```

Class2 = 0;
missClass1 = 0;

for i=1:100
    if sqrt((W2(:,i)-Ew1)' * inv(S) * (W2(:,i)-Ew1)) < sqrt((W2(:,i)-Ew2)' *
        inv(S) * (W2(:,i)-Ew2))
%% if Mahalanobis distance of W2 from Ew1 less than Mahalanobis distance of W2 from
Ew2
        missClass1 = missClass1 + 1;
        missMat1 = [missMat1 , W2(:,i)];
    else
        Class2 = Class2 + 1;
    end
end

fprintf('The number of elements in Class2 is %d and the number of elements that
should be in Class1 is %d , according to Mahalanobis distance\n', Class2,
missClass1)
fprintf('The classification error is %d , according to Mahalanobis distance \n\n',
(missClass1+missClass2)/(length(W2)+length(W1))) %% error is No of missclassified
% divided by the total No of samples

hold off;
plot(missMat1(1,:),missMat1(2,:), 'r*');
hold on
axis([0 9 0 6]);
legend({'Class W1'}, 'Location', 'northwest')
title('Mahalanobis Distance');

```

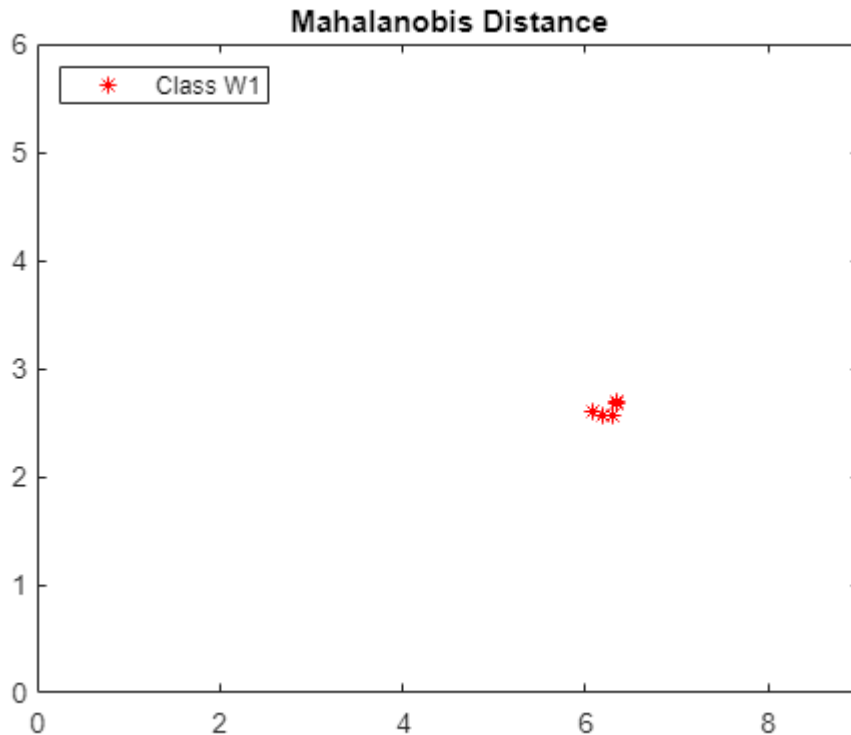
Ο κώδικάς μας εκτυπώνει τα εξής:

```

The number of elements in Class1 is 400 and the number of elements that should be in Class2 is 0 , according to Mahalanobis distance
The number of elements in Class2 is 95 and the number of elements that should be in Class1 is 5 , according to Mahalanobis distance
The classification error is 1.000000e-02 , according to Mahalanobis distance

```

Άρα, έχουμε ένα classification error της τάξεως του 1% . Επίσης, το διάγραμμα το οποίο εμφανίζει ο κώδικας είναι :



Ουσιαστικά , η διαδικασία που ακολουθούμε, είναι ίδια με την Ευκλείδεια απόσταση. Το μόνο που αλλάζει είναι η συνθήκη του loop , όπου πλέον έχουμε τη Mahalanobis distance και τα αποτελέσματα. Εύκολα παρατηρούμε , ότι ενώ η Ευκλείδεια έκανε classify πολλά στοιχεία στην κλάση w2 , εδώ η Mahalanobis δεν κάνει κανένα και τα μόνα στοιχεία που κάνει classify είναι στην κλάση w1. Αυτό δικαιολογείται από την ύπαρξη του covariance matrix στη συνθήκη των επαναλήψεων.

## ΜΕΡΟΣ Δ: Γραμμική Ταξινόμηση με Διάφορες Συναρτήσεις Κόστους

### Δ.1

Βρείτε τον γραμμικό ταξινομητή (ευθεία) που ελαχιστοποιεί το κριτήριο ελαχίστων τετραγώνων για τα δεδομένα εκπαίδευσης του μέρους Α, όπου η κλάση w1 αντιστοιχεί στην ετικέτα  $y = +1$  και η κλάση w2 στην ετικέτα  $y = -1$  (Κεφ. 3.4.3, εξ. (3.45) του βιβλίου). Ποιο είναι το σφάλμα ταξινόμησης και το σφάλμα τετραγώνων (εξ. (3.42) του βιβλίου) του γραμμικού ταξινομητή που προκύπτει; Σχεδιάστε επίσης την ευθεία στον χώρο, μαζί με τα δεδομένα εκπαίδευσης.

Απάντηση

Κώδικας



```

%% Meros D1 %%
%% We need to define X and y
X = [w1 , w2]' ;
X(:,3) = ones(500,1); %appending X matrix with ones

y = [ones(400,1) ; -ones(100,1)] % y has the value 1 for class w1 , which has 400
samples and -1 for w2 , which has 100 samples

w = inv((X'*X)) * X'*y

x1 = linspace(0,6,2000);
x2 = linspace(0,9,2000);

line = w(1)*x1 + w(2)*x2 + w(3);

hold on
plot(line,'g')

%% Lathos taksinomisis 0 giati opws vlepoume k sto plot, ola diaxwrizontai opws
prepei

sqError = sum((y - X*w).^2); % den antistrefoume ton X , giati einai hdh sth morfhi
pou prepei

fprintf('The sum of square errors is %f', sqError)

```

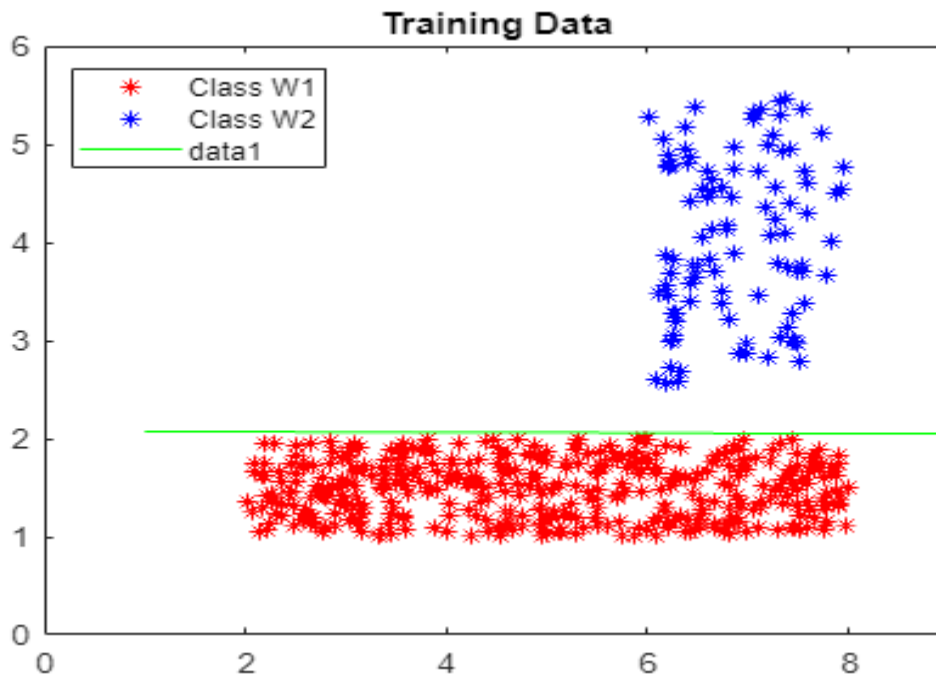
Ο κώδικάς μας εκτυπώνει τα εξής:

$w =$

-0.037548  
-0.63539  
2.0719

The sum of square errors is 49.935745

Άρα η ευθεία μας είναι της μορφής  $y = -0.037548 \cdot x_1 - 0.63539 + 2.0719$ . Παρακάτω την παρουσιάζουμε και σε διάγραμμα:



Από το plot φαίνεται πως η πράσινη αυτή γραμμή διαχωρίζει τα στοιχεία κάθε κλάσης. Επίσης, απαντώντας και σε ερώτημα της άσκησης, το σφάλμα τετραγώνων όπως είδαμε είναι 49.935745.

## Δ.2

Βρείτε τον γραμμικό ταξινομητή (ευθεία) με βάση τον αλγόριθμο perceptron, χρησιμοποιώντας κάποια αρχική ευθεία που δεν ταξινομεί σωστά όλα τα δεδομένα εκπαίδευσης (π.χ. της κλάσης  $\omega_2$ ), υλοποιώντας τον με χρήση κατάλληλης παραμέτρου ρυθμού εκμάθησης,  $\rho$  (Εξ. (3.9) του βιβλίου), ώστε ο αλγόριθμος να συγκλίνει σε σχετικά μικρό αριθμό βημάτων. Ποιο είναι το σφάλμα ταξινόμησης του γραμμικού ταξινομητή που προκύπτει; Σχεδιάστε επίσης την ευθεία στον χώρο, μαζί με τα δεδομένα εκπαίδευσης. (Θεωρήστε τις ετικέτες του συνόλου εκπαίδευσης όπως στο Βήμα Δ.1.)

## Απάντηση

### Κώδικας

```
% Meros D2 %%
rng('default') %% seed, so we get the same results every time we run the simulation
W1x = 2 + (8-2)*rand(1,400); %% ο τυπος είναι xmin + (xmax -xmin)*rand(1,size)
%,γιατι το rand επιστρεφει τιμη στο [0,1]
W1y = 1 + (2-1)*rand(1,400);
W1 = [W1x ; W1y]; %%first row consists of the x variable, second row consists of
the y variable
```

```

W2x = 6 + (8-6)*rand(1,100);
W2y = 2.5 + (5.5-2.5)*rand(1,100);
W2 = [W2x ; W2y]; %% same as above

```

```

plot(W1x,W1y, 'r*');
hold on
plot(W2x,W2y, 'b*');
hold on
axis([0 9 0 6]);
legend({'Class W1', 'Class W2'}, 'Location', 'northwest')
title('Training Data');

```

```

%% Meros D2 %%

```

```

X = [W1 , W2]' ;
X(:,3) = ones(500,1); %appending X matrix with ones
%%Estw h eftheia y=4 , dhladh x2 = 4 , ara tha exoume w = [0 4 0]
w = [-2 ; 4 ; 0];
epochs = 0; % arxikopoioume ta epochs
allClassified = false(1); % logikh metavliti gia na doume an xreiazetai na
ksanakanoume loop
r = 3.11;

```

```

while allClassified == false(1)
wrongClass = []; % pinakas pou tha periexei ta lathos classified stoixeia tou kathe
epoch
count = 0 ; % metritis, o opoios einai to synolo twn stoixeiwn pou den einai swsta
classified se kathe epoch
epochs = epochs + 1;
for i= 1:400
if (X(i,:)* w <= 0)
wrongClass = [wrongClass, r*(X(i,:)')];
count = count + 1;

end
end

for i= 401:500
if (X(i,:)*w > 0)
wrongClass = [wrongClass, r*(-X(i,:)')];
count = count + 1;
end
end

w = w + sum(wrongClass)';

if count == 0
allClassified = true(1);
end
end

```

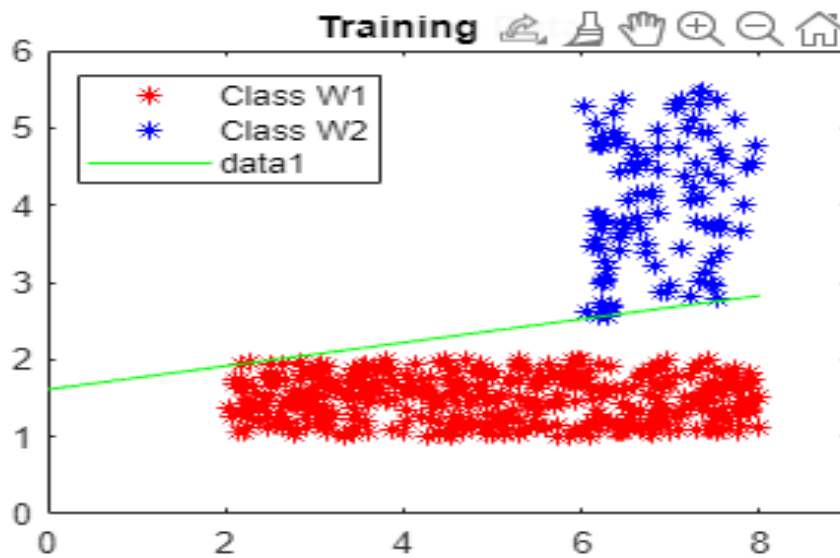
```

x1 = linspace(0,8,1000);

```

```
x2 = -(w(1)*x1 + w(3))/w(2);
```

```
hold on  
plot(x1,x2,'g')
```



Το perceptron που περιγράφει ο κώδικας παρουσιάζει την εξής γραμμή για όλες τις τιμές του  $r$ . Τα epochs που παίρνει για να συγκλίνει είναι 140 και φυσικά δεν έχουμε σφάλμα, αλλιώς ο αλγόριθμος θα έτρεχε μέχρι να τα διορθώσει όλα ή για πάντα.