ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ

Σειρά ασκήσεων 2

Ακαδημαϊκό έτος 2023

10^{o}εξάμηνο

Αγγλογάλλος Αναστάσιος ΑΜ : el18641

✅Άσκηση 1

1.
(p ⇒ ¬ (p ∨ q)) ⇔ ((r ∧ ¬q) ∨ r)

Βήμα 1ο , Αντικαθιστώ τις ισοδυναμίες (⇔) με (⇒ ∧ ⇐)

[(p ⇒ ¬ (p ∨ q)) ⇒ ((r ∧ ¬q) ∨ r)] ∧ [((r ∧ ¬q) ∨ r) ⇒ (p ⇒ ¬ (p ∨ q))]

Βήμα 2ο , Αντικαθιστώ τις συνεπαγωγές (p ⇒ q) με (¬p ∨ q)

[¬(¬ p ∨ ¬ (p ∨ q)) ∨ ((r ∧ ¬q) ∨ r)] ∧ [¬((r ∧ ¬q) ∨ r) ∨ (¬ p ∨ ¬ (p ∨ q))]

Βήμα 3ο, Μετακινούμε την άρνηση μπροστά στις ατομικές προτάσεις

[(¬¬ p ∧ ¬¬¬ (p ∨ q)) ∨ ((r ∧ ¬q) ∨ r)] ∧ [(¬(r ∧ ¬q) ∧ ¬r) ∨ (¬ p ∨ (¬p ∧ ¬q))]
[(p ∧ (p ∨ q)) ∨ ((r ∧ ¬q) ∨ r)] ∧ [((¬r ∨ q) ∧ ¬r) ∨ (¬ p ∨ (¬p ∧ ¬q))]

Βήμα 4ο, Επιμερούμε τις διαζεύξεις

[(((p ∧ (p ∨ q)) ∨ (r ∧ ¬q)) ∧ ((p ∧ (p ∨ q)) ∨ r] ∧
[(((¬r ∨ q) ∧ ¬r) ∨ (¬p ∧ ¬q)) ∧ (((¬r ∨ q) ∧ ¬r) ∨ (¬ p)]

(((p ∧ (p ∨ q)) ∨ r) ∧ (((p ∧ (p ∨ q)) ∨ ¬q) ∧
((p ∨ r) ∧ (r ∨ (p ∨ q)) ∧
(((¬r ∨ q) ∧ ¬r) ∨ ¬p) ∧ (((¬r ∨ q) ∧ ¬r) ∨ ¬q) ∧ (((¬r ∨ q) ∧ ¬r) ∨ (¬ p)

(p ∨ r) ∧ (p ∨ q ∨ r) ∧ (p ∨ ¬q) ∧ (p ∨ q ∨ ¬q) ∧
(r ∨ p ∨ r) ∧ (p ∨ q ∨ r) ∧
(¬r ∨ q ∨ ¬p) ∧ (¬p ∨ ¬r) ∧
(¬r ∨ q ∨ ¬q) ∧ (¬r ∨ ¬q) ∧
(¬r ∨ q ∨ ¬p) ∧ (¬r ∨ ¬ p)

Βήμα 5, Απλοποίηση

(p ∨ r) ∧ (p ∨ q ∨ r) ∧ (p ∨ ¬q) ∧ (p ∨ *q ∨ ¬q*) ∧
(r ∨ p) ∧ (p ∨ q ∨ r) ∧

$(\neg r \vee q \vee \neg p) \wedge (\neg p \vee \neg r) \wedge$
$(\neg r \vee q \vee \neg q) \wedge (\neg r \vee \neg q) \wedge$
$(\neg r \vee q \vee \neg p) \wedge (\neg r \vee \neg p)$

CNF form : { [p, r], [p, q, r], [p, ¬q], [p, q, ¬q], [r, p], [p, q, r], [¬r, q, ¬p], [¬p, ¬r], [¬r, q, ¬q],
[¬r, ¬q], [¬r, q, ¬p], [¬r, ¬p] }

Απαλοίφουμε τους ίδιους όρους

CNF form : { [p, r], [p, q, r], [p, ¬q], [p, q, ¬q], [¬r, q, ¬p], [¬p, ¬r], [¬r, q, ¬q],
[¬r, ¬q]}

Συνεχίζουμε τις απαλοιφές τον συμπληρωματικών

CNF form : { [p, q, r], [p, ¬q], [p, q, ¬q], [¬r, q, ¬p], [¬r, q, ¬q],
[¬r, ¬q] }

CNF form : { [p, q, r], [p, ¬q], [¬r, q, ¬p], [¬r, ¬q] }

CNF form : { [p, q, r], [p, ¬q], [¬r, ¬q] }

2.
Στην περίπτωση όπου το w δεν έχει εμβέλεια μέχρι το “δεύτερο” w, κάνουμε τα παρακάτω :
Το “δεύτερο” w είναι σταθερά σε αυτή την περίπτωση και για να μην μπερδευόμαστε θα το μετονομάσουμε σε c. Οπότε προκύπτει
$\forall x.\forall y.\exists z. (\forall w.(p(x, y) \Rightarrow q(w)) \vee (p(y, z) \Rightarrow \neg q(c)))$

Βήμα 1, Αντικατάσταση συνεπαγωγών

$\forall x.\forall y.\exists z. (\forall w.(\neg p(x, y) \vee q(w)) \vee (\neg p(y, z) \vee \neg q(c)))$

Βήμα 2, Μετακινούμε την άρνηση προς τα μέσα

-

Βήμα 3, Δώσε μοναδικά ονόματα σε όλες τις μεταβλητές που εμφανίζονται στην φόρμουλα
(Το έχουμε κάνει ήδη)
$\forall x.\forall y.\exists z. (\forall w.(\neg p(x, y) \vee q(w)) \vee (\neg p(y, z) \vee \neg q(c)))$

Βήμα 4, Αφαίρεσε όλους τους υπαρξιακούς ποσοδείκτες μέσω Skolemization
Αντικαθιστούμε το υπαρξιακό ποσοδείκτη με μια νέα συνάρτηση Skolem

$\forall x.\forall y.(\forall w.(\neg p(x, y) \vee q(w)) \vee \neg p(y, f(x,y)) \vee \neg q(c)$

Βήμα 5, Μετακίνηση καθολικών ποσοδεικτών στην αρχή και απαλοιφή

$\forall x.\forall y.\forall w.(\neg p(x, y) \vee q(w)) \vee \neg p(y, f(x,y)) \vee \neg q(c)$

Όλοι οι καθολικοί ποσοδείκτες βρίσκονται στην αρχή οπότε προχωράμε στην απαλοιφή.

$\neg p(x, y) \vee q(w) \vee \neg p(y, f(x,y)) \vee \neg q(c)$

Βήμα 6, Επιμερίζουμε τις διαζεύξεις

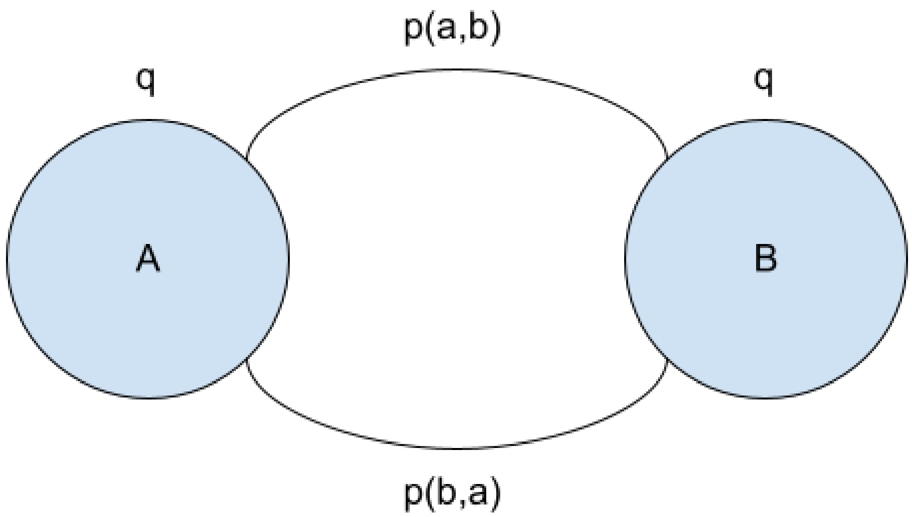
-

CNF form : $\{ [\neg p(x, y), q(w), \neg p(y, f(x,y)), \neg q(c)] \}$

✅Άσκηση 2

Για να βρούμε ένα μοντέλο της γνώσης K, πρέπει να βρούμε μια ερμηνεία για τα αντικείμενα και τις σχέσεις που ικανοποιεί όλες τις προτάσεις της K.

Ας εξετάσουμε τις προτάσεις μία-μία:
p(a, b) , p(b, a)
q(a) , q(b)
Σύμφωνα με τις παραπάνω προτάσεις φτιάχνουμε το παρακάτω σχήμα και προκύπτει η παρακάτω ερμηνεία :
Δ^I: { a, b }
p^I: { (a,b) , (b,a) }
q^I: { a , b }



Έπειτα ελέγχουμε εάν ικανοποιεί την παρακάτω πρόταση:
$\forall x. \forall y. \exists z. (\forall w.(p(x, z) \Rightarrow q(w)) \vee (p(y, z) \Rightarrow \neg q(w)))$:
Αυτή η πρόταση λέει ότι για κάθε x και y υπάρχει ένα z, τέτοιο ώστε για κάθε w, αν ισχύει η σχέση p(x, z), τότε ισχύει η σχέση q(w), ή αν ισχύει η σχέση p μεταξύ του y και του z, τότε ισχύει η άρνηση της σχέσης q για ένα w (σταθερά).

Οι μεταβλητές x,y,z,w μπορούν να πάρουν όλες τις τιμές (a,b).
Αρκεί να εξετάσουμε εάν το $\forall w.(p(x, z) \Rightarrow q(w))$ είναι αληθές για όλα τα σενάρια.
Εάν x=z τότε προκύπτει F → T άρα αληθές

Εάν $x \neq z$ τότε προκύπτει $T \to T$ άρα αληθές
Επειδή το z έχει υπαρξιακό ποσοδείκτη δεν χρειαζόταν να ελένξουμε όλα τα σενάρια αρκούσε να βρούμε εάν υπάρχει z ώστε όταν $x = a$ ή όταν $x = b$ η πρόταση είναι αληθής.
Συνεπώς για κάθε τιμή ισχύει η πρόταση άρα βρήκαμε μια ερμηνεία που ικανοποιεί ολες τις προτάσεις.

Άσκηση 3

Αντίστοιχα με το Ερώτημα 1.2 προκύπτει :

CNF form : $\{ [\neg p(f(x,y), x), q(w), \neg p(y, f(x,y)), \neg q(c)] \}$
Για $w = c$
Τότε η πρόταση είναι ταυτολογία.

✅Άσκηση 4

Για να εξετάσουμε εάν υπάρχει μια ερμηνεία που ικανοποιεί την πρόταση $\forall x.(\exists y.p(x, y) \Rightarrow q(a))$ και δεν ικανοποιεί την πρόταση $(\forall x. \exists y.p(x, y)) \Rightarrow q(a))$, πρέπει να αξιολογήσουμε τις δύο προτάσεις σε μια συγκεκριμένη ερμηνεία.

Πρόταση 1 : $\forall x.(\exists y. p(x, y) \Rightarrow q(a))$

Πρόταση 2 : $(\forall x. \exists y.p(x, y)) \Rightarrow q(a))$

Για να κάνουμε την ζωή μας πιο εύκολη θα ξεκινήσουμε από ένα σενάριο στο οποίο δεν ικανοποιείται η δεύτερη πρόταση. Καθώς εξετάζουμε ένα από τα τέσσερα σενάρια της συνεπαγωγής

P1	**P2**	**P1 ⇒ P2**
True	True	True
True	**False**	**False**
False	True	True
False	False	True

Συνεπώς στην συνεπαγωγή που υπάρχει στην Π2 πρέπει να ισχύει :
$(\forall x. \exists y.p(x, y))$: True
$q(a))$: False
Έστω
Δ^I : { a, b }
p^I : { (a,b) , (b,a) }

q^I : { }
Με αυτήν την ερμηνεία ισχύουν τα παραπάνω
Ας εξετάσουμε τώρα την Π1 πρέπει να ικανοποιείται :
$(\forall x. \exists y.p(x, y))$: -
q(a)) : Γνωρίζουμε ότι είναι False

Συνεπώς πρέπει το αριστερό μέλος της συνεπαγωγής $(\forall x. \exists y.p(x, y))$, να είναι και αυτό False για να ικανοποιείται.
Εξετάζουμε το σενάριο αυτό και παρατηρούμε ότι δεν είναι False αλλά True καθώς υπάρχει για κάθε x (a,b) υπάρχει ένα y (b,a) αντίστοιχα τέτοιο ώστε να ισχύει η σχέση p(x,y).

Άρα αποδείξω πως δεν υπάρχει ερμηνεία η οποία να ικανοποιεί την Π1 και να μην ικανοποιεί την Π2.

✅Άσκηση 5

Για κάθε ζεύγος πρέπει να βρούμε μία ερμηνεία που να ικανοποιεί τις αντιστοιχες δύο προτάσεις.
Το σύμπαν του λογικού προγράμματος περιλαμβάνει όλα τα πιθανά αντικείμενα που μπορούν να αντιστοιχηθούν στις μεταβλητές του προγράμματος. Στην περίπτωσή μας, τα αντικείμενα που μπορούν να χρησιμοποιηθούν είναι:

Το σύνολο όλων των βασικών όρων που μπορούν να κατασκευαστούν από ρις

a, b, c, d, e, f

Η βάση Herbrand περιλαμβάνει όλα τα πιθανά γεγονότα που μπορούν να παραχθούν από τις προτάσεις του προγράμματος. Στην περίπτωσή μας, η βάση Herbrand περιλαμβάνει τις ακόλουθες προτάσεις:

parent(a, a), parent(a, b), parent(a, c), parent(a, d), parent(a, e), parent(a,f),
parent(b, a), parent(b, b), ...
parent(c, a), ...
...

sibling(a, a), sibling(a, b), sibling(a, c), ...
...

grandparent(a, a), grandparent(a, b), ...
...

cousin(a, a), cousin(a, b), cousin(a, c), ...
...

mother(a, a), mother(a, b), mother(a, c), ...
...

father(a, a), father(a, b), father(a, c), ...
...

Αλλάζουμε τα ονόματα των σχέσεων για να είναι πιο κατανοητά

parent (x, y) ≡ hasparent (x, y)
father (x, y) ≡ hasfather (x, y)
mother (x, y) ≡ hasmother (x, y).
sibling (x, y) ≡ hassibling (x, y)
cousin (x, y) ≡ hascousin (x, y)
grandparent (x, y) ≡ hasgrandparent (x, y)

5.2 a). forward chaining.

i). cousin (a, e) ; cousin (a, f)

hasmother (a, b) → hasparent (a, b) → hasgrandparent (a, c) → cousin (a, d)
hasfather (b, c) → hasparent (b, c) → hasgrandparent (d, c) → cousin (a, f)
hasmother (d, b) → hasparent (d, b) → hasgrandparent (f, c) → cousin (d, f)
hasmother (e, c) → hasparent (e, c)
hasmother (f, b) → hasparent (f, b)

...
Δεν χρειάζεται να γράψουμε τα αντιμεταθετικά.

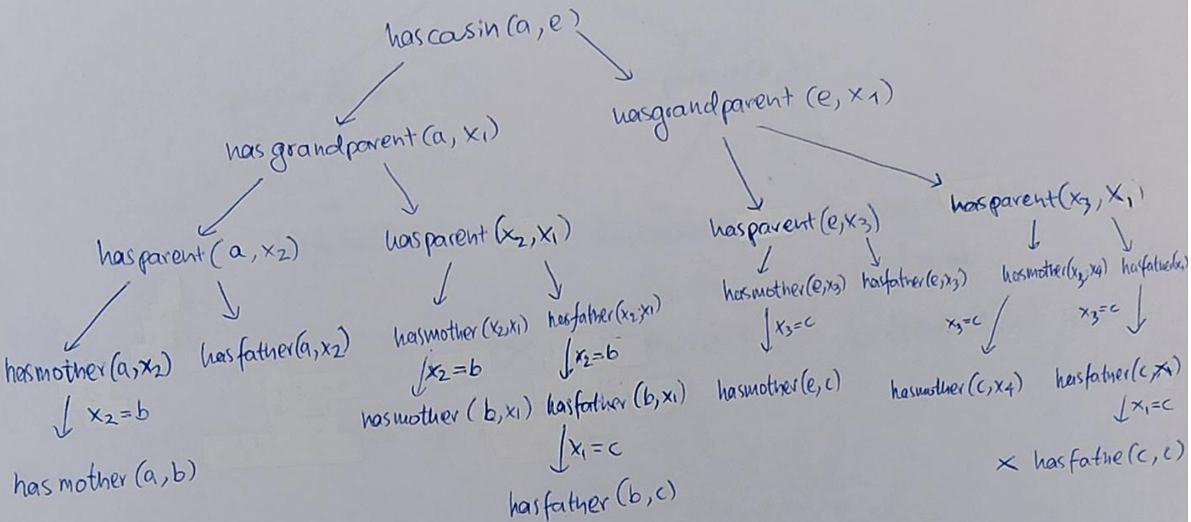
Αποτυχία για το cousin (a, e)
Επιτυχία για το cousin (a, f)

sibling (a, e)
hasmother (a, b) → hasparent (a, b) → sibling (a, d)
hasfather (b, c) → hasparent (b, c) → sibling (a, f)
hasmother (d, b) → hasparent (d, b) → sibling (b, e).
hasmother (e, c) → hasparent (e, c)
hasmother (f, b) → hasparent (f, b)

Επιτυχία για το sibling (b, e).

b) Backward chaining.

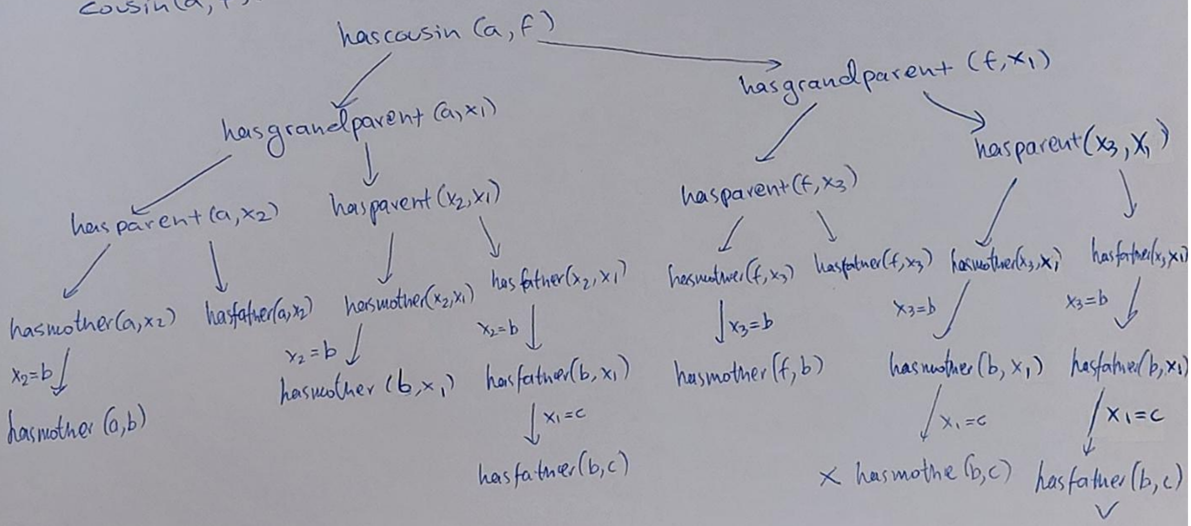
cousin(a,e).



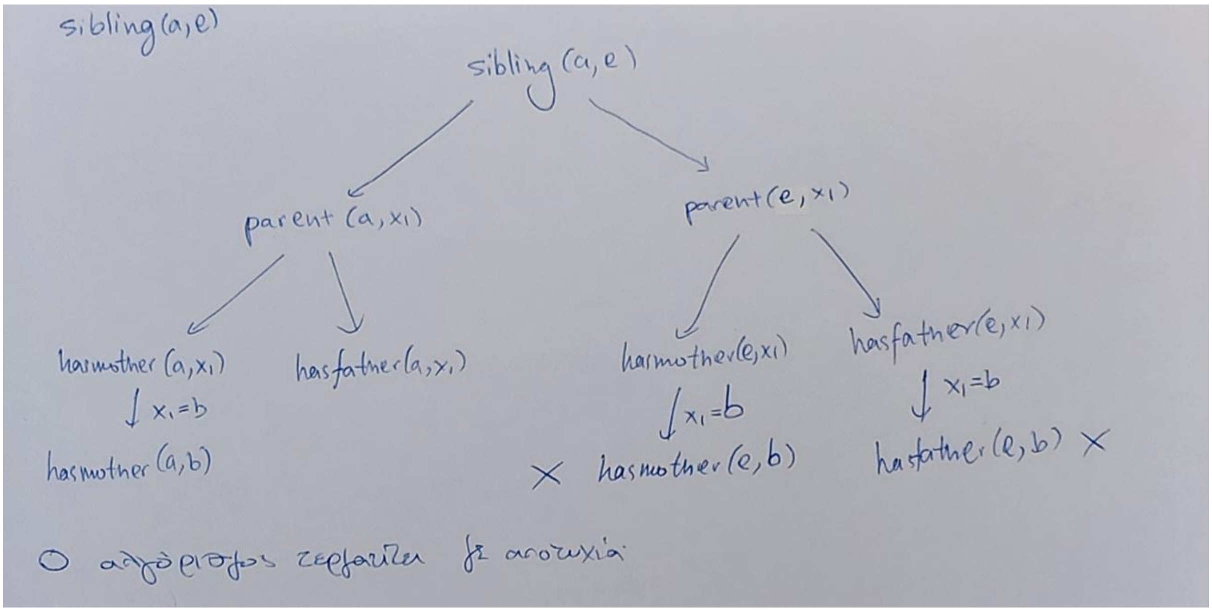
Ο αλγόριθμος τερματίζει με αποτυχία

hascousin(a,f)

cousin(a,f).



Ο αλγόριθμος τερματίζει με επιτυχία.



c) Υπάρχει πρόβλημα με το πως έχουν οριστεί οι βασικές προτάσεις. Για παράδειγμα υπάρχει στη γνώση father(b,c) και mother(e,c). Είναι βιολογικά αδύνατο ένας άνθρωπος να είναι μητέρα και πατέρας.
Γι'αυτό το λόγο θα έκανα την εξής αλλαγή :
hasparent(x,y) → hasfather(x,y), hasmother(x,z)
hasparent(x,z) → hasfather(x,y), hasmother(x,z)
Δεν εξετάζω μακάβρια σενάρια που έχει πεθάνει κάποιος από τους γονείς, ακόμα και νεκροί να είναι οι γονείς η συγγένεια παραμένει.

✅Άσκηση 6

a.

Το w2 ανάλογα το η μπορεί να πάρει άπειρες τιμές μέσα στο διάστημα σύγκλισης του αλγορίθμου. Αρκεί να διαχωρίζει σωστά τα δεδομένα μεταξύ τους δηλαδή να συγκλίνει ο αλγόριθμος. Το οποίο στην περίπτωση του perceptron για $0 \leq η \leq 1$ είναι εξασφαλισμένο.

b.

Ο αριθμός των απαιτούμενων επαναλήψεων μέχρι την σύγκλιση του αλγόριθμου είναι διαφορετικός ανάλογα την τιμή του η.

Η σύγκλιση του αλγορίθμου του Perceptron είναι σίγουρη σε ένα γραμμικά διαχωρίσιμο σύνολο δεδομένων. Το πόσο μεγάλα θα είναι τα βήματα προς την σύγκλιση εξαρτάται και από την τιμή του ρυθμού μάθηση η.

Όσο πιο μεγάλα βήματα (μεγαλύτερη τιμή του η) τόσο πιο γρήγορα μπορούν να αλλάξουν οι τιμές των βαρών που μπορεί να οδηγήσει σε πιο γρήγορη σύγκλιση του αλγορίθμου. Όμως μεγάλες τιμές του η μπορεί να προκαλέσουν το αλγόριθμο να αυξάνει ή να μειώνει τα βάρη

πέρα από το σημείο σύγκλισης. Μια σημαντικά μικρότερη τιμή του η πιθανόν να χρειαστεί περισσότερα βήματα να φτάσει στην σύγκλιση.
Για τα παραπάνω παίζει αρκετό ρόλο και η αρχικοποίηση των βαρών.

✅Άσκηση 7

Για να σχεδιάσουμε τα δέντρα αποφάσεων για το πρόβλημα δυαδικής ταξινόμησης στο {0, 1}, ακολουθούμε τα εξής βήματα:

Επιλέγουμε ένα χαρακτηριστικό (feature) που μας επιτρέπει να κάνουμε διαχωρισμό μεταξύ των κλάσεων 0 και 1.

Δημιουργούμε έναν κόμβο στο δέντρο αποφάσεων με βάση αυτό το χαρακτηριστικό.

Αποφασίζουμε την κατηγορία (0 ή 1) που αντιστοιχεί στον κόμβο αυτό, με βάση την πλειοψηφία των παραδειγμάτων εκπαίδευσης που ανήκουν σε αυτόν τον κόμβο.

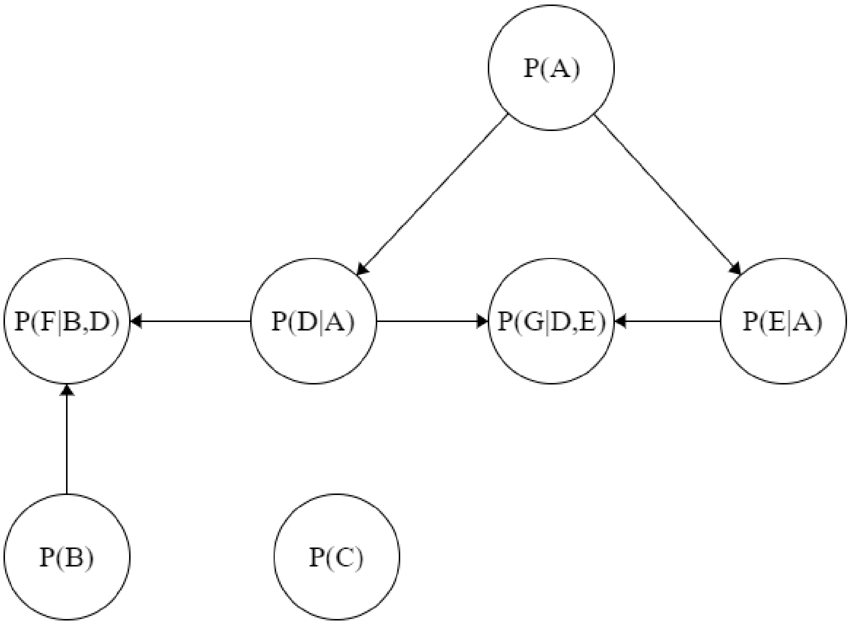
Συνεχίζουμε τη διαδικασία για τους υποκόμβους του δέντρου, επιλέγοντας νέα χαρακτηριστικά και διαχωρίζοντας τα παραδείγματα με βάση αυτά.

Το μέγιστο βάθος των δέντρων αποφάσεων εξαρτάται από τον αριθμό των χαρακτηριστικών που χρησιμοποιούνται για τον διαχωρισμό και τον αριθμό των παραδειγμάτων εκπαίδευσης. Συνήθως, το μέγιστο βάθος των δέντρων περιορίζεται για να αποφευχθεί η υπερπροσαρμογή (overfitting) στα δεδομένα εκπαίδευσης.

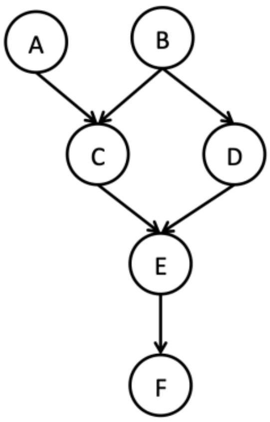
Η VC-διάσταση (VC-dimension) της κλάσης των δέντρων αποφάσεων στο {0, 1}d είναι ίση με τον μέγιστο αριθμό παραδειγμάτων που μπορούν να "σπάσουν" από ένα δέντρο αποφάσεων. Η VC-διάσταση αυξάνει με τον αριθμό των χαρακτηριστικών που χρησιμοποιούνται στο δέντρο απόφασης.

✅Άσκηση 8

1. Σχεδίαση Δικτύου Πίστης



2. Γράψτε την J(⟨A, B, C, D, E, F⟩) που αντιστοιχεί στο παρακάτω δίκτυο πίστης.



Σχήμα 1: Δίκτυο πίστης (belief network)

J(⟨A, B, C, D, E, F⟩) = Pr(P(A)) * Pr(P(B)) * Pr (P(A|C)) * Pr(P(C|B)) * Pr(P(D|B)) * Pr(P(E|C ∧ D)) * Pr(P(F|E))

3. . Θεωρήστε ότι οι τυχαίες μεταβλητές του δικτύου του προηγούμενου ερωτήματος είναι τύπου boolean. Εξηγήστε πόσες παράμετροι χρειάζεται κατ' ελάχιστο να είναι γνωστές για να προσδιοριστεί πλήρως το δίκτυο πίστης.

Δεν καταλαβαίνω ακριβώς τι εννοεί η άσκηση τις "παραμέτρους" εάν εννοεί τις πιθανότητες. Τότε πρέπει να υπολογίσουμε : 1+1+2+2+2+4+2 = 14 πιθανότητες.