

## Прв дел: Баесови мрежи

### Подем на вештачката интелигенција

а)

AAI – случајна променлива која го претставува индексот на развој и примена на ВИ, прима вредности висок, умерен, низок. Оваа променлива ни е целната променлива, т.е. класата.

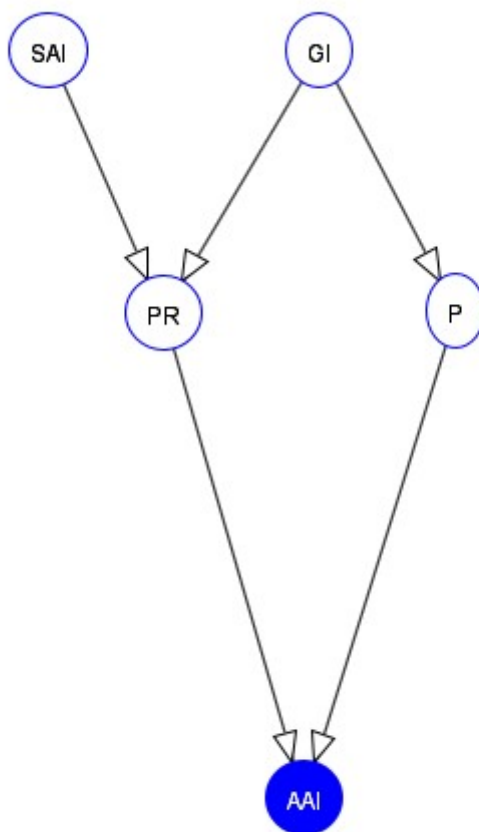
PR – случајна променлива која е индикатор на тоа дали бројот на истражувачки трудови е низок, среден или висок.

P – логичка(boolean) случајна променлива која кажува дали има или нема пораст.

GI - логичка(boolean) случајна променлива која кажува дали порастот се или не се должи на порастот на владини иницијативи

SAI - логичка(boolean) случајна променлива која кажува дали е висок или низок интересот за предметот ВИ

б)



в)

1. за јазлите SAI и GI, потребни се само нивните маргинални веројатности, односно по две вредности за секој јазол
2. за јазолот PR, во неговата табела влегуваат секако самиот тој јазол, како и јазлите SAI и GI. За да се дефинираат веројатносните распределби потребни се сите можни комбинации на овие 3 јазли, односно 3 вредности има PR, јазлите од кои тој зависи по 2,  $\rightarrow 3 \cdot 2^2 = 12$
3. јазолот P зависи од GI, двата јазли имаат по 2 можни вредности  $\rightarrow 2^2 = 4$
4. AAI: Јазолот PR има 3 вредности, јазолот AAI 3 вредности, јазолот P две вредности  $\rightarrow 2 \cdot 3^2 = 18$
5. Вкупната бројка на параметри е:  $2+2+12+4+18=38$

6. Секако, кај бинарните променливи може да скратиме половина од параметрите поради тоа што можеме да ја добиеме другата веројатност одземајќи ја преостанатата од 1. На пример кај SAI ако го скратиме параметарот ко соодветствува на SAI=низок, веројатноста за SAI=низок се добива од  $1-0.6 = 0.4$ .

1. Соодветно, кај променливите со по 3 можни вредности, можеме да скратиме третина од параметрите, со што веројатноста за скратените параметри ќе ја добиеме одземајќи ги останатите 2 од 1. Пример кај PR=висок,  $1-0.25-0.25=0.5$ , за првиот ред, и соодветно за сите други

7. По објаснување кај 6., минималниот број на параметри е:

$$1+1+8+2+12=24$$

г)

P(SAI=висок)	P(SAI=низок)
0.6	0.4

(за GI не е дадена информација па внесувам вредности по мој избор, истово ќе важи и понатаму за останатите веројатности за кои не е дадена информација)

P(GI=има)	P(GI=нема)
0.4	0.6

SAI	GI	P(PR=низок SAI,GI)	P(PR=среден SAI,GI)	P(PR=висок SAI,GI)
висок	има	0.25	0.25	0.5
висок	нема	0.15	0.4	0.45
низок	има	0.15	0.4	0.45
низок	нема	0.3	0.5	0.2

GI	P(P=ИМА GI)	P(P=НЕМА GI)
има	0.35	0.65
нема	0.2	0.8

PR	P	P(AAI=висок PR,P)	P(AAI=умерен PR,P)	P(AAI=низок PR,P)
низок	има	0.43	0.2	0.37
низок	нема	0.4	0.2	0.4
среден	има	0.83	0.1	0.07
среден	нема	0.8	0.1	0.1
висок	има	0.87	0.1	0.03
висок	нема	0.85	0.1	0.05

д)  $P(\text{SAI, GI, PR, P, AAI}) = P(\text{SAI}) * P(\text{GI}) * P(\text{P}|\text{GI}) * P(\text{PR}|\text{SAI,GI}) * P(\text{AAI}|\text{PR,P})$

ѓ)

PR ⊥ P | GI – заедничка причина

$PR \perp\!\!\!\perp P$  – заедничка причина со  $AAI$

$SAI \perp\!\!\!\perp GI$  – заедничка последица

$SAI \perp\!\!\!\perp AAI | PR$  – каузален синџир

$GI \perp\!\!\!\perp AAI | P$  – каузален синџир

$GI \perp\!\!\!\perp AAI | PR$  – каузален синџир

$$\text{е) } P(GI=\text{има}, P=\text{има}) = P(GI=\text{има}) * P(P=\text{има} | GI=\text{има}) = 0.4 * 0.35 = 0.14$$

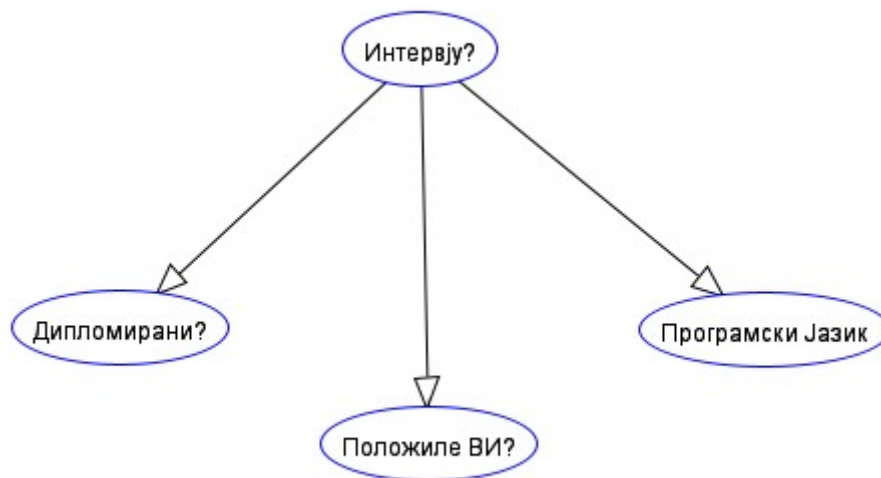
ж)

$$P(PR=\text{низок} | SAI=\text{висок}) = P(PR=\text{низок} | SAI=\text{висок}, GI=\text{има}) + P(PR=\text{низок} | SAI=\text{висок}, GI=\text{нема}) = 0.25 + 0.15 = 0.4$$

## Втор дел: Машинско учење

А)

а)



б)

1. Најпрво треба да се најдат maximum likelihood estimates за сите карактеристики:

<u>интервју</u>	<u>P(интервју)</u>
<u>да</u>	0.5
<u>не</u>	0.5

<u>дипломирани</u>	<u>интервју</u>	<u>P(дипломирани интервју)</u>
<u>да</u>	<u>да</u>	0.6
<u>да</u>	<u>не</u>	0.4
<u>не</u>	<u>да</u>	0.4
<u>не</u>	<u>не</u>	0.6

<u>положиле ви</u>	<u>интервју</u>	<u>P(положиле ви интервју)</u>
<u>да</u>	<u>да</u>	0.8
<u>да</u>	<u>не</u>	0.4
<u>не</u>	<u>да</u>	0.2
<u>не</u>	<u>не</u>	0.6

<u>јазик</u>	<u>интервју</u>	<u>P(јазик интервју)</u>
<u>java</u>	<u>да</u>	0.4
<u>java</u>	<u>не</u>	0.6
<u>python</u>	<u>да</u>	0.6
<u>python</u>	<u>не</u>	0.4

Потоа да се изврши лапласовото порамнување со  $k=2$ :

интервју	P(интервју)	
да	0.5	
не	0.5	
дипломирани	интервју	P(дипломирани интервју)
да	да	0.5555555555555556
да	не	0.4444444444444444
не	да	0.4444444444444444
не	не	0.5555555555555556
положиле ви	интервју	P(положиле ви интервју)
да	да	0.6666666666666667
да	не	0.4444444444444444
не	да	0.3333333333333333
не	не	0.5555555555555556
јазик	интервју	P(јазик интервју)
java	да	0.4444444444444444
java	не	0.5555555555555556
python	да	0.5555555555555556
python	не	0.4444444444444444

потоа треба да пресметаме здружени веројатности за секој исход со новиот примерок, па да споредиме која веројатност е поголема, тогаш така е биде класифициран новиот примерок

$$P(\text{интервју}=\text{да}, \text{дипломиран}=\text{да}, \text{полагал ви}=\text{не}, \text{јазик}=\text{java})=0.04115226$$

$$P(\text{интервју}=\text{да}) * P(\text{дипломиран}=\text{да} | \text{интервју}=\text{да}) * P(\text{положил ви}=\text{да} | \text{интервју}=\text{да}) * P(\text{јазик}=\text{java} | \text{интервју}=\text{да})=0.04115226$$

$$P(\text{интервју}=\text{не}, \text{дипломиран}=\text{да}, \text{полагал ви}=\text{не}, \text{јазик}=\text{java})=0.06858711$$

$$P(\text{интервју}=\text{не}) * P(\text{дипломиран}=\text{да} | \text{интервју}=\text{не}) * P(\text{положил ви}=\text{да} | \text{интервју}=\text{не}) * P(\text{јазик}=\text{java} | \text{интервју}=\text{не})=0.06858711$$

Според добиениот резултат, примерокот ќе се класифицира во класата интервју=не

Овој конкретен примерок би се класифицирал во истата класа дури и да немавме лапласово порамнување

(табелите и пресметките за веројатноста се направени во LibreOffice Calc за полесно и побрзо прикажување и пресметување, но постапката за добивање на сите е запазена исто како што ни е учено)

## Б) Перцептрон

а) Соодветен перцептрон за овој проблем би бил перцептрон со три влезни неврони и уште еден за BIAS влез, кој ќе биде сетиран на 1 бидејќи не е наведено во дефиницијата на проблемот. Перцептронот ќе има 1 излезен неврон, кој исто како и влезните, ќе биде бинарен.

Ефектот на параметри еднакви на нула точно не може да се согледа, поради постоењето на контрадикторност во резултатите за секој влез. На пример, за кандидати 6 и 7, кои имаат идентични влезови, се добива различна класа. Ова ми укажува на тоа дека постојат и други фактори кои влијаат врз одлуката дали еден кандидат ќе добие интервју или не.

$w=[1, 0, 0, 0]$

Ефектот од вредностите поставени во почетниот тежински вектор  $w$  ќе биде дека се преферираат позитивни класи.

б)

Почетно, треба влезовите да се трансформираат во бинарни влезови. Така, влез од „да“ или „java“ нека биде 1, а влез од „не“ или „python“ нека биде 0.

#кандидат	Дипломирани?	Положиле ви?	Програмски јазик	Интервју?
1	1	1	1	1
2	1	1	1	0
3	1	1	1	1
4	0	1	1	0
5	0	1	0	1
6	0	0	0	0
7	0	0	0	1
8	0	0	1	0
9	1	0	0	0
10	1	1	0	1

Потоа можеме да ја покажеме епохата:

$f(x)=[BIAS, d, v, j]$

#1

$w=[1, 0, 0, 0]$

точна класа=1 -> +1

предвидена:

$w*f(x) = 1 * bias + 0*1 + 0*1 + 0*1 = 1$

предвидена==точна класа -> нема промена во векторот на тежини

#2

точна = 0 -> -1

предвидена:

$w*f(x) = 1*bias + 0*1+0*1+0*1 = 1$

предвидена!=точна -> векторот на тежини мора да се намали

$w = w-f = [1,0,0,0]-[1,1,1,1] = [0,-1,-1,-1]$

#3

точна=1 -> +1

предвидена:

$w*f(x)=0*bias-1*1-1*1-1*1=-3$

треба да се ажурира тежинскиот вектор

$w=w+f=[0,-1,-1,-1]+[1,1,1,1]=[1,0,0,0]$

#4

точна класа=0 ->-1

предвидена:

$$w*f(x)=1*bias+0*0+0*1+0*1=1$$

треба да се ажурира тежинскиот вектор

$$w=w-f=[1,0,0,0]-[1,0,1,1]=[0,0,-1,-1]$$

#5

точна класа=1->+1

предвидена:

$$w*f(x)=0*bias+0*0+1*1+0*0=1$$

нема потреба од ажурирање

#6

точна класа=0->-1

предвидена:

$$w*f(x)=0*bias+0*0+-1*0+-1*0=0$$

бидејќи преферираме позитивна класа, резултат од 0 го сметам како позитивен резултат, што значи дека и сега ќе треба ажурирање на векторот

$$w=w-f=w-[0,0,0,0]=[0,0,-1,-1] \text{ (во овој случај останува ист поради влезовите од кандидатот)}$$

#7

точна класа=1->

предвидена: (влезот е ист како претходниот кандидат, векторот останува ист, и нема потреба од „ажурирање“ бидејќи овојпат е погодена класата)

$$w*f(x)$$

#8

точна класа=0

предвидена:

$$w*f(x)=[0,0,-1,-1]*[1,0,0,1]=-1$$

нема потреба од ажурирање

#9

точна класа=0->-1

предвидена:

$$w*f(x)=[0,0,-1,-1]*[1,1,0,0]=0$$

треба ажурирање:

$$w=w-f=[0,0,-1,-1]-[1,1,0,0]=[-1,-1,-1,-1]$$

#10

точна класа=1->+1

предвидена:

$$w*f(x)=[-1,-1,-1,-1]*[1,1,1,0]=-3$$

треба ажурирање:

$$w=w+f=[-1,-1,-1,-1]+[1,1,1,0]=[0,0,0,-1]$$

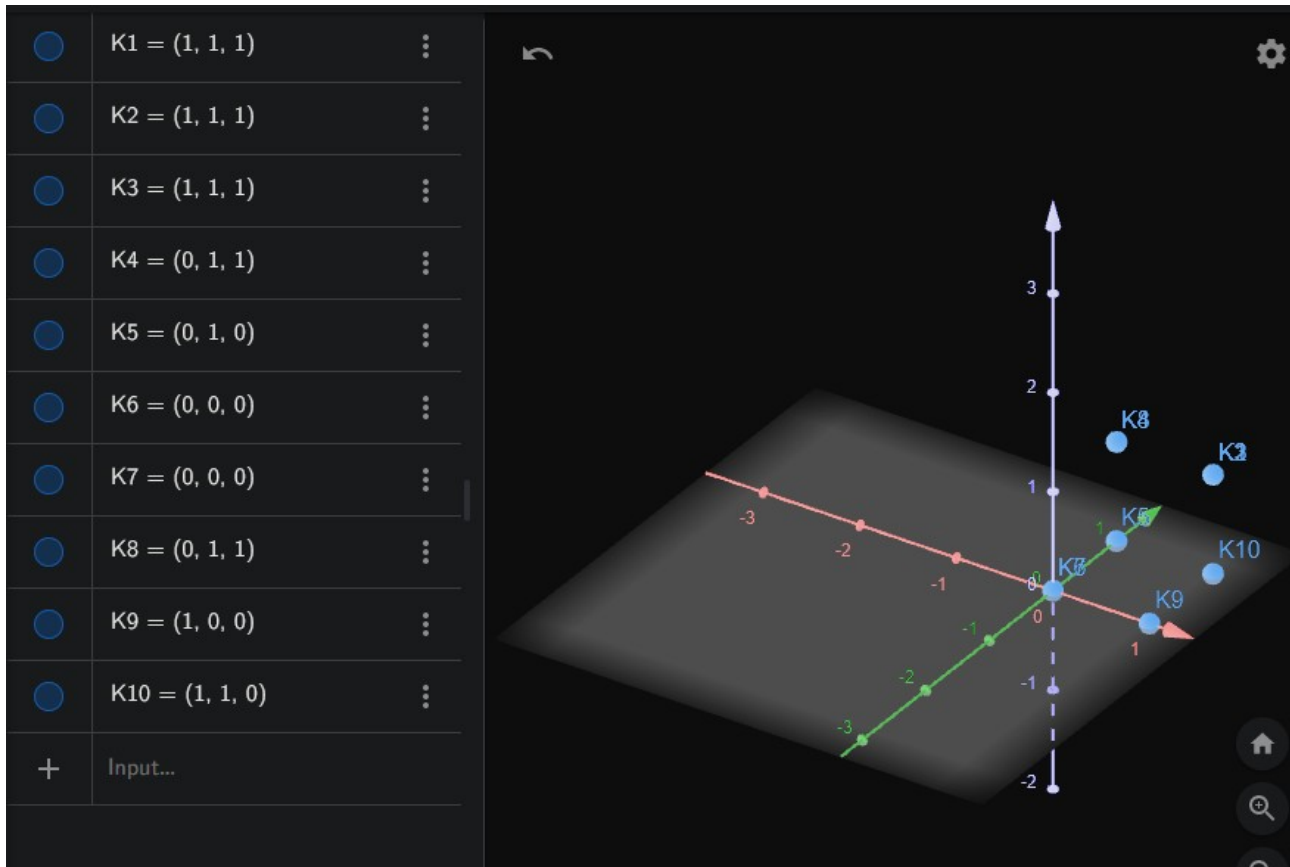
со тоа покажавме една епоха од учењето на овој перцептрон



в)

Еден начин за покажување дали перцептронот ќе конвергира е користејќи хиперрамнина. Доколку податоците се линеарно сепарабилни со таа хиперрамнина, тогаш перцептронот може да конвергира.

Ќе искористам 3Д графички приказ користејќи Geogebra, за да ги прикажам податоците визуелно.



Како што може да се согледа, не можат сите кандидати да се поделат со некоја хиперрамнина, најочигледно е кај кандидатите K1 до K3, кои имаат ист влез, а се со класи 1,0,1 соодветно, па се наоѓаат на иста локација во 3Д просторот.

Тоа значи дека овој перцептрон НЕМА да конвергира

## Б)Дрво на одлучување

а)

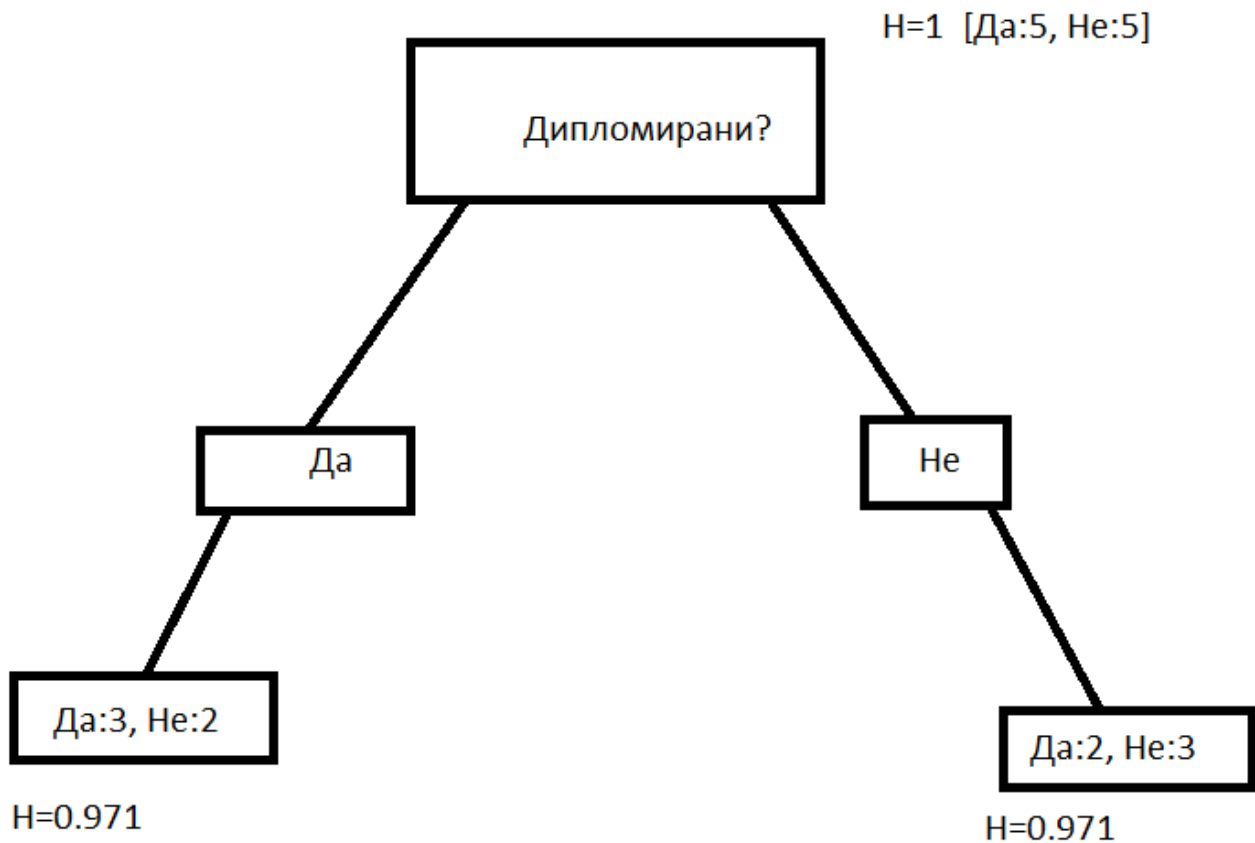
Треба да ја пресметаме информациската добивка за сите атрибути, и потоа го земаме за корен оној атрибут за кој сме пресметале најголема информациска добивка

(Дефинирам своја функција во LibreOffice Calc за да ги убрзам пресметките, да не треба да ги внесувам одделно во калкулатор)

```
REM ***** BASIC *****
```

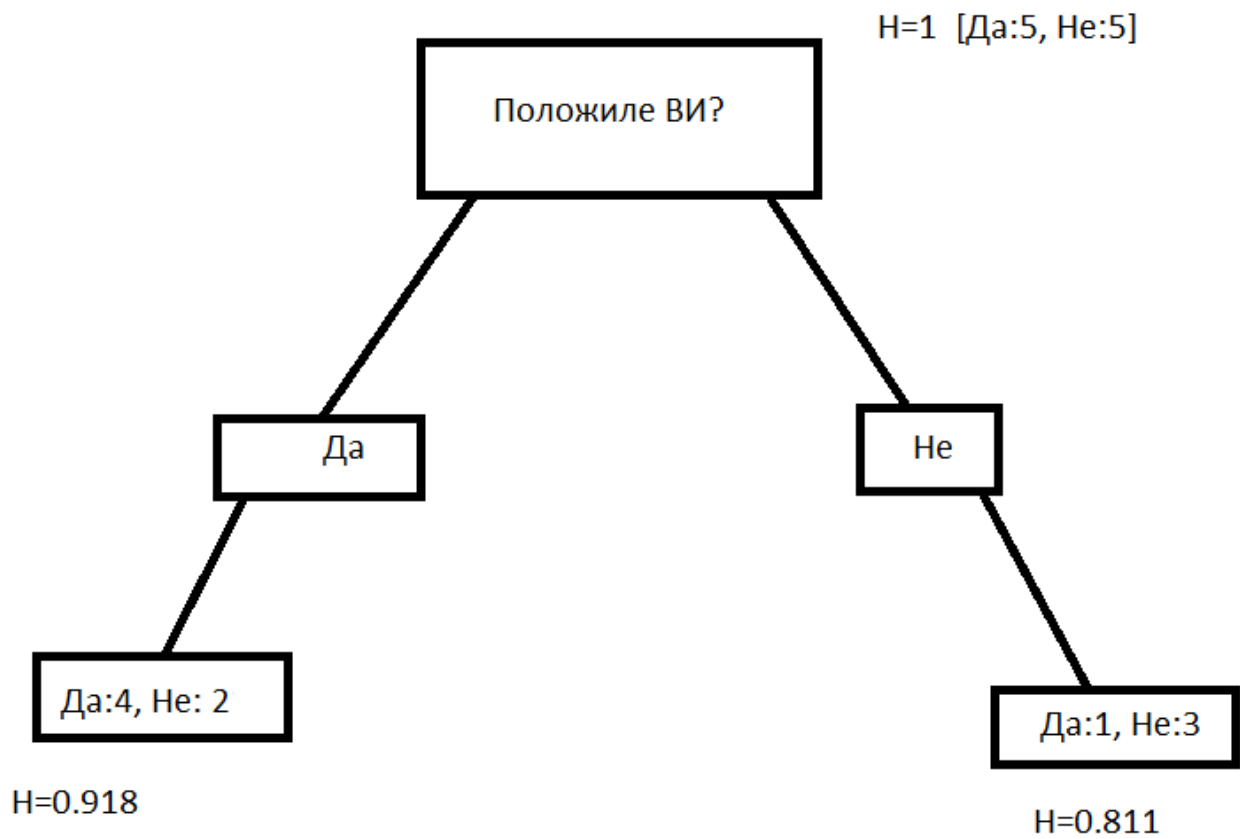
```
Function ENTROPY(a,b)  
    ENTROPY=(-a)*(Log(a)/Log(2))-b*(Log(b)/Log(2))  
End Function
```

1. И.Д. за атрибут „Дипломирани?“



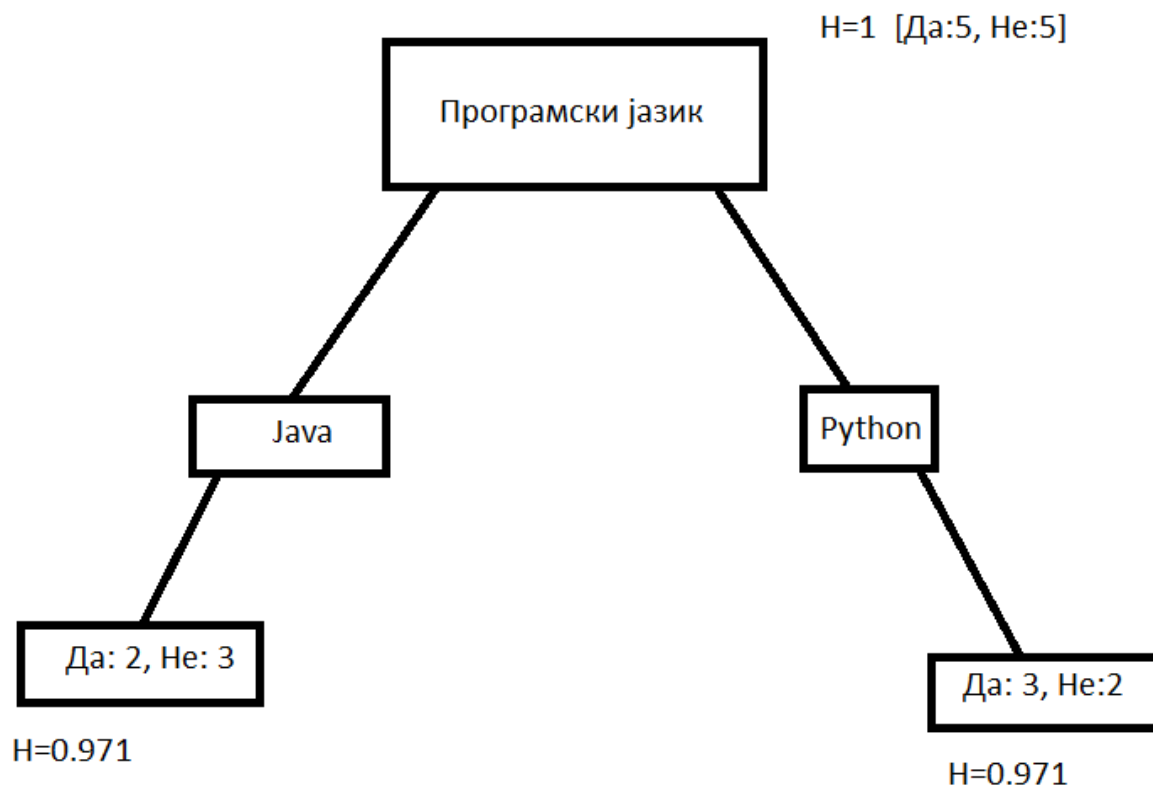
$$GAIN=1-5/10*0.971-5/10*0.971=0.029$$

2. И.Д. за атрибутот „Положиле ВИ?“



$$\text{GAIN} = 1 - 6/10 * 0.918 - 4/10 * 0.811 = 0.1248$$

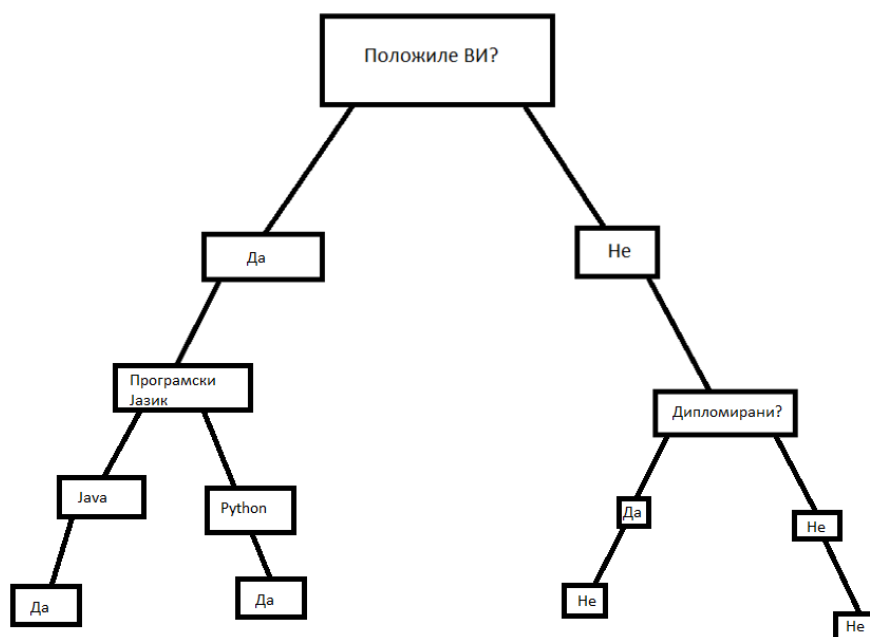
3. И.Д. за атрибутот „Програмски јазик“



$$\text{GAIN}=1-5/10*0.971-5/10*0.971=0.029$$

Јасно, најголема информациска добивка имаме кај атрибутот „Положиле ВИ?“, па него го бираме за корен во дрвото

б)



На левата страна од дрвото, како дополнителен услов за гранење го зедов податокот за програмски јазик бидејќи имаше поголема информациска добивка во однос на податоците за дипломирани.

На десната страна од дрвото, земено е случајно дипломирани, бидејќи и програмски јазик и дипломирани имаа исти информациски добивки, и дополнително кај двете поддрва се добиваше класификација „Не“ независно од тоа дали програмскиот јазик бил јава или пајтон, или пак дали кандидатот бил дипломиран или не.

Сите терминиращки јазли се наоѓаат во второто ниво.

Терминиращкиот јазол „да“ кај програмски јазик Јава го добив преку мнозинска одлука враќајќи се едно ниво погоре, кај родителот, бидејќи од тие што положили ви и имале јазик Јава, 2 кандидати имаа интервју, и 2 немаа. Кај родителот, од тие што положили ВИ, 4 од нив добиле интервју, а 2 не.

Следниот терминиращки јазол „да“ кај јазикот Python го добив по совршена класификација.

Имено, сите кандидати со јазик python, кои положили ВИ, добиле интервју,

Следниот терминиращки јазол „не“ од оние кои не положили ВИ и дипломирале, го добив по совршена класификација, бидејќи единствениот дипломиран кој не положил ВИ, беше класифициран во класата „не“, т.е. немаше интервју.

И последниот терминиращки јазол „не“ го добив со мнозинска одлука, бидејќи од оние кои не дипломирале и не положили ВИ, двајца не добиле интервју, а еден добил.

Дополнително, финалното дрво може да се поткастри со тоа што може целосно да се поткастрат двата дела, бидејќи независно од тоа дали јазикот им е Јава или Python, или пак дипломирале или не, дрвото на одлука ќе класифицира нов примерок со „да“ или со „не“, соодветно.

