

# Επεξεργασία Εικόνας & Βίντεο



09. Τεχνικές Μηχανικής  
Μάθησης στην Εικόνας

Εισηγητής: Νικόλαος Γιαννακέας

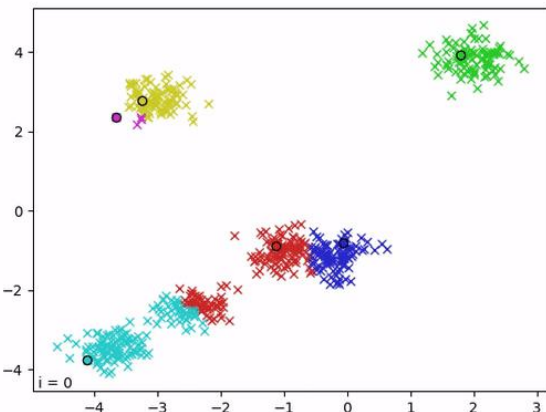
*Επίκουρος Καθηγητής, Σημάτων & Συστημάτων*





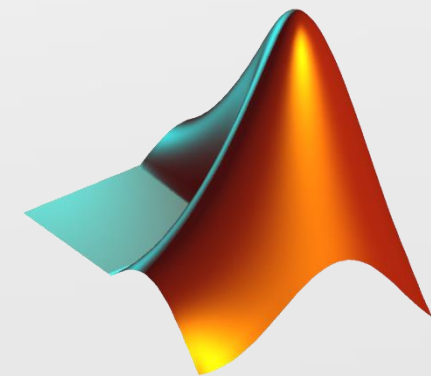
# Εισαγωγή

Μια από τις βασικές κατηγορίες **μεθόδων Μηχανικής Μάθησης** (Machine Learning) είναι η **ομαδοποίηση** ή **συσταδοποίηση**. **Σκοπός** των τεχνικών αυτών είναι η **ομαδοποίηση** αντικειμένων (δειγμάτων) σύμφωνα με **συγκεκριμένα χαρακτηριστικά**. Ο **αριθμός** των ομάδων στις οποίες δύναται να **διαχωριστούν** τα αντικείμενα **προκαθορίζεται** ανάλογα με το **πρόβλημα**. Ο περισσότερο γνωστός **αλγόριθμος ομαδοποίησης** είναι ο αλγόριθμος **K-Μέσων**. Στο παρακάτω σχήμα φαίνονται τα βήματα του αλγορίθμου K.



## Βασικός αλγόριθμος

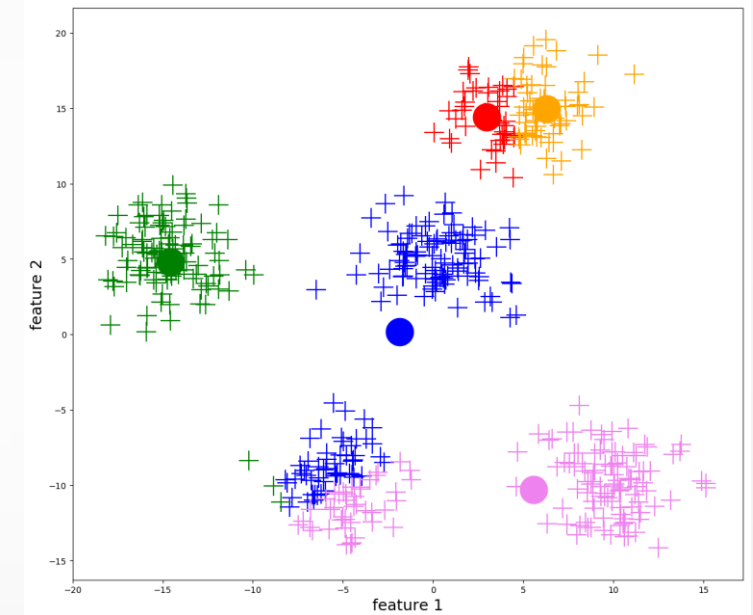
- 1: Επιλογή  $K$  σημείων ως τα αρχικά κεντρικά σημεία
- 2: **Repeat**
- 3: Ανάθεση όλων των αρχικών σημείων στο *κοντινότερο* τους από τα  $K$  κεντρικά σημεία
- 4: Επανα-υπολογισμός του *κεντρικού σημείου* κάθε συστάδας
- 5: **Until** τα κεντρικά σημεία να μην αλλάζουν



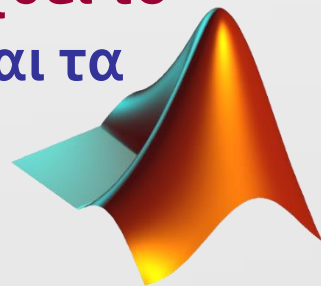


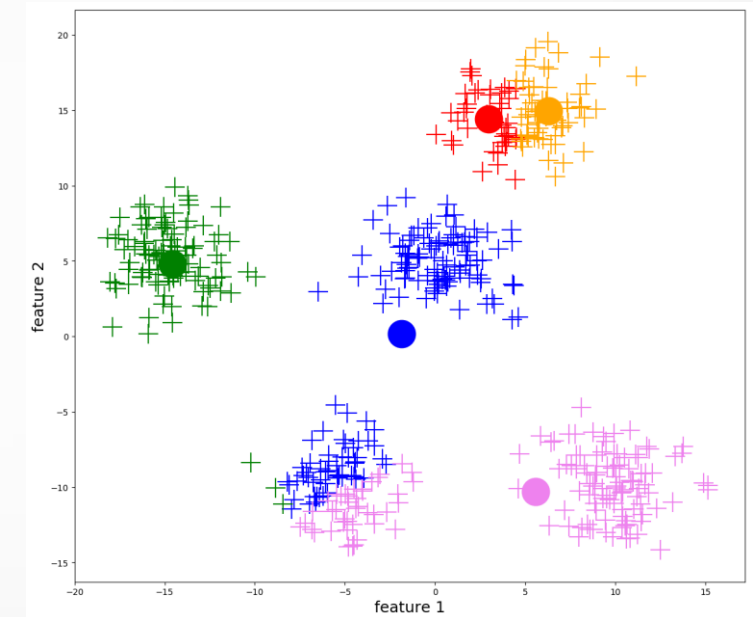
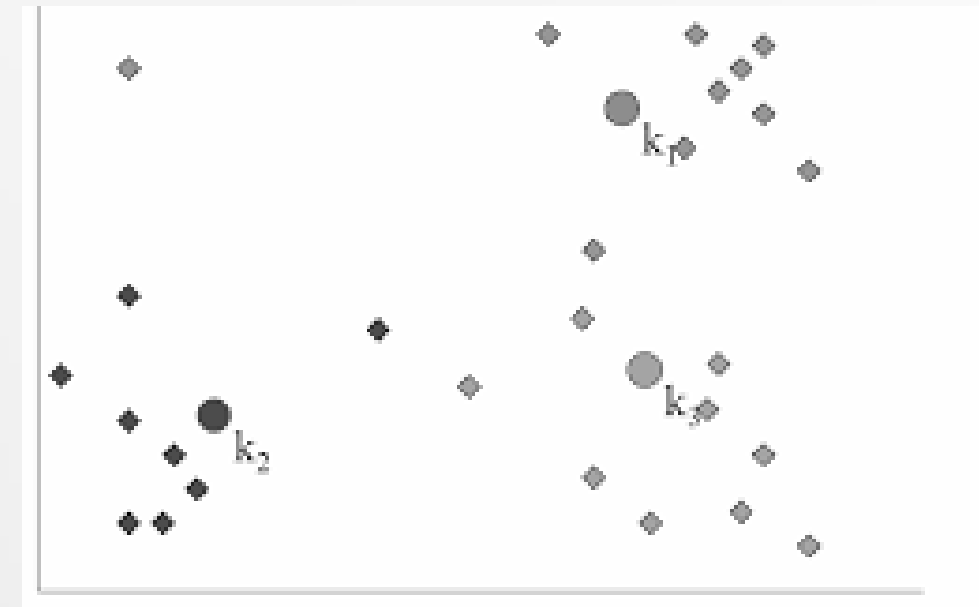
## Βασικός αλγόριθμος

- 1: Επιλογή  $K$  σημείων ως τα αρχικά κεντρικά σημεία
- 2: **Repeat**
- 3:     Ανάθεση όλων των αρχικών σημείων στο *κοντινότερο* τους από τα  $K$  κεντρικά σημεία
- 4:     Επανα-υπολογισμός του *κεντρικού σημείου* κάθε συστάδας
- 5: **Until** τα κεντρικά σημεία να μην αλλάζουν

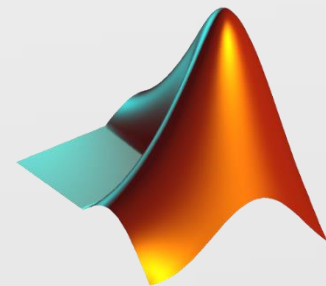


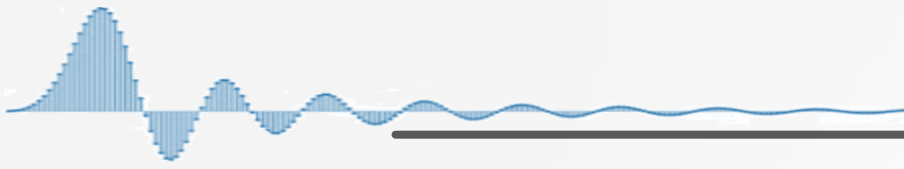
Ο αλγόριθμος αρχικά **επιλέγει τυχαία τόσα δείγματα όσοι είναι ο αριθμός των ομάδων** και στην συνέχεια **σε κάθε επανάληψη** του αλγορίθμου **εντάσσει ένα νέο δείγμα σε μια από τις ομάδες σύμφωνα με κοντινότερή απόσταση από τα κεντροειδή (centroids)**. **Αφού ενταχθεί το νέο δείγμα όλα τα κεντροειδή επαναπροσδιορίζονται ώστε να είναι τα κέντρα βάρους των μελών της ομάδας**





Στο τελευταίο εργαστήριο επιχειρείται ο **διαχωρισμός** των περιοχών της εικόνας **shapes.png**, σύμφωνα με **χαρακτηριστικά χρώματος**, και σύμφωνα με χαρακτηριστικά **σχήματος**.

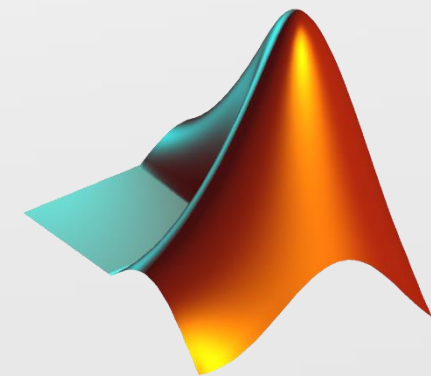




# ΠΕΙΡΑΜΑΤΙΚΗ ΔΙΑΔΙΚΑΣΙΑ

## Άσκηση 1

Δημιουργείστε ένα νέο script, το οποίο να εισάγει την εικόνα του `'shapes.png'` από τον φάκελο `datasets` στο workspace του λογισμικού MATLAB και να **εμφανίζει** την εικόνα σε νέο **παράθυρο**. Η συγκεκριμένη εικόνα έχει απολύτως **μαύρο υπόβαθρο** και επομένως είναι **εύκολος** ο εντοπισμός **όλων των σχημάτων** και ο **διαχωρισμός** τους από το **υπόβαθρο**. Χρησιμοποιήστε τον κώδικα της προηγούμενης Εργαστηριακής Άσκησης, με σκοπό να **εξαχθούν τα χαρακτηριστικά**.





# ΠΕΙΡΑΜΑΤΙΚΗ ΔΙΑΔΙΚΑΣΙΑ

## Άσκηση 1

Τοποθετείτε όλα τα χαρακτηριστικά των εντοπισμένων περιοχών σε ένα **πίνακα** με την ακόλουθη σειρά στηλών

**1η στήλη:** Εμβαδόν

**2η στήλη:** Εκκεντρότητα

**3η στήλη:** Περίμετρο

**4η στήλη:** Φωτεινότητα στο κόκκινο κανάλι

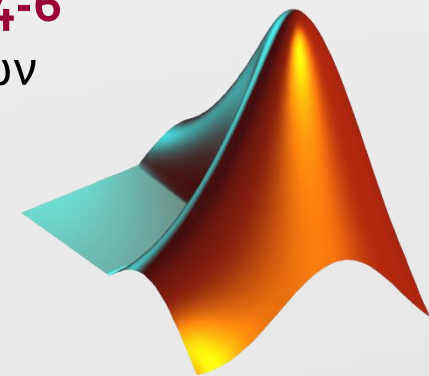
**5η στήλη:** Φωτεινότητα στο πράσινο κανάλι

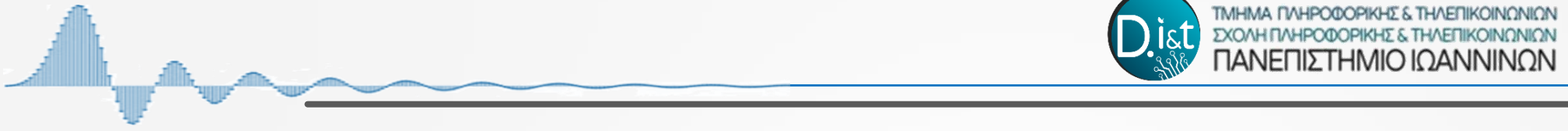
**6η στήλη:** Φωτεινότητα στο μπλε κανάλι

**7η στήλη:** Υπολογίστε το μέτρο της στρογγυλότητας (roundness) =  $4\pi(\text{εμβαδόν}) / (\text{Περίμετρος})^2$

**A)** Εκτελέστε τον αλγόριθμο **K-means** με **χαρακτηριστικά** μόνο τις **στήλες 4-6** των χαρακτηριστικών και **4 ομάδες** (όσες είναι τα διαφορετικά χρώματα των περιοχών). Εμφανίστε μια **binary** εικόνα για **κάθε χρώμα** αντικειμένων.

**B)** Εκτελέστε τον αλγόριθμο **K-means** με **μόνο** χαρακτηριστικό την **στρογγυλότητα**. Θέστε **K=3**. Εμφανίστε μια **binary** εικόνα **για κάθε ομάδα**.





# ΠΕΙΡΑΜΑΤΙΚΗ ΔΙΑΔΙΚΑΣΙΑ

## Άσκηση 1

### Υπόδειξη:

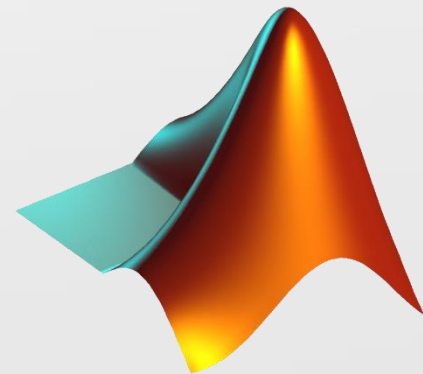
Για την εκτέλεση του **k-means** χρησιμοποιήστε την εντολή `[idx, C] = kmeans(features, K)`

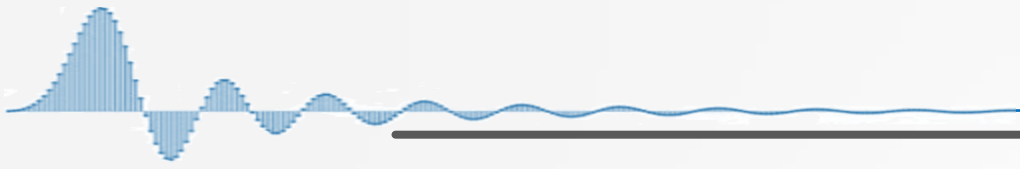
Η εντολή δέχεται σαν είσοδο

- α) τον **πίνακα** των **χαρακτηριστικών** όπου στην κάθε **γραμμή** έχει **διαφορετικό δείγμα** και στην κάθε **στήλη** **διαφορετικό χαρακτηριστικό**,
- β) τον **αριθμό** των **ομάδων K** σε **ακέραιο αριθμό**.

Η εντολή επιστρέφει

- α) την μεταβλητή **idx** το οποίο είναι ένα **διάνυσμα** με τον **δείκτη** της **ομάδας** στην οποία καταχωρείται το **κάθε δείγμα**, και
- β) την μεταβλητή **C** η οποία περιέχει τις **συντεταγμένες** των **τελικών κεντροειδών**.





# ΛΥΣΗ

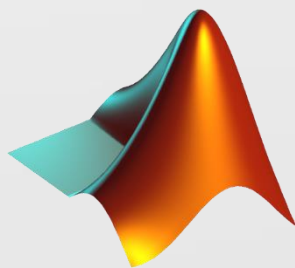
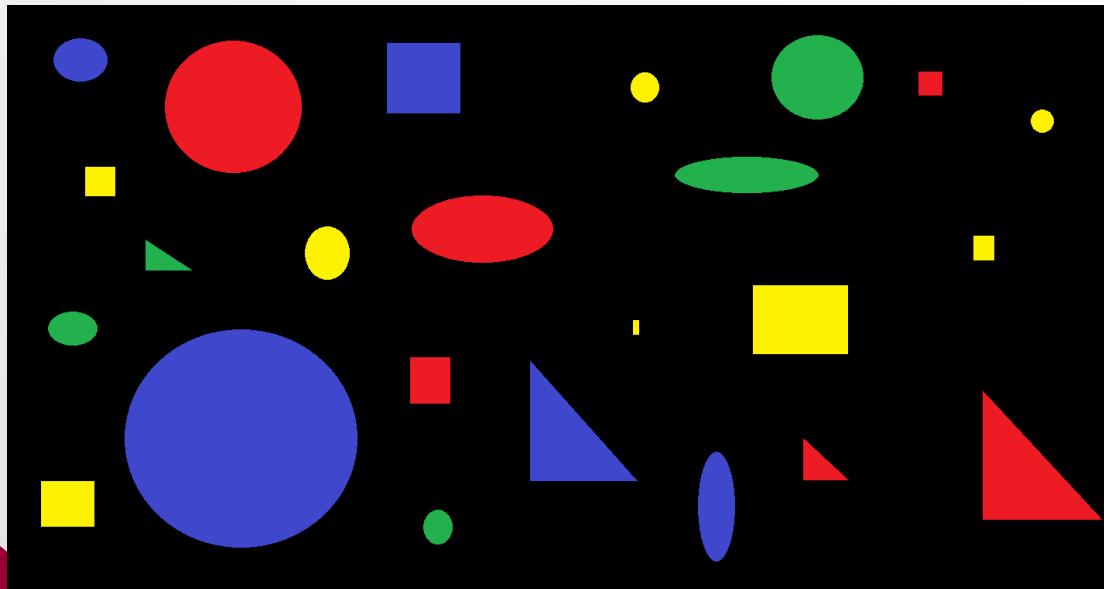
```
I = imread('shapes.png');  
figure, imshow(I);
```



Διάβασμα εικόνας



Εμφάνιση εικόνας







## ΛΥΣΗ

```
I = imread('shapes.png');
```

```
figure, imshow(I);
```

```
I_bin = (I(:,:,1) > 0) | (I(:,:,2) > 0) | (I(:,:,3) > 0);
```

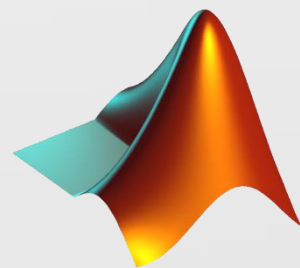
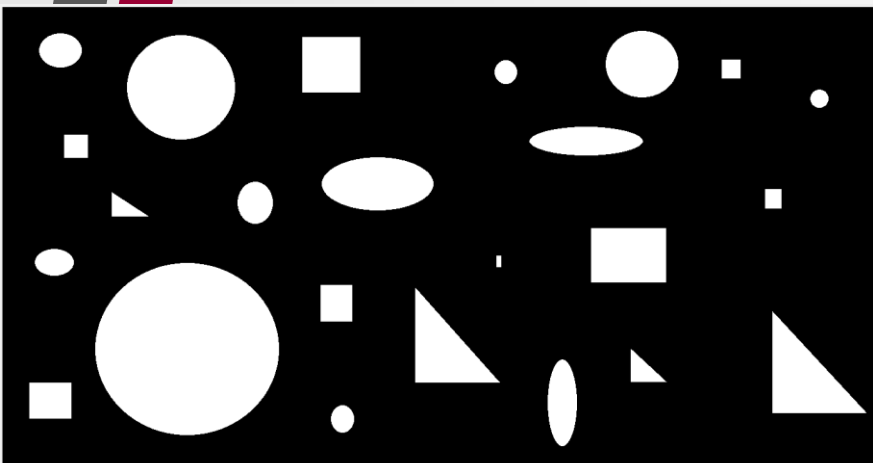
```
figure, imshow(I_bin)
```

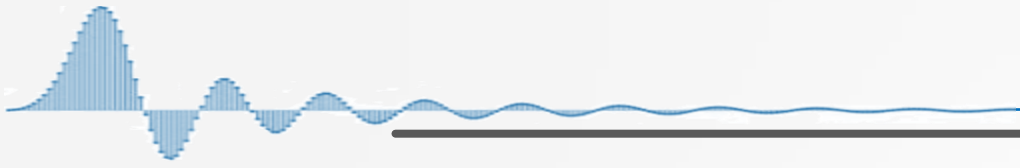
→ Διάβασμα εικόνας

→ Εμφάνιση εικόνας

→ Εφαρμόζω μια απλή χειροκίνητη μέθοδο κατάτμησης ώστε να βρω τα αντικείμενα

→ Εμφάνιση εικόνας





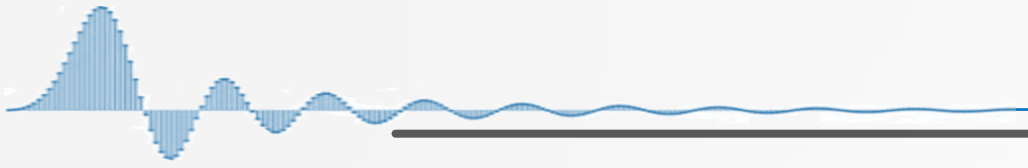
# ΛΥΣΗ

## `bwlabel`

Επισημάνετε τα συνδεδεμένα στοιχεία σε δυαδική εικόνα 2-D

`L = bwlabel(BW)`

Επιστρέφει τον πίνακα ετικετών `L` που περιέχει ετικέτες για τα 8-συνδεδεμένα αντικείμενα που βρίσκονται στο `BW`.



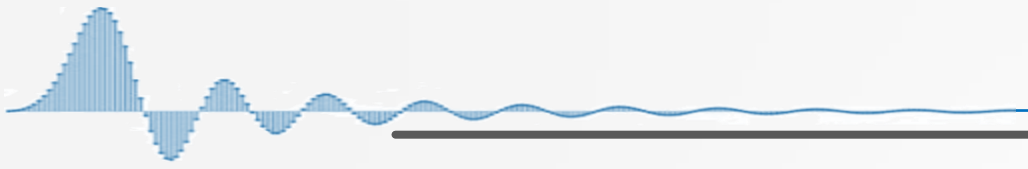
# ΛΥΣΗ

## regionprops

Μετρήστε τις ιδιότητες των περιοχών εικόνας

```
stats = regionprops(___, I, properties)
```

Επιστρέφει μετρήσεις για το σύνολο ιδιοτήτων που καθορίζονται από `properties` για κάθε περιοχή με ετικέτα στην εικόνα `I`. Η πρώτη είσοδος στο `regionprops` (`BW`, `CC` ή `L`) προσδιορίζει τις περιοχές σε `I`.



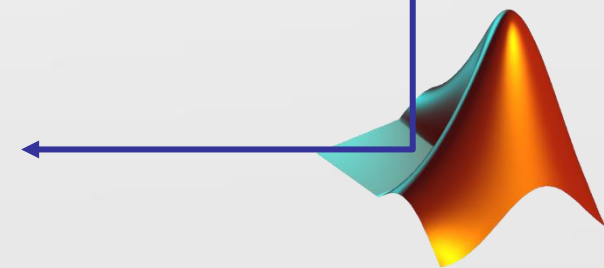
## ΛΥΣΗ

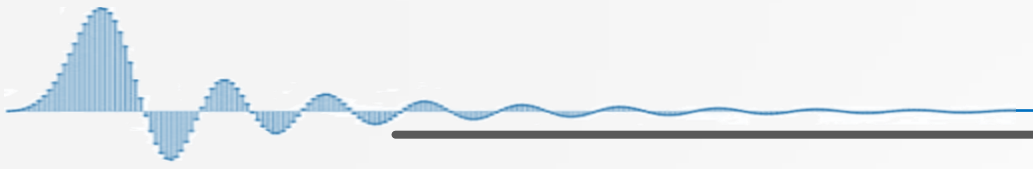
```
I = imread('shapes.png');  
figure, imshow(I);  
I_bin = (I(:,:,1) > 0) | (I(:,:,2) > 0) | (I(:,:,3) > 0);  
figure, imshow(I_bin)
```

```
I_labeled = bwlabel(I_bin);  
stats_red = regionprops(I_labeled, I(:,:,1), 'Area', 'Eccentricity', 'Perimeter', 'MeanIntensity',  
'PixelValues');  
stats_green = regionprops(I_labeled, I(:,:,2), 'MeanIntensity', 'PixelValues');  
stats_blue = regionprops(I_labeled, I(:,:,3), 'MeanIntensity', 'PixelValues');
```

→ Τοποθετούμε ετικέτα σε κάθε συνεκτικό αντικείμενο

Εξάγουμε το εμβαδό (Area), την περίμετρο (Perimeter), την εκκεντρότητα (Eccentricity), καθώς επίσης και της μέσες φωτεινότητες (MeanIntensity) σε κάθε κανάλι της εικόνας.



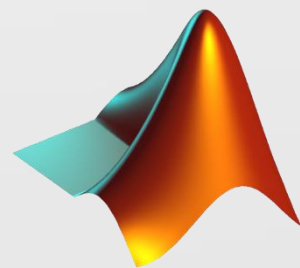


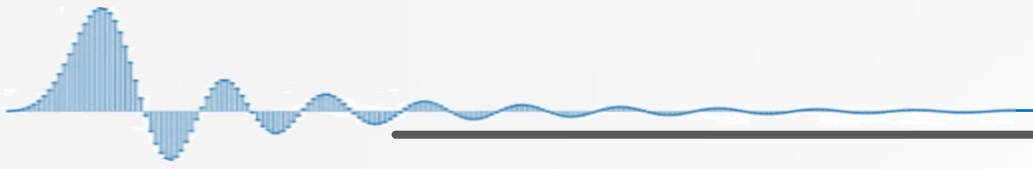
## ΛΥΣΗ

```
I = imread('shapes.png');  
figure, imshow(I);  
I_bin = (I(:,:,1) > 0) | (I(:,:,2) > 0) | (I(:,:,3) > 0);  
figure, imshow(I_bin)  
I_labeled = bwlabel(I_bin);  
stats_red = regionprops(I_labeled,I(:,:,1),'Area','Eccentricity','Perimeter', 'MeanIntensity', 'PixelValues');  
stats_green = regionprops(I_labeled,I(:,:,2),'MeanIntensity', 'PixelValues');  
stats_blue = regionprops(I_labeled,I(:,:,3),'MeanIntensity', 'PixelValues');  
  
I_max_area = (I_labeled == 5);  
features = zeros(length(stats_red),5);
```

Επιλέγουμε την περιοχή που θέλουμε

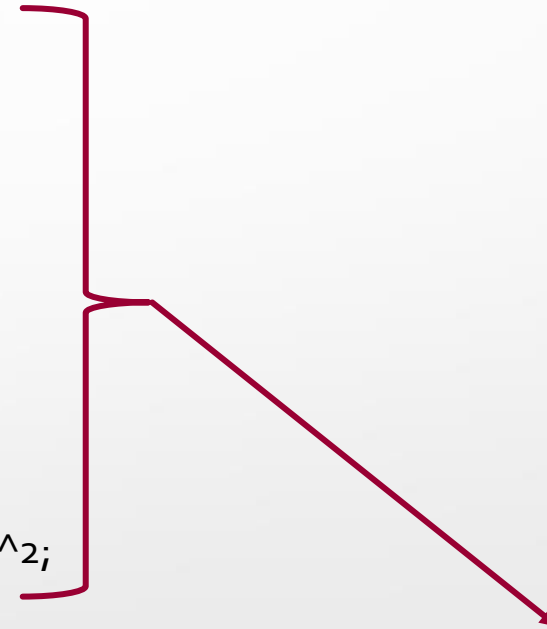
Δημιουργία μηδενικού πίνακα με μήκος τις γραμμές του πίνακα stats\_red



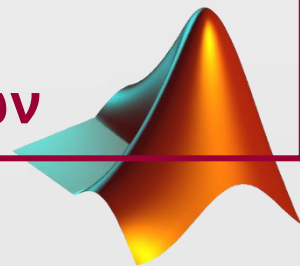


# ΛΥΣΗ

```
I = imread('shapes.png');  
figure, imshow(I);  
L_bin = (I(:,:,1) > 0) | (I(:,:,2) > 0) | (I(:,:,3) > 0);  
figure, imshow(L_bin)  
L_labeled = bwlabel(L_bin);  
stats_red = regionprops(L_labeled,I(:,:,1),'Area','Eccentricity','Perimeter', 'MeanIntensity', 'PixelValues');  
stats_green = regionprops(L_labeled,I(:,:,2),'MeanIntensity', 'PixelValues');  
stats_blue = regionprops(L_labeled,I(:,:,3),'MeanIntensity', 'PixelValues');  
L_max_area = (L_labeled == 5);  
features = zeros(length(stats_red),5);  
  
for i=1:length(stats_red)  
  
    features(i,1) = stats_red(i).Area;  
    features(i,2) = stats_red(i).Eccentricity;  
    features(i,3) = stats_red(i).Perimeter;  
    features(i,4) = stats_red(i).MeanIntensity;  
    features(i,5) = stats_green(i).MeanIntensity;  
    features(i,6) = stats_blue(i).MeanIntensity;  
    features(i,7) = (4*stats_red(i).Area*pi)/(stats_red(i).Perimeter)^2;  
  
end
```



Εξαγωγή χαρακτηριστικών





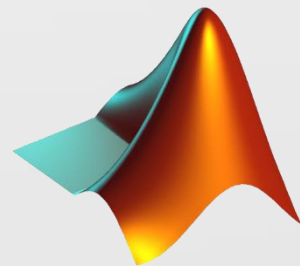
```
I = imread('shapes.png');  
figure, imshow(I);  
I_bin = (I(:,:,1) > 0) | (I(:,:,2) > 0) | (I(:,:,3) > 0);  
figure, imshow(I_bin)  
I_labeled = bwlabel(I_bin);  
stats_red =  
regionprops(I_labeled,I(:,:,1),'Area','Eccentricity','Perimeter',  
'MeanIntensity', 'PixelValues');  
stats_green = regionprops(I_labeled,I(:,:,2),'MeanIntensity',  
'PixelValues');  
stats_blue = regionprops(I_labeled,I(:,:,3),'MeanIntensity',  
'PixelValues');  
I_max_area = (I_labeled == 5);  
features = zeros(length(stats_red),5);  
for i=1:length(stats_red)  
    features(i,1) = stats_red(i).Area;  
    features(i,2) = stats_red(i).Eccentricity;  
    features(i,3) = stats_red(i).Perimeter;  
    features(i,4) = stats_red(i).MeanIntensity;  
    features(i,5) = stats_green(i).MeanIntensity;  
    features(i,6) = stats_blue(i).MeanIntensity;  
    features(i,7) = (4*stats_red(i).Area*pi)/(stats_red(i).Perimeter)^2;  
end
```

```
[idx,C] = kmeans(features(:,4:6),4,'emptyaction','singleton');
```

```
xrwm1 = zeros(size(I_bin,1),size(I_bin,2));  
xrwm2 = zeros(size(I_bin,1),size(I_bin,2));  
xrwm3 = zeros(size(I_bin,1),size(I_bin,2));  
xrwm4 = zeros(size(I_bin,1),size(I_bin,2));
```

Εκτέλεση του k-means

Αρχικοποίηση μηδενικών πινάκων





```

I = imread('shapes.png');
figure, imshow(I);
I_bin = (I(:,:,1) > 0) | (I(:,:,2) > 0) | (I(:,:,3) > 0);
figure, imshow(I_bin)
I_labeled = bwlabel(I_bin);
stats_red =
regionprops(I_labeled,I(:,:,1),'Area','Eccentricity','Perimeter',
'MeanIntensity', 'PixelValues');
stats_green = regionprops(I_labeled,I(:,:,2),'MeanIntensity',
'PixelValues');
stats_blue = regionprops(I_labeled,I(:,:,3),'MeanIntensity',
'PixelValues');
I_max_area = (I_labeled == 5);
features = zeros(length(stats_red),5);
for i=1:length(stats_red)
    features(i,1) = stats_red(i).Area;
    features(i,2) = stats_red(i).Eccentricity;
    features(i,3) = stats_red(i).Perimeter;
    features(i,4) = stats_red(i).MeanIntensity;
    features(i,5) = stats_green(i).MeanIntensity;
    features(i,6) = stats_blue(i).MeanIntensity;
    features(i,7) = (4*stats_red(i).Area*pi)/(stats_red(i).Perimeter)^2;

```

end

```

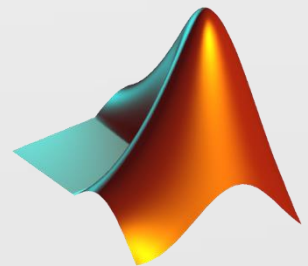
[idx,C] = kmeans(features(:,4:6),4,'emptyaction','singleton');
xrwma1 = zeros(size(I_bin,1),size(I_bin,2));
xrwma2 = zeros(size(I_bin,1),size(I_bin,2));
xrwma3 = zeros(size(I_bin,1),size(I_bin,2));
xrwma4 = zeros(size(I_bin,1),size(I_bin,2));

```

```

for i=1:length(idx)
    if idx(i) == 1
        xrwma1 = xrwma1 + (I_labeled == i);
    elseif idx(i) == 2
        xrwma2 = xrwma2 + (I_labeled == i);
    elseif idx(i) == 3
        xrwma3 = xrwma3 + (I_labeled == i);
    else
        xrwma4 = xrwma4 + (I_labeled == i);
    end
end

```







```

I = imread('shapes.png');
figure, imshow(I);
I_bin = (I(:,:,1) > 0) | (I(:,:,2) > 0) | (I(:,:,3) > 0);
figure, imshow(I_bin)
I_labeled = bwlabel(I_bin);
stats_red =
regionprops(I_labeled,I(:,:,1),'Area','Eccentricity','Perimeter',
'MeanIntensity', 'PixelValues');
stats_green = regionprops(I_labeled,I(:,:,2),'MeanIntensity',
'PixelValues');
stats_blue = regionprops(I_labeled,I(:,:,3),'MeanIntensity',
'PixelValues');
I_max_area = (I_labeled == 5);
features = zeros(length(stats_red),5);
for i=1:length(stats_red)
    features(i,1) = stats_red(i).Area;
    features(i,2) = stats_red(i).Eccentricity;
    features(i,3) = stats_red(i).Perimeter;
    features(i,4) = stats_red(i).MeanIntensity;
    features(i,5) = stats_green(i).MeanIntensity;
    features(i,6) = stats_blue(i).MeanIntensity;
    features(i,7) = (4*stats_red(i).Area*pi)/(stats_red(i).Perimeter)^2;

```

end

```
[idx,C] = kmeans(features(:,4:6),4,'emptyaction','singleton');
```

```

xrwma1 = zeros(size(I_bin,1),size(I_bin,2));
xrwma2 = zeros(size(I_bin,1),size(I_bin,2));
xrwma3 = zeros(size(I_bin,1),size(I_bin,2));
xrwma4 = zeros(size(I_bin,1),size(I_bin,2));

```

```

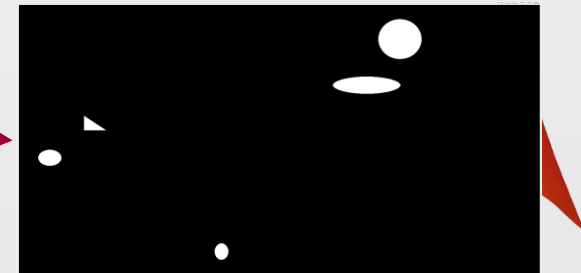
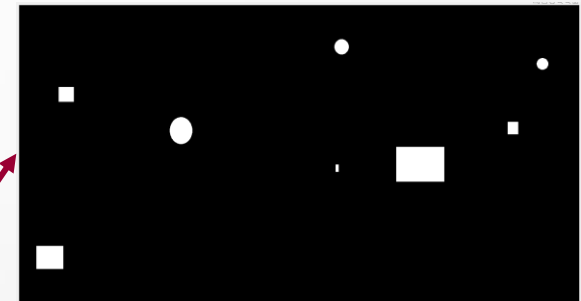
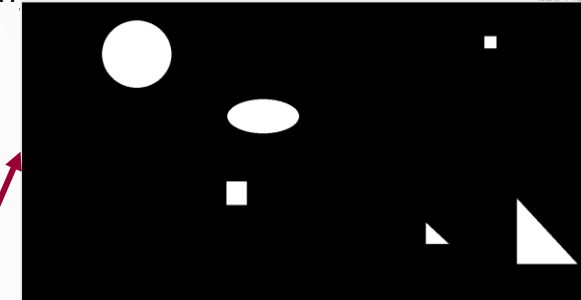
for i=1:length(idx)
    if idx(i) == 1
        xrwma1 = xrwma1 + (I_labeled == i);
    elseif idx(i) == 2
        xrwma2 = xrwma2 + (I_labeled == i);
    elseif idx(i) == 3
        xrwma3 = xrwma3 + (I_labeled == i);
    else
        xrwma4 = xrwma4 + (I_labeled == i);
    end
end

```

```

figure, imshow(xrwma1)
figure, imshow(xrwma2)
figure, imshow(xrwma3)
figure, imshow(xrwma4)

```





```

I = imread('shapes.png');
figure, imshow(I);
I_bin = (I(:,:,1) > 0) | (I(:,:,2) > 0) | (I(:,:,3) > 0);
figure, imshow(I_bin)
I_labeled = bwlabel(I_bin);
stats_red =
regionprops(I_labeled,I(:,:,1),'Area','Eccentricity','Perimeter',
'MeanIntensity', 'PixelValues');
stats_green = regionprops(I_labeled,I(:,:,2),'MeanIntensity',
'PixelValues');
stats_blue = regionprops(I_labeled,I(:,:,3),'MeanIntensity',
'PixelValues');
I_max_area = (I_labeled == 5);
features = zeros(length(stats_red),5);
for i=1:length(stats_red)
    features(i,1) = stats_red(i).Area;
    features(i,2) = stats_red(i).Eccentricity;
    features(i,3) = stats_red(i).Perimeter;
    features(i,4) = stats_red(i).MeanIntensity;
    features(i,5) = stats_green(i).MeanIntensity;
    features(i,6) = stats_blue(i).MeanIntensity;
    features(i,7) = (4*stats_red(i).Area*pi)/(stats_red(i).Perimeter)^2;

```

end

```
[idx,C] = kmeans(features(:,7),3);
```

```

sxhma1 = zeros(size(I_bin,1),size(I_bin,2));
sxhma2 = zeros(size(I_bin,1),size(I_bin,2));
sxhma3 = zeros(size(I_bin,1),size(I_bin,2));

```

```

for i=1:length(idx)
    if idx(i) == 1
        sxhma1 = sxhma1 + (I_labeled == i);
    elseif idx(i) == 2
        sxhma2 = sxhma2 + (I_labeled == i);
    else
        sxhma3 = sxhma3 + (I_labeled == i);
    end
end

```

```

figure, imshow(sxhma1)
figure, imshow(sxhma2)
figure, imshow(sxhma3)

```

