Класификација

Ненад Митић

Математички факултет
nenad.mitic@matf.bg.ac.rs

Увод

Улазни материјал садржи слогове облика
$S_i = (x_i^1, x_i^2, ..., x_i^k, y_i)$ где је атрибут y од посебног интереса

x_1^1	x_1^2	x_1^3	x_1^4	...	y_1
x_2^1	x_2^2	x_2^3	x_2^4	...	y_2
x_3^1	x_3^2	x_3^3	x_3^4	...	y_3
x_4^1	x_4^2	x_4^3	x_4^4	...	y_4
x_5^1	x_5^2	x_5^3	x_5^4	...	y_5
x_6^1	x_6^2	x_6^3	x_6^4	...	y_6
...	...	...	...	...	...
x_n^1	x_n^2	x_n^3	x_n^4	...	y_n

Циљ је уочити да ли се на основу вредности атрибута x_i може одредити вредност атрибута y

Увод

- Задатак класификације: одредити функцију (класификациони модел) која пресликава сваки скуп атрибута $X = (x^1, x^2, ..., x^k)$ у једну од предефинисаних вредности y
- y - ознака класе
  - y категорички атрибут $\Longrightarrow$ класификација
  - y је нумерички atribut $\Longrightarrow$ регресија
- Циљ формирања модела: примена на материјал код кога је вредност атрибута y непозната ради што прецизнијег предвиђања вредности y

Увод

- Како проверити карактеристике модела?
- Улазни подаци се обично деле у два дисјунктна дела
  - податке за тренинг помоћу којих се формира модел и који се користе за иницијално одређивање тачности модела
  - податке за тестирање који се користе за проверу исправности модела
- При провери се упоређују предвиђене вредности класа са стварним вредностима

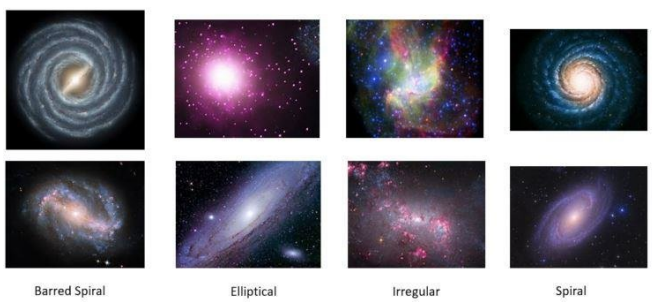
Увод


Tid Attrib1 Attrib2 Attrib3 Class
1 Yes Large 125K No
2 No Medium 100K No
3 No Small 70K No
4 Yes Medium 120K No
5 No Large 95K Yes
6 No Medium 60K No
7 Yes Large 220K No
8 No Small 85K Yes
9 No Medium 75K No
10 No Small 90K Yes
Подаци за формирање модела (тренинг)
Алгоритам (алгоритми) за учење
Индукција
Учење модела
Модел(и)
Примена модела
Закључивање
Tid Attrib1 Attrib2 Attrib3 Class
11 No Small 55K ?
12 Yes Medium 80K ?
13 Yes Large 110K ?
14 No Small 95K ?
15 No Large 67K ?
Подаци за проверу модела (тест)

Увод

Примери класификација

- Ћелије тумора (бенигни, малигни)
- Исправност коришћења кредитних картица
- Предвиђање секунарне структуре протеина
- Одређивање типа текстова (спорт, политика, ...)
- Одређивање типа галаксија



- ...

Технике класификације

- Основне технике/методе класификације
  - Дрвета одлучивања
  - Правила
  - Неуронске мреже
  - Статистички засноване методе
  - Машине са потпорним векторима
  - Одређивање најближег суседа
  - ...
- Методе ансамбла
  - Појачавање (енг. *Boosting*)
  - Паковање (енг. *Bagging*)
  - Насумична шума (енг. *Random forest*)
  - ...

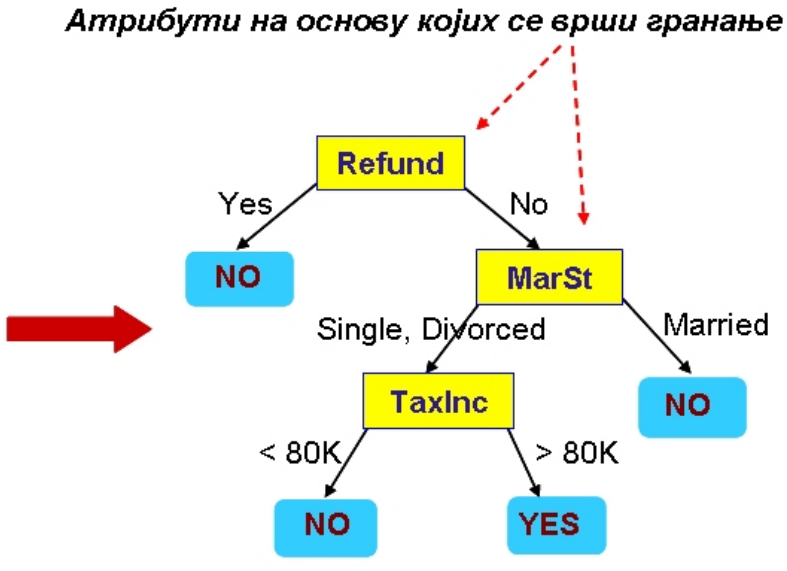
Основна идеја

- Модел се формира у облику дрвета
- У не-лист чвору се налазе питања на основу којих се врши гранање
- Улазни материјал се дели на основу вредности атрибута и питања у чвору
- Листови дрвета одређују ознаке класа

Пример дрвета одлучивања

	категорички	категорички	непрекидни	класа
Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

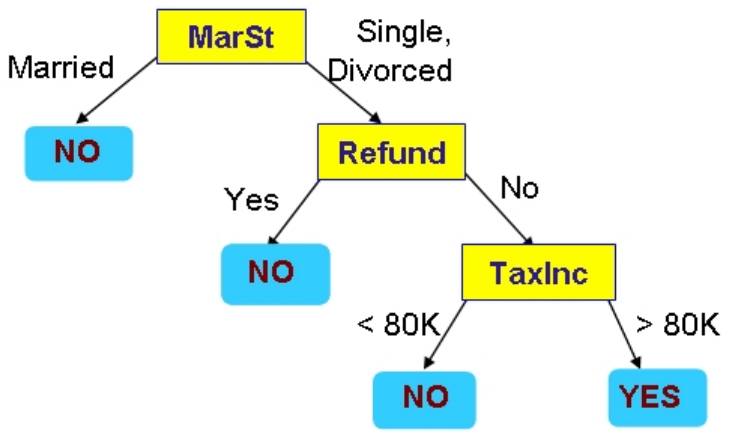
Подаци за тренинг (прављење модела



Модел: дрво одлучивања

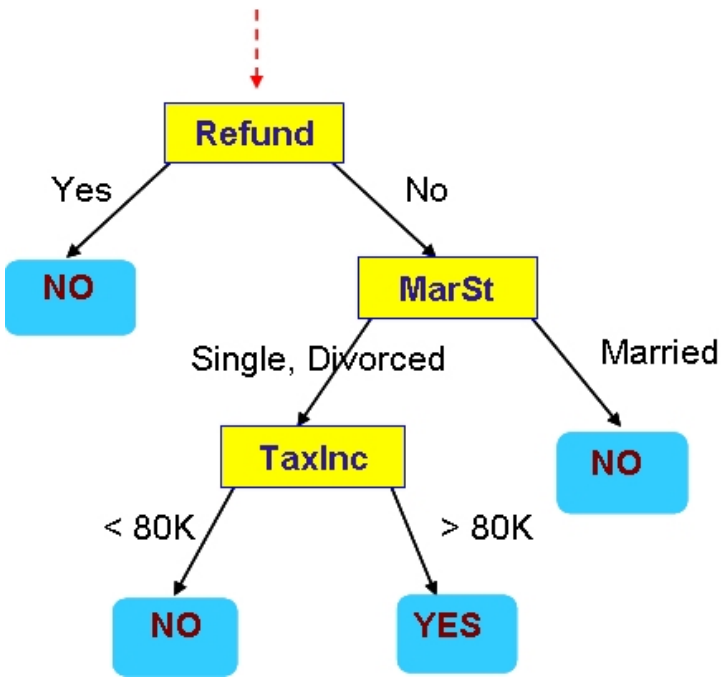
Пример дрвета одлучивања (наставак)

	категорички	категорички	непрекидни	класа
Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



Могуће је да постоји више дрвета (модела) која одговарају истим подацима

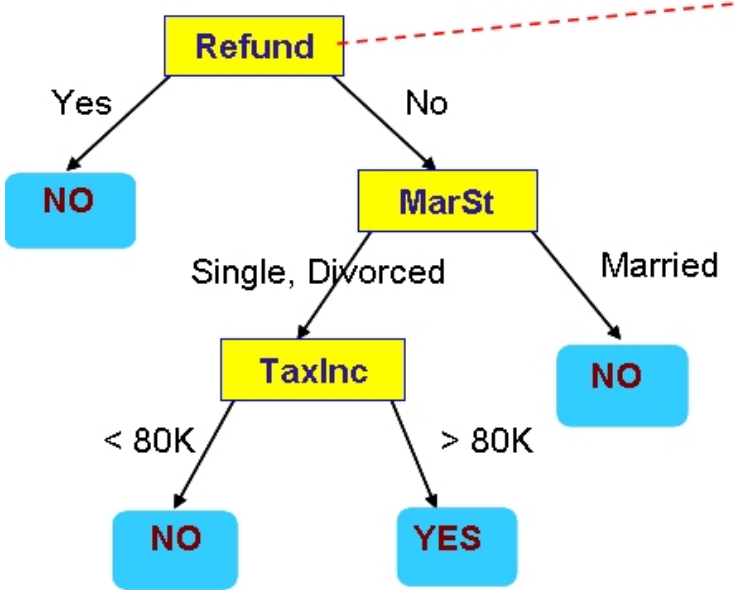
Примена модела - пример



Подаци за проверу
(тестни подаци)

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

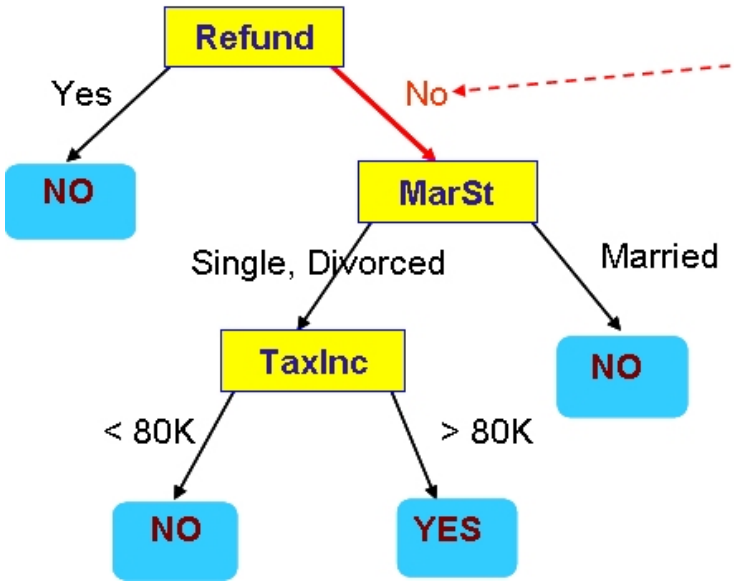
Примена модела - пример (наставак)



Подаци за проверу
(тестни подаци)

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

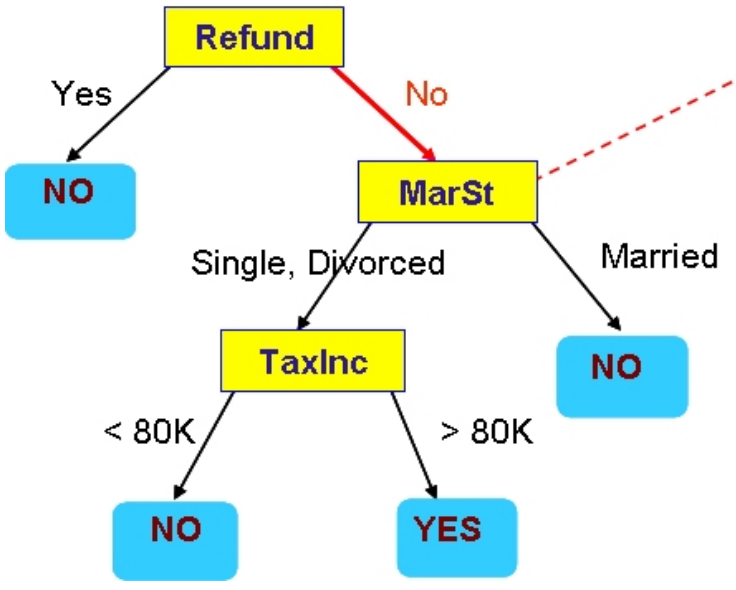
Примена модела - пример (наставак)



Подаци за проверу
(тестни подаци)

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

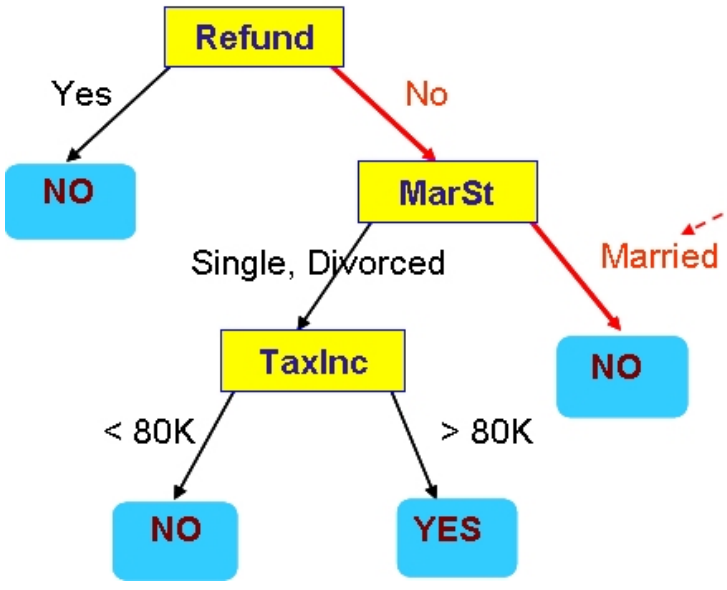
Примена модела - пример (наставак)



Подаци за проверу
(тестни подаци)

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

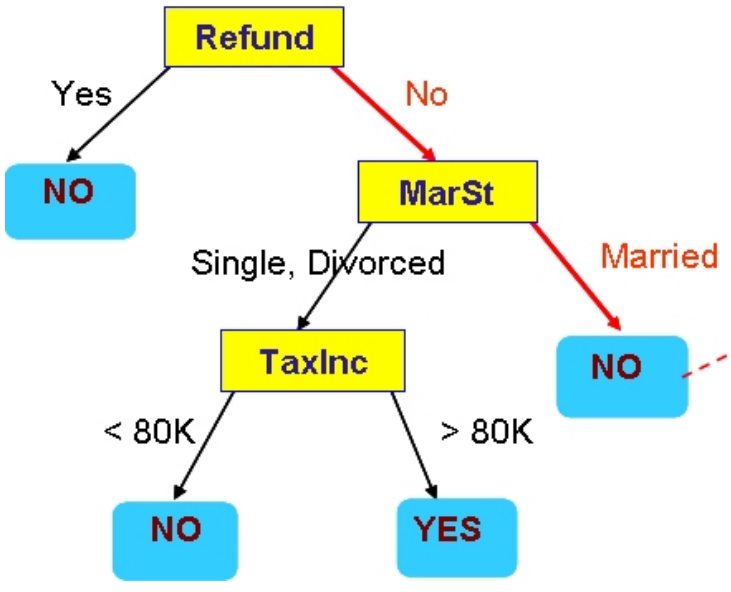
Примена модела - пример (наставак)



Подаци за проверу
(тестни подаци)

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Примена модела - пример (наставак)



Подаци за проверу
(тестни подаци)

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Доделити “Не”
атрибуту “Cheat”

Конструкција дрвета одлучивања

Изазови

- Како одабрати атрибут(е) по коме се врши подела?

Конструкција дрвета одлучивања

Изазови

- Како одабрати атрибут(е) по коме се врши подела?
- Како формирати упит за различите типове атрибута?

Конструкција дрвета одлучивања

Изазови

- Како одабрати атрибут(е) по коме се врши подела?
- Како формирати упит за различите типове атрибута?
- Како одредити најбољу поделу?

Конструкција дрвета одлучивања

Изазови

- Како одабрати атрибут(е) по коме се врши подела?
- Како формирати упит за различите типове атрибута?
- Како одредити најбољу поделу?
- На који начин индуктивно применити претходне критеријуме на дрво по дубини?

Конструкција дрвета одлучивања

Изазови

- Како одабрати атрибут(е) по коме се врши подела?
- Како формирати упит за различите типове атрибута?
- Како одредити најбољу поделу?
- На који начин индуктивно применити претходне критеријуме на дрво по дубини?
- Када стати са конструкцијом дрвета?

Конструкција дрвета одлучивања

Изазови

- Како одабрати атрибут(е) по коме се врши подела?
- Како формирати упит за различите типове атрибута?
- Како одредити најбољу поделу?
- На који начин индуктивно применити претходне критеријуме на дрво по дубини?
- Када стати са конструкцијом дрвета?
- Начин рада са недостајућим вредностима?

Конструкција дрвета одлучивања

Изазови

- Како одабрати атрибут(е) по коме се врши подела?
- Како формирати упит за различите типове атрибута?
- Како одредити најбољу поделу?
- На који начин индуктивно применити претходне критеријуме на дрво по дубини?
- Када стати са конструкцијом дрвета?
- Начин рада са недостајућим вредностима?
- Који је критеријум за процену грешке у генерализацији?

Конструкција дрвета одлучивања

Изазови

- Како одабрати атрибут(е) по коме се врши подела?
- Како формирати упит за различите типове атрибута?
- Како одредити најбољу поделу?
- На који начин индуктивно применити претходне критеријуме на дрво по дубини?
- Када стати са конструкцијом дрвета?
- Начин рада са недостајућим вредностима?
- Који је критеријум за процену грешке у генерализацији?
- Цена и перформансе модела?

Конструкција дрвета одлучивања

Изазови

- Како одабрати атрибут(е) по коме се врши подела?
- Како формирати упит за различите типове атрибута?
- Како одредити најбољу поделу?
- На који начин индуктивно применити претходне критеријуме на дрво по дубини?
- Када стати са конструкцијом дрвета?
- Начин рада са недостајућим вредностима?
- Који је критеријум за процену грешке у генерализацији?
- Цена и перформансе модела?
- ...

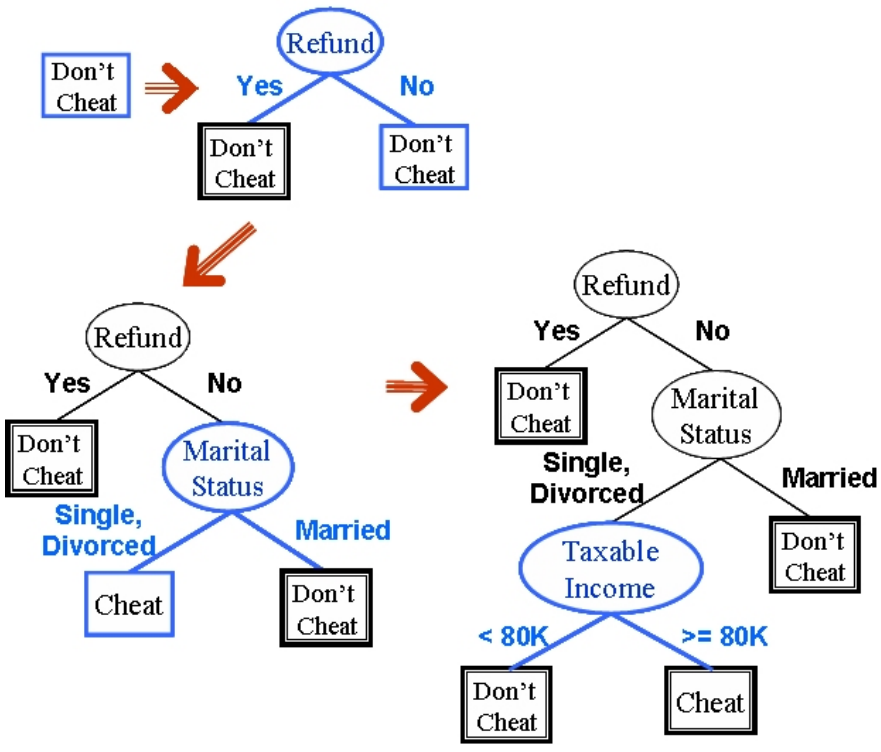
Алгоритми

- ID3 (Iterative Dichotomiser 3)
- C4.5
- C5.0
- CART (Classification And Regression Trees)
- CHAID (CHi-squared Automatic Interaction Detection)
- Exhaustive CHAID
- QUEST (Quick, Unbiased, Efficient Statistical Trees)
- SLIQ (Supervised Learning In Quest)
- SPRINT (Scalable PaRallelizable INduction and decision Trees)
- ...

Општи модел

- Општа процедура за конструкцију дрвета
- Описује начин поделе материјала у чвору
- Ако је D_t скуп слогова за тренинг који се налазе у чвору t и нека су $y = \{y_1, y_2, ..., y_c\}$ ознаке класа. Општи поступак је
  - Ако D_t садржи само слогове који припадају истој класи y_t тада је тај чвор лист означен по називу класе y_t
  - Ако D_t садржи слогове који се налазе у више од једне класе, тада се на основу тест атрибута врши подела материјала у мање подскупове на које се рекурзивно примењује комплетна процедура

Општи модел - илустрација



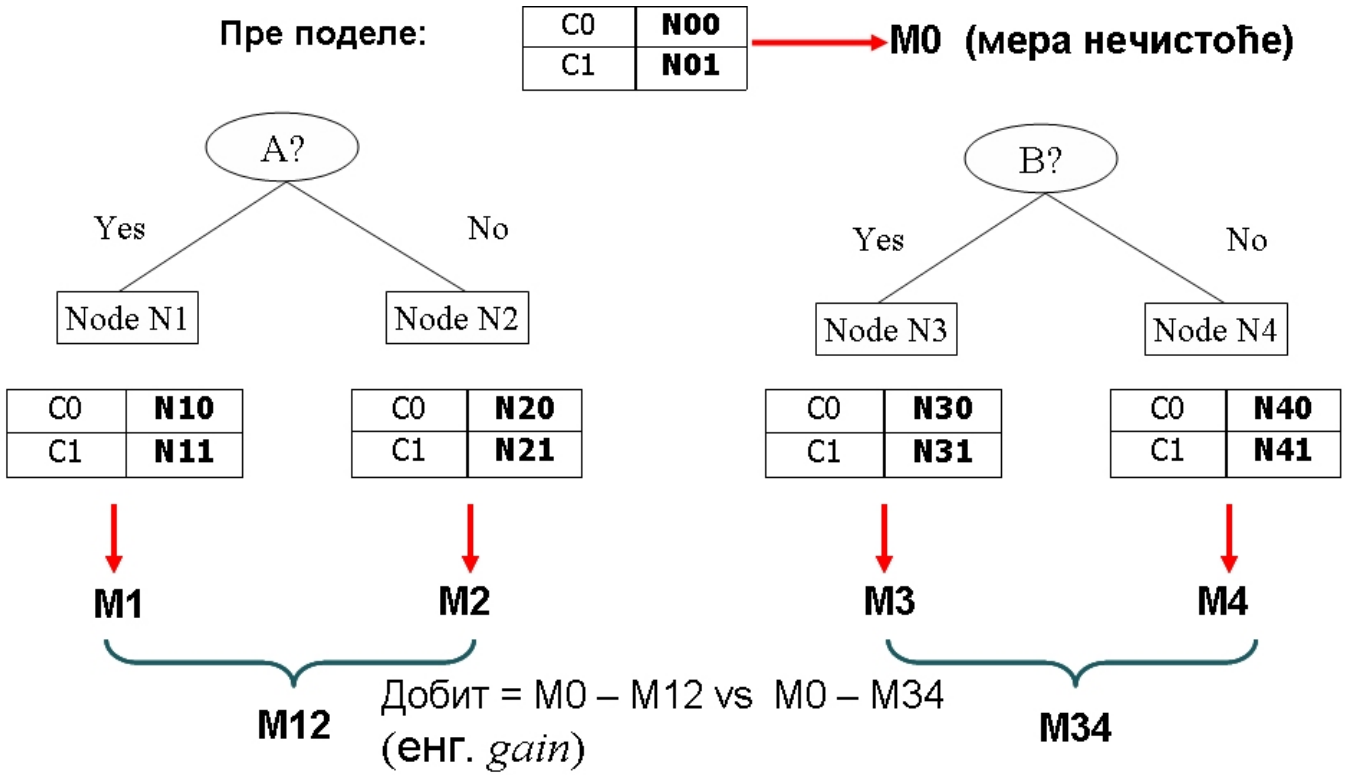
Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Избор атрибута за поделу

- Атрибут за поделу материјала се бира стратегијом похлепе (енг. *greedy*)
  - Поделити слогове према атрибуту који оптимизује одређени критеријум
- Одлуке које треба донети
  - Како поделити слогове и која је најбоља подела?
  - Како навести услове за тестирање атрибута
  - Када стати са деобом

Одређивање најбоље поделе

Најбоља подела се одређује помоћу *мере нечистоће* - мера хомогености дистрибуције класа у чвору



Мере нечистоће

$$\text{Gini}(t) = 1 - \sum_{i=1}^{c} [p(i \mid t)]^2$$

$$\text{Ентропија}(t) = -\sum_{i=1}^{c} p(i \mid t) log_2 p(i \mid t)$$

$$\text{Грешка класификације}(t) = 1 - \max_i p(i \mid t)$$

gde $p(j \mid t)$ означава релативну учесталост класе j у чвору t, c број класа, и важи $0 * log_2 0 = 0$ при рачунању ентропије

Мера нечистоће

- Гинијев индекс (*Corrado Gini*, италијански статистичар)
- Максимум ($1 - 1/n_c$ za Ginijev indeks i grešku klasifikacije, $log_2 n_c$ за ентропију) када су слогови равномерно распоређени у свим класама садржи најмање интересантне информације
- Минимум (0.0) када сви слогови припадају једној класи садржи најинтересантније информације

C_1	0
C_2	6
Gini=0.000	

C_1	1
C_2	5
Gini=0.278	

C_1	2
C_2	4
Gini=0.444	

C_1	3
C_2	3
Gini=0.500	

Мере нечистоће - примери израчунавања

Чвор H_1	Број
Класа 1	0
Класа 2	6

Гини $=1-(0/6)^2-(6/6)^2=0$
Ентропија $=-(0/6)log_2(0/6)-(6/6)log_2(6/6)=0$
Грешка класификације$=1-max[0/6,6/6]=0$

Чвор H_1	Број
Класа 1	1
Класа 2	5

Гини $=1-(1/6)^2-(5/6)^2=0.278$
Ентропија $=-(1/6)log_2(1/6)-(5/6)log_2(5/6)=0.650$
Грешка класификације$=1-max[1/6,5/6]=0.167$

Чвор H_1	Број
Класа 1	3
Класа 2	3

Гини $=1-(3/6)^2-(3/6)^2=0.5$
Ентропија $=-(3/6)log_2(3/6)-(3/6)log_2(3/6)=1$
Грешка класификације$=1-max[3/6,3/6]=0.5$

Критеријум поделе у чвору

Посматра се разлика мере нечистоће у родитељ чвору и деци чворовима. Што је разлика већа то је подела квалитетнија
Нека $I(.)$ означава нечистоћу чвора, N укупан број слогова у родитељ чвору, $N(v_j)$ број слогова у дете чвору v_j и број различитих вредности атрибута

- Добит за Гини

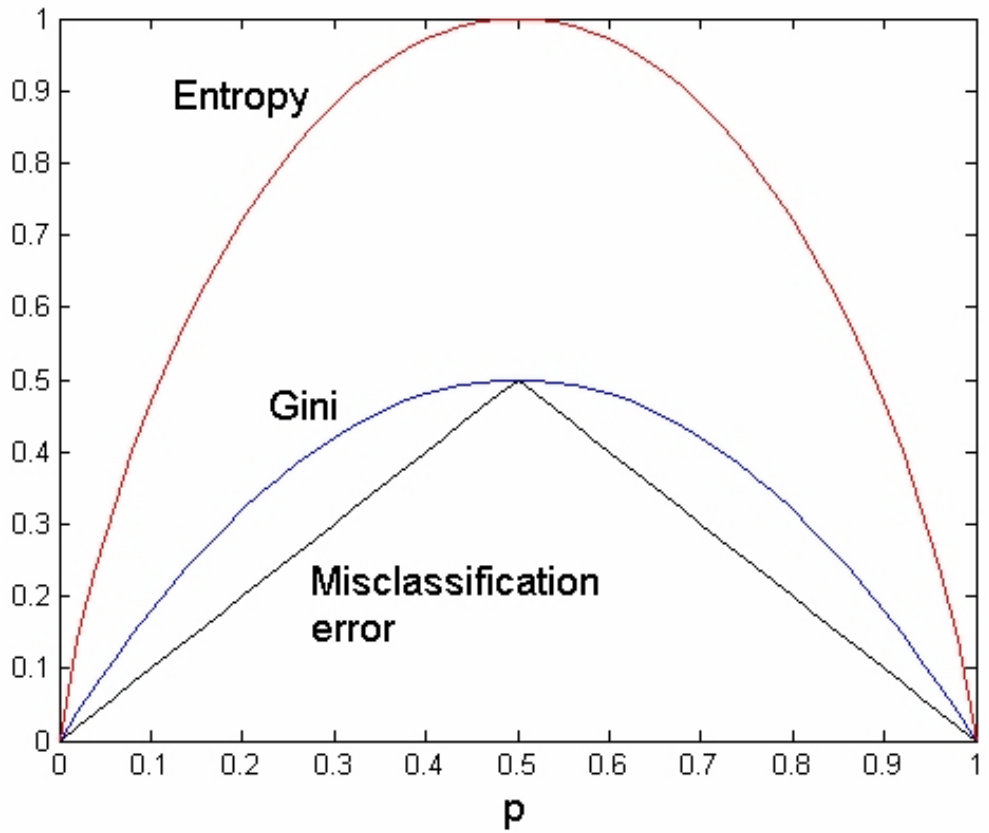
$$\Delta = \text{Гини}(\text{родитељ чвор}) - \sum_{j=1}^{k} \frac{N(v_j)}{N} I(v_j)$$

- Добит за Ентропију (информациона добит)

$$\Delta = \text{Ентропија}(\text{родитељ чвор}) - \left(\sum_{j=1}^{k} \frac{N(v_j)}{N} \text{Ентропија}(v_j) \right)$$

Поређење мера нечистоће

Пример за проблем бинарне класификације



Услови тестирања атрибута за поделу

- Зависност од типа атрибута
  - Бинарни
  - Именски
  - Редни
  - Непрекидни
- Зависност од броја начина за поделу
  - Подела на две гране
  - Подела на више грана

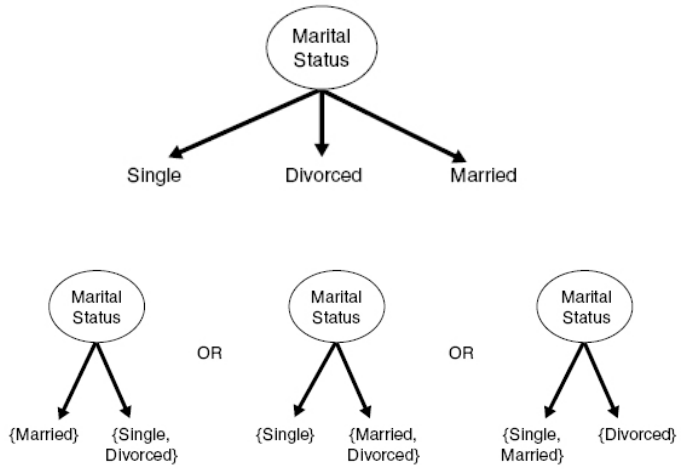
Подела заснована на бинарним и именским атрибутима

Бинарни атрибути

- Увек се дели на две гране

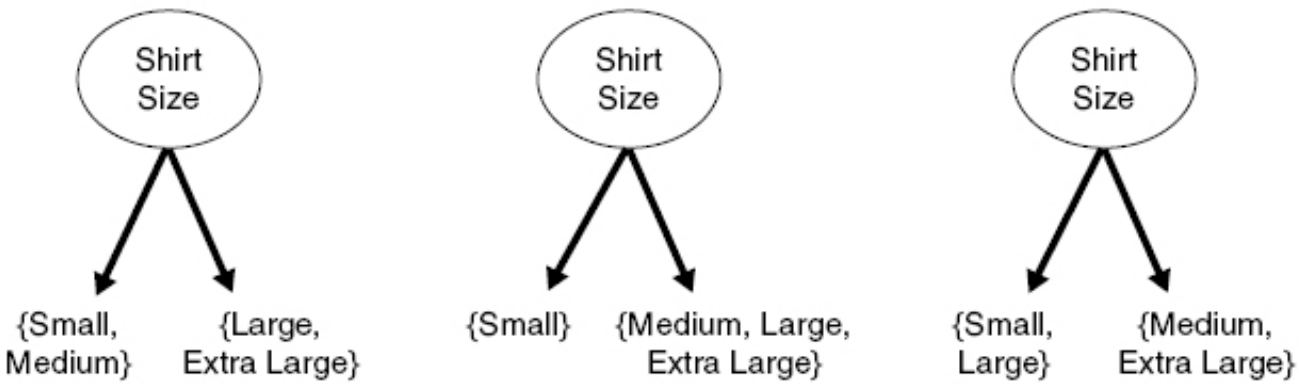
Именски атрибути

- Подела на више грана (колико атрибут има различитих вредности)
- Подела на две гране - подела у два подскупа. Треба наћи оптималну поделу



Подела заснована на редним атрибутима

- Бинарна подела или подела на више грана
- Да ли су све поделе коректне? Очување редоследа!



Подела заснована на непрекидним атрибутима

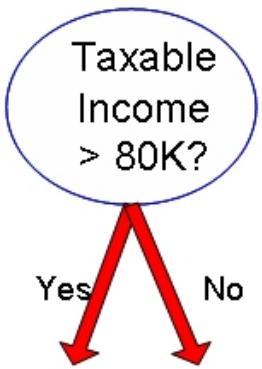
Различити начини рада

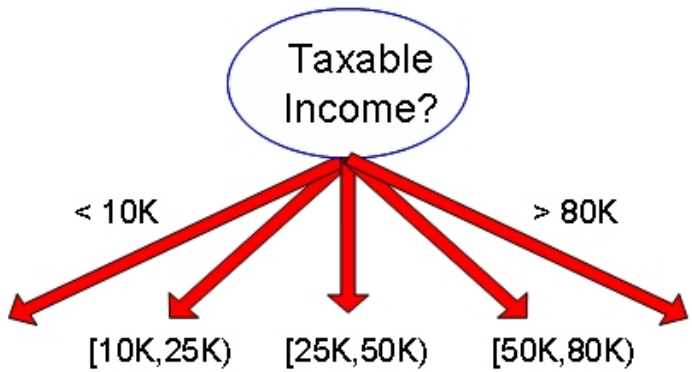
- Дискретизацијом - формирање редних категоричких атрибута
- Бинарна подела: $A < v$ или $A \geq v$
  - Разматрају се све могуће поделе у зависности од v
  - Рачунарски интензиван посао

Подела заснована на непрекидним атрибутима (наставак)

Различити начини рада

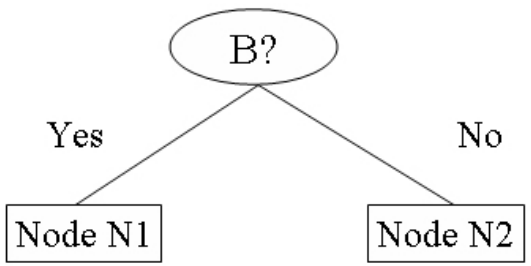
- Подела на k интевала: $v_i \leq A < v_{i+1}, i = 1, ..., k$
  - Разматрају се сви могући интервали
  - Рачунарски интензиван посао





Подела по бинарним атрибутима - израчунавање Гини индекса

- Скуп се дели у две партиције
- Ефекти тежина партиција - пожељне су веће и чистије



	Parent
C1	6
C2	6
Gini = 0.500	

Gini(N1)
= 1 – (5/7)² - (2/7)²
= 0.408

Gini(N2)
= 1 – (1/5)² – (4/5)²
= 0.320

	N1	N2
C1	5	1
C2	2	4
Gini=0.371		

Gini(dete)
= 7/12 * 0.408 +
5/12 * 0.320
= 0.371

Подела по категоричким атрибутима - израчунавање Гини индекса

- За сваку од различитих вредности атрибута израчуна се број слогова у свакој класи података
- Одреди се Гинијев индекс за сваки чвор и пронађе минимална вредност

Подела у више грана

	CarType		
	Family	Sports	Luxury
C1	1	2	1
C2	4	1	1
Gini	0.393		

Подела у две групе
(налази се најбоља подела по вредности)

	CarType	
	{Sports, Luxury}	{Family}
C1	3	1
C2	2	4
Gini	0.400	

	CarType	
	{Sports}	{Family, Luxury}
C1	2	2
C2	1	5
Gini	0.419	

Подела по непрекидним атрибутима - израчунавање Гини индекса

За ефикасно израчунавање за сваки атрибут се

- врши сортирање по вредностима
- добијене вредности линеарно скенирају уз ажурирање броја и слогова и рачунање Гинијевог индекса
- бира се позиција за поделу са најмањим Гинијевим индексом

Cheat	No		No		No		Yes		Yes		Yes		No		No		No		No	
	Taxable Income																			
Сортиране вредности →	60		70		75		85		90		95		100		120		125		220	
Позиције поделе →	55		65		72		80		87		92		97		110		122		172	230
	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>
Yes	0	3	0	3	0	3	0	3	1	2	2	1	3	0	3	0	3	0	3	0	3	0
No	0	7	1	6	2	5	3	4	3	4	3	4	3	4	4	3	5	2	6	1	7	0
Gini	0.420		0.400		0.375		0.343		0.417		0.400		*0.300*		0.343		0.375		0.400		0.420	

Алтернативни критеријум поделе

- Мере као што су Гинијев индекс и ентропија фаворизују атрибуте са великим бројем различитих вредности
- У неким случајевима се добијају погрешни резултати (нпр. број индекса, матични број, ...)
- Стратегије превазилажења проблема
  - употреба искључиво бинарне поделе
  - укључивање броја могућих вредности

$$\text{Однос добити за поделу} = \frac{\Delta_{podela}}{Split\_info}$$

$$Split\_info = -\sum_{i=1}^{k} P(v_i) log_2 P(v_i)$$

где је k број подела атрибута

Модел алгоритма за формирање дрвета одлучивања

```
/* Skup atributa F; skup trening slogova E */
Formiranje drveta(E, F )
if uslov_zaustavljanja(E,F) then
   list=formiraj_cvor()
   labela lista=Klasifikuj(E)
   return list
else
   koren=formiraj_cvor()
   koren.test_uslov = nadji_najbolju_podelu(E,F)
   let V={v|v je moguci ishod za koren.test_uslov}
   for each v in V do
       E_v = {e | koren.test_uslov(e) = v i e pripada E}
       dete = Formiranje_drveta(E_v,F)
       dodaj dete cvor kao potomak korenog cvora i
             oznaci granu (koren-dete) sa v
    end for
end if
return koren

/* Posle formiranj drvo treba da se potkrese */
```

Критеријум заустављања

Формирање нових чвора деобом слога текућег чвора се зауставља (функција uslov_zaustavljanja из претходног алгоритма) када

- сви слогови у текуће, чвору припадају истој класи
- сви слогови имају исте вредности (свих) атрибута
- испуњен критеријум ранијег заустављања (достигнута дубина дрвета, ...)

Карактеристике дрвета одлучивања

- Јевтина за конструисање
- Јако брза у класификацији непознатог материјала
- Лака за интерпретацију за дрвета мале величине
- Прецизност је упоредива са осталим техникама класификације за једноставне типове података
- Избор мере нечистоће нема велики утицај на перформансе

Карактеристике дрвета одлучивања

- Применљивост (на све типове података)
- Изражајност (могу да представе сваку функцију дискретних атрибута)
- Ефикасност израчунавања
- Рад са недостајућим вредностима
- Рад са ирелевантним атрибутима и редундантним атрибутима

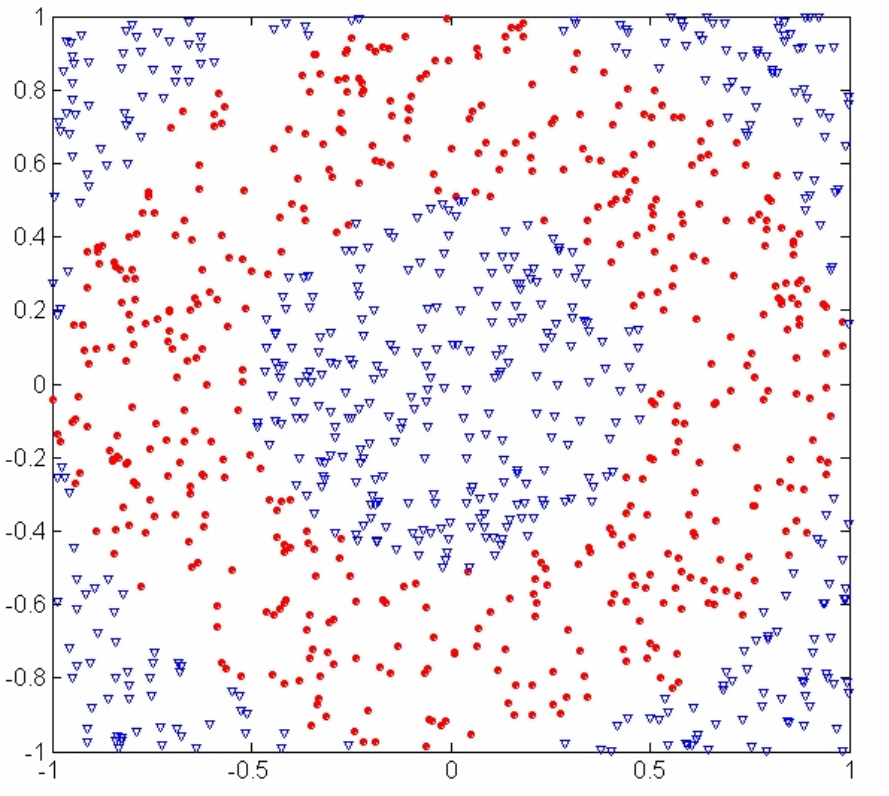
Карактеристике дрвета одлучивања

- Слаба могућност рада са повезаним атрибутима (спајање два атрибута даје значајну информацију)
- Критеријум поделе само један атрибут $\longrightarrow$ интервал је правоугаоног облика
- Критеријум поделе комбинација више атрибута $\longrightarrow$ могуће је обрадити и нпр. *косе* податке
- Избор начина поткресивања значајно утиче на резултате
- Проблем преприлагођавања и потприлагођавања

Преприлагођеност и потприлагођеност

- Модел који исувише добро класификује податке за тренинг може да има лошије карактеристике при генерализацији од модела који има већу грешку у процесу прављења модела - превише прилагођен модел (преприлагођен модел, енг. *model overfitting*)
- Ако је модел исувише једноставан грешке при тренирању и уопштавању могу да буду јако високе - премало прилагођен модел (потприлагођен модел, енг. *model underfitting*)

Преприлагођеност и потприлагођеност



500 кружних и 500 троугаоних тачака

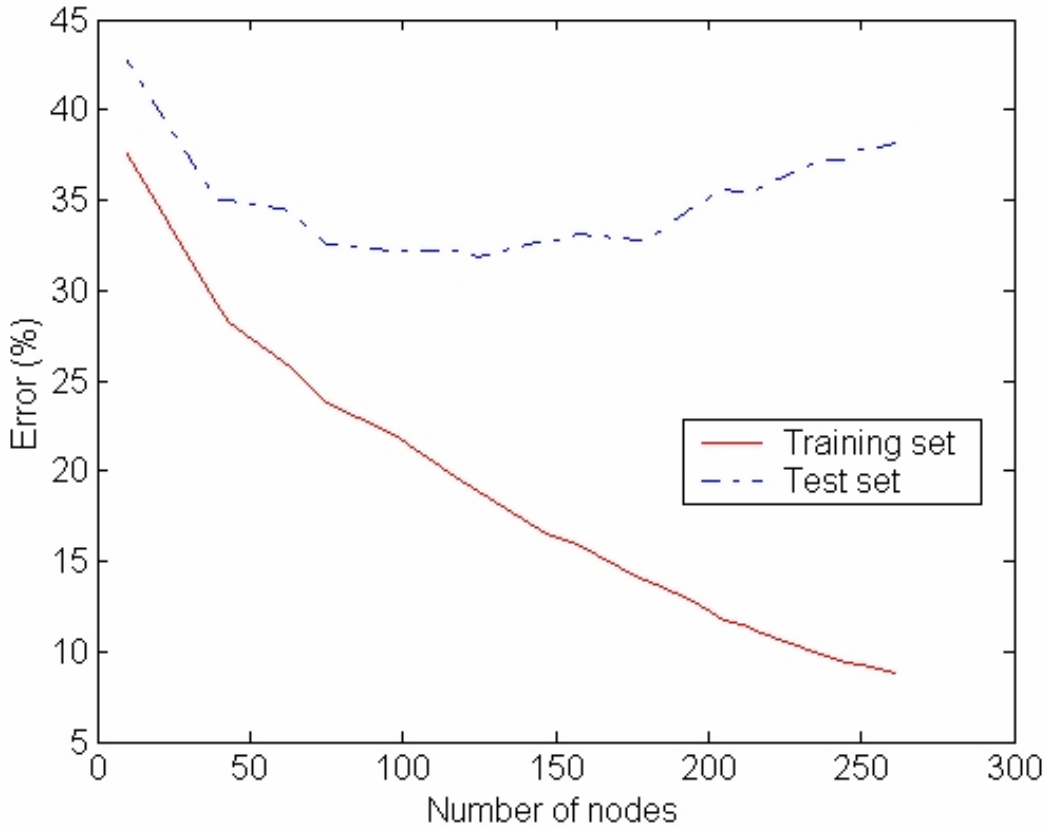
Кружне тачке:
$0.5 \leq \text{sqrt}(x_1^2+x_2^2) \leq 1$

Троугаоне тачке:
$\text{sqrt}(x_1^2+x_2^2) < 0.5$ или
$\text{sqrt}(x_1^2+x_2^2) > 1$

30% тачака се бира за тренинг, остале за тест

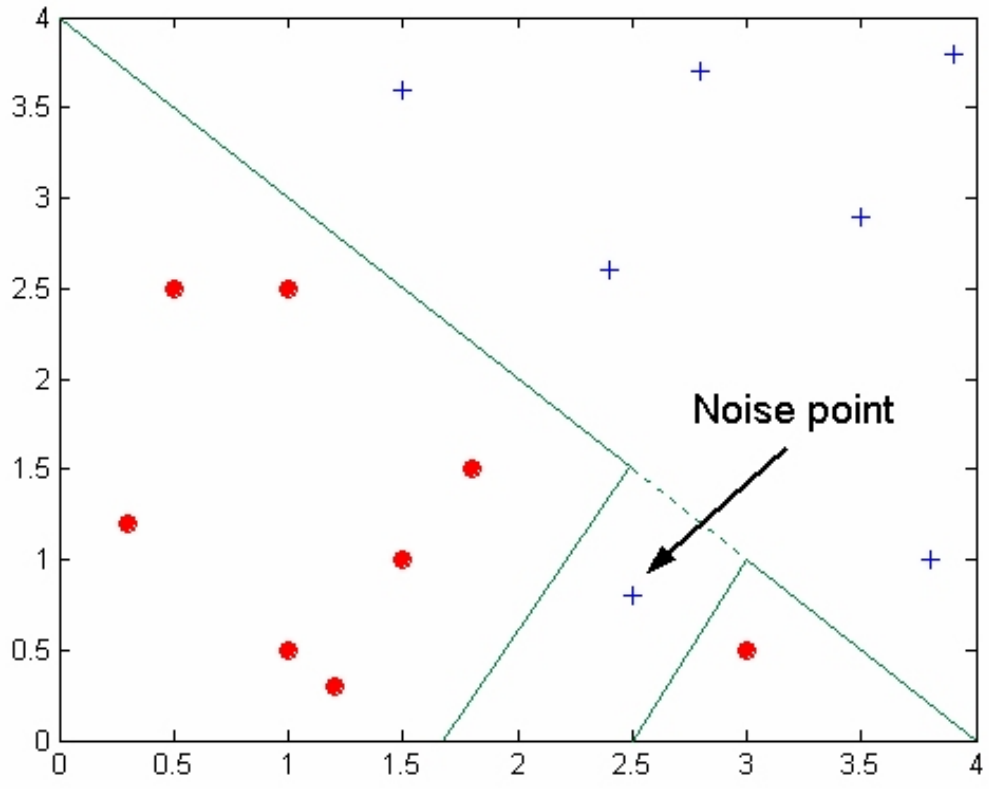
Дрво са Гинијевим индексом као мером се примењује на тренинг податке

Преприлагођеност и потприлагођеност



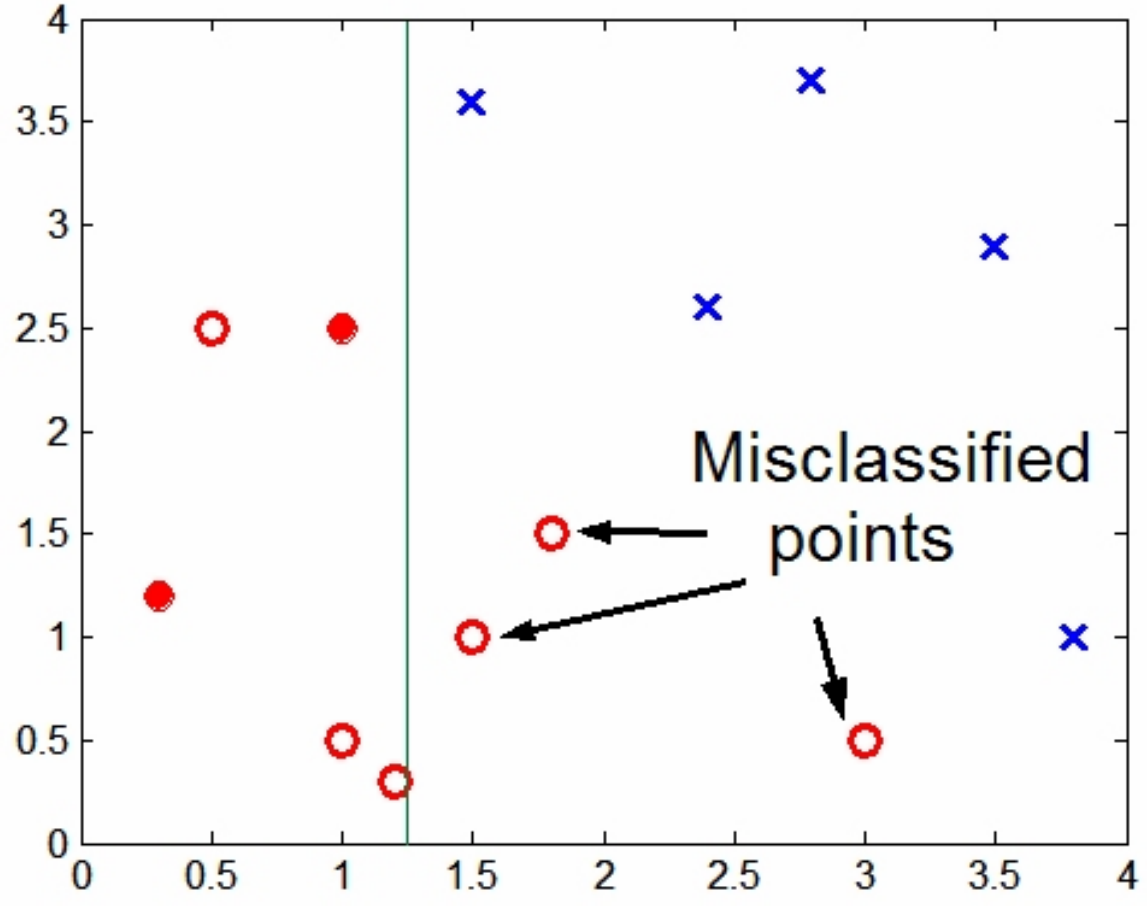
Потприлагођавање: ако је модел исувише једноставан и тренинг и тест грешка су велике.
Поткресивањем дрвета на различитим нивоима

Преприлагођеност и потприлагођеност



Границе поделе могу да се искриве и због постојања шума

Преприлагођеност и потприлагођеност



Преприлагођеност може да се јави због непостојања репрезентативних примера

Преприлагођеност и потприлагођеност

- Преприлагођеност се јавља код дрвета која су сложенија него што је потребно
- У том случају грешка при тренирању не даје коректну процену понашања дрвета у случају примене на претходно непознате податке
- Потребно је наћи начин за процену грешке при примени на тестне и непознате податке

Процена грешке у генерализацији

- Нека је T дрво, t чвор, N број листова у дрвету T, $e(t)$ број погрешно класификованих слогова у t, и $e(T)$ укупан број грешака у класификацији по дрвету T
- Грешка поновне замене (грешка при тренирању) $min(\sum e(t))$ - бира се модел са најмањом грешком при тренирању
- Грешка при генерализацији: грешка при тестирању $\sum e'(t)$

Процена грешке у генерализацији

Методе за процену грешке у генерализацији

- Оптимистички приступ: $e'(t) = e(t)$
- Песимистички приступ: $e'(t) = e(t) + 0.5$ (лист)

$$e'(T) = \frac{\sum_{i=1}^{k} [e(t_i) + \Omega]}{\sum_{i=1}^{k} n(t_i)} = e(T) + \frac{k\Omega}{N} \quad \text{(дрво)}$$

где је k broj listova, Ω цена придружена сваком листу t, $e(T)$ укупна грешка дрвета у фази тренинрања, а N укупан број тренинг слогова

Процена грешке у генерализацији

На пример, за $\Omega = 0.5$ укупан број грешака је $e'(T) = e(T) + N \times 0.5$. За дрво са 30 листова и 10 грешака на тренинг скупу од 1000 ставки:

- Грешка тренирања = 10/1000 (1%)
- Грешка уопштавања = (10+ 30× 0.5)/1000 (2.5%)

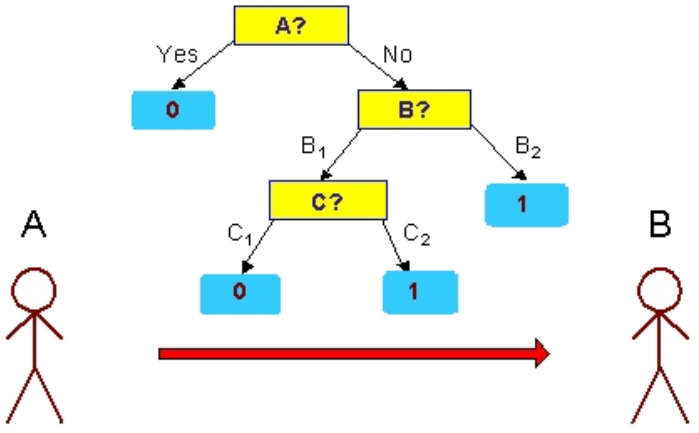
- Смањење грешке поткресивањем (дрвета)
  - Користи се посебан скуп података за процену грешке уопштавања

Процена грешке у генерализацији

- Што је модел сложенији веће су шансе за преприлагођеност
- Параметар процене је и сложеност модела
- Модел не треба непотребно усложњавати
- Од два модела са сличном грешком генерализације изабрати онај који је једноставнији - Окамов (*Occm*) принцип (жилет)
- Код сложених модела већа је шанса да случајно дође до уклапања због грешаак у подацима

Принцип најмање дужине описа (*MDL*)

X	y
X_1	1
X_2	0
X_3	0
X_4	1
...	...
X_n	1

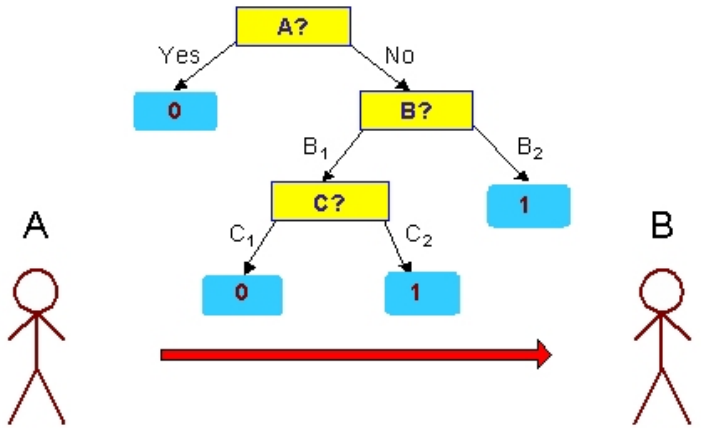


X	y
X_1	?
X_2	?
X_3	?
X_4	?
...	...
X_n	?

- Начин да се укључи сложеност модела
- Дат је скуп слогова са познатим атрибутом X
- Особа A познаје све вредности класа атрибута y
- Особа B нема ту информацију и може да пошаље захтев A да проследи све ознаке класа - $\Theta(n)$ битова где је n број слогова
- Алтернативно, A може да направи модел, кодира и пошаље B

Принцип најмање дужине описа (*MDL*)

X	y
X_1	1
X_2	0
X_3	0
X_4	1
...	...
X_n	1



X	y
X_1	?
X_2	?
X_3	?
X_4	?
...	...
X_n	?

- Ако је модел 100% тачан цена преноса је једнака цени кодирања модела. Ако није, потребне су и информације о некоректно класификованим слоговима
- Цена (модел,подаци)= Цена (модел) + Цена (подаци|модел), где је цена број битова потребан за енкодирање
- Према *MDL*-у тражи се модел који минимизује укупну цену

Баратање преприлагођавањем у индукцији по дрвету

Пре-поткресивање (правило ранијег заустављања)

- Алгоритам се зауставља пре него што дрво нарасте до максималне величине
- Типични услови заустављања:
  - све инстанце припадају истој класи
  - вредности свих атрибута (сем атрибута класе) су исте
  - број инстанци је мањи од неке унапред задате границе
  - дистрибуција инстанци је независна од расположивих атрибута (нпр. види се применом χ^2 теста)
  - ширење текућег чвора не побољшава меру чистоће

Баратање преприлагођавањем у индукцији по дрвету

Поткресивање по завршетку

- Дрво одлучивања расте до крајњих граница
- Исеку се чворови у дрвету од дна ка врху
- Ако се грешка генерализације побољша после отсецања поддрво се замени са чвором који је лист
- Ознака класе листа се одређују према већини класа инстанци поддрвета
- За поткресивање по завршетку се може користити и *MDL*

Пример поткресивања по завршетку

Class = Yes	20
Class = No	10
Error = 10/30	

Тренинг грешка (пре деобе) = 10/30

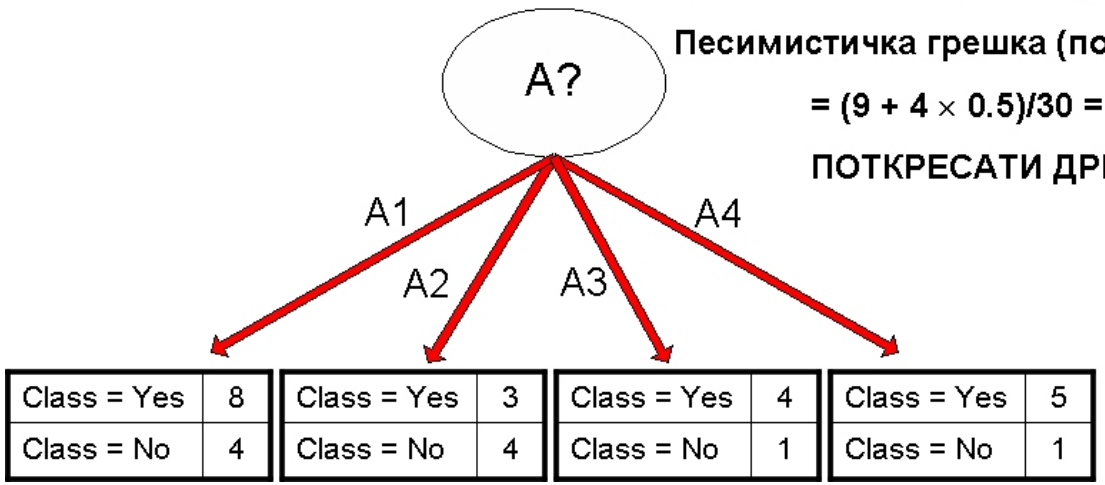
Песимистичка грешка = (10 + 0.5)/30 = 10.5/30 за Ω=0.5

Тренинг грешка (после деобе) = 9/30

Песимистичка грешка (после деобе)

= (9 + 4 $\times$ 0.5)/30 = 11/30

ПОТКРЕСАТИ ДРВО!



Руковање атрибутима са недостајућим вредностима

Утицај НВ на дрво одлучивања

- Шта ако атрибут по коме се дели чвор има НВ (како рачунати меру нечистоће)
- Како дистрибуирати инстанце са НВ на децу чворове
- Како класификовати тест инстанцу са НВ

Руковање атрибутима са недостајућим вредностима

Tid	Refund	Marital Status	Taxable Income	Class
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	?	Single	90K	Yes

Недостајућа вредност

Пре поделе: Ентропија (родитељ)
= -0.3 log(0.3)-(0.7)log(0.7) = 0.8813

	Class = Yes	Class = No
Refund=Yes	**0**	**3**
Refund=No	**2**	**4**
Refund=?	**1**	**0**

Подела на REFUND:

Ентропија(Refund=Yes) = 0

Ентропија(Refund=No)
= -(2/6)log(2/6) – (4/6)log(4/6) = 0.9183

Ентропија (дете)
= 0.3 (0) + 0.6 (0.9183) = 0.551

Добит = 0.9 × (0.8813 – 0.551) = 0.3303

Мерење преформанси израчунавања

Утицај НВ на дрво одлучивања

- Нагласак је на предвиђачким особинама модела
  - предност у односу на брзину класификације, изградње модела, скалабилност, ...
- Матрица конфузије

		Предвиђена класа	
		Класа=Да	Класа=Не
Стварна класа	Класа=Да	a	b
	Класа=Не	c	d

- a: TP (true positive)
- b: FN (false negative)
- c: FP (false positive)
- d: TN (true negative)

Мерење преформанси израчунавања

		Предвиђена класа	
		Класа=Да	Класа=Не
Стварна класа	Класа=Да	a TP	b FN
	Класа=Не	c FP	d TN

Најчешће коришћена метрика

$$\text{Тачност} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+FN+FP+TN}$$

Ограничења прецизности

- Размотримо проблем 2 класе
  - Број примерака класе 0 = 9990
  - Број примерака класе 1 = 10
- Ако модел предвиђа да ће сви слогови бити класе 0 тада је прецизност 9990/10000 = 99.9 %
  - Прецизност нема значаја (ни смисла) јер модел не открива ни један слог класе

Матрица цене

		Предвиђена класа	
		Класа=Да	Класа=Не
Стварна класа	Класа=Да	Цена(Да|Да)	Цена(Не|Да)
	Класа=Не	Цена(Да|Не)	Цена(Не|Не)

Цена(а|б) - цена погрешне класификације слога класе а као слога класе б

Израчунавање цене класификације

Матрица цене	Предвиђена класа		
	Цена(а|б	+	-
Стварна класа	+	-1	100
	-	1	0

Модел M1	Предвиђена класа		
		+	-
Стварна класа	+	150	40
	-	60	250

Модел M2	Предвиђена класа		
		+	-
Стварна класа	+	250	45
	-	5	200

Модел M1: Прецизност 80%, Цена 3910

Модел M2: Прецизност 90%, Цена 4255

Мере осетљиве на цену

- Прецизност $p = \frac{a}{a+c}$
- Одзив $r = \frac{a}{a+b}$
- Ф - мера $F = \frac{2rp}{r+p} = \frac{2a}{2a+b+c}$
- Тежинска прецизност $= \frac{w_1 a + w_4 d}{w_1 a + w_2 b + w_3 c + w_4 d}$
- F_β мера $F_\beta = (1+\beta)^2 \times \frac{p \times r}{\beta^2 \times p + r}$

Проверити:

- Да ли је F мера специјалан случај F_β мере
- Шта се добија за $\beta = 0$ а шта за $\beta = \infty$?