DECISION TREES, GINI, GAIN κλπ

Για να βρω **GINI** πχ (απο εργασια 1)

ID	a1	a2	a3	class
id1	T	T	1.0	+
id2	T	T	6.0	+
id3	T	F	5.0	–
id4	F	F	4.0	+
id5	F	T	7.0	–
id6	F	T	3.0	–
id7	F	F	8.0	–
id8	T	F	7.0	+
id9	F	T	5.0	–

Βρισκω gini κλασης για αρχη δηλαδη του class ως εξης:

Σύνολο κλασσ στοιχείων = 9

Τα + ειναι 4

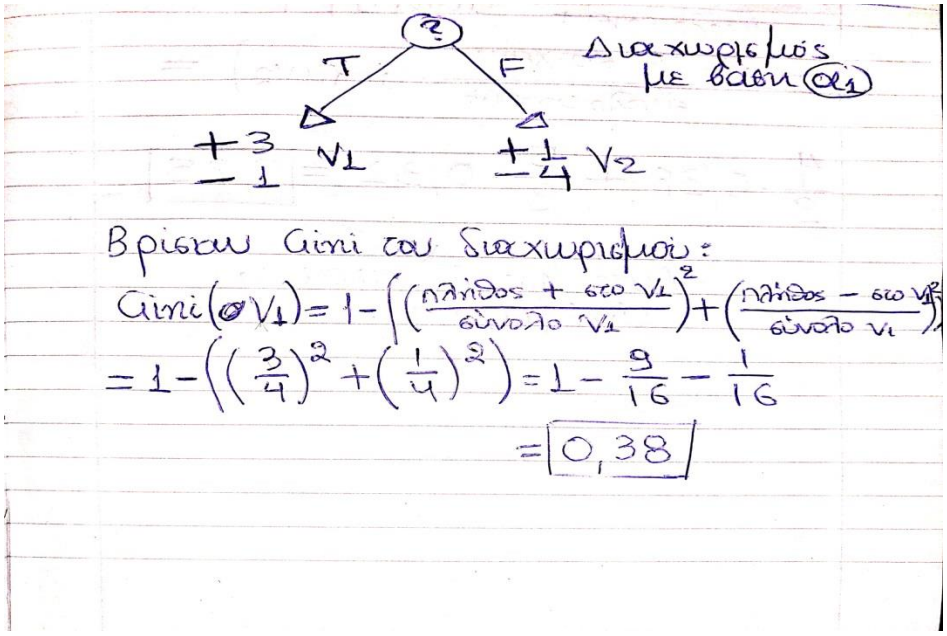
Τα - ειναι 5

Αρα έχω: Gini(parent=class) = 1 – ((πληθος ΣΥΝ/σύνολο)^2 + (πλήθος ΜΕΙΟΝ / σύνολο)^2)

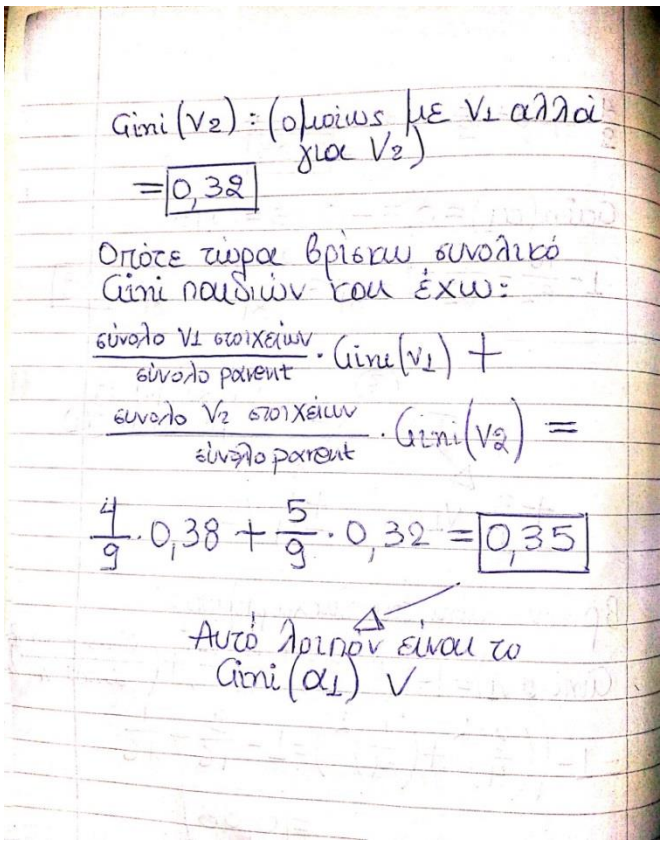
= 1 – $(4/9)^2$ – $(5/9)^2$ = 1 – 16/81 – 25/81 = 1 – 41/81 = 40/81 = **0.49** Θα το χρειαστούμε μετά.

Τώρα για κάθε στήλη ψάχνω τζινι:

α1:



Συνολικο Τζινι Παιδιων = Weighted



Το ***Gain*** της α1 θα είναι απλά η αφαίρεση των Gini(parent) – Gini(a1) = 0.49 – 0.35 = ***0.14***

<u>Ομοίως βγαίνουν για α2 και α3</u>

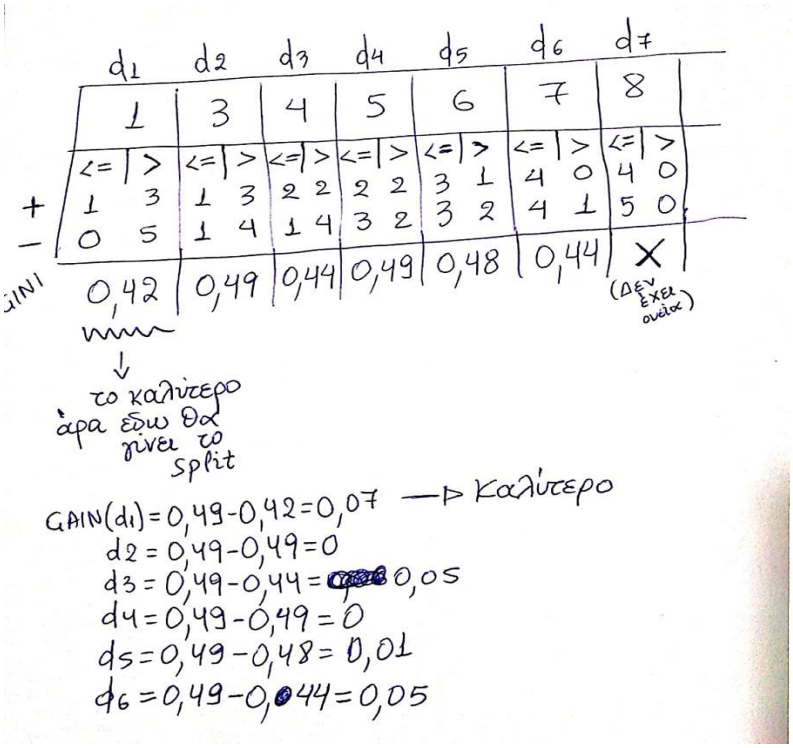
Αν ρωτήσει γιατι δεν παίρνουμε για ρίζα το ID θα πούμε:

> Δεν επιλέγουμε το ID ως ρίζα γιατί στην ουσία δεν μας προσφέρει κάποια πληροφορία αφού πάντα τα id θα είναι διαφορετικά και μελλοντική εισαγωγή κάποιου id δε θα ανήκει σε καμία κατηγορία.

Τωρα όσον αφορά συνεχείς τιμές και διάσπαση τους κλπ (εργασία 1 πάλι)

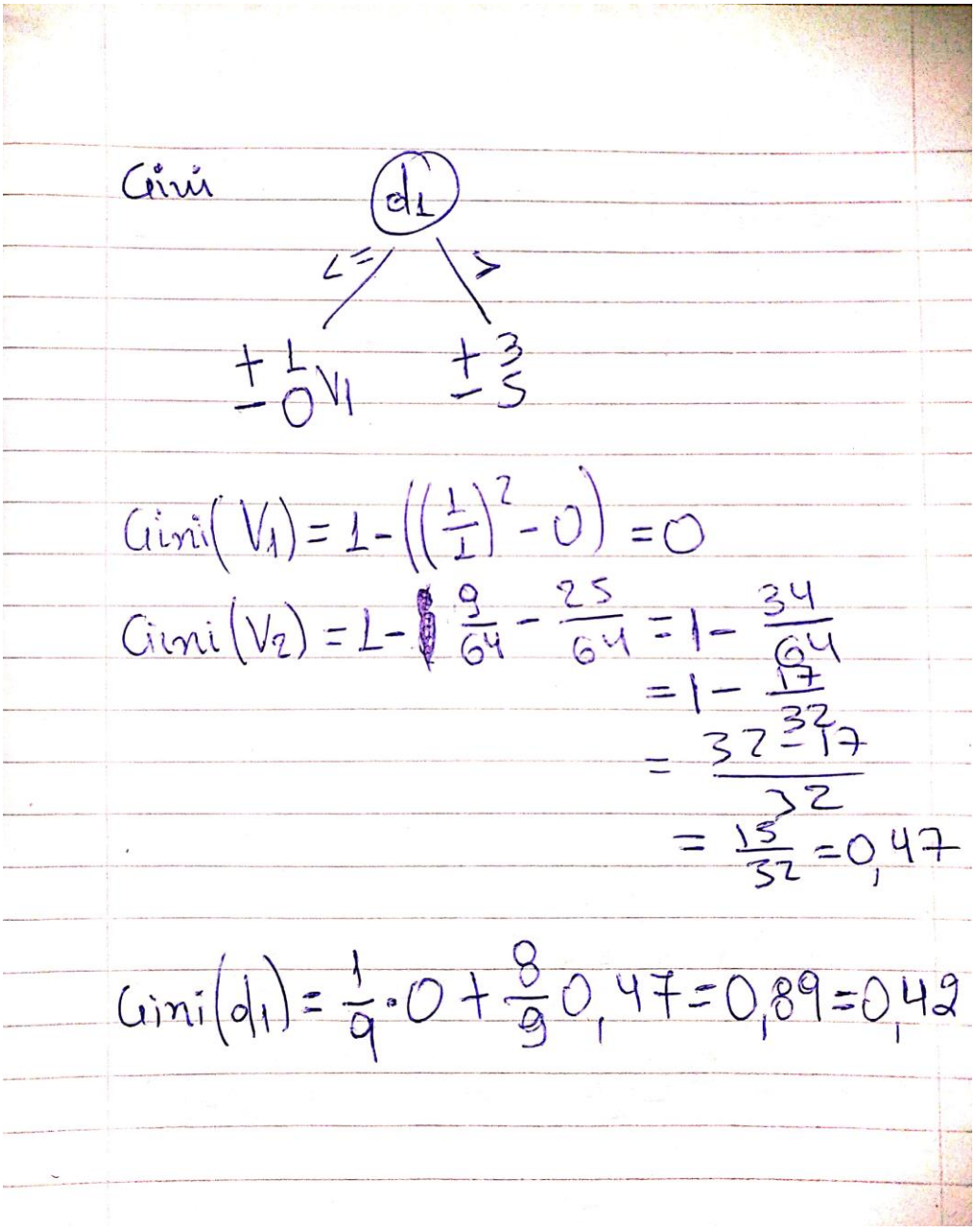
<u>(Κανονικα θέλει σπλιτ τιμες δηλαδη αναμεσα απο 1-3 είναι το 2, το γράφω, ανάμεσα απο 3-4 είναι το 3,5 κοκ)</u>

Αλλά μάλλον είναι σωστό και με τη μέθοδο της φωτογραφίας

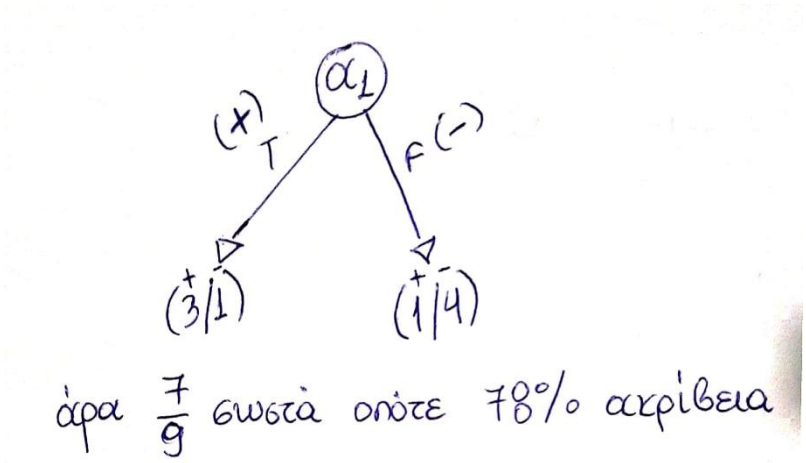


Ενδεικτικά το Τζινι = 0.42:

Gini(d1):


Gini
d1
<=
>
+ 1
- 0
V1
+ 3
- 5
Gini(V1) = 1 - ((1/1)^2 - 0) = 0
Gini(V2) = 1 - 9/64 - 25/64 = 1 - 34/64
= 1 - 17/32
= (32 - 17)/32
= 15/32 = 0,47
Gini(d1) = 1/9 · 0 + 8/9 0,47 = 0,89 = 0,42

Αν ρωτήσει ποια βάζουμε ρίζα λέμε **<u>αυτή με το μεγαλύτερο GAIN.</u>** Στην προκειμένη περίπτωση είναι η **α1**

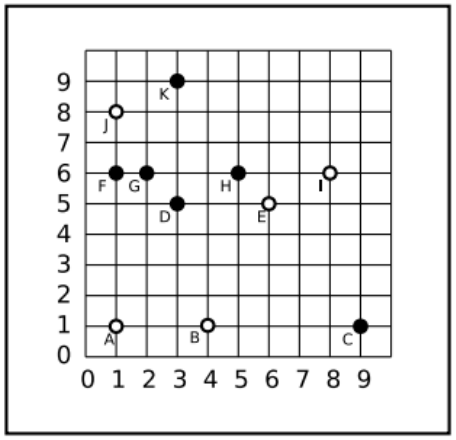


KNN, IB2, KMEANS, DATA REDUCTION, CLUSTERING κλπ

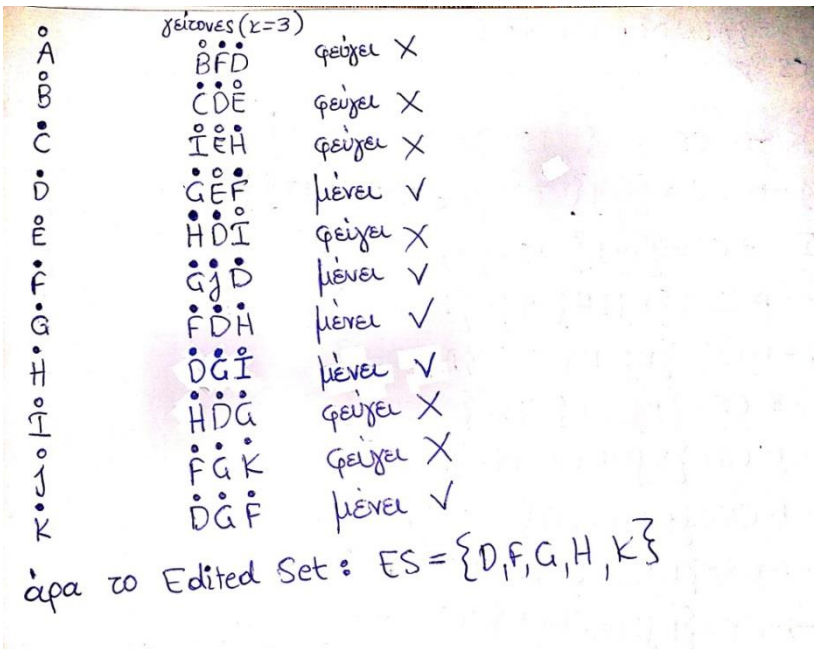
Παίρνω πάλι την εργασία 2 ως παράδειγμα

DATA REDUCTION

KNN(εδώ ειναι κ=3) *!!Σε περίπτωση ισοπαλίας εγγύτερος είναι αυτός που είναι πιο μικρός αλφαβητικά*



Επί της ουσίας παίρνεις τους Κ κοντινότερους γείτονες και αν η πλειοψηφία συμφωνεί με αυτό που ψάχνω να προσθέσω, το κρατάω αλλιώς μπούλο:



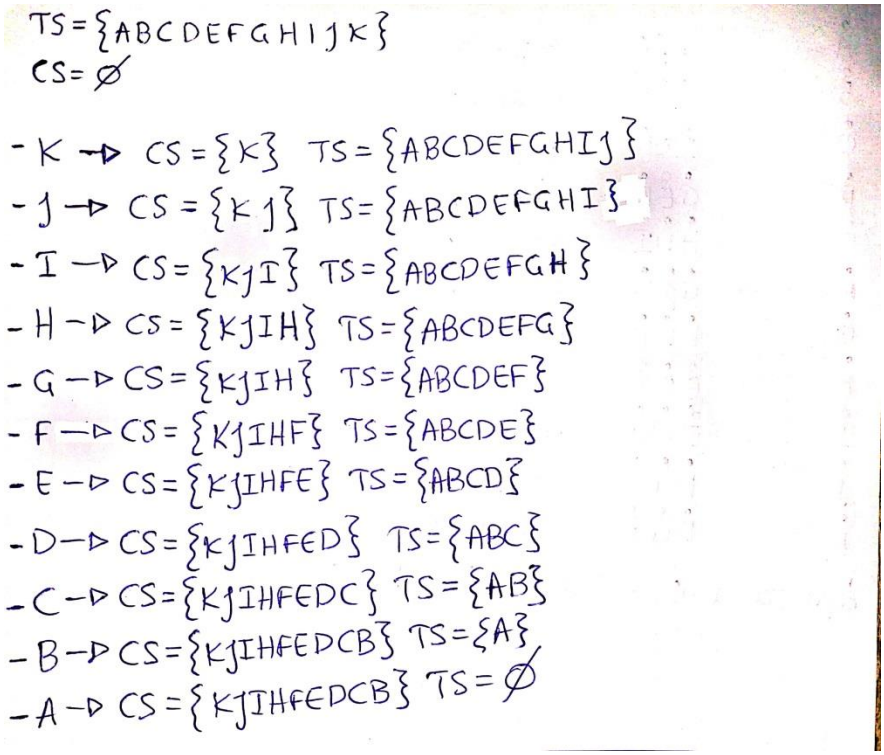
IB2: Αυτο που κάνει είναι οτι στην ουσία παίρνεις ένα ένα τα γράμματα και συγκρίνεις με τους γείτονες του υποψηφίου γράμματος ***ΠΟΥ ΒΡΙΣΚΟΝΤΑΙ ΗΔΗ ΣΤΟ CS.*** Λίγο πιο αναλυτικά:

Υποψήφιο το J. Έχω CS={K}. J με K διαφορετική κλάση (άσπρο-μαύρο) άρα ΒΑΖΩ ΤΟ J ΣΤΟ CS ΚΑΙ ΤΟ ΑΦΑΙΡΩ ΑΠΟ ΤΟ TS.

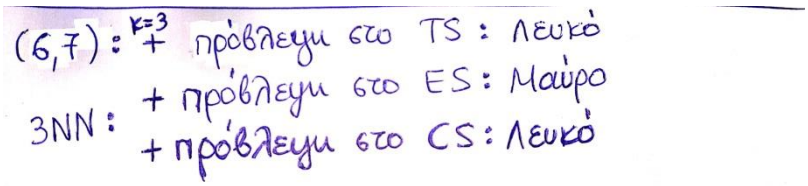
Υποψήφιο το G. Έχω CS={K J I H}. Κοντινότερος γείτονας του G απ' αυτά τα 4 είναι το H και το J (απόσταση 3 και τα δύο αλλά κρατάω το αλφαβητικά μικρότερο άρα H). G με H ίδια κλάση(μαύρο-μαύρο) άρα ΤΟ G ΠΑΙΡΝΕΙ ΜΠΟΥΛΟ. Το αφαιρώ από το TS και συνεχίζω.

Υποψήφιο το A, κοντινότερος στο CS το B -> ΙΔΙΑ ΚΛΑΣΗ ΑΡΑ ΜΠΟΥΛΟ ΣΤΟ A

Στις φώτο το CS που κοιτάω για να δω αν θα εισέλθει το υποψήφιο είναι της απο πάνω γραμμής!



Πρόβλεψη κατα σειρά για TS,ES,CS



CLUSTERING – ΣΥΣΤΑΔΟΠΟΙΗΣΗ

a				b	c			d				e			f		g	h	i		j
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	

Στα παρακάτω όποτε υπάρχει περίπτωση ισοβαθμίας να επιλέγετε πάντα την επιλογή στα αριστερά.

1. Εφαρμόστε Ιεραρχική συσταδοποίηση με μέτρο ομοιότητας συστάδων τη μέθοδο MIN distance (simple linkage). Δώστε τα διαδοχικά βήματα του αλγορίθμου:

ΒΗΜΑ 1:	a(0)	b(4)	c(5)	d(8)	e(12)	f(14)	g(16)	h(17)	i(18)	j(20)
ΒΗΜΑ 2:	a(0)	bc(4,5)		d(8)	e(12)	f(14)	g(16)	h(17)	i(18)	j(20)
ΒΗΜΑ 3:	a(0)	bc(4,5)		d(8)	e(12)	f(14)	gh(16,17)		i(18)	j(20)

Βήμα 4: a(0) bc(4,5) d(8) e(12) f(14) ghi(16,17,18) j(20)
Βήμα 5: a(0) bc(4,5) d(8) ef(12,14) ghi(16,17,18) j(20)
Βήμα 6: a(0) bc(4,5) d(8) efghi(12,14,16,17,18) j(20)
Βήμα 7: a(0) bc(4,5) d(8) efghij(12,14,16,17,18,20)
Βήμα 8: a(0) bcd(4,5,8) efghij(12,14,16,17,18,20)
Βήμα 9: abcd(0,4,5,8) efghij(12,14,16,17,18,20)
Βήμα 10: abcdefghij

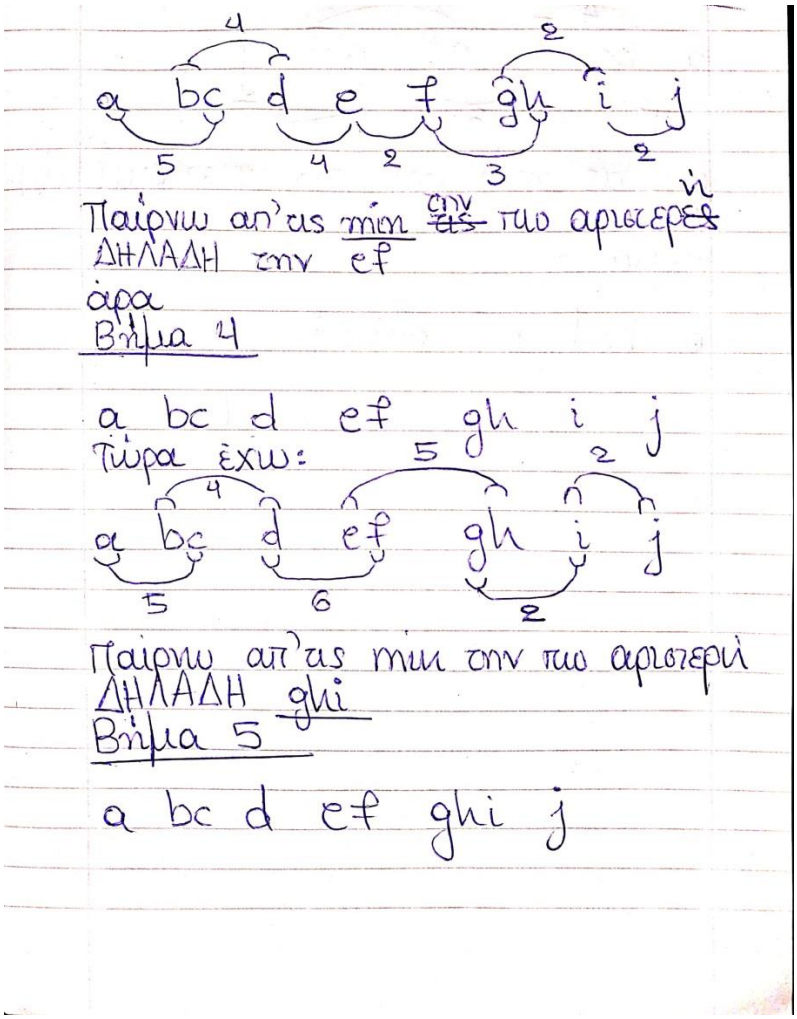
Επεξήγηση ενδεικτικά: Στο ΒΗΜΑ 6→ Προσαρτώ το {e,f} στο {g,h,ι} ως εξής: Βλέπω τις ακραίες τιμές του {e,f} πόσο απέχουν απ'τους γείτονες (τις min αποστάσεις από τις γειτονικές συστάδες). Δηλαδή το f από το g (distance = 2) και το e από το d (distance = 4). Οπότε προσθέτω τη συστάδα {e,f} στην {g,h,i}

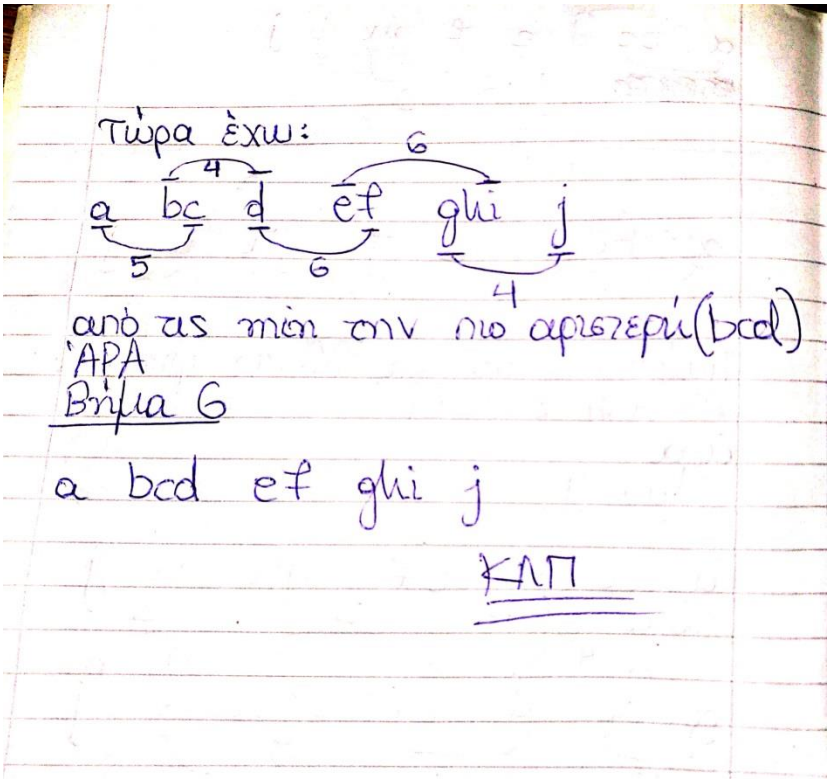
a			b	c			d				e		f		g	h	i		j
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

2. Εφαρμόστε Ιεραρχική συσταδοποίηση με μέτρο ομοιότητας συστάδων τη μέθοδο MAX distance (Complete linkage). Δώστε τα διαδοχικά βήματα του αλγορίθμου.

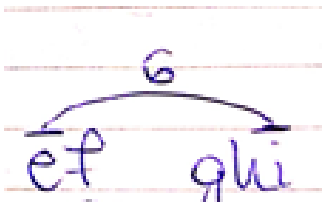
54:08 ΣΤΗ ΔΙΑΛΕΞΗ 6

```
(B2)
ΒΗΜΑ  1:   a(0)    b(4)   c(5)    d(8)    e(12)   f(14)   g(16)   h(17)   i(18)   j(20)
ΒΗΜΑ  2:   a(0)    bc(4,5)        d(8)    e(12)   f(14)   g(16)   h(17)   i(18)   j(20)
ΒΗΜΑ  3:   a(0)    bc(4,5)        d(8)    e(12)   f(14)   gh(16,17)       i(18)   j(20)
ΒΗΜΑ  4:   a(0)    bc(4,5)        d(8)    ef(12,14)       gh(16,17)       i(18)   j(20)
ΒΗΜΑ  5:   a(0)    bc(4,5)        d(8)    ef(12,14)       ghi(16,17,18)           j(20)
ΒΗΜΑ  6:   a(0)    bcd(4,5,8)             ef(12,14)       ghi(16,17,18)           j(20)
ΒΗΜΑ  7:   a(0)    bcd(4,5,8)             ef(12,14)       ghij(16,17,18,20)
ΒΗΜΑ  8:   abcd(0,4,5,8)                  ef(12,14)       ghij(16,17,18,20)
ΒΗΜΑ  9:   abcd(0,4,5,8)                  efghij(12,14,16,17,18,20)
ΒΗΜΑ 10:   abcdefghij (0,4,5,812,14,16,17,18,20)
```

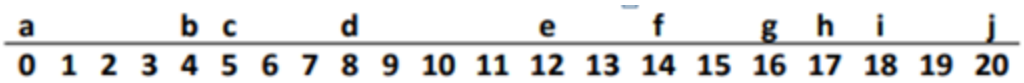




Στην ουσία το ΜΑΞ έχει να κάνει με τις <u>αποστάσεις ανάμεσα στις συστάδες</u> και τίποτα παραπάνω. Η **συγχώνευση** θα γίνει πάλι με το **MIN** (Η MIN απόασταση ανάμεσα στις ΜΑΞ)



Εδώ πχ πριν στο MIN θα έπαιρνε ως απόσταση την απόσταση του f με g ενώ εδώ που ζητάει ΜΑΞ παίρνεις των πιο μακρινών σημείων μεταξύ των δύο συστάδων δηλαδή e και i .



KMEANS: Το Κ είναι οι συνολικές συστάδες που θα έχουμε χωρίσει τα δεδόμενα μας. Εδώ 3 (15, 17, 19)

3. Εφαρμόστε τον k-means με k=3 και αρχικά centroids τα σημεία 15, 17, 19. Δώστε τα διαδοχικά βήματα του αλγορίθμου (Ci είναι οι συστάδες):

```
(B3)
ΒΗΜΑ 1:            C1Centroid=15     C2Centroid=17     C3Centroid=19
ανάθεση σημείων:   C1(abcdefg)       C2(hi)            C3(j)
ΒΗΜΑ 2:            C1Centroid=8.43   C2Centroid=17.5   C3Centroid=20
ανάθεση σημείων:   C1(abcde)         C2(fghi)          C3(j)
ΒΗΜΑ 3:            C1Centroid=5.8    C2Centroid=16.25  C3Centroid=20
ανάθεση σημείων:   C1(abcd)          C2(efghi)         C3(j)
ΒΗΜΑ 4:            C1Centroid=4.25   C2Centroid=15.4   C3Centroid=20
ανάθεση σημείων:   C1(abcd)          C2(efgh)          C3(ij)
ΒΗΜΑ 5:            C1Centroid=4.25   C2Centroid=14.75  C3Centroid=19
ανάθεση σημείων:   C1(abcd)          C2(efg)           C3(hij)
ΒΗΜΑ 6:            C1Centroid=4.25   C2Centroid=14     C3Centroid=18.33
ανάθεση σημείων:   C1(abcd)          C2(efg)           C3(hij) ΔΕΝ ΥΠΑΡΧΕΙ ΑΛΛΑΓΗ
```

!Στο ΒΗΜΑ 1 το g μπαίνει στο C1 επειδή στην εκφώνηση λέει ότι προηγείται ότι είναι αριστερά! (*Υποψήφια για το g είναι το 15 και 17. Το 15→αριστερά του g άρα πάει εκεί*)

Η διαδικασία είναι να βάζεις το κάθε σημείο στη συστάδα που είναι πιο κοντά. Centroid βρίσκουμε αν προσθέσουμε όλους τους αριθμούς της συστάδας και διαιρέσουμε με το πλήθος της συστάδας.

ΒΗΜΑ 3: C1centroid(5,8). Το 5,8 βγαίνει ως εξής: C1(a,b,c,d,e) άρα **0+4+5+8+12 = 29 / 5 = 5,8**

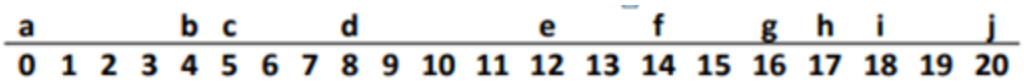
Άρα το νέο **C1** με βάση το 5,8 είναι **(a,b,c,d)**

Ομοίως βγαίνουν και τα άλλα

ΒΗΜΑ 4: C1centroid = 4,25 από (a,b,c,d) → **0+4+5+8 = 17 / 4 = 4,25**

Άρα με βάση το νέο centroid έχω νέο C1 = (a,b,c,d) → ΔΕΝ ΑΛΛΑΖΕΙ

Σταματάω τον αλγόριθμο όταν δω ότι δεν έχω αλλαγές στα C και τα centroids!



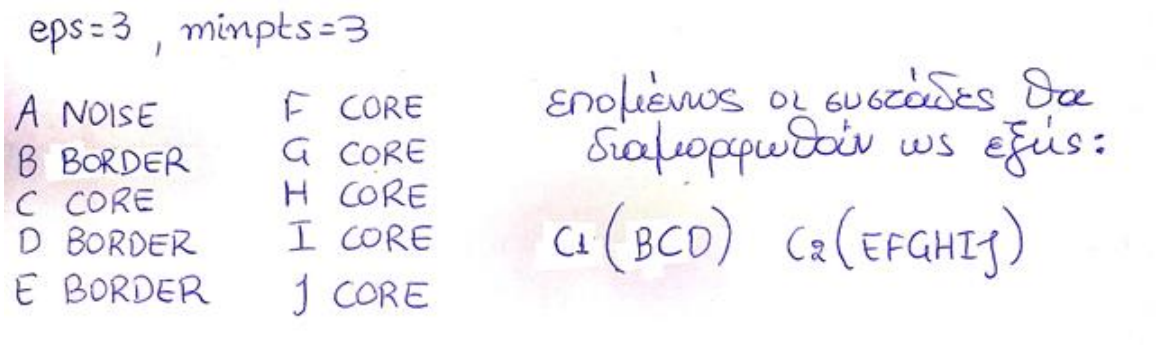
DBSCAN: Δωσμένου του EPSILON και του MINPTS θα χωρίσεις τα σημεία σε CORE και NOISE

CORE είναι ένα σημείο όταν σε απόσταση *EPSILON* έχει για γείτονες τουλάχιστον *MINPTS* σημεία.

BORDER είναι ένα σημείο όταν δεν ειναι CORE αλλά στη γειτονιά του (απόσταση EPS) έχει κάποιο **CORE** point

NOISE είναι όταν δεν είναι κάτι από τα παραπάνω

4. Εφαρμόστε τον DBSCAN με eps=3 και minpts=3.



Στα MINPTS συμπεριλαμβάνουμε και ΤΟ ΙΔΙΟ ΤΟ ΣΗΜΕΙΟ

Πχ για Α: σε απόσταση EPS=3 έχει τον εαυτό του και τίποτα άλλο άρα ΔΕΝ ΕΙΝΑΙ **CORE** και (σύμφωνα με φώτο) δεν έχει CORE point για γείτονα άρα είναι **NOISE.**

Για C: Έχει στη γειτονιά του (απόσταση EPS=3) 3 σημεία(MINPTS=3), το Β, το C και το D (ΒΑΖΟΥΜΕ ΚΑΙ ΤΟΝ ΕΑΥΤΟ ΤΟΥ). Άρα είναι **CORE**

Τώρα τα clusters δημιουργούνται ως εξής:

C1: Παίρνω για αρχή όποιο CORE point μου καυλώσει. Έστω το πρώτο που βλέπω δηλαδή το C και βλέπω σε απόσταση EPS=3 αν έχει άλλα CORE points. Αν ναι, τα προσθέτω στο cluster.

C1: {C} Για αρχή μπαίνει μόνο του το C γιατί στη γειτονιά δεν έχει αλλο CORE point

Φτιάχνω καινούργιο κλαστερ τώρα

C2→Έστω ότι ξεκινάω από το F

C2: {F, G, H, I, J). Κάθε φορά που προσθέτω ένα σημείο στο κλάστερ κοιτάω έπειτα τη γειτονιά αυτού του σημείου, δηλαδή:

Βάζω το F → Κοιτάω αν το F έχει κάποιο CORE στη γειτονιά και βάζω το κοντινότερο. Είναι το G

Άρα έχω C2: {F, G}. Τώρα κοιτάω τη γειτονιά του G και βρίσκω το κοντινότερο CORE point στο G που είναι το H

Άρα έχω C2: {F, G, H} ΚΛΠ

Τελικά θα έχω μετά από αυτή τη διαδικασία:

C1 → {C}

C2 → {F, G, H, I, J}

Αφού κοίταξα όλα τα CORE τώρα πάω στα BORDER points

Τα BORDER τα βάζουμε στο κοντινότερο κλάστερ άρα

B → C1 , D→C1, E→C2

Τα **NOISE points** *όλα* παίρνουν μπούλο εντελώς!

Άρα τελικά τα δύο clusters θα είναι:

C1 → {BCD} και C2→{E, F, G, H, I, J}

ΚΑΝΟΝΕΣ ΣΥΣΧΕΤΙΣΗΣ, APRIORI, LIFT ΚΛΠ

APRIORI: Ξεκινάμε και παίρνουμε τα Ν-στοιχειοσύνολα ξεκινώντας από τα 1-στοιχειοσύνολο, 2-στχσνλ κλπκλπ

Transaction ID	Items
T1	Κραγιόν, Μάσκαρα, Ρουζ
T2	Κραγιόν, Σκιές, Πούδρα
T3	Κραγιόν, Μάσκαρα
T4	Πούδρα, Ρουζ
T5	Πούδρα, Σκιές
T6	Κραγιόν, Σκιές, Πούδρα, Ρουζ

1. Apriori με το χέρι
Εκτελέστε τον αλγόριθμο Apriori πάνω στα παραπάνω δεδομένα καταστήματος καλλυντικών με όρια υποστήριξης s=33% και εμπιστοσύνης c=60%.
Δείξτε τα υποψήφια και τα συχνά στοιχειοσύνολα σε κάθε πέρασμα της βάσης.
Δώστε τη λίστα με τα τελικά συχνά στοιχειοσύνολα και τη λίστα με τους κανόνες συσχέτισης που παράγονται από αυτά ταξινομημένους ως προς την εμπιστοσύνη.
Για τους τελικούς κανόνες συσχέτισης (δηλαδή για αυτούς που έχουν c>=60%) υπολογίστε και το lift.
Ποιον κανόνα θεωρείτε τον πιο ισχυρό και γιατί; Σχολιάστε τα αποτελέσματα σας.

Υπολογίζω για το καθένα τη συχνότητα εμφάνισης δηλαδή το **SUPPORT.** Ο υπολογισμός της συχνότηατς αυτής γίνεται ως εξής:

ΠΧ: Το 1-στοιχειόσυνολο {κραγιόν} εμφανίζεται 4 φορές στα 6 αρχεία άρα **S = 4/6**

Το 2-στοιχειοσύνολο {κραγιόν, σκιές} εμφανίζεται 2 φορές στα 6 αρχεία άρα **S = 2/6** κλπκλπ

Πρέπει να δω ποια είναι **ΣΥΧΝΑ** στοιχειοσύνολα για κάθε Ν-στοιχειοσύνολο που παίρνω δηλαδή πάω πρώτα στα 1-στοιχειοσύνολα και <u>βλέπω ποια είναι συχνά και με βάση αυτά φτιάχνω τα 2-στοιχειοσύνολα</u> κλπκλπ

<u>ΣΥΧΝΟ</u> *λέγεται ενα στοιχειοσύνολο όταν η συχνότητα εμφάνισης S είναι μεγαλύτερη από το min supp που θα μας δίνει στην εκφώνηση (s=33% εκφώνηση)*

Υποψήφια 1-στοιχειοσύνολα

	s
{Κραγιόν}	**4/6**
{Μάσκαρα}	**2/6**
{Πούδρα}	**4/6**
{Ρουζ}	**3/6**
{Σκιές}	**3/6**

Απ' αυτά ΣΥΧΝΑ είναι ΟΛΑ. Γιατί όλα τα κλάσματα είναι *μεγαλύτερα* απο s=33%

Τώρα για να σημιουργήσω τα 2-στοιχειοσύνολα βλέπω τα συχνά 1-στοιχ και δημιουργω τα 2-στοιχ ως εξής:

{Κραγιόν} + {Μάσκαρα} → {Κραγιόν, Μάσκαρα}

{Κραγιόν} + {Πούδρα} → {Κραγιόν, Πούδρα}

{Πούδρα} + {Σκιές} → {Πούδρα, Σκιές} κλπκλπ

Εδώ τα παίρνω όλα με όλα επειδή έτυχε ολα να είναι συχνά. Στην ουσία συνδυάζω ανά δύο

Υποψήφια 2-στοιχειοσύνολα	s
{Κραγιόν, Μάσκαρα}	**2/6**
{Κραγιόν, Πούδρα}	**2/6**
{Κραγιόν, Ρουζ}	**2/6**
{Κραγιόν, Σκιές}	**2/6**
{Μάσκαρα, Πούδρα}	0/6
{Μάσκαρα, Ρουζ}	1/6
{Μάσκαρα, Σκιές}	0/6
{Πούδρα, Ρουζ}	**2/6**
{Πούδρα, Σκιές}	**3/6**
{Ροζ, Σκιές}	1/6

<u>Αυτά με το έντονο είναι τα συχνά!</u>

ΠΧ {Κραγιόν, Μάσκαρα} + {Κραγιόν, Πούδρα} → {Κραγιόν, Μάσκαρα, Πούδρα}

Όμως {Κραγιόν, Μάσκαρα} + {Μάσκαρα, Πούδρα} → ΔΕ ΓΙΝΕΤΑΙ επειδή

{Μάσκαρα, Πούδρα} = <u>ΜΗ ΣΥΧΝΟ</u>

Συνσυάζω πάντα μόνο τα συχνά μεταξύ τους!

Ας πούμε στα 3-στοιχ δε θα συμπεριλάβω κάποιο το οποίο συνδυάζει κάποιο 2-στοιχ ΜΗ ΣΥΧΝΟ. Δηλαδή {Μάσκαρα, Πούδρα} → ΜΗ ΣΥΧΝΟ άρα στα 3-στοιχ δε θα είναι υποψήφιο κάποιο που το περιέχει, πχ {Κραγιόν, Μάσκαρα, Πούδρα} → ΟΧΙ ΥΠΟΨΗΦΙΟ

Υποψήφια 3-στοιχειοσύνολα

	s
{Κραγιόν, Πούδρα, Ρουζ}	1/6
{Κραγιόν, Πούδρα, Σκιές}	**2/6**

Άρα, δεν υπάρχουν συχνά 4-στοιχειοσύνολα.

Και σταματάω στα 3-στοιχειοσύνολα!

Δώστε τη λίστα με τα τελικά συχνά στοιχειοσύνολα και τη λίστα με τους κανόνες συσχέτισης που παράγονται από αυτά ταξινομημένους ως προς την εμπιστοσύνη.

Απλά γράφω όλα τα συχνά που βρήκα

Λίστα με συχνά:

Κραγιόν

Μάσκαρα

Πούδρα

Ρουζ

Σκιές

Κραγιόν, Μάσκαρα

Κραγιόν, Πούδρα

Κραγιόν, Ρούζ

Κραγιόν, Σκιές

Πούδρα, Ρούζ

Πούδρα, Σκιές

Κραγιόν, Πουδρα, Σκιές

Τώρα οι κανόνες συσχέτισης είναι:

Κανόνες συσχέτισης

Μάσκαρα -> Κραγιόν: c = 2/2 = 1 ==ΣΥΧΝΟ

Σκιές-> Πούδρα: c = 3/3 = 1 ==ΣΥΧΝΟ

{Κραγιόν, Πούδρα} -> Σκιές: c = 2/2 = 1 ==ΣΥΧΝΟ

{Κραγιον, Σκιές} -> Πούδρα: c = 2/2 = 1 ==ΣΥΧΝΟ

Πούδρα -> Σκιές: c = 3/4 = 0.75 ==ΣΥΧΝΟ

Σκιές -> Κραγιόν: c = 2/3 = 0.66 == ΣΥΧΝΟ

Ρούζ -> Κραγιόν: c =2/3 = 0.66 ==ΣΥΧΝΟ

{Πούδρα, Σκιές} -> Κραγιον: c = 2/3 = 0.66 == ΣΥΧΝΟ

Ρουζ -> Πούδρα: c = 2/3 = 0.66 ==ΣΥΧΝΟ

Σκιές -> {Κραγιόν, Πούδρα} : c = 2/3 = 0.66 == ΣΥΧΝΟ

Κραγιόν -> Μάσκαρα: c = 2/4 = 0.5

Κραγιόν -> Πούδρα: c = 2/4 = 0.5

Πούδρα -> Κραγιόν: c = 2/4 = 0.5

Κραγιόν ->Ρουζ: c = 2/4 = 0.5

Κραγιόν -> Σκιές: c =2/4 = 0.5

Πούδρα -> Ρουζ: c = 2/4 = 0.5

Κραγιόν -> Πούδρα, Σκιες: c = 2/4 = 0.5

Πούδρα -> Κραγιόν, Σκιές: c = 2/4 = 0.5

Ξεκινάω απο τα ΣΥΧΝΑ 2-στοιχ και βρίσκω το C=confidence που βγαίνει ως εξής:

ΠΧ: Μάσκαρα → Κραγιόν:

C = (Το S του 2-στοιχ που τα περιλαμβάνει) / (το S του αριστερού μέλους)

= (2/6) / (2/6) = 2/2 = 1 ΚΑΙ ΣΥΧΝΟΣ

{Κραγιόν, Σκιές} → Πούδρα: C = (2/6) / (2/6) = 1 ΚΑΙ ΣΥΧΝΟΣ

ΣΥΝΧΟΣ ένας κανόνας είναι όταν το κλάσμα βγαίνει μεγαλύτερο από ένα C που θα δίνει στην εκφώνηση (στο παράδειγμα αυτό το C>=60%)

Για τους τελικούς κανόνες συσχέτισης (δηλαδή για αυτούς που έχουν c>=60%) υπολογίστε και το lift.

Τελικοί κανόνες είναι προφανώς οι ΣΥΧΝΟΙ που βρήκαμε από πριν

LIFTS

Μάσκαρα -> Κραγιόν: Lift = 1/(4/6) = 1/0.66 = 1.52

Ρουζ -> Κραγιόν: Lift = 0.66/(4/6) = 0.66/0.66 = 1

Σκιές -> Κραγιόν: Lift = 0.66/(4/6) = 0.66/0.66 = 1

Ρουζ -> Πούδρα: Lift = 0.66/(4/6) = 0.66/.066 = 1

Πούδρα -> Σκιές: Lift = 0.75/(3/6) = 0.75/0.5 = 1.5

Σκιές-> Πούδρα: Lift = 1/(4/6) = 1/0.66 = 1.5

{Κραγιόν, Πούδρα} -> Σκιές: Lift = 1/(3/6) = 1/0.5 = 2

{Κραγιόν, Σκιές} -> Πούδρα: Lift = 1/(4/6) = 1/0.66 = 1.5

{Πούδρα, Σκιές} -> Κραγιόν: Lift = 0.66/(4/6) = 0.66/0.66 = 1

Σκιές -> {Κραγιόν, Πούδρα} : Lift = 0.66/(2/6) = 0.66/0.33 = 2

Ο τύπος του LIFT είναι LIFT: (το C που βρήκαμε πριν) / (το S του δεξιού μέλους)

Ποιον κανόνα θεωρείτε τον πιο ισχυρό και γιατί; Σχολιάστε τα αποτελέσματα σας.

Παρατηρώ ότι ο ισχυρότερος κανόνας συσχέτισης είναι ο ***{Κραγιόν, Πούδρα} -> Σκιές*** διότι όχι μόνο έχει απ'τα μεγαλύτερα Confidence (C=1) αλλά και το Lift του είναι απ τα μεγαλύτερα (Lift=2)

ΚΑΤΗΓΟΡΙΟΠΟΙΗΣΗ ΚΕΙΜΕΝΟΥ, NAIVE BAYES ΚΛΠ

NAIVE BAYES MULTINOMIAL:

	docID	Λέξεις του εγγράφου	τηλεόραση
Σύνολο εκπαίδευσης	1	πρόγραμμα πρόγραμμα επεισόδιο σειρά κανάλι	ναι
	2	πρόγραμμα επεισόδιο ταινία επεισόδιο	ναι
	3	επεισόδιο κανάλι επεισόδιο κανάλι ειδήσεις	ναι
	4	επεισόδιο γήπεδο ομάδα	όχι
	5	γήπεδο ομάδα ειδήσεις	όχι
Σύνολο ελέγχου	6	επεισόδιο επεισόδιο γήπεδο γήπεδο γήπεδο ειδήσεις	

Για αρχή βρίσκω το P της κλάσης:

P(Τηλ) = 3/5

P(όχιΤηλ) = 2/5

Αυτό προφανώς βγαίνει ως εξής:

P(Τηλ) = (πλήθος εγγράφων όπου Τηλ = ΝΑΙ) / (πλήθος εγγράφων)

P(όχιΤηλ) = (πλήθος εγγράφων όπου Τηλ = ΟΧΙ) / (πλήθος εγγράφων)

Τώρα κοιτάω ποιες λέξεις περιλαμβάνει το ζητούμενο και μόνο γι' αυτές βρίσκω τα P:

P(επεισόδιο | Τηλεόραση) = (5+1) / (14+8) = 6 / 22 = **3 / 11**

P(επεισόδιο | όχιΤηλεόραση) = (1+1) / (6+8) = 2 / 14 = **1 / 7**

Ο τύπος είναι:

Πχ P(Επεισ | Τηλ) =

Αριθμητής → (Πόσες φορές υπάρχει η λέξη Επεισόδιο στα αρχεία όπου Τηλ = ΝΑΙ) **+ 1**

Παρονομ →(πλήθος λέξεων όπου Τηλ = ΝΑΙ) + (πλήθος λέξεων ΜΙΑ ΦΟΡΑ Η ΚΑΘΕΜΙΑ και ΣΕ ΟΛΑ ΤΑ ΕΓΓΡΑΦΑ ΕΙΤΕ **Τηλ** ΕΙΤΕ **όχιΤηλ**)

Άρα: P(επεισόδιο | Τηλεόραση) = (5+1) / (14+8) = 6 / 22 = **3 / 11**

Τα υπόλοιπα είναι:

P(γήπεδο | Τηλεόραση) = (0+1) / (14+8) = **1 / 22**

P(γήπεδο | όχιΤηλεόραση) = (2+1) / (6+8) = **3 / 14**

P(ειδήσεις | Τηλεόραση) = (1+1) / (14+8) = 2 / 22 = **1 / 11**

P(ειδήσεις | όχιΤηλεόραση) = (1+1) / (6+8) = 2 / 14 = **1 / 7**

Επομένως έχω:

$P(\text{Τηλεόραση} \mid d6) = 3/5 * (3/11)^2 * (1/22)^3 * (1/11) \approx 0.0000004$

Ο τύπος είναι:

$$P(\text{Τηλ} \mid d6) = P(\text{Τηλ}) * P(\text{Επεισ} \mid \text{Τηλ})^2 * P(\text{γήπεδο} \mid \text{Τηλ})^3 * P(\text{ειδήσεις} \mid \text{Τηλ})$$

Εξηγώ:

Τα ^{2} και ^{3} είναι αναλόγα το πόσες φορές βρίσκεται στο υποψήφιο αρχείο η λέξη που ψάχνω δηλαδή Επεισόδιο → 2 φορές ΑΡΑ εις το τετράγωνο και Γήπεδο → 3 φορές ΑΡΑ εις τον κύβο και προφανώς Ειδήσεις → 1 φορά

Για όχιΤηλ έχω:

$$P(\text{όχιΤηλεόραση} \mid d6) = 2/5 * (1/7)^2 * (3/14)^3 * (1/7) \approx 0.00001$$

Συμπεραίνουμε ότι το *d6* ανήκει στην κατηγορία **<u>όχιΤηλεόραση</u>** εφόσον

<u>P(όχιΤηλεόραση | d6) > P(Τηλεόραση | d6)</u>

NAIVE BAYES BERNOULLI

Εδώ θα βρω τύπους για ΟΛΕΣ ΤΙΣ ΛΕΞΕΙΣ και όχι μόνο για τις λέξεις του υποψήφιου εγγράφου

P(Τηλεόραση) = **3 / 5**

P(όχιΤηλεόραση) = **2 / 5**

P(επεισόδιο | Τηλεόραση) = (3+1) / (3+2) = **4 / 5**

P(επεισόδιο | όχιΤηλεόραση) = (1+1) / (2+2) = 2 / 4 = **1 / 2**

P(γήπεδο | Τηλεόραση) = (0+1) / (3+2) = **1 / 6**

P(γήπεδο | όχιΤηλεόραση) = (2+1) / (2+2) = **3 / 4**

P(ειδήσεις | Τηλεόραση) = (1+1) / (3+2) = **2 / 5**

P(ειδήσεις | όχιΤηλεόραση) = (1+1) / (2+2) = 2 / 4 = **1 / 2**

P(πρόγραμμα | Τηλεόραση) = (2+1) / (3+2) = **3 / 5**

P(πρόγραμμα | όχιΤηλεόραση) = (0+1) / (2+2) = **1 / 4**

P(σειρά | Τηλεόραση) = (1+1) / (3+2) = **2 / 5**

P(σειρά | όχιΤηλεόραση) = (0+1) / (2+2) = **1 / 4**

P(ομάδα | Τηλεόραση) = (0+1) / (3+2) = **1 / 5**

P(ομάδα | όχιΤηλεόραση) = (2+1) / (2+2) = **3 / 4**

P(κανάλι | Τηλεόραση) = (3+1) / (3+2) = **4 / 5**

P(κανάλι | όχιΤηλεόραση) = (0+1) / (2+2) = **1 / 4**

P(ταινία | Τηλεόραση) = (1+1) / (3+2) = **2 / 5**

P(ταινία | όχιΤηλεόραση) = (0+1) / (2+2) = **1 / 4**

Εξηγώ:

P(Επεισ | Τηλ):

Αριθμητής: (Πλήθος εγγράφων που εμφανίζ η λέξη Επεισόδιο ΧΩΡΙΣ ΔΙΠΛΟΤΥΠΑ) **+ 1**

Παρονομ: (Πλήθος Τηλ = ΝΑΙ εγγράφων) **+ 2**

Ομοίως και για τα όχιΤηλ κλπκλπ

Επομένως προκύπτει,

<u>P(τηλεόρασηεόραση|d6)</u> = P(Τηλεόραση) * P(επεισόδιο|Τηλεόραση) * P(γήπεδο|Τηλεόραση) * P(ειδήσεις|Τηλεόραση) * (1- P(προγραμμα|Τηλεόραση)) * (1- P(σειρά|Τηλεόραση)) * (1- P(ομάδα|Τηλεόραση)) * (1- P(κανάλι|Τηλεόραση)) * (1- P(ταινία|Τηλεόραση)) = 3/5 * 4/5 * 1/5 * 2/5 * 2/5 * 3/5 * 4/5 * 2/5 * 3/5 = ***<u>0.002</u>***

<u>P(όχιΤηλεόρασηεόραση|d6)</u> = P(όχιΤηλεόραση)*P(επεισόδιο|όχιΤηλεόραση)*P(γήπεδο|όχιΤηλεόραση) * P(ειδήσεις|όχιΤηλεόραση) *(1-P(προγραμμα|όχιΤηλεόραση)) * (1-P(σειρά|όχιΤηλεόραση)) * (1- P(ομάδα|όχιΤηλεόραση)) * (1- P(κανάλι|όχιΤηλεόραση)) * (1- P(ταινία|όχιΤηλεόραση)) = 2/5 * 1/2 * 3/4 * 1/2 * 3/4 * 3/4 * 1/4 * 3/4 * 3/4 = ***<u>0.006</u>***

Εξηγώ:

Παίρνω τον πρώτο τύπο: (Που έχει Τηλ = ΝΑΙ)

Είναι επί της ουσίας το P(Τηλ) * (τα P για τις λέξεις που υπάρχουν στο υποψήφιο έγγραφο→Για μας είναι Επεισ, Γήπεδο, ειδήσεις και προφανώς για την κατηγορία που ψάχνω με αυτόν τον τύπο → Για Τηλ=ΝΑΙ) * (1 - Το P των λέξεων που ΔΕΝ βρίσκονται στο υποψήφιο έγγραφο για Τηλ =ΝΑΙ). *Τους έχουμε όλους στην προηγούμενη σελίδα!*

Άρα έχω: 3/5 * **4/5 * 1/5 * 2/5** * <u>2/5 * 3/5 * 4/5 * 2/5 * 3/5</u>

Με **έντονα** είναι αυτά που υπάρχουν στο υποψήφιο

Με <u>υπογράμμιση</u> είναι τα 1 – κάτι

Ομοίως για Τηλ = ΟΧΙ

Αυτά!

Ο ΘΕΟΣ ΜΑΖΙ ΜΑΣ ΚΑΡΔΟΥΛΕΣ ΜΟΥ