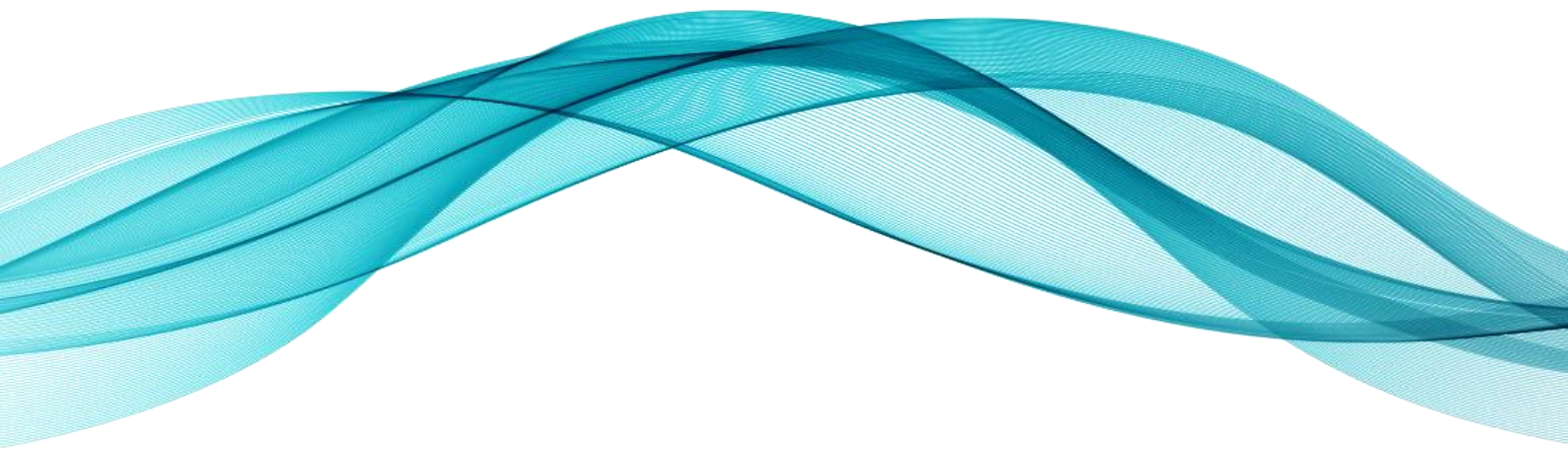


ΑΝΑΓΝΩΡΙΣΗ ΠΡΟΤΥΠΩΝ

***ΤΑΞΙΝΟΜΗΤΗΣ BAYES ΓΙΑ ΕΙΚΟΝΕΣ ΗΜΕΡΑΣ
ΚΑΙ ΝΥΧΤΑΣ ΜΕ ΧΡΗΣΗ MATLAB***



Τασούλας Θεοφάνης

Κέρκυρα, Ιούνιος του 2018

α) Περιγραφή του σκοπού

Κύριος στόχος του παρόντος είναι η απόκτηση γνώσεων και εμπειρίας στον κλάδο της Αναγνώρισης Προτύπων (Pattern Recognition). Αυτό επιτυγχάνεται μέσα από την εκμάθηση της στοιχειώδους θεωρίας του Bayes, της υλοποίησης και εκτέλεσης του αλγορίθμου ταξινόμησης Naive Bayes Classifier, της αντίληψης του τρόπου λειτουργίας και της σημασίας της ταξινόμησης στον τομέα αυτό. Ειδικότερα, το παρόν απαιτεί την ταξινόμηση 20 εικόνων ημέρας και 20 εικόνων νύχτας, όπου αποτελούν τη βάση γνώσης. Έπειτα, καθορίστηκαν ως μετρήσιμα χαρακτηριστικά οι τιμές της φωτεινότητας ή και της διακύμανσης της φωτεινότητας των εικόνων. Σύμφωνα με αυτές τις παραμέτρους εισόδου, σχηματίζεται μια καμπυλόμορφη κατανομή συχνοτήτων εμφάνισης. Με αυτήν ο αλγόριθμος ταξινόμησης θα επιτελεί τον καθορισμό των κλάσεων, δηλαδή θα ξεχωρίζει τις εικόνες ημέρας από της νύχτας με το μικρότερο δυνατό σφάλμα. Τέλος, γίνεται μια τυπική ανάληψη ρίσκου. Αυτά αποτελούν τα 5 βασικά βήματα για την ταξινόμηση κλάσεων στην αναγνώριση προτύπων, δοθέντος ενός προβλήματος.

- 1) Υλοποίηση βάσης γνώσης/δεδομένων
- 2) Καθορισμός μετρήσιμων χαρακτηριστικών (διάνυσμα χαρακτηριστικών)
- 3) Καθορισμός των κλάσεων
- 4) Λήψη αποφάσεων (εμπεριεχομένου της ελαχιστοποίησης σφάλματος)
- 5) Ανάληψη ρίσκου

Είναι σημαντικό να επισημανθεί ότι με μικρή παραλλαγή του παρόντος, είναι δυνατή η εισαγωγή μιας νέας τυχαίας εικόνας στον αλγόριθμο και η εξαγωγή του συμπεράσματος, εάν η φωτογραφία απεικονίζει ημέρα ή νύχτα.

β) Αλγόριθμος Naive Bayes και εφαρμογή του

Οι ταξινομητές Naive Bayes αποτελούν ένα σύνολο απλών «πιθανοτικών ταξινομητών» βασισμένων στην εφαρμογή του θεωρήματος του Bayes. Το θεώρημα εκφράζει την πιθανότητα του προτύπου να ταξινομείται στην κλάση ω_i , δεδομένου του χαρακτηριστικού διανύσματος \underline{x} , ίση με την πιθανότητα που περιγράφει την κατανομή του διανύσματος \underline{x} στην κλάση ω_i , επί την πιθανότητα του προτύπου να ανήκει στην κλάση ω_i , προς την πιθανότητα που ορίζει τη συνάρτηση πιθανοφάνειας του \underline{x} .

$$P(\omega_i|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i)P(\omega_i)}{p(\mathbf{x})}$$

Ο αλγόριθμος ενός ταξινομητή Naive Bayes επιδιώκει να ταξινομήσει ένα πρότυπο. Αρχικά δέχεται ένα σύνολο χαρακτηριστικών (έστω x_1, x_2, \dots, x_M) και προσδιορίζονται τα μετρήσιμα χαρακτηριστικά. Δεδομένων αυτών, ταξινομείται το πρότυπο σε κλάσεις (έστω $\omega_1, \omega_2, \dots, \omega_M$) ως εξής:

Αν $P(\omega_1|x) > P(\omega_2|x)$ τότε το x ανήκει στην κλάση ω_1

Αν $P(\omega_2|x) > P(\omega_1|x)$ τότε το x ανήκει στην κλάση ω_2

Να τονισθεί ότι ο αλγόριθμος θεωρεί πάντοτε τα χαρακτηριστικά του ως ανεξάρτητα μεταξύ τους, ακόμα και κατά την ύπαρξη τυχόν συσχετισμών. Αυτός είναι και ο λόγος, που ονομάζεται Naive (αφελής). Ωστόσο, οι ταξινομητές Naive Bayes παραμένουν αποτελεσματικοί σε πολλές πολύπλοκες πραγματικές καταστάσεις.

Στην προκείμενη περίπτωση, χρησιμοποιήθηκε το περιβάλλον αριθμητικής υπολογιστικής Matlab. Σε αυτό εισήχθησαν 20 φωτογραφίες ημέρας και 20 νύχτας. Για καθεμία από αυτές, μετασχηματίσθηκε το μοντέλο χρώματος RGB (Red, Green, Blue) σε HSV (Hue, Saturation, Value). Σε αυτό το σημείο ας σημειωθεί ότι κάθε εικόνα στον υπολογιστή αποτελείται από 3 δισδιάστατους πίνακες ή αλλιώς 3 επίπεδα (layers). Σε κάθε επίπεδο υπάρχει ένα πλήθος αριθμών, βάσει του προτύπου χρώματος. Λόγου χάριν, στο πρότυπο RGB το 1ο επίπεδο εμπεριέχει τις (αριθμητικές) τιμές του κόκκινου για μια εικόνα, το 2ο του πράσινου και το 3ο του μπλε. Αντίστοιχα στο μοντέλο HSV, τα τρία επίπεδα του πίνακα που αναπαριστούν την εικόνα, εμπεριέχουν τις τιμές απόχρωσης, κορεσμού και φωτεινότητας (το 3ο layer ονομάζεται Value που σημαίνει «τιμή», υπονοώντας τις τιμές φωτεινότητας).

Προκειμένου να ταξινομηθούν οι εικόνες σε ημέρας και νύχτας, παράγουμε τη μέση τιμή της φωτεινότητας των εικόνων με μορφή HSV. Με τις τιμές αυτές, απεικονίζουμε τη δισδιάστατη γραφική παράσταση της κατανομής συχνοτήτων εμφάνισης των τιμών μέσης φωτεινότητας. Έτσι, εξάγουμε συμπεράσματα για το διαχωρισμό των δυο κλάσεων (ημέρα και νύχτα), ώστε, εάν ο αλγόριθμος δεχόταν μια νέα εικόνα, να την ταξινομούσε στην κατάλληλη κλάση. Τέλος, συνυπολογίζεται ως δεύτερο χαρακτηριστικό η διακύμανση της φωτεινότητας, με απώτερο σκοπό την ελαχιστοποίηση της πιθανότητας σφάλματος. Βάσει αυτών των δυο χαρακτηριστικών (μέση τιμή, διακύμανση) εξάγουμε και ένα τρισδιάστατο γράφημα, που περιγράφει την κατανομή συχνοτήτων εμφάνισης των τιμών μέσης φωτεινότητας και διακύμανσης.

γ) Αναμενόμενα – επιθυμητά αποτελέσματα

Η αρχική έξοδος του αλγορίθμου είναι η δισδιάστατη γραφική παράσταση όπου, απεικονίζεται η κατανομή συχνοτήτων εμφάνισης των τιμών μέσης φωτεινότητας. Με το παραπάνω γράφημα προσδοκάται ο διαχωρισμός της περίπτωσης α) η εικόνα να είναι ημέρα, β) η εικόνα να είναι νύχτα και γ) να είναι απροσδιόριστη η περίπτωση μέρας ή νύχτας (πιθανότητα σφάλματος). Ακόμα, επιθυμητή είναι η ελάττωση της πιθανότητας σφάλματος, προσθέτοντας τη διακύμανση φωτεινότητας των φωτογραφιών ως χαρακτηριστικό εισόδου. Αυτό εκτιμάται με την εξαγωγή της αντίστοιχης τρισδιάστατης γραφικής παράστασης. Στη συνέχεια, συγκρίνονται οι δυο πιθανότητες σφάλματος και αναμένεται η 2η πιθανότητα (βάσει μέσης τιμής και διακύμανσης) να είναι μικρότερη της 1ης (βάσει μέσης τιμής μόνο). Εν ολίγοις στοχεύεται ο πιο ευδιάκριτος διαχωρισμός μιας εικόνας μέρας και νύχτας, καθώς αυξάνονται οι καλοί παράμετροι εισόδου. Αυτό γίνεται κατανοητό, αν σκεφθεί κανείς δυο αισθητήρες φωτιάς. Εάν ο πρώτος πετάει νερό, ελέγχοντας μόνο τη θερμοκρασία δωματίου, και ο δεύτερος ελέγχοντας και τη θερμοκρασία δωματίου και το διοξείδιο του άνθρακα και το εάν υπάρχει καπνός στο δωμάτιο, ποιος θα πάρει πιο σωστές αποφάσεις; Η απάντηση είναι οφθαλμοφανής, εφόσον οι παράμετροι αποτελούν καλά παραδείγματα εισόδου (π.χ. δεν θα εξετάζαμε το εάν υπάρχει φωτιά ή όχι, βάσει της ραδιενέργειας του χώρου).

δ) Ο κώδικας και η ανάλυσή του

Προχωρώντας στην ανάλυση του αλγορίθμου, ας σημειωθεί ότι ο κώδικας αποτελείται από ξεχωριστά τμήματα, αναλόγως του επιθυμητού στόχου. Το πρώτο τμήμα του αλγορίθμου αρχικοποιεί τα απαραίτητα διανύσματα ή αλλιώς τους πίνακες για να εκχωρηθούν οι κατάλληλες τιμές των εικόνων.

Στο δεύτερο μέρος, ξεκινάει μια επανάληψη από το 1 μέχρι το 20, ώστε να διαβαστούν οι 20 εικόνες νύχτας από το πρόγραμμα Matlab με την εντολή `imread`. Αυτή δέχεται ως παράμετρο τη διεύθυνση των εικόνων, μαζί με το αρχικό γράμμα της κάθε εικόνας (d για day ημέρα, n για night νύχτα) και τον αριθμό της επανάληψης και τέλος την κατάληξη της '.jpg' εικόνας. Συνεπώς πρώτα θα διαβαστεί η εικόνα n1.jpg, έπειτα η n2.jpg και ούτω καθεξής. Όταν διαβάζεται μια εικόνα, μετά μετατρέπεται το μοντέλο χρώματός της από RGB σε HSV με την εντολή `rgb2hsv`. Στη συνέχεια, παίρνουμε τις τιμές του 3ου layer της εικόνας, δηλαδή τις τιμές φωτεινότητας, και μετά υπολογίζουμε τη μέση τιμή και τη διακύμανσή της. Αυτό πετυχαίνεται, χρησιμοποιώντας την κάθε συνάρτηση εμφολιασμένη στον εαυτό της μια φορά. Τουτέστιν, **`mean(mean(value))`** και **`var(var(value))`**, όπου value ο πίνακας με τις τιμές φωτεινότητας των εικόνων. Τούτο, διότι κάθε layer της εικόνας αποτελεί ένα δισδιάστατο πίνακα (γραμμές και στήλες). Συνεπώς, με τη χρήση της μιας συνάρτησης εξάγουμε αποτέλεσμα για κάθε γραμμή ή στήλη ατομικά (π.χ. Μέση τιμή για κάθε γραμμή), δημιουργώντας μια κάθετη ή οριζόντια σειρά από αριθμητικές τιμές. Άρα με τη δεύτερη χρήση

βρίσκουμε το επιθυμητό αποτέλεσμα. Τέλος, αποθηκεύονται οι τιμές στους αντίστοιχους πίνακες. Ομοίως και για τις εικόνες ημέρας.

Το τρίτο κομμάτι του αλγορίθμου είναι προαιρετικό και στρογγυλοποιεί τους πίνακες, όπου μόλις δημιουργήθηκαν. Αυτό γίνεται με τη συνάρτηση `round`, όπου παίρνει ως πρώτο όρισμα τον πίνακα και ως δεύτερο τον αριθμό των ψηφίων μετά την υποδιαστολή – στην προκειμένη περίπτωση 1.

Το επόμενο τμήμα δημιουργεί τα σχεδιαγράμματα δύο διαστάσεων. Εμφανίζει ένα παράθυρο με την εντολή `figure` και προσθέτει σε αυτό ένα ιστόγραμμα με καμπάνες των τιμών φωτεινότητας της ημέρας, μέσω της `histfit` συνάρτησης. Μετά γίνεται το ίδιο για τις εικόνες νύχτας. Επίσης, χρησιμοποιείται η `grid on` εντολή, ώστε να δημιουργήσει γραμμές, που διαχωρίζουν το χώρο. Με την `xlim ([0 1])` περιορίζουμε τον προβαλλόμενο άξονα x , από το 0 μέχρι το 1 και με την `ylabel` και την `xlabel` εντολή, δίνεται μια ονομασία σε κάθε άξονα. Βέβαια, αφού προβληθεί το ιστόγραμμα και οι καμπάνες, διαγράφουμε το ιστόγραμμα και κρατάμε μόνο τις καμπάνες.

Το πέμπτο μέρος αναπαράγει τις τιμές μέσης φωτεινότητας ανά διακύμανση φωτεινότητας, με εντολές παρόμοιες με τις προηγούμενες. Για κάθε πλαίσιο/πλέγμα του σχεδιαγράμματος, μετράμε τα 'x' σύμβολα που απεικονίζονται και στη συνέχεια σημειώνουμε σε δυο αρχεία `excel` τη μέση φωτεινότητα (1η στήλη), τη διακύμανσή της (2η στήλη) και τον αριθμό των 'x' (3η στήλη). Για το αρχείο `NIGHT.csv` σημειώνονται τα κόκκινα σημεία, που αναπαριστούν τις τιμές της νύχτας και για το `DAY.csv` τα μπλε, τα οποία αντιπροσωπεύουν τις τιμές ημέρας. Αυτή η διαδικασία γίνεται εκτός του Matlab με χειροκίνητο τρόπο.

Στο πέμπτο τμήμα διαβάζεται το καινούργιο αρχείο `excel` και στο έκτο εκχωρούνται οι τιμές των στηλών του σε νέες μεταβλητές.

Τέλος, σχηματίζεται το διάγραμμα τριών διαστάσεων, δηλαδή της μέσης τιμής, διακύμανσης και συχνότητας εμφάνισης της φωτεινότητας των εικόνων. Αυτό υλοποιείται με την εντολή `load franke`, όπου φορτώνει τη συνάρτηση αλληλουχίας (bivariate) του Franke. Έπειτα η `fit` τοποθετεί την κατάλληλη διαφάνεια των δεδομένων με γραμμικό τρόπο (δέχεται ως τρίτη παράμετρο τη 'linear' γραμματοσειρά) και η `plot` σχεδιάζει το γράφημα.

```

1 % Initialization
2 DayBr=[]; % Day Brightness Values
3 DayVa=[]; % Day Variance Values
4 NightBr=[]; % Night Brightness Values
5 NightVa=[]; % Night Variance Values
6
7
8 % Read the images files and take the necessary values
9 for i=1:20 % Operating System may have a different way to specify
10 % the location of files! This command is for windows.
11 img_color = imread(['D:\day-night\n' num2str(i) '.jpg']); % Path name and file name. All night images start with 'n'
12 % and ends with a number (n1, n2, ..., n20). All images are .jpg.
13 hsv_img = rgb2hsv(img_color); % Conversion to hsv
14 value = hsv_img(:,:,3); % Take the 3rd layer
15 brightness=mean(mean(value)); % As long as we have a 2 x 2 table we need to use nested functions
16 NightBr=[NightBr,brightness]; % Save the value and do the same for the Variance
17 Variance = var(var(value));
18 NightVa=[NightVa,Variance];
19 end;
20 for i=1:20
21 img_color = imread(['D:\day-night\d' num2str(i) '.jpg']); % All day images start with 'd' and ends with a number (d1, d2, ..., d20).
22 hsv_img = rgb2hsv(img_color);
23 value = hsv_img(:,:,3);
24 brightness=mean(mean(value));
25 DayBr=[DayBr,brightness];
26 Variance = var(var(value));
27 DayVa=[DayVa,Variance];
28 end;
29
30
31 % That step is optional and it limits decimal points to 1 (for example the number 5.65128 is being converted to 5.7)
32 round(NightVa,1);
33 round(NightBr,1);
34 round(DayVa,1);
35 round(DayBr,1);
36
37
38 % Make the 2D chart of Brightness Vs Frequency of Occurrence
39 figure % New figure. do not throw it away any previous one
40 histfit(DayBr) % Design a histogram with the frequency of occurrence of Day Brightness values
41 grid on % Design some lines for more accurate appearance
42 hold all % Keep that histogram, do not throw it away if a new one will be added
43 histfit(NightBr) % Design a histogram with the frequency of occurrence of Night Brightness values
44 xlim ([0 1]) % Limitate x label. Starts from 0 and ends to 1.
45 ylabel 'Frequency of Occurrence' % Name the labels
46 xlabel 'Brightness'
47
48
49 % Make the 2D chart of Brightness Vs Variance (based on this chart we will count the 'x' signs per grid and add them to a csv file)
50 figure
51 plot(DayBr,DayVa,'x') % Creates a 2D line plot of the data
52 grid on
53 hold all
54 plot(NightBr,NightVa,'x')
55 grid on
56 hold all
57 xlabel 'Brightness mean'
58 ylabel 'Brightness variance'
59
60
61 % Read the files that contains the Brightness per Variance values + the number of 'x' signs per grid
62 DAY = csvread('D:\day-night\DAY.csv');
63 NIGHT = csvread('D:\day-night\NIGHT.csv');
64
65
66 % Define collumns
67 a=NIGHT(:,1); % Take the repetitively Brightness values
68 b=NIGHT(:,2); % Take the repetitively Variance values
69 c=NIGHT(:,3); % Take the number of 'x' signs
70 d=DAY(:,1);
71 e=DAY(:,2);
72 f=DAY(:,3);
73
74
75 % Make the 3D chart
76 figure
77 load franke % Franke bivariate test function
78 sf = fit([a, b],c,'linear') % Fit linear surface to data
79 plot(sf,[a,b],c) % Creates a 3D plot of a set of data points
80 hold on
81 sf = fit([d, e],f,'linear')
82 plot(sf,[d,e],f)
83 hold off % Sets the hold state to off so that new plots added to the axes
84 % clear existing plots and reset all axes properties

```


% Initialization

```
DayBr=[ ];  
DayVa=[ ];  
NightBr=[ ];  
NightVa=[ ];
```

% Read the images files and take the necessary values

```
for i=1:20  
    img_color = imread(['D:\day-night\n' num2str(i) '.jpg']);  
    hsv_img = rgb2hsv(img_color);  
    value = hsv_img(:,:,3);  
    brightness=mean(mean(value));  
    NightBr=[NightBr,brightness];  
    Variance = var(var(value));  
    NightVa=[NightVa,Variance];  
end;  
for i=1:20  
    img_color = imread(['D:\day-night\d' num2str(i) '.jpg']);  
    hsv_img = rgb2hsv(img_color);  
    value = hsv_img(:,:,3);  
    brightness=mean(mean(value));  
    DayBr=[DayBr,brightness];  
    Variance = var(var(value));  
    DayVa=[DayVa,Variance];  
end;
```

% Limits decimal points to 1

```
round(NightVa,1);  
round(NightBr,1);  
round(DayVa,1);  
round(DayBr,1);
```

% 2D chart of Brightness Vs Frequency of Occurrence

```
figure  
histfit(DayBr)  
grid on  
hold all  
histfit(NightBr)  
xlim ([0 1])
```

ylabel 'Frequency of Occurrence'
xlabel 'Brightness'

% 2D chart of Brightness Vs Variance

figure
plot(DayBr,DayVa,'x')
grid on
hold all
plot(NightBr,NightVa,'x')
grid on
hold all
xlabel 'Brightness mean'
ylabel 'Brightness variance'

% Read the files with the Brightness per Variance values

DAY = csvread('D:\day-night\DAY.csv');
NIGHT = csvread('D:\day-night\NIGHT.csv');

% Define collumns

a=NIGHT(:,1);
b=NIGHT(:,2);
c=NIGHT(:,3);
d=DAY(:,1);
e=DAY(:,2);
f=DAY(:,3);

% Make the 3D chart

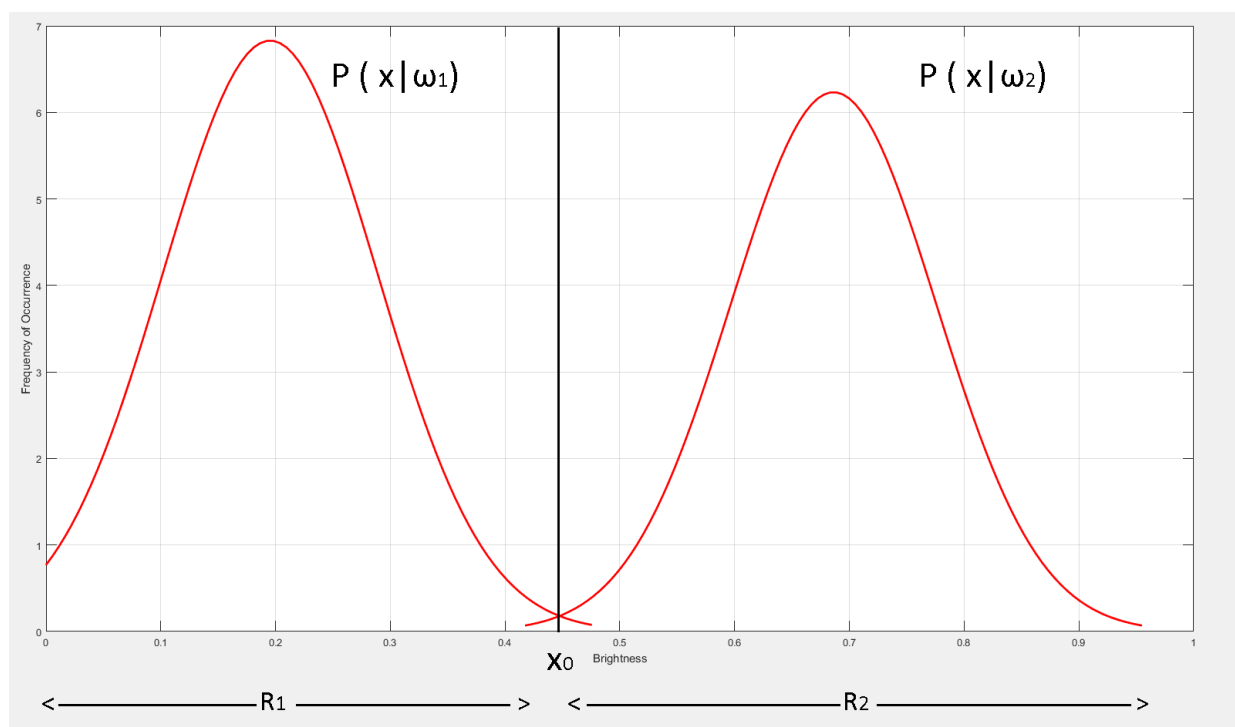
figure
load franke
sf = fit([a, b],c,'linear')
plot(sf,[a,b],c)
hold on
sf = fit([d, e],f,'linear')
plot(sf,[d,e],f)
hold off
xlabel 'Brightness mean'
ylabel 'Brightness variance'
zlabel 'Frequency of Occurrence'

ε) Εξαγωγή αποτελεσμάτων και συμπερασμάτων

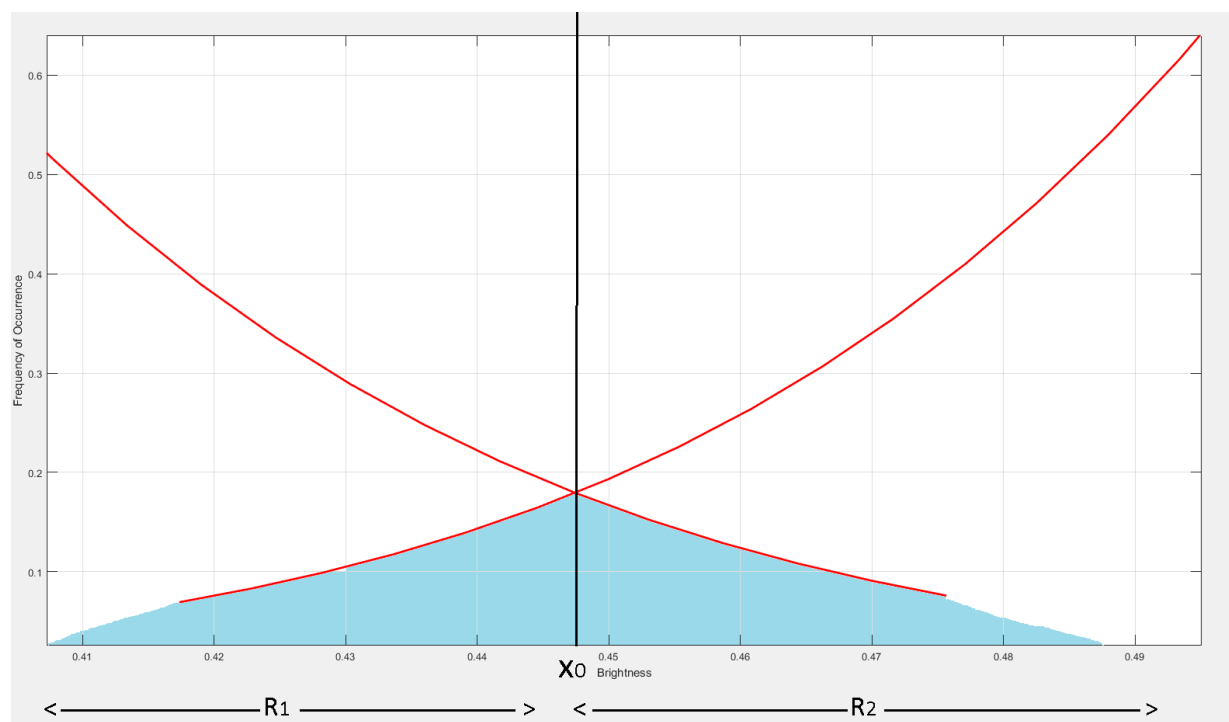
Αποτέλεσμα της πρώτης γραφικής παράστασης είναι δυο καμπύλες. Η δεξιά καμπύλη εκφράζει τη συχνότητα εμφάνισης της μέσης τιμής φωτεινότητας για τις εικόνες ημέρας και η αριστερή για της νύχτας. Αυτό είναι λογικό, εφόσον οι εικόνες ημέρας έχουν προφανώς μεγαλύτερη φωτεινότητα, έναντι των εικόνων νύχτας.

Στα πραγματικά προβλήματα δεν είναι σύνηθες το ιδανικό αποτέλεσμα. Παρομοίως και στο παρόν, δεν υπάρχει ισοδυναμία στην κατανομή. Το γεγονός ότι η δεξιά καμπάνα είναι λίγο μικρότερη της δεξιάς, υποδηλώνει την πιο συχνή εμφάνιση χαμηλής φωτεινότητας σε μια εικόνα. Με πιο απλά λόγια, από το σύνολο των εικόνων που συγκεντρώθηκαν, η μέση εικόνα είναι πιο σκοτεινή. Σε κάποιο άλλο σετ, θα μπορούσε να ισχύει το αντίθετο ή να έχουν ίσο ύψος οι καμπάνες. Ακόμη, δε θα πρέπει να λησμονείται ο μικρός αριθμός εικόνων. Με τη λήψη περισσότερων εικόνων, τα αποτελέσματα βελτιώνονται και προφανώς (αλλά όχι σίγουρα) οι καμπάνες μειώνουν την απόκλισή τους.

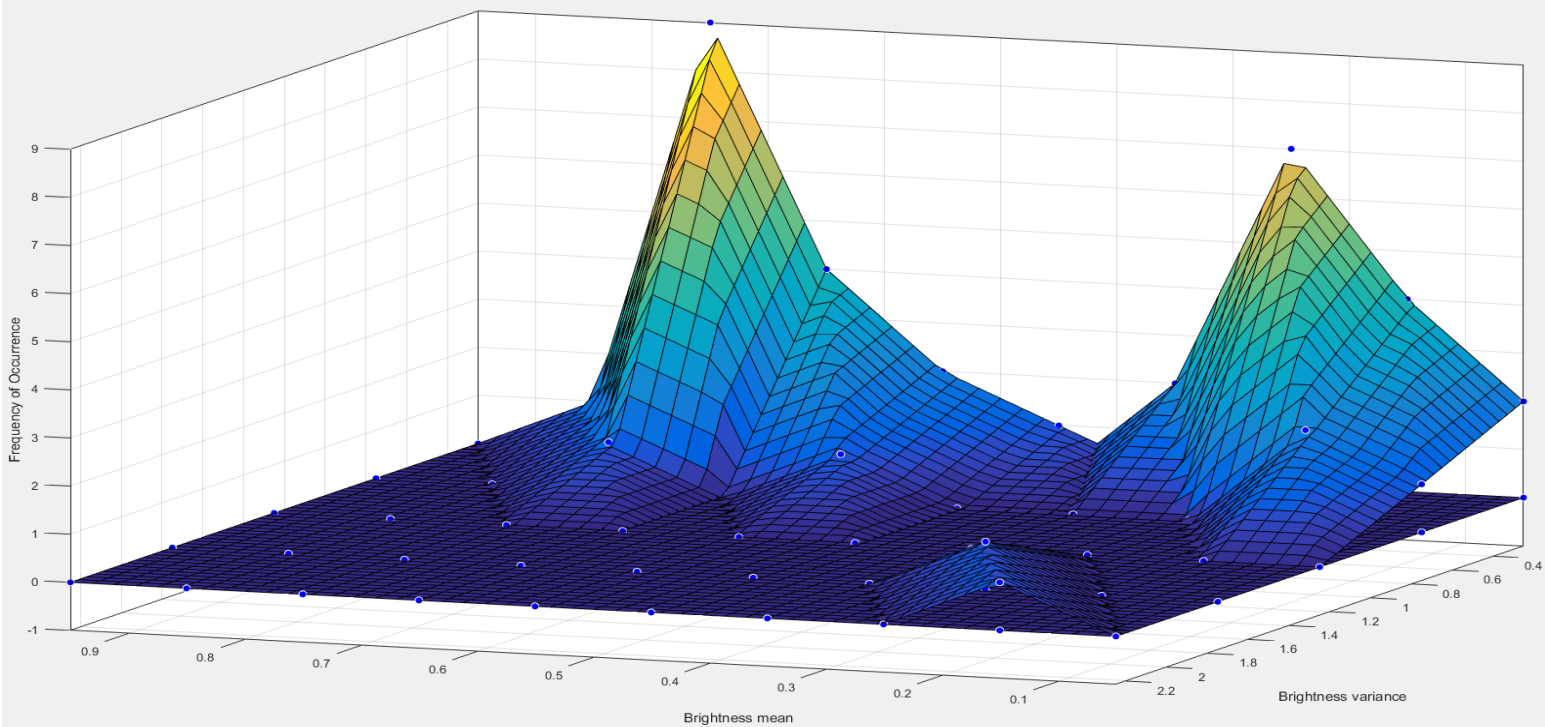
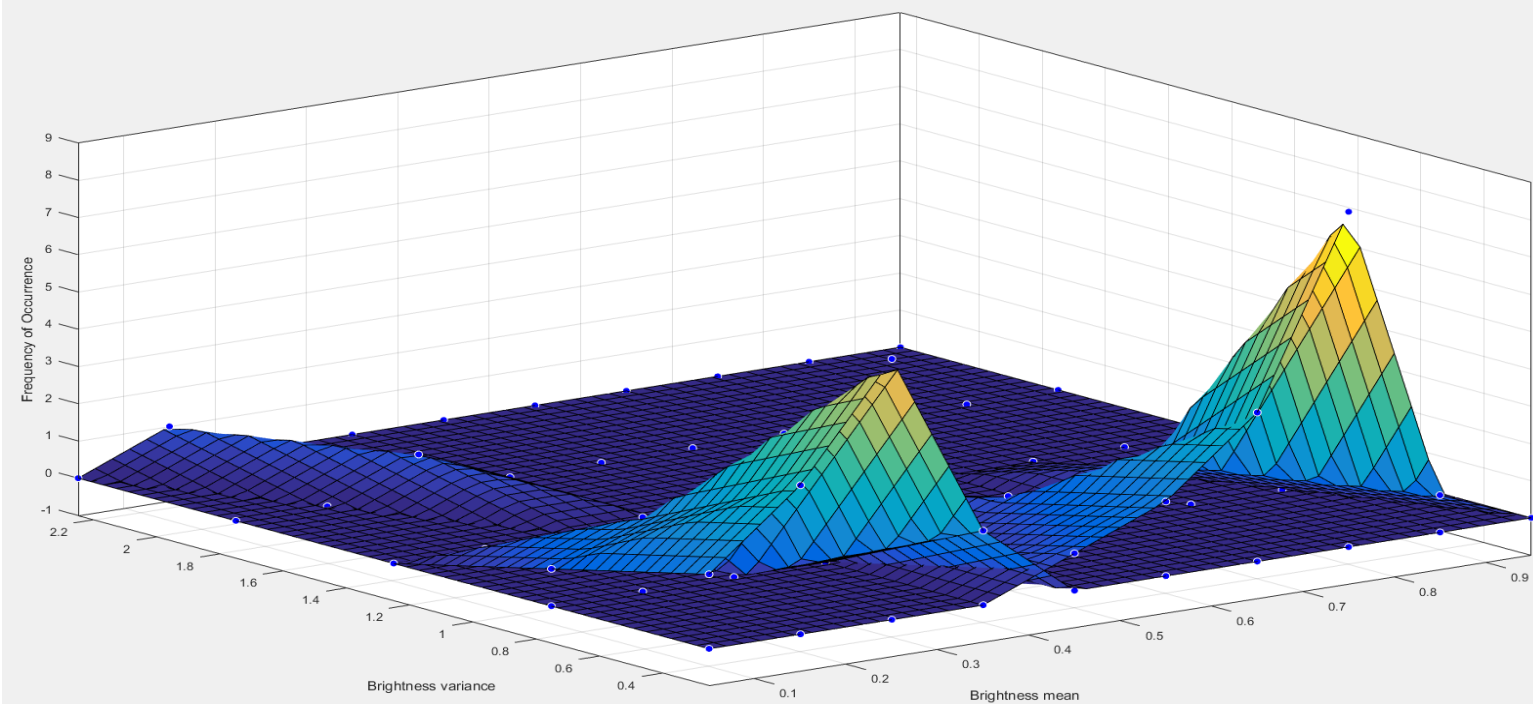
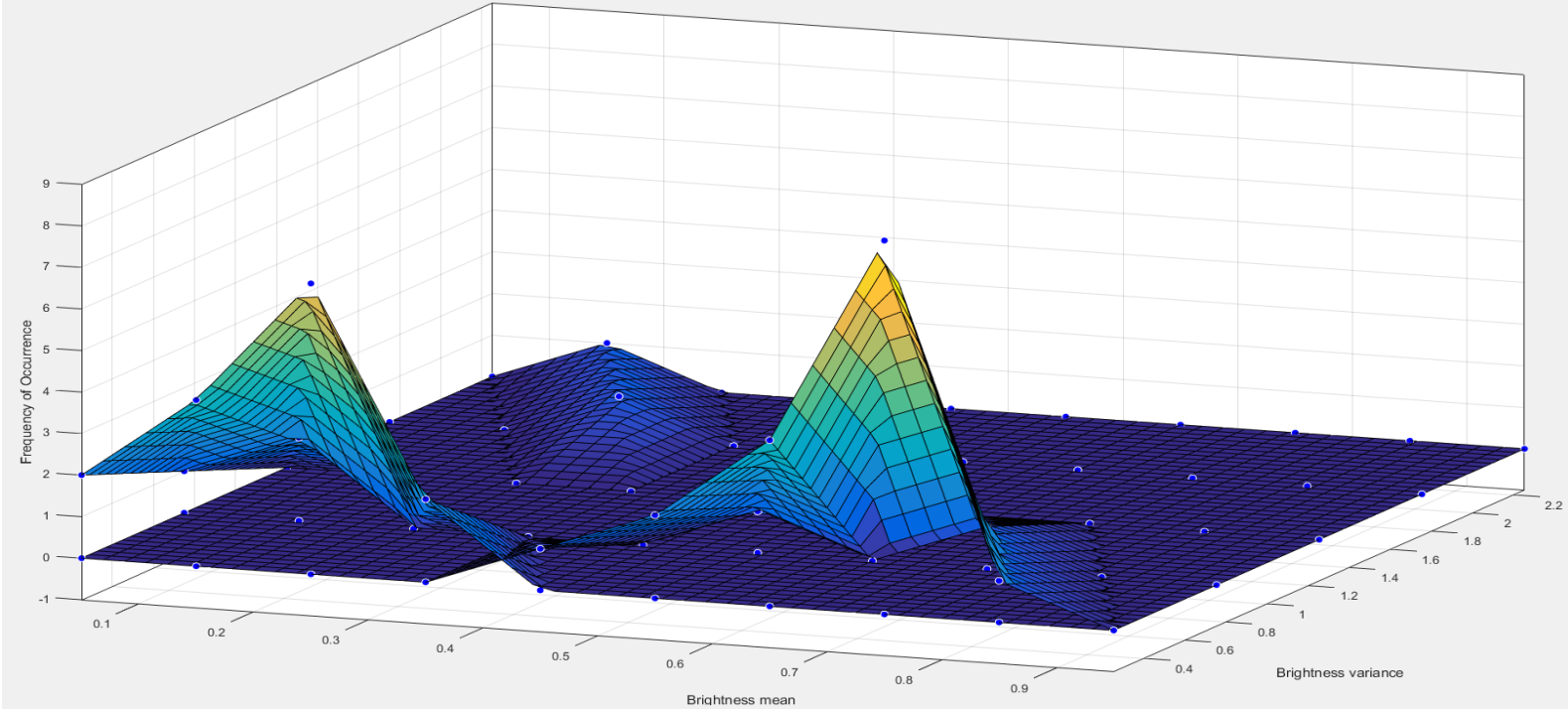
Το διάνυσμα \underline{x} εκφράζει τη μέση φωτεινότητα. Η μαύρη κάθετη γραμμή στο x_0 είναι ένα κατώφλι, που χωρίζει τον χώρο σε δύο περιοχές R_1 και R_2 , με σημείο τομής το x_0 . Κατά κανόνα του ταξινομητή Bayes, όταν το \underline{x} μιας εικόνας ανήκει στο διάστημα R_1 , θα κατατάσσεται στην κλάση ω_1 , δηλαδή θα θεωρείται εικόνα νύχτας. Αν ανήκει στο διάστημα R_2 , τότε θα ταξινομείται ως εικόνα ημέρας στην κλάση ω_2 . Τα διαστήματα R , $P(X|\omega)$, X_0 και κατώφλι δεν εξήχθησαν από το Matlab, αλλά σχεδιάστηκαν στο Photoshop.

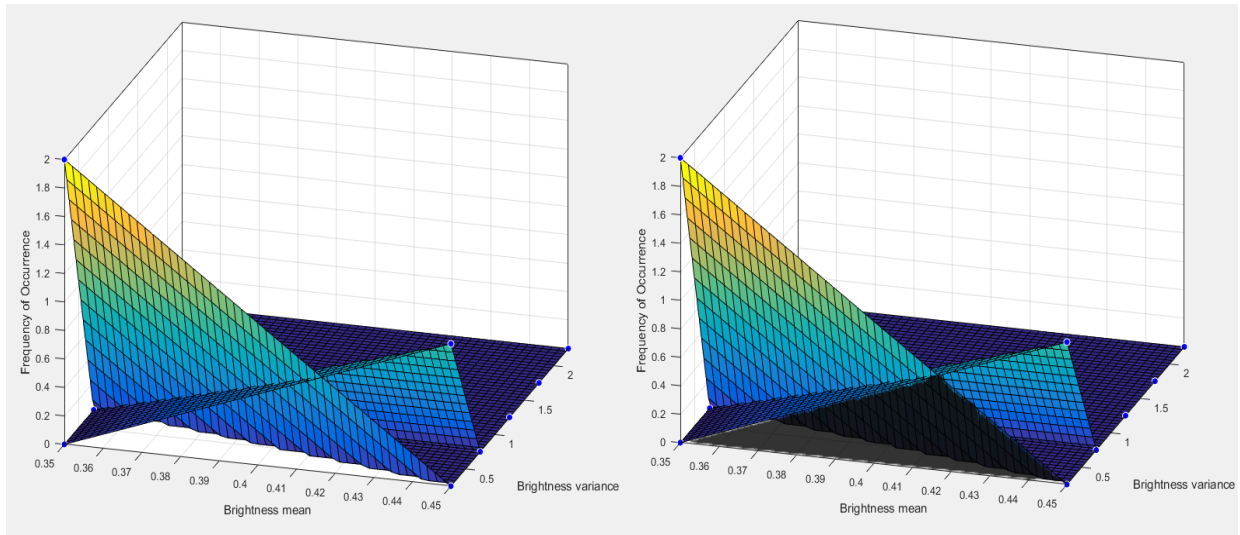


Ωστόσο, στην περίπτωση που ανήκει στην περιοχή, όπου οι δυο καμπύλες ενώνονται/συγκρούονται, τότε είναι απροσδιόριστη η κλάση ημέρας ή νύχτας. Αυτή είναι η πιθανότητα σφάλματος. Το παρακάτω γράφημα αποτελεί μεγέθυνση του παραπάνω, με επιπρόσθετο χρωματισμό στην περιοχή σφάλματος.



Μετάπειτα, το τρισδιάστατο σχεδιάγραμμα προβάλλεται σε τρεις διαφορετικές οπτικές γωνίες. Ο άξονας x είναι μέση φωτεινότητα, ο y είναι η διακύμανσή της και ο z είναι η συχνότητα εμφάνισης των τιμών της. Επομένως, η επιφάνεια που σχηματίζεται αντιπροσωπεύει την πιθανότητα η εικόνα να ανήκει στην κλάση ημέρα ή στην κλάση νύχτα, συσχετίζοντας τις τιμές διακύμανσης, συχνότητας εμφάνισης και μέσης τιμής της φωτεινότητας.





Αξιοπρόσεκτο είναι το κενό τμήμα που δημιουργείται στις τιμές μέσης φωτεινότητας από 0.35 έως και 0.45 (στη δεξιά εικόνα δείχνει σκιερό). Αυτό παρουσιάζει τη νέα πιθανότητα σφάλματος.

Συγκρίνοντας τις δυο πιθανότητες σφάλματος, συμπεραίνεται ότι, εάν η πιθανότητα σφάλματος μειώθηκε με την προσθήκη της νέας παραμέτρου, τότε η διακύμανση φωτεινότητας αποτελεί ένα καλό παράδειγμα για την ανίχνευση ημέρας ή νύχτας σε μια φωτογραφία. Όμως, η αντίθετη περίπτωση θα σήμαινε ότι αυτό το χαρακτηριστικό δεν αποτελεί καλό παράδειγμα.

Τελειώνοντας, θα παρθεί μια τυπική ανάληψη ρίσκου. Εν ολίγοις με αυτήν ενσωματώνεται η πιθανότητα σφάλματος σε μια κλάση, προκειμένου να αποφευχθεί η μέγιστη δυνατή ζημία. Επομένως, εάν με την εισαγωγή μιας νέας εικόνας στον αλγόριθμο δεν υπάρξει σαφής ταξινόμηση σε κάποια κλάση, τότε θα καταταχθεί στην κλάση νύχτας, εάν θεωρηθεί η περίπτωση νύχτας πιο ζημιοφόρα έναντι της ημέρας. Ωστόσο, εάν θεωρηθεί πιο ζημιοφόρα η κλάση ημέρας, θα ισχύσει το αντίθετο. Σίγουρα στην προκειμένη περίπτωση δεν είναι ευδιάκριτη η αξία της ανάληψης ρίσκου. Αλλά σε έναν ταξινομητή, που διακρίνει εάν ένας ασθενής έχει καλοήγη ή κακοήγη όγκο, υπάρχει ιδιαίτερα μεγάλη σημασία στην κάθε απόφασή του. Διότι, εάν ο ασθενής έχει στην πραγματικότητα καλοήγη όγκο και ο ταξινομητής τον κατατάζει σε κακοήγη, θα υπάρξουν κάποιες οικονομικές χρεώσεις για φάρμακα, θεραπείες και εξετάσεις, αλλά σίγουρα δε θα υπάρξουν ανεπανόρθωτες συνέπειες. Αντιθέτως, αν έχει στην πραγματικότητα κακοήγη όγκο και ταξινομηθεί σε καλοήγη, ενδέχεται ακόμα και ο θάνατος για αυτόν, επειδή δε θα υπάρξει η κατάλληλη περίθαλψη. Συνεπώς, για να αποφευχθεί η δεύτερη και πιο ζημιοφόρα περίπτωση, η πιθανότητα σφάλματος οφείλει να οδηγεί στην κλάση «κακοήγης όγκος».

ΒΙΒΛΙΟΓΡΑΦΙΑ – ΔΙΚΤΥΟΓΡΑΦΙΑ

***Αναγνώριση Προτύπων, S. Theodoridis, K. Koutroumbas, 2011,
ΕΚΔΟΣΕΙΣ ΠΑΣΧΑΛΙΔΗΣ***

<https://opencourses.ionio.gr/courses/DDI133/>

<https://www.mathworks.com/help/matlab/>