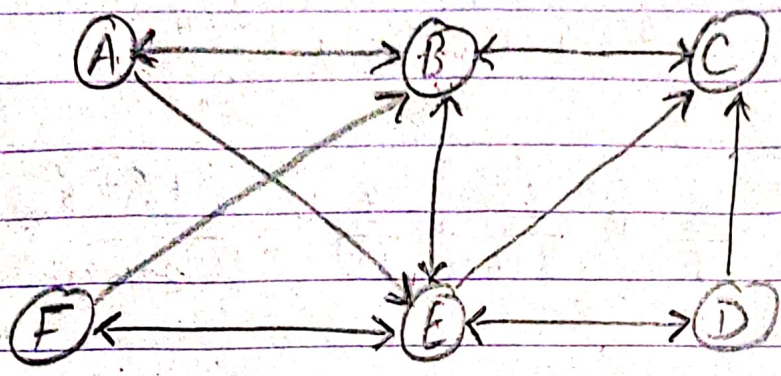
# * Page Rank Algorithm:-




upto round 2.

Formula: $$PR(u) = \sum_{V \in B_u} \frac{PR(V)}{L(V)}$$

Initially, $PR(A) = PR(B) = PR(C) = PR(D) = PR(E) = PR(F) = 1/n = 1/6$

where, $n$ = number of nodes.

Iteration : 1

Inbound links / outbound links [Links from page V]

$$PR(A) = \frac{PR(B)}{L(B)} = \frac{1/6}{3} = \frac{1}{18} = 0.055$$

$$PR(B) = \frac{PR(A)}{L(A)} + \frac{PR(C)}{L(C)} + \frac{PR(E)}{L(E)} + \frac{PR(F)}{L(F)}$$

$$= \frac{1/6}{2} + \frac{1/6}{1} + \frac{1/6}{4} + \frac{1/6}{2}$$

$$= \frac{1}{12} + \frac{1}{6} + \frac{1}{24} + \frac{1}{12} = \frac{2+4+1+2}{24} = \frac{9}{24}$$

$$= 0.375$$

$$PR(C) = \frac{PR(B)}{L(B)} + \frac{PR(E)}{L(E)} + \frac{PR(D)}{L(D)}$$

$$= \frac{1/6}{3} + \frac{1/6}{4} + \frac{1/6}{2}$$

$$= \frac{1}{18} + \frac{1}{24} + \frac{1}{12} = \frac{4+3+6}{72} = \frac{13}{72}$$

$$= 0.18$$

$$PR(D) = \frac{PR(E)}{L(E)}$$

$$= \frac{1/6}{4} = 1/24 = 0.041$$

$$PR(E) = \frac{PR(A)}{L(A)} + \frac{PR(B)}{L(B)} + \frac{PR(D)}{L(D)} + \frac{PR(F)}{L(F)}$$

$$= \frac{1/6}{2} + \frac{1/6}{3} + \frac{1/6}{2} + \frac{1/6}{2}$$

$$= \frac{1}{12} + \frac{1}{18} + \frac{1}{12} + \frac{1}{12}$$

$$= \frac{3+2+3+3}{36} = \frac{11}{36} = 0.305$$

$$PR(F) = \frac{PR(E)}{L(E)}$$

$$= \frac{1/6}{4}$$

$$= \frac{1}{24} = 0.0416$$

Iteration: 2

$$PR(A) = \frac{PR(B)}{L(B)} = \frac{0.375}{3} = 0.125$$

$$PR(B) = \frac{PR(A)}{L(A)} + \frac{PR(C)}{L(C)} + \frac{PR(E)}{L(E)} + \frac{PR(F)}{L(F)}$$

$$= \frac{0.055}{2} + \frac{0.18}{1} + \frac{0.305}{4} + \frac{0.0416}{2}$$

$$= ?$$

$$PR(C) = \frac{PR(B)}{L(B)} + \frac{PR(E)}{L(E)} + \frac{PR(D)}{L(D)}$$

$$= \frac{0.375}{3} + \frac{0.305}{4} + \frac{0.041}{2}$$

$$= ?$$

$$PR(D) = \frac{PR(E)}{L(E)} = \frac{0.305}{4} = 0.076$$

$$PR(E) = \frac{PR(A)}{L(A)} + \frac{PR(B)}{L(B)} + \frac{PR(D)}{L(D)} + \frac{PR(F)}{L(F)}$$

$$= \frac{0.055}{2} + \frac{0.375}{3} + \frac{0.041}{2} + \frac{0.0416}{2} = ?$$

$$PR(F) = \frac{PR(E)}{L(E)} = \frac{0.305}{4} = 0.076$$

## * HITS Algorithm :

'Hypertext Induced Topic Search'

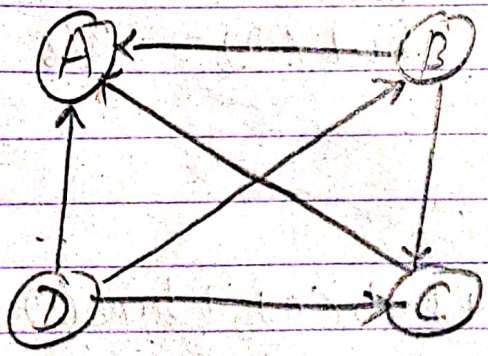
Formula;

Authority Update Rule:

- Each node's authority score is the sum of hub scores of each node that points to it. (Inbound links)

Hub Update Rule:

- Each node's hub score is the sum of authority scores of each node that it points to. (outbound links)




Initially;

✗ auth(A) = auth(B) = auth(C) = auth(D) = 1
hub(A) = hub(B) = hub(C) = hub(D) = 1

Iteration: 1

auth(A) = hub(B) + hub(C) + hub(D) = 3
auth(B) = hub(D) = 1
auth(C) = hub(B) + hub(D) = 2
auth(D) = 0

hub(A) = 0
hub(B) = auth(A) + auth(C) = 2
hub(C) = auth(A) = 1
hub(D) = auth(A) + auth(B) + auth(C) = 3

After applying normalization:

auth(A) = 3/6
auth(B) = 1/6
auth(C) = 2/6
auth(D) = 0

hub(A) = 0
hub(B) = 2/6
hub(C) = 1/6
hub(D) = 3/6

Iteration: 2

auth(A) = hub(B) + hub(C) + hub(D)

$= 2/6 + 1/6 + 3/6 = \frac{6}{6} = 1$

auth(B) = hub(D) = 3/6

$auth(C) = hub(B) + hub(D) = \frac{2}{6} + \frac{3}{6} = \frac{5}{6}$

auth(D) = 0

hub(A) = 0

hub(B) = auth(A) + auth(C)

$= \frac{3}{6} + \frac{2}{6} = \frac{5}{6}$

hub(C) = auth(A) = 3/6

hub(D) = auth(A) + auth(B) + auth(C)

$= \frac{3}{6} + \frac{1}{6} + \frac{2}{6} = \frac{6}{6} = 1$

After applying normalization:

$auth(A) = \frac{1}{1 + 3/6 + 5/6 + 0} = \frac{6}{14}$

$auth(B) = \frac{3}{6} \times \frac{6}{14} = \frac{3}{14}$

$auth(C) = \frac{5}{6} \times \frac{6}{14} = \frac{5}{14}$

auth(D) = 0

Now, hub(A) = 0

$hub(B) = \frac{5}{6} \times \frac{6}{14} = \frac{5}{14}$

$hub(C) = \frac{3}{6} \times \frac{6}{14} = \frac{3}{14}$

$hub(D) = \frac{6}{14}$

## * Collaborative Filtering Recommender Systems:-

Similarity functions:

1. $$\text{Pearson}(x,y) = \frac{\sum (x-\bar{x})(y-\bar{y})}{\sqrt{\sum (x-\bar{x})^2}\sqrt{\sum (y-\bar{y})^2}}$$

2. $$\text{Cosine}(x,y) = \frac{\sum xy}{\sqrt{\sum x^2}\sqrt{\sum y^2}}$$

(User-based)

Example 1; Suppose the ratings of five users U1, U2, U3, U4 and U5 are drawn for six items denoted by I1, I2, I3, I4, I5 and I6. Each rating is drawn from the range ~~basis~~ of {1, ... 7}.

Find the predictions of user U3 for items I1 and I6 on the basis of the ratings in the table below. Use pearson correlation coefficient to find similarity between users and consider top-2 closest user.

Row-wise Row-wise

| User/Item | I1 | I2 | I3 | I4 | I5 | I6 |
|---|---|---|---|---|---|---|
| U1 | 7 | 6 | 7 | 4 | 5 | 4 |
| U2 | 6 | 7 | ? | 4 | 3 | 4 |
| U3 | ? | 3 | 3 | 1 | 1 | ? |
| U4 | 1 | 2 | 2 | 3 | 3 | 4 |
| U5 | 1 | ? | 1 | 2 | 3 | 3 |

Soln:

The first step is to compute the similarity between user U3 and all the other users.

$$\text{Pearson}(U1,U3) = \frac{(6-5.5)(3-2)+(7-5.5)(3-2)+(4-5.5)(1-2)+(5-5.5)(1-}{\sqrt{1.5^2+1.5^2+(-1.5)^2+(-0.5)^2}\sqrt{1^2+1^2+(-1)^2+(-1)^2}}$$

$$= \frac{4}{\sqrt{7}\times\sqrt{4}} = \frac{4}{2.64\times 2} = 0.756$$

Mean
$33/6 = 5.5$

4.8

$8/4 = 2$

2.5

2

Similarly;

$$\text{Pearson}(U2, U3) = \frac{(7-4.8)(3-2)+(4-4.8)(1-2)+(3-4.8)(1-2)}{\sqrt{2.2^2+(-0.8)^2+(-1.8)^2} \times \sqrt{1^2+(-1)^2+(-1)^2}}$$

$$= \frac{4.8}{\sqrt{8.72} \times \sqrt{3}} = 0.938$$

$$\text{Pearson}(U4, U3) = \frac{(2-2.5)(3-2)+(2-2.5)(3-2)+(3-2.5)(1-2)+(3-2.5)(1-2)}{\sqrt{(-0.5)^2+(-0.5)^2+(0.5)^2+(0.5)^2} \times \sqrt{1^2+1^2+(-1)^2+(-1)^2}}$$

$$= \frac{-2}{\sqrt{1} \times \sqrt{4}} = -\frac{2}{2} = -1$$

$$\text{Pearson}(U5, U3) = \frac{(1-2)(3-2)+(2-2)(1-2)+(3-2)(1-2)+\cdots}{\sqrt{(-1)^2+1^2} \times \sqrt{1^2+(-1)^2+(-1)^2+\cdots}}$$

$$= \frac{-2}{\sqrt{2} \times \sqrt{3}} = -0.816$$

Hence, the top-2 closest users to user U3 are users U1 and U2 according to pearson correlation coefficient.

By using the pearson - weighted average of the raw ratings

Rank (U3, I1)

$$= \frac{(7 * 0.756 + 6 * 0.938)}{(0.756 + 0.938)} = 6.446$$

Rank (U3, I6)

$$= \frac{(4 * 0.756 + 4 * 0.938)}{(0.756 + 0.938)} = 4$$

Cosine similary;

$$\text{Cosine}(U1, U3) = \frac{6 \times 3 + 7 \times 3 + 4 \times 1 + 5 \times 1}{\sqrt{6^2 + 7^2 + 4^2 + 5^2} \times \sqrt{3^2 + 3^2 + 1^2 + 1^2}}$$

$$= \frac{48}{50.199} = 0.9561$$

(Item based)

Example 2; Column-wise

Suppose the ratings of five users U1, U2, U3, U4 and U5 rating is drawn from the range $\{1, \ldots 7\}$. Find the predictions of Item I1 for user U3 on the basis of the ratings in the table below. Use pearson Correlation Coefficient to find Similarity between users and Consider top-2 closest items.

| User/Item | I1 | I2 | I3 | I4 | I5 | I6 |
|---|---|---|---|---|---|---|
| U1 | 7 | 6 | 7 | 4 | 5 | 4 |
| U2 | 6 | 7 | ? | 4 | 3 | 4 |
| U3 | ? | 3 | 3 | 1 | 1 | ? |
| U4 | 1 | 2 | 2 | 3 | 3 | 4 |
| U5 | 1 | ? | 1 | 2 | 3 | 3 |
| mean | 3.75 | 4.5 | 3.25 | 2.8 | 3 | 3.75 |

Soln. The first step is to Compute the Similarity between item I1 and all the other items.

Pearson (I1, I2) =

$$= \frac{(7-3.75)(6-4.5) + (6-3.75)(7-4.5) + (1-3.75)(2-4.5)}{\sqrt{3.25^2 + 2.25^2 + (-2.75)^2} \times \sqrt{1.5^2 + 2.5^2 + (-2.5)^2}}$$

$$= \frac{17.375}{18.493} = 0.94$$

Pearson (I1, I3) =

$$= \frac{(7-3.75)(7-3.25) + (1-3.75)(2-3.25) + (1-3.75)(1-3.25)}{\sqrt{3.25^2 + (-2.75)^2 + (-2.75)^2} \times \sqrt{3.75^2 + (-1.25)^2 + (-2.25)^2}}$$

$$= \frac{21.8125}{21.052} = 0.946$$

Pearson (I1, I4) =

## * Content - based Recommender systems:

Example; Find the predictions of movie M4 for user U1 on the basis of the information given in the table below.

| Movie/User | U1 | U2 | U3 | U4 | U5 | U6 | Genre |
|---|---|---|---|---|---|---|---|
| M1 | 1 | 1 | ? | 1 | 1 | ? | Romance |
| M2 | 1 | 1 | 1 | 1 | 1 | ? | Thriller |
| M3 | 1 | 1 | 1 | 1 | 1 | 1 | Action |
| M4 | ? | 1 | 1 | 1 | 1 | 1 | Romance |
| M5 | 1 | 1 | 1 | 1 | 1 | ? | Crime |
| M6 | 1 | 1 | 1 | 1 | 1 | 1 | Crime |
| Watched | 5 | 6 | 5 | 6 | 6 | 5 | |

Item profile generation:

| Movie/Genre | Romance | Thriller | Action | Crime |
|---|---|---|---|---|
| M1 | 1 | 0 | 0 | 0 |
| M2 | 0 | 1 | 0 | 0 |
| M3 | 0 | 0 | 1 | 0 |
| M4 | 1 | 0 | 0 | 0 |
| M5 | 0 | 0 | 0 | 1 |
| M6 | 0 | 0 | 0 | 1 |

## User profile generation:

| User/Genre | Romance | Thriller | Action | Crime |
|---|---|---|---|---|
| U1 | 1/5 = 0.2 | 0.2 | 0.2 | 2/5 = 0.4 |
| U2 | 0.33 | 0.17 | 0.17 | 0.33 |
| U3 | 0.2 | 0.2 | 0.2 | 0.4 |
| U4 | 0.33 | 0.17 | 0.17 | 0.33 |
| U5 | 0.33 | 0.17 | 0.17 | 0.33 |
| U6 | 0.33 | 0 | 0.33 | 0.33 |

$\therefore$ Sim (m4, U1) $\quad X = [1, 0, 0, 0]$

= Cosine (m4, U1) $\quad Y = [0.2, 0.2, 0.2, 0.4]$

$$= \frac{\sum xy}{\sqrt{\sum x^2} \times \sqrt{\sum y^2}}$$

$$= \frac{1 \times 0.2 + 0 \times 0.2 + 0 \times 0.2 + 0 \times 0.4}{\sqrt{1^2} \quad \sqrt{0.2^2 + 0.2^2 + 0.2^2 + 0.4^2}}$$

$$= \frac{0.2}{\sqrt{0.2^2 + 0.2^2 + 0.2^2 + 0.4^2}} = 0.38$$

## * Mining Frequent Patterns:

1. Apriori
2. FP growth

Support = $\frac{\text{Frequency}}{\text{Transaction}} \times 100$

1. Apriori Algorithm  Confidence = $\frac{S(A \cup B)}{S(A)}$

Eg:

| TID | Items |
|---|---|
| 100 | 1 3 4 |
| 200 | 2 3 5 |
| 300 | 1 2 3 5 |
| 400 | 2 5 |

Min Support = 50 %.
Threshold Confidence = 70 %.

| Itemset | Support (Count) / Frequency |
|---|---|
| 1 | 2/4 → 50 %. |
| 2 | 3/4 → 75 %. |
| 3 | 3/4 → 75 %. |
| 4 | 1/4 → 25 %. X |
| 5 | 3/4 → 75 %. |

Itemset → 1, 2, 3, 5

| Itemset | Support | |
|---|---|---|
| {1,2} | 1/4 → 25% | X |
| {1,3} | 2/4 → 50% | |
| {1,5} | 1/4 → 25% | X |
| {2,3} | 2/4 → 50% | |
| {2,5} | 3/4 → 75% | |
| {3,5} | 2/4 → 50% | |

| Itemset | Support | |
|---|---|---|
| {1,3,5} | 1/4 → 25% | X |
| {2,3,5} | 2/4 → 50% | |
| {1,2,3} | 1/4 → 25% | X |

| Rules | Support | Confidence |
|---|---|---|
| (2 ∧ 3) → 5 | 2 | 2/2 = 100% |
| (3 ∧ 5) → 2 | 2 | 2/2 = 100% |
| (2 ∧ 5) → 3 | 2 | 2/3 = 66% |
| 2 → (3 ∧ 5) | 2 | 2/3 = 66% |
| 5 → (2 ∧ 3) | 2 | 2/3 = 66% |
| 3 → (2 ∧ 5) | 2 | 2/3 = 66% |

Eg: Confidence = $S(A \cup B) / S(A)$

$(2 \wedge 3) \rightarrow 5 = S((2 \wedge 3) \cup 5) / S(2 \wedge 3)$

A B

$= 2/2 = 100\%$

The association rules will be;

$(2 \wedge 3) \rightarrow 5$ & $(3 \wedge 5) \rightarrow 2$ Ans

2. FP growth Algorithm:

Example)

| TID | Items |
|---|---|
| 1 | A, B, C, D, E, F |
| 2 | B, C, D, E, F, G |
| 3 | A, D, E, H |
| 4 | A, D, F, I, J |
| 5 | B, D, E, K |

min. support = 60 %.
i.e. support Count = 60/100 * 5 = 3
Confidence = 80 %.

Step 1: 1-item set

| Item | Count | Item | Count |
|---|---|---|---|
| A | 3 | H | 1 |
| B | 3 | I | 1 |
| C | 2 | J | 1 |
| D | 5 | K | 1 |
| E | 4 | | |
| F | 3 | | |
| G | 1 | | |

Step: 2 1-item frequent set

| Item | Count |
|---|---|
| A | 3 |
| B | 3 |
| D | 5 |
| E | 4 |
| F | 3 |

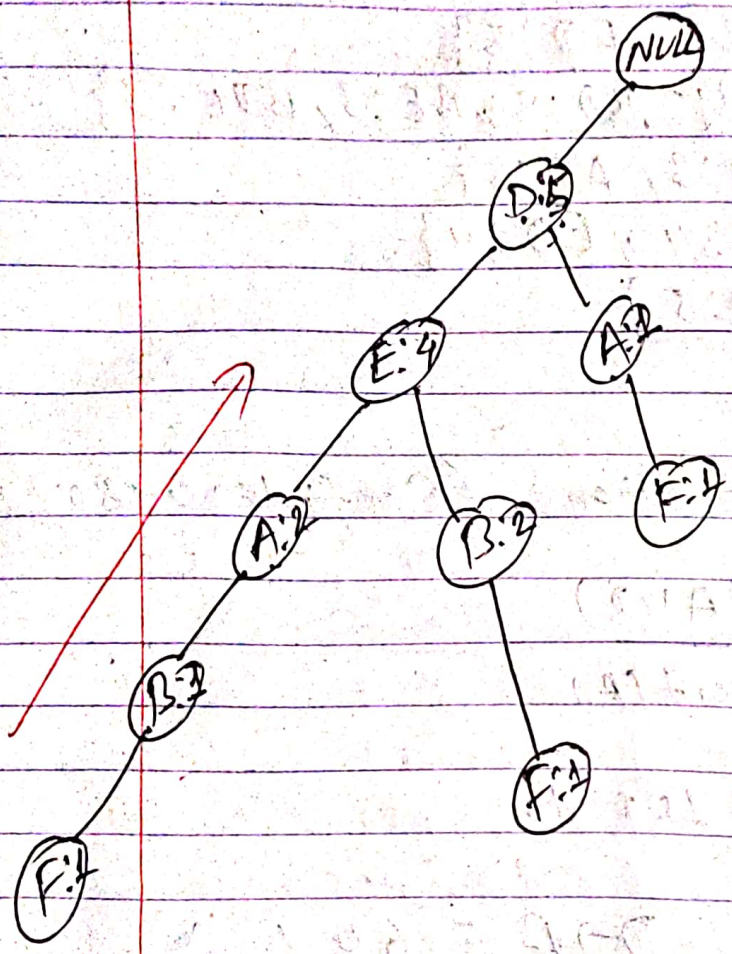
Step: 3 Reording 1-item frequent set in descending order w.r.t. count value.

| Item | Count |
|---|---|
| D | 5 |
| E | 4 |
| A | 3 |
| B | 3 |
| F | 3 |

Step: 4 Reording original itemset in descending order w.r.t count value.

| TID | Items |
|---|---|
| 1 | D, E, A, B, F |
| 2 | D, E, B, F |
| 3 | D, E, A |
| 4 | D, A, F |
| 5 | D, E, B |

## Step 5: Construction of FP Tree




## Step 6: Generation of Frequent Patterns w.r.t Conditional pattern base.

| Item | Conditional Pattern Base | Conditional FP (count) Tree |
|---|---|---|
| F | {{D,E,A,B:1},{D,E,B:1}, {D,A:1}} | {D:3, E:2, A:2, B:2} |
| B | {{D,E,A:1},{D,E:2}} | {D:3, E:3, A:1} |
| A | {{D,E:2},{D:1}} | {D:3, E:2} |
| E | {D:4} | {D:4} |
| D | — | |

| Items | Frequent Patterns generated |
|---|---|
| F | {F:3, FD:3} |
| B | {B:3, BD:3, BE:3, BDE:3} |
| A | {A:3, AD:3} |
| E | {E:4, ED:4} |
| D | {D:5} |

Step 7: Rules generation (Confidence = 80%)

$$A \rightarrow D = \frac{\text{Support}(A \cup D)}{\text{Support}(A)}$$

$$= 3/3 = 1 = 100\%$$

$B \rightarrow D = 100\%$ $\quad$ $D \rightarrow B = 60\%$ ✗
$B \rightarrow E = 100\%$ $\quad$ $E \rightarrow B = 75\%$ ✗
$D \rightarrow E = 80\%$ $\quad$ $E \rightarrow D = 100\%$
$D \rightarrow F = 60\%$ ✗ $\quad$ $F \rightarrow D = 100\%$

$B \rightarrow DE = 100\%$ $\quad$ $DE \rightarrow B = 100\%$
$D \rightarrow BE = 60\%$ ✗ $\quad$ $BE \rightarrow D = 100\%$
$E \rightarrow BD = 75\%$ ✗ $\quad$ $BD \rightarrow E = 100\%$

$$E.D = \sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$$

## * K-means

Divide the data points of {(1,1), (2,1), (4,3), (5,4)} into two clusters.

Sol^n:

Let $P1 = (1,1)$, $P2 = (2,1)$
$P3 = (4,3)$, $P4 = (5,4)$

Let $C1 = (1,1)$ and $C2 = (2,1)$ are two initial cluster center.

Iteration: 1 Distance bet^n cluster center and each data points are;

$d(C1,P1) = 0$ ✓
$d(C1,P2) = 1$
$d(C1,P3) = 3.6$
$d(C1,P4) = 5$

$d(C2,P1) = 1$
$d(C2,P2) = 0$ ✓
$d(C2,P3) = 2.83$ ✓
$d(C2,P4) = 4.24$ ✓

Thus, after first iteration

cluster 1 = {P1}, cluster 2 = {P2, P3, P4}

Iteration: 2

New clusters centers are: C1 (1,1) and C2 (11/3, 8/3).

Now, Calculate distance bet^n new cluster centers and each data points.

$d(c_1, p_1) = 0$ ✓ $d(c_2, p_1) = 3.14$

$d(c_1, p_2) = 1$ ✓ $d(c_2, p_2) = 2.35$

$d(c_1, p_3) = 3.6$ $d(c_2, p_3) = 0.46$ ✓

$d(c_1, p_4) = 5$ $d(c_2, p_4) = 1.88$ ✓

Thus, after second iteration;

Cluster 1 = $\{p_1, p_2\}$, Cluster 2 = $\{p_3, p_4\}$.

Iteration: 3 New cluster centers are:

$c_1 (3/2, 1)$ and $c_2 = (9/2, 7/2)$

$d(c_1, p_1) = 0.5$ ✓ $d(c_2, p_1) = 5.54$

$d(c_1, p_2) = 0.5$ ✓ $d(c_2, p_2) = 4.33$

$d(c_1, p_3) = 4.5$ $d(c_2, p_3) = 0.70$ ✓

$d(c_1, p_4) = 5.5$ $d(c_2, p_4) = 0.70$ ✓

Thus, after third iteration.

Cluster 1 = $\{p_1, p_2\}$

Cluster 2 = $\{p_3, p_4\}$

## * Robust clustering algorithm for categorical attributes (ROCK):

- $Sim(P_i, P_j) = \frac{|P_i \cap P_j|}{|P_i \cup P_j|}$, $\theta = 0.3$ (Threshold)

Number of clusters = 2

- $g(P_i, P_j) = \frac{Link[P_i, P_j]}{(n+m)^{1+2f(\theta)} - n^{1+2f(\theta)} - m^{1+2f(\theta)}}$

- $f(\theta) = \frac{1-\theta}{1+\theta}$

Example;

$P_1 = \{Judgement, faith, prayer, fair\}$

$P_2 = \{fasting, faith, prayer\}$

$P_3 = \{fair, fasting, faith\}$

$P_4 = \{fasting, prayer, pilgrimage\}$

Step 1: Similarity Table: (0-1)

| | P1 | P2 | P3 | P4 |
|---|---|---|---|---|
| P1 | 1 | 2/5 = 0.4 | 2/5 = 0.4 | 1/6 = 0.17 |
| P2 | | 1 | 2/4 = 0.5 | 2/4 = 0.5 |
| P3 | | | 1 | 1/5 = 0.2 |
| P4 | | | | 1 |

Taking Threshold, $\theta = 0.3$

Now,

Step 2: Adjacency Table ($\theta = 0.3$)

| | P1 | P2 | P3 | P4 |
|---|---|---|---|---|
| P1 | 1 | 1 | 1 | 0 |
| P2 | 1 | 1 | 1 | 1 |
| P3 | 1 | 1 | 1 | 0 |
| P4 | 0 | 1 | 0 | 1 |

(Consist of 0's & 1's) Similarity $\geq \theta = 0.3$
$\Rightarrow$ 1 otherwise 0.

Step 3: Link Table (A * A)
(Row * column) multiplication

| | P1 | P2 | P3 | P4 |
|---|---|---|---|---|
| P1 | — | 3 | 3 | 1 |
| P2 | | — | 3 | 2 |
| P3 | | | — | 1 |
| P4 | | | | — |

| Pair | Goodness measure | |
|---|---|---|
| {P1, P2} | 1.3825 | Search for highest values. |
| {P1, P3} | 1.3825 | |
| {P1, P4} | 0.46 | |
| {P2, P3} | 1.3825 | |
| {P2, P4} | 0.92 | |
| {P3, P4} | 0.46 | |

$$g(P_1, P_2) = \frac{Link(P_1, P_2)}{(n+m)^{1+2f(\theta)} - n^{1+2f(\theta)} - m^{1+2f(\theta}}$$

Assuming $f(\theta) = \frac{1-\theta}{1+\theta} = \frac{1-0.3}{1+0.3} = 0.538$

$$= \frac{3}{(1+1)^{1+2\times 0.53} - 1 - 1}$$

$$= \frac{3}{2^{2.06} - 2} = 1.3825$$

For (P1, P3);

$$= \frac{3}{(1+1)^{1+2\times 0.53} - 1 - 1} = 1.3825$$

For (P1, P4);

$$= \frac{1}{2^{2.06} - 2} = 0.46$$

For (P2, P3);

$$= \frac{3}{2^{2.06} - 2} = 1.3825$$

For (P2, P4);

$$= \frac{2}{2^{2.06} - 2} = 0.921$$

For $(P_3, P_4)$;

$$= \frac{1}{2^{2.06} - 2} = 0.46$$

The pairs $(P_1, P_2)$, $(P_1, P_3)$ and $(P_2, P_3)$ have highest goodness measure among all other pairs. So, Next pair will be;

Link Table

| ~~Pair~~ $\{P_1, P_2\}$ | $P_3$ | $P_4$ |
|---|---|---|
| $\{P_1, P_2\}$ | — | $3+3=6$ |
| $P_3$ | — | $1+2=3$ |
| $P_4$ | — | 1 |

(Link Table)

| Pair | Goodness Measure |
|---|---|
| $\{P_1, P_2\}, P_3$ | 1.35 |
| $\{P_1, P_2\}, P_4$ | 0.67 |
| $\{P_3, P_4\}$ | 0.46 |

For $\{P_1, P_2\}, P_3$;

$$= \frac{6}{(2+1)^{2.06} - 2^{2.06} - 1} = 1.35$$

For $\{P_1, P_2\}, P_4$

$$= \frac{3}{(2+1)^{2.06} - 2^{2.06} - 1} = 0.67$$

For $\{P_3, P_4\}$,

$$= \frac{1}{(1+1)^{2.06} - 1 - 1} = 0.46$$

$\{P_1, P_2, P_3\}$ $P_4$

$\{P_1, P_2, P_3\}$ — $3+1=4$

$\{P_4\}$ —

| Pair | Goodness measure |
|---|---|
| $\{P_1, P_2, P_3\}, P_4$ | 0.59 |

$$= \frac{4}{(3+1)^{2.06} - 3^{2.06} - 1}$$

$$= 0.59$$

There is only two clusters. So, stop the process.

## * DBSCAN:

DBSCAN is a density-based clustering algorithm that groups together points that are closely packed together while marking points in low-density regions as outliers.

It requires two parameters: epsilon(ε), which defines the radius of the neighborhood around a point, and minPts., which specifies the minimum number of points required to form a dense region.

→ It stands for Density-Based Spatial Clustering of Applications with Noise.

E.g: P1:(3,7), P2:(4,6), P3:(5,5), P4:(6,4), P5:(7,3)

| | P1 | P2 | P3 | P4 | P5 |
|---|---|---|---|---|---|
| P1 | 0 | | | | |
| P2 | 1.41 | 0 | | | |
| P3 | 2.83 | 1.41 | 0 | | |
| P4 | 4.24 | 2.83 | 1.41 | 0 | |
| P5 | 5.66 | 4.24 | 2.83 | 1.41 | 0 |

Given,

minPts = 3

epsilon(ε) = 1.9

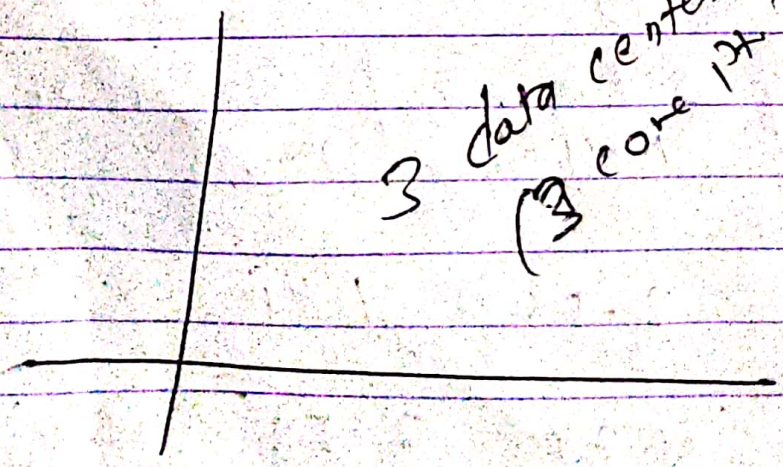
Step: 1 Find distance less than or equal to epsilon(ε) i.e. 1.9.
(Search horizontally & vertically).

For:

P1 : P2, ~~P3~~ P3 : P2, P4
P2 : ~~P1~~, P3 ~~P4~~ P4 : P3, P5
& so on. P5 : P4.

| Point | Status | |
|---|---|---|
| P1 | Noise | Border |
| P2 | Core | Core |
| P3 | Core | Core |
| P4 | Core | Core |
| P5 | Noise | Border (Noise) |

P1 should be a part of anyone Core data points.



## * Bayesian classification:

Bayesian Classification is based on Bayes' theorem. It is also called Naive Bayes classification or Naive Bayesian classification. Bayes Theorem is given by:

$$P(H|X) = \frac{P(X|H)\,P(H)}{P(X)}$$

Bayes' theorem is useful in that it provides a way of calculating the posterior probability, $P(H|X)$, from $P(H)$, $P(X|H)$, and $P(X)$. Here $P(X)$ and $P(H)$ are prior probability.

Let D be a database and $C_1, C_2, \ldots C_m$ are m classes. Now above Bayes rule can be written as:

$$P(C_i|X) = \frac{P(X|C_i)\,P(C_i)}{P(X)}.$$

Let X is the set of attributes $\{X_1, X_2, X_3, \ldots X_n\}$ where attributes are independent of one another. Now, the probability $P(X|C_i)$ is given by the equation given below:

$$P(X/C_i) = \prod_{k=1}^{n} P(X_k/C_i)$$

$$= P(X_1/C_i) \times P(X_2/C_i) \ldots \times P(X_n/C_i)$$

Example;

| RID | age | income | student | credit | buys-Comput class: |
|---|---|---|---|---|---|
| 1 | youth | high | no | fair | no |
| 2 | youth | high | no | excellent | no |
| 3 | middle-aged | high | no | fair | yes |
| 4 | Senior | medium | no | fair | yes |
| 5 | Senior | low | yes | fair | yes |
| 6 | senior | low | yes | excellent | no |
| 7. | middle-aged | low | yes | excellent | yes |
| 8 | youth | medium | no | fair | no |
| 9 | youth | low | yes | fair | yes |
| 10 | Senior | medium | yes | fair | yes |
| 11 | youth | medium | yes | excellent | yes |
| 12 | middle-aged | medium | no | excellent | yes |
| 13 | middle-aged | high | yes | fair | yes |
| 14 | senior | medium | no | excellent | no |
| 15. | youth | medium | yes | fair | ? |

Predict class level of the tuple : X

= (age = youth, income = medium, student = yes, credit_rating = fair) using Bayesian classification.

Soln: Prior probability of each class can be computed based on the training tuples:

P(buys-computer = yes) = 9/14 = 0.643
P(buys-Computer = no) = 5/14 = 0.357

Compute the following conditional probabilities:

P(age = youth | buys-computer = yes) = 2/9 = 0.222

P(age = youth | buys-computer = no) = 3/5 = 0.6

P(income = medium | buys-computer = yes) = 4/9 = 0.444

P(income = medium | buys-computer = no) = 2/5 = 0.4

P(student = yes | buys-computer = yes) = 6/9 = 0.667

P(student = yes | buys-computer = no) = 1/5 = 0.

P(Credit-rating = fair | buys-computer = yes) = 6/9 = 0.667

P(Credit-rating = fair | buys-computer = no) = 2/5 = 0.4

Using the above probabilities, we obtain

$P(X / buys\_computer = yes)$

$= P(age = youth \mid buys\_computer = yes) \times$
$P(income = medium \mid buys\_computer = yes) \times$
$P(student = yes \mid buys\_computer = yes) \times$
$P(credit\_rating = fair \mid buys\_computer = yes)$

$= 0.222 \times 0.444 \times 0.667 \times 0.667 = 0.044$

$P(X / buys\_computer = no)$

$= P(age = youth \mid buys\_computer = no) \times$
$P(income = medium \mid buys\_computer = no) \times$
$P(student = yes \mid buys\_computer = no) \times$
$P(credit\_rating = fair \mid buys\_computer = no)$

$= 0.6 \times 0.4 \times 0.2 \times 0.4 = 0.019$

Now,

$P(buys\_computer = yes \mid X)$

$= P(X \mid buys\_computer = yes) \times P(buys\_computer = yes) = 0.044 \times 0.643 = 0.028$

$P(buys\_computer = no \mid X)$

$= P(X \mid buys\_computer = no) \times P(buys\_computer = no)$

## ID3 Decision Tree:

ID3 Stands for Iterative Dichotomiser 3. It uses top-down greedy approach to build decision tree model.

This algorithm computes information gain for each attribute and then selects the attribute with the highest information gain.

Information gain measures reduction in entropy after data transformation. It is calculated by comparing entropy of the dataset before and after transformation.

Entropy is the measure of homogeneity of the Sample. Entropy or expected information of dataset D is calculated by using equation 1.

$$E(D) = -\sum_{i=1}^{m} P_i \log_2 P_i \quad \text{---- (1)}$$

where,

$P_i$ is the probability of a tuple in D belonging to class $C_i$ and is estimated using Equation 2.

$$p_i = \frac{|C_{i,D}|}{|D|} \quad \text{..... (2)}$$

where, $|C_{i,D}|$ is the number of tuples in D belonging to class $C_i$ and $|D|$ is the number of tuples in D.

Suppose we have to partition the tuples in D on some attribute A having v distinct values. The attribute A can be used to split D into v partitions $\{D_1, D_2, \ldots D_v\}$.

Now, Entropy of attribute, A is calculated as,

$$E_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times E(D_j)$$

Finally, the information gain achieved after partioning D on attribute A is calculated as,

$$IG(A) = E(D) - E_A(D).$$

Example:

| | age | income | student | credit-rating | Buys-C class |
|---|---|---|---|---|---|
| 1 | youth | high | no | fair | no |
| 2 | youth | high | no | excellent | no |
| 3 | middle-aged | high | no | fair | yes |
| 4 | Senior | medium | no | fair | yes |
| 5 | Senior | low | yes | fair | yes |
| 6 | Senior | low | yes | excellent | no |
| 7 | middle-aged | low | yes | excellent | yes |
| 8 | youth | medium | no | fair | no |
| 9 | youth | low | yes | fair | yes |
| 10 | Senior | medium | yes | fair | yes |
| 11 | youth | medium | yes | excellent | yes |
| 12 | middle-aged | medium | no | excellent | yes |
| 13 | middle-aged | high | yes | fair | yes |
| 14 | Senior | medium | no | excellent | no |

Construct decision tree from above data using information gain.

- Predict class level of the tuple: X = (age = youth, income = medium, student = yes, credit-rating = fair).

Soln.

Expected information needed to classify a tuple in D is:

$$E(D) = -\frac{9}{14}\log_2\left(\frac{9}{14}\right) - \frac{5}{14}\log_2\left(\frac{5}{14}\right)$$

$$= 0.94$$

$$E_{age}(D) = \frac{5}{14}\left[-\frac{2}{5}\log_2\left(\frac{2}{5}\right) - \frac{3}{5}\log_2\left(\frac{3}{5}\right)\right] +$$

$$\frac{4}{14}\left[-\frac{4}{4}\log_2\left(\frac{4}{4}\right) - \frac{0}{4}\log_2\left(\frac{0}{4}\right)\right] + \frac{5}{14}\left[\right.$$

$$\left.-\frac{3}{5}\log_2\left(\frac{3}{5}\right) - \frac{2}{5}\log_2\left(\frac{2}{5}\right)\right]$$

$$= 0.694$$

$$\therefore IG(Age) = E(D) - E_{age}(D)$$

$$= 0.94 - 0.694 = 0.246$$

Similarly,

$$E_{income}(D) = \frac{4}{14}\left[-\frac{2}{4}\log_2\frac{2}{4} - \frac{2}{4}\log_2\frac{2}{4}\right]$$

$$+ \frac{6}{14}\left[-\frac{4}{6}\log_2\frac{4}{6} - \frac{2}{6}\log_2\frac{2}{6}\right] + \frac{4}{14}$$

$$\left[-\frac{3}{4}\log_2\frac{3}{4} - \frac{1}{4}\log_2\frac{1}{4}\right] = 0.91$$

$\therefore IG(Income) = E(D) - E_{income}(D)$

$= 0.94 - 0.91 = 0.03$

Again,

$$E_{student}(D) = \frac{7}{14}\left[-\frac{6}{7}\log_2\frac{6}{7} - \frac{1}{7}\log_2\frac{1}{7}\right] + \frac{7}{14}\left[-\frac{3}{7}\log_2\frac{3}{7} - \frac{4}{7}\log_2\frac{4}{7}\right]$$

$= 0.2958 + 0.4926 = 0.788$

$\therefore IG(student) = E(D) - E_{student}(D)$

$= 0.94 - 0.788 = 0.152$

$$E_{credit\_rat}(D) = \frac{8}{14}\left[-\frac{6}{8}\log_2\frac{6}{8} - \frac{2}{8}\log_2\frac{2}{8}\right] + \frac{6}{14}\left[-\frac{3}{6}\log_2\frac{3}{6} - \frac{3}{6}\log_2 3/6\right]$$

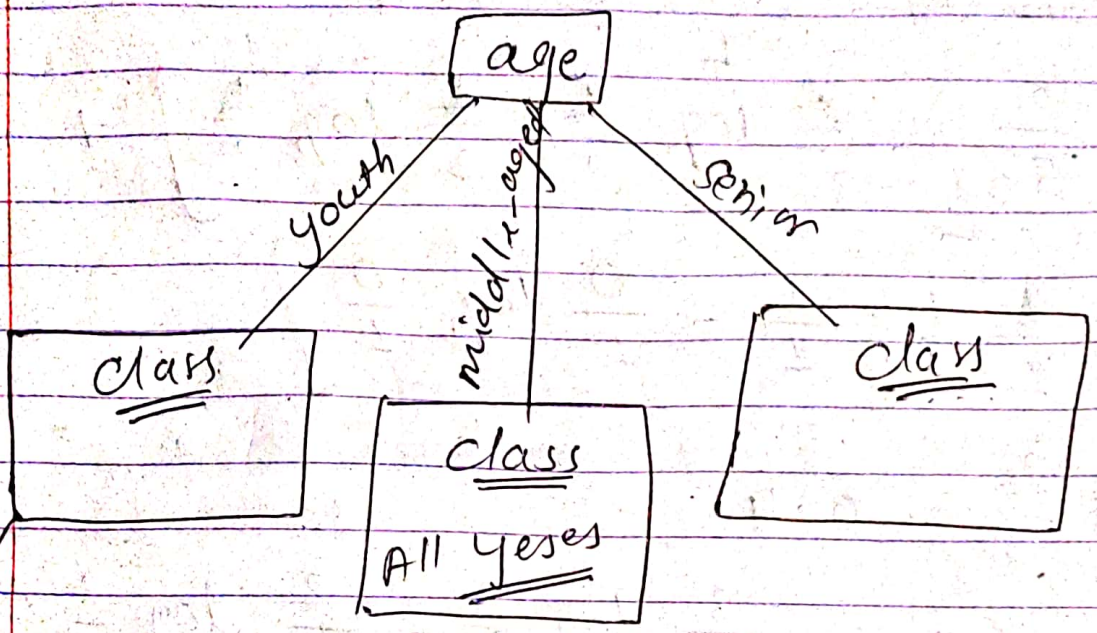
$= 0.4635 + 0.4285 = 0.892$

$\therefore IG(credit\_rat) = E(D) - E_{credit\_rat}(D)$

$= 0.94 - 0.892$

$= 0.048$

Because age has the highest information gain among the attributes, it is selected as the splitting attribute. The decision tree now looks like below:




| income | student | credit-rating | class |
|---|---|---|---|
| high | no | fair | no |
| high | no | excellent | no |
| medium | no | fair | yes |
| low | yes | fair | yes |
| medium | yes | excellent | yes |

Now, Calculate information gain of income, student and credit_rating in case of left children (youth) & right children (Senior). And repeat the above process.

$\{\log 0 = 0$
$\log 1 = 0\}$

IG(income) = E~~gyouth~~(D) − Eincome(D)

~~= 0.691~~ 0.97 − 0.4 = ~~0.97~~ 0.57

$$E_{income}(D) = \frac{2}{5}\left[-\frac{0}{2}\log_2\frac{0}{2} - \frac{2}{2}\log_2\frac{2}{2}\right]$$

$$+\frac{2}{5}\left[-\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}\right] + \frac{1}{5}$$

$$\left[-\frac{1}{1}\log_2\frac{1}{1} - \frac{0}{1}\log_2\frac{0}{1}\right]$$

$$= \frac{2}{5} = 0.4$$

$$E_{youth}(D) = -\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}$$

$$= 0.97$$

Again,

$$E_{student}(D) = \frac{2}{5}\left[\frac{2}{2}\log_2\frac{2}{2} - \frac{0}{2}\log_2\frac{0}{2}\right]$$

$$+\frac{3}{5}\left[-\frac{3}{3}\log_2\frac{3}{3} - \frac{0}{3}\log_2\frac{0}{3}\right]$$

$$= \text{~~0.5509~~}\ 0$$

$\therefore$ IG(Student) = $E_{youth}(D) - E_{student}(D)$

= 0.97 ~~− 0.5509~~ [illegible]

$$E_{credit\_rat}(D) = \frac{3}{5}\left[-\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3}\right]$$

$$+\frac{2}{5}\left[-\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}\right]$$

$$= 0.5TO97 + 0.4 = 0.950$$

4. $IG(credit\_rat) = E_{youth}(D) - E_{credit\_rat}(D)$

$$= 0.97 - 0.95 = 0.02$$

Because ~~income~~ student has the highest information gain.