Додатне методе класификације

Ненад Митић

Математички факултет
nenad@matf.bg.ac.rs

Увод

- Слогови се класификују помоћу скупа правила облика "ако...онда..."
- Правило: (Услов) $\longrightarrow$ y
  - *Услов* је конјункција атрибута
  - y је ознака класе
  - Лева страна правила: (пред)услов
  - Десна страна правила: последица
  - Пример правила за класификацију:
    - (Тип крви=Топла $\wedge$ (Носи јаја=Да) $\longrightarrow$ Птице
    - (Опорезиви приход $<$ 50K) $\wedge$ (Враћа=Да) $\longrightarrow$ избегава=Не

Пример

Назив	Тип крви	Живорођено	Може да лети	Живи у води	Класа
човек	топла	да	не	не	сисар
питон	хладна	не	не	не	рептил
лосос	хладна	не	не	да	риба
кит	топла	да	не	да	сисар
жаба	хладна	да	не	повремено	амфибија
комодо	хладна	не	не	не	рептил
слепи миш	топла	да	да	не	сисар
голуб	топла	не	да	не	птица
мачка	топла	да	не	не	сисар
леопард ајкула	хладна	да	не	да	риба
корњача	хладна	не	не	повремено	рептил
пингвин	топла	не	не	повремено	птица
бодљикаво прасе	топла	да	не	не	сисар
јегуља	хладна	не	не	да	риба
саламандер	хладна	не	не	повремено	амфибија
гуштер гила	хладна	не	не	не	рептил
кљунар	топла	не	не	не	сисар
сова	топла	не	да	не	птица
делфин	топла	да	не	да	сисар
орао	топла	не	да	не	птица

Класификатори засновани на правилима (пример)
П1: (Живорођено = не) $\wedge$ (Може да лети = да) $\longrightarrow$ Птица
П2: (Живорођено = не) $\wedge$ (Живи у води = да) $\longrightarrow$ Риба
П3: (Живорођено = да) $\wedge$ (Тип крви = топла) $\longrightarrow$ Сисар
П4: (Живорођено = не) $\wedge$ (Може да лети = не) $\longrightarrow$ Рептил
П5: (Живи у води = повремено) $\longrightarrow$ Амфибија

Прецизност и одзив правила

- Одзив правила: проценат броја слогова који задовољавају леву страну правила
- Прецизност правила: проценат броја слогова који задовољавају десну страну правила од слогова који задовољавају леву страну правила
- Пример: за правило

  П5: (Живи у води = повремено) $\longrightarrow$ Амфибија

  одзив=25% (5/20), прецизност=50% (2/4)

Правило и покривање инстанци

Правило П покрива(обухвата) инстанцу x ако вредност атрибута инстанце задовољавају услов правила

Скуп правила
П1: (Живорођено = не) $\wedge$ (Може да лети = да) $\longrightarrow$ Птица
П2: (Живорођено = не) $\wedge$ (Живи у води = да) $\longrightarrow$ Риба
П3: (Живорођено = да) $\wedge$ (Тип крви = топла) $\longrightarrow$ Сисар
П4: (Живорођено = не) $\wedge$ (Може да лети = не) $\longrightarrow$ Рептил
П5: (Живи у води = повремено) $\longrightarrow$ Амфибија

Назив	Тип крви	Живорођено	Може да лети	Живи у води	Класа
соко	топла	не	да	не	??
медвед	топла	да	не	не	??
лемур	топла	да	не	не	??
корњача	хладна	не	не	повремено	??
леопард ајкула	хладна	да	не	да	??

правило П1 покрива: соко $\longrightarrow$ птица
правило П3 покрива: медвед $\longrightarrow$ сисар
правило П3 покрива: лемур $\longrightarrow$ сисар
правила П4, П5 покривају: корњача $\longrightarrow$ рептил или амфибија??
леопард ајкулу не покрива ни једно правило $\longrightarrow$ класа=??

Ограничења скупа правила

- Узајамно искључива правила
  - Не постоје два правила која покривају исту инстанцу
  - Сваки слог је покривен највише једним правилом
- Правила покривају све могућности
  - Класификатор поседује потпуно покривање ако садржи правило за све могуће комбинације вредности атрибута
  - Сваки слог је покривен бар једним правилом

Могући проблеми и решења

- Правила нису узајамно искључива

Могући проблеми и решења

- Правила нису узајамно искључива
- Неке слогове може да покрива више правила. Решење?

Могући проблеми и решења

- Правила нису узајамно искључива
- Неке слогове може да покрива више правила. Решење?
  - Скуп правила уређен по редоследу

Могући проблеми и решења

- Правила нису узајамно искључива
- Неке слогове може да покрива више правила. Решење?
  - Скуп правила уређен по редоследу
  - Неуређен скуп правила – користи се избор (гласачки систем)

Могући проблеми и решења

- Правила нису узајамно искључива
- Неке слогове може да покрива више правила. Решење?
  - Скуп правила уређен по редоследу
  - Неуређен скуп правила – користи се избор (гласачки систем)
- Правила не морају да покривају све могућности

Могући проблеми и решења

- Правила нису узајамно искључива
- Неке слогове може да покрива више правила. Решење?
  - Скуп правила уређен по редоследу
  - Неуређен скуп правила – користи се избор (гласачки систем)
- Правила не морају да покривају све могућности
  - Може да се деси да неки слог није покривен нити једним правилом

Могући проблеми и решења

- Правила нису узајамно искључива
- Неке слогове може да покрива више правила. Решење?
  - Скуп правила уређен по редоследу
  - Неуређен скуп правила – користи се избор (гласачки систем)
- Правила не морају да покривају све могућности
  - Може да се деси да неки слог није покривен нити једним правилом
  - Користи се предефинисана класа

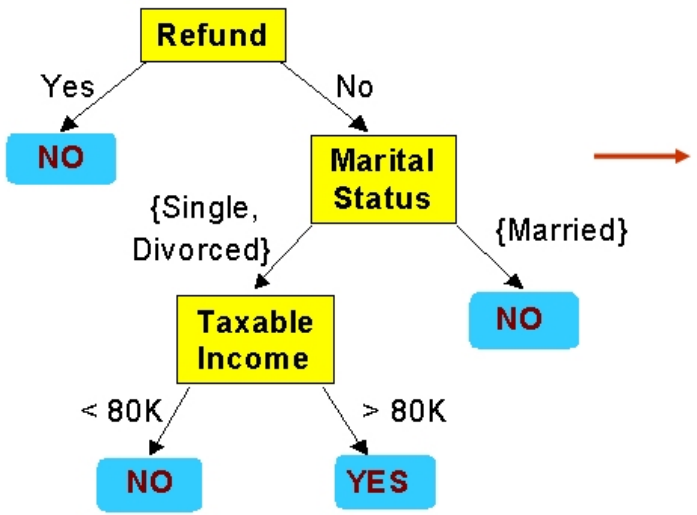
Скуп правила уређен према редоследу

- Правила се рангирају према приоритету
- Када се тестни слог преда класификатору
  - Додели му се ознака класе највишег ранга за коју постоји неко правило које покрива тај слог
  - Ако такво правило не постоји, додељује му се предефинисана класа
- Уређење засновано на правилима
  - Појединачна правила се рангирају према њиховом квалитету
- Уређење засновано на класама
  - Правила која припадају истој класи се групишу једно до другог

Формирање правила за класификацију

- Индиректна метода
  - Правила се издвајају из других класификационих модела (нпр. дрвета одлучивања, неуронских мрежа, итд)
  - Пример алгоритам C4.5rules
- Директна метода
  - Правила се издвајају директно из података
  - Алгоритми RIPPER, CN2, 1R, ...

Од дрвета одлучивања до правила



Classification Rules

(Refund=Yes) ==> No

(Refund=No, Marital Status={Single,Divorced}, Taxable Income<80K) ==> No

(Refund=No, Marital Status={Single,Divorced}, Taxable Income>80K) ==> Yes

(Refund=No, Marital Status={Married}) ==> No

Правила су међусобно искључива и покривају све могућности

Скуп правила садржи исту количину информација као и дрво

Индиректна метода

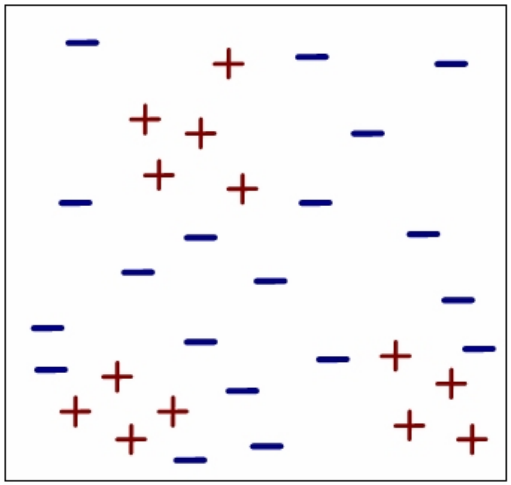
- Најпознатији алгоритам C4.5rules
- Формира правила на основу дрвета одлучивања формираног C4.5 алгоритмом
- Пример на сајту: ИРИС скуп; предвиђање класе перунике
- Алгоритам C5.0 рулес у SPSS Modeler V18.3

Директна метода

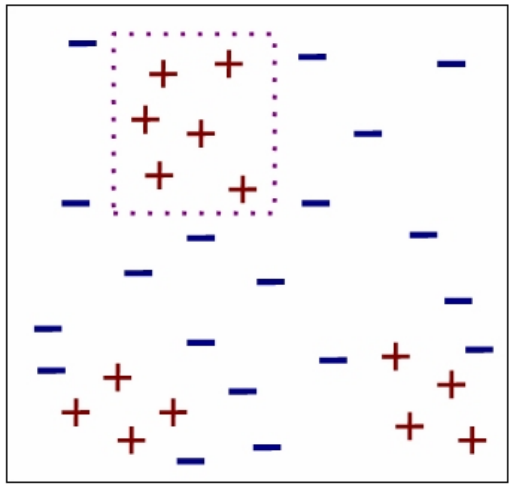
Секвенцијално покривање

1. Почиње се од празног скупа правила
2. Издвајају се правила за наредну класу
   - Скуп правила се проширује коришћењем функције *Научи-једно-правило* (енг. *Learn- One-Rule*)
3. Уклањају се слогови за тренинг који су покривени додатим правилом (позитивни слогови)
   - Остају негативни примери – сви остали слогови
4. Понављају се кораци од (2) до достизања критеријума заустављања

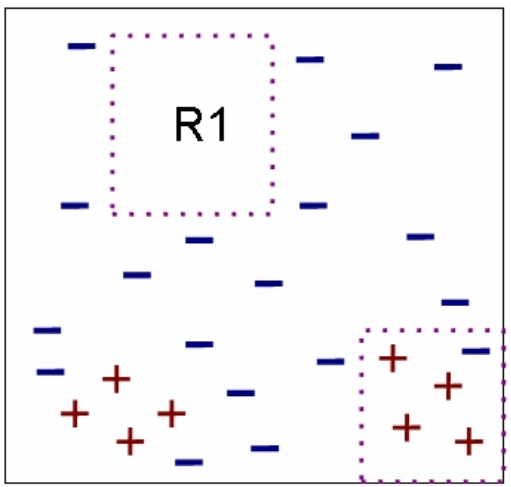
Пример секвенцијалног покривања



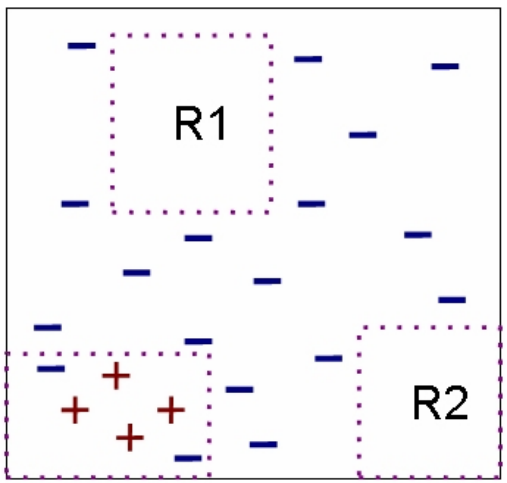
Оригинални подаци



Корак 1



Корак 2



Корак 3

Директна метода - алгоритми

1R (*One rule*)

- за сваки атрибут у скупу података формира једно правило, и затим бира правило са највећом прецизношћу
- ради са атрибутима који имају дискретне вредности, док непознате вредности третира као издвојене (појединачне) вредности у скупу вредности атрибута

CN2

- формира уређен скуп правила (према квалитету)
- користи секвенцијално покривање и одређује наредно правило без фиксирања класе унапред
- последица - правила за различите класе су помешана
- НВ замењује са најчешћим (категорички атр.) односно средњим (нумерички атр.) вредностима

Директна метода - алгоритми

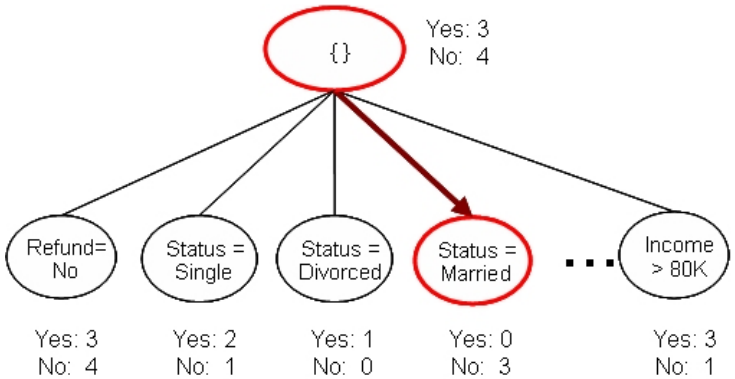
RIPPER

- користи секвенцијално покривање
- формира уређен скуп правила
- унапред се фиксира класа и одреде сва правила за ту класу
- на наредну класу се прелази тек када се комплетира претходна класа
- код бинарне класификације за предефинисану класу се узима бројнија класу и одређују правила за мање бројну класу
- код вишекласне класификације, прво се одређују правила за најмање бројну класу, итд.
- редослед правила унутар групе која одређују једну класу није битан, док је редослед правила у уређеној листи диктиран величином класе, односно редоследом одређивања правила

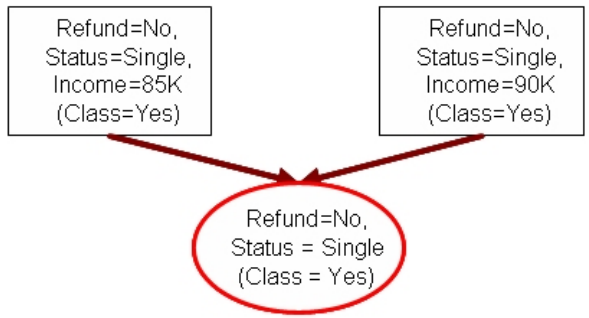
Особине секвенцијалног покривања

- Пораст скупа правила
- Уклањање обрађених инстанци
- Провера правила
- Критеријум заустављања
- Поткресивање (скупа) правила

Пораст скупа правила



Од општег ка појединачном



Од појединачног ка општем

- формира се иницијално правило $r : \{\} \rightarrow y$. Покрива све инстанце - низак квалитет
- додавањем нових конјуката повећава се квалитет; додавање се врши док се не достигне критеријум заустављања, односно све док се повећава квалитет

- једна од позитивних инстанци се бира методом случајног избора као иницијално правило које се оптимизује уклањањем неког од конјуката тако да новодобијено правило покрива што више позитивних инстанци

Пораст скупа правила

CN2

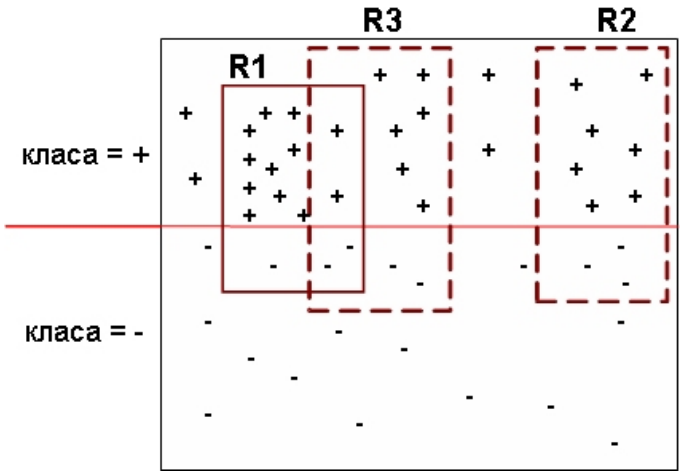
- Почни од празног конјункта: $\{\}$
- Додај конјункте који минимизују ентропију: $\{A\}, \{A, B\}, \ldots$
- Одреди редослед правила узимајући у најбројније класе инстанци које покривају правило

RIPPER

- Почни од празног конјункта: $\{\} \rightarrow$ класа
- Додај конјункте који максимизују FOIL-ову меру добити квалитета информације
  - $R_0 : \{\} \rightarrow$ класа (почетно правило)
  - $R_1 : \{\} \rightarrow$ класа (правило по додавању конјункта)
  - Добит $(R_0, R_1) = t[log_2(p_1/(p_1 + n_1)) - log_2(p_0/(p_0 + n_0))]$ где је
    - t број позитивних инстанци покривених са оба правила R_0 и R_1
    - p_0 број позитивних инстанци покривен са R_0
    - n_0 број негативних инстанци покривен са R_0 (нису покривене са R_0)
    - p_1 број позитивних инстанци покривен са R_1
    - n_1 број негативних инстанци покривен са R_1 (нису покривене са R_1)

Уклањање инстанци

- Без елиминације инстанци наредно правило би било идентично претходном правилу
- Елиминацијом позитивних инстанци се обезбеђује да је следеће правило различито од постојећих
- Елиминацијом негативних инстанци онемогућено је смањење прецизности правила
- Поредити правила R2 и R3 на дијаграмима



Мере за проверу квалитета правила

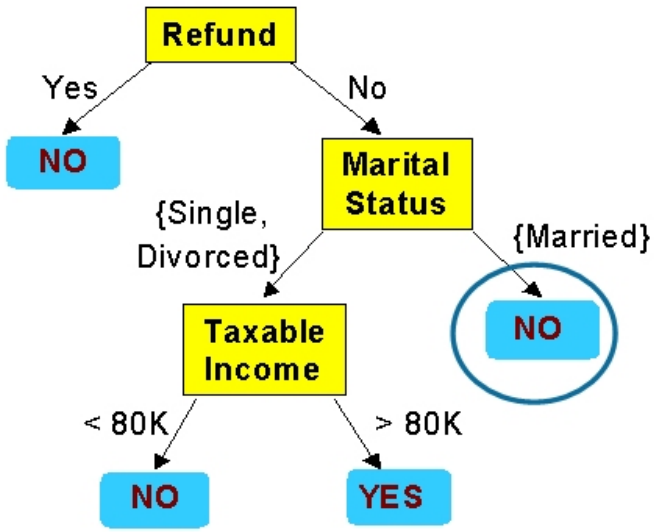
- Прецизност $= \frac{n_+}{n}$
- Laplas $= \frac{n_+ + 1}{n + k}$
- M-procena $= \frac{n_+ + kp_+}{n + k}$

где је

- n број инстанци покривен правилом
- n_+ број позитивних инстанци покривен правилом
- k укупан број класа
- p_+ претходна вероватноћа за позитивну класу

ДЗ: испитати да ли су претходно наведене мере и метрике

Упрошћавање правила



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

Почетно стање: $(\text{Refund=No}) \wedge (\text{Status=Married}) \rightarrow \text{No}$

Упрошћено правило: $(\text{Status=Married}) \rightarrow \text{No}$

Критеријум заустављања и поткресивање правила

Критеријум заустављања

- Израчунавање добити
- Ако добит није значајна, одбаци ново правило

Поткресивање правила

- Слично покресивању дрвета одлучивања
- Смањивање грешке потресивањем
  - Уклонити један од конјуката у правилу
  - Поредити стари и нови ниво грешке
  - Ако се грешка смањује, искључити конјункт

Резиме директних метода

- Раст (проширење) појединачног правила
- Уклањање инстанци из правила
- Покресивање правила (по потреби)
- Додати правило у скуп правила
- Поновити поступак

Предности класификатора заснованих на правилима

- Иста изражајна моћ као и дрвета одлучивања
- Једноставна интерпретација
- Једноставно формирање
- Могу брзо да класификују нове инстанце
- Перформансе су упоредиве са дрветима одлучивања

Класификатори засновани на инстанцама

- Не постоји модел у класичном смислу
- Модел чине сви тренинг слогови који се чувају и поново користе у обради сваки пут када се класификује непозната инстанца
- Мања брзина класификације ако је скуп за тренинг јако велики
- *Лењи* класификатори

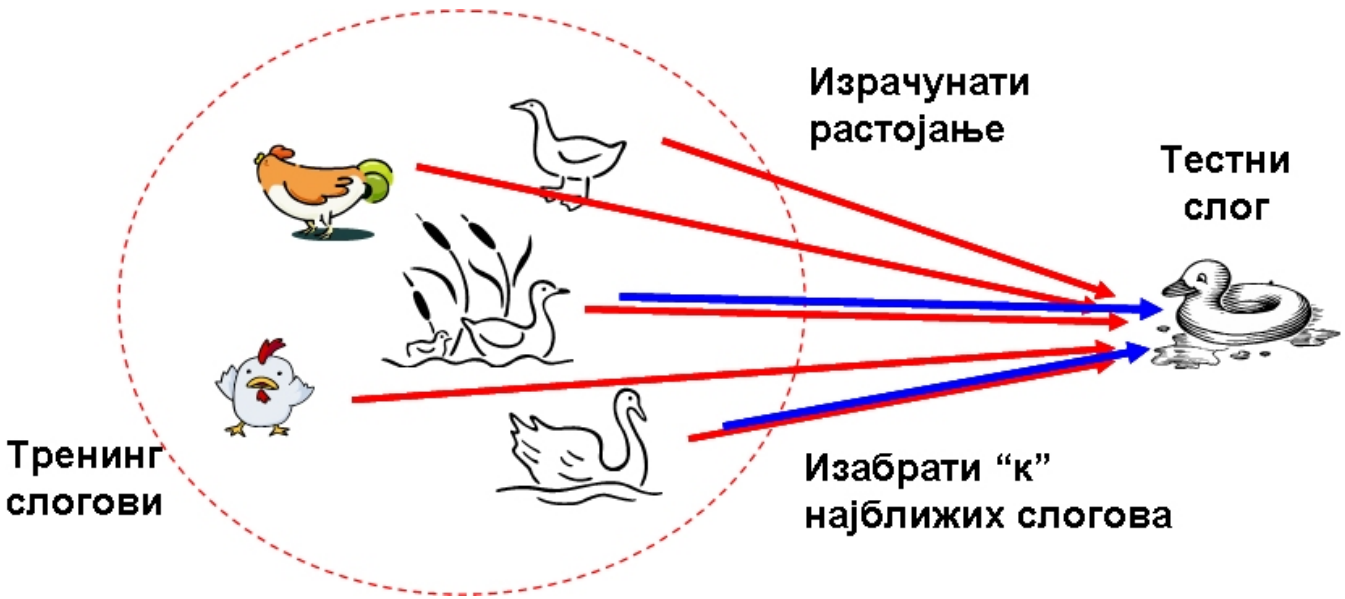
Класификатори засновани на инстанцама

Примери

- Учење напамет
  - чува целокупан скуп слогова за тренинг и спроводи класификацију само ако се атрибути нових слогова потпуно поклопе са атрибутима тренинг слогова
- Најближи сусед (комшија)
  - користи k "најближих" тачака (најближих комшија) за обављање класификације

Основна идеја

- С ким си, такав си
- Ако шета као патка, кваче као патка, личи на патку, онда је вероватно у питању патка



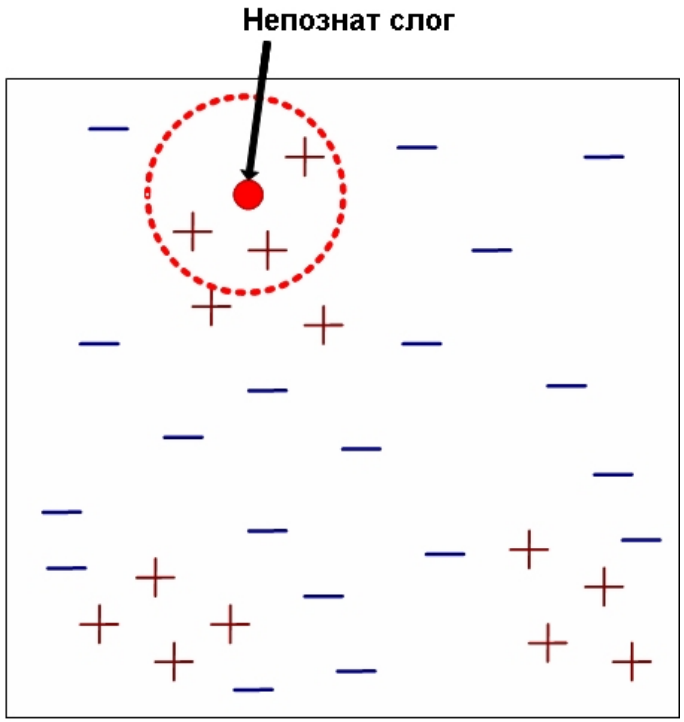
Основна идеја (наставак)

Потребно

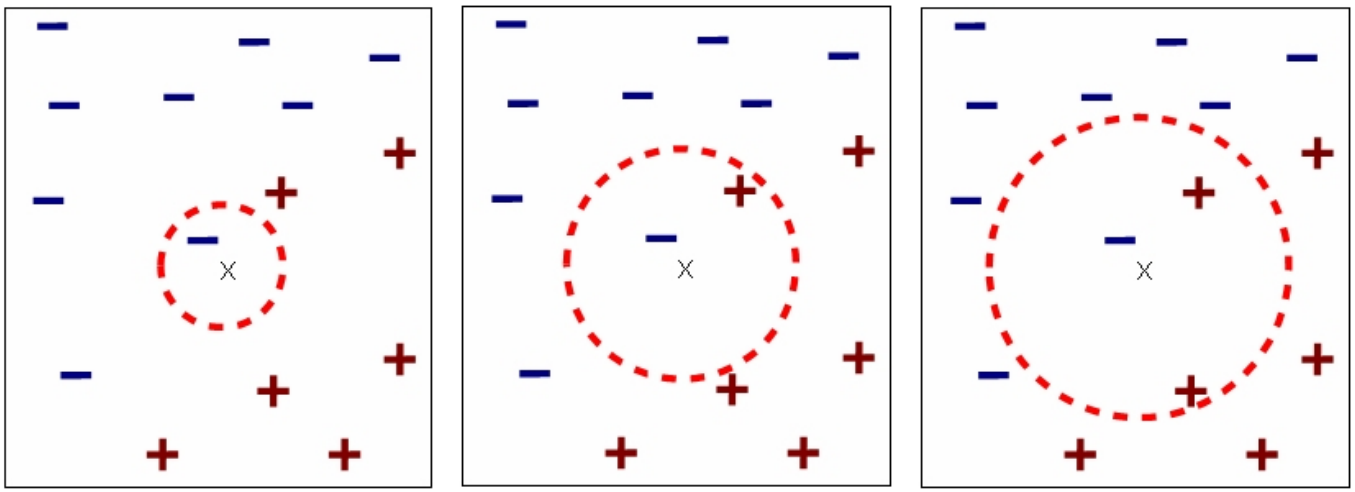
- Скуп сачуваних слогова
- Метрика за израчунавање растојања између слогова
- Вредност к која представља број најближих суседа које треба разматрати

За класификацију непознатих слогова

- Израчунати растојање до осталих слогова за тренинг Користећи ознаке класа најближих суседа одредити ознаку класе непознатог слога (нпр. узети ознаку већине)
- Одредити к најближих суседа
- Користећи ознаке класа најближих суседа одредити ознаку класе непознатог слога (нпр. узети ознаку већине)

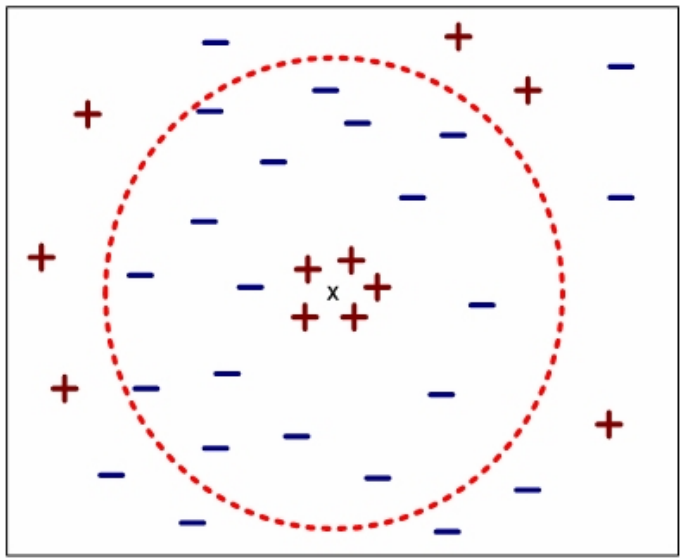


Дефиниција најближег суседа



1-најближи сусед 2-најближа суседа 3-најближа суседа

Класификација помоћу најближег суседа

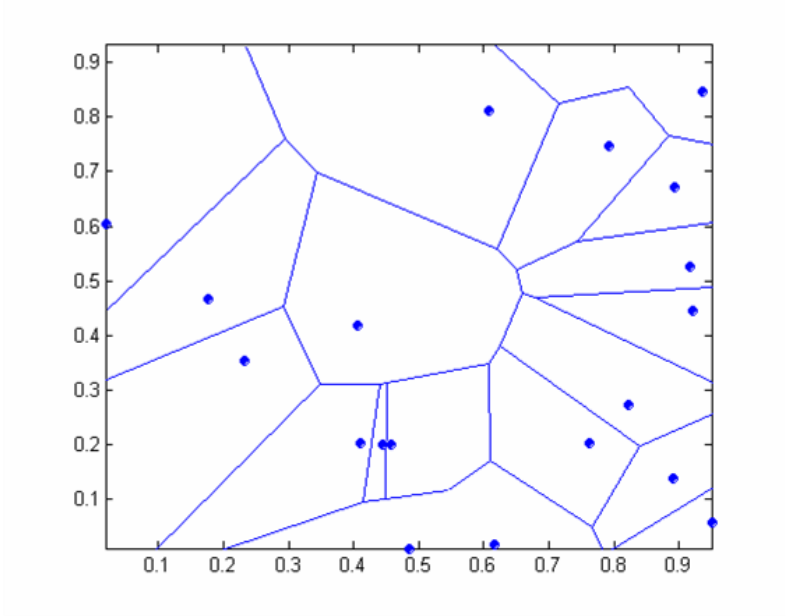


Избор вредности к:

- Ако је к јако мало класификација је осетљива на шум
- Ако је к јако велико суседи могу да укључе тачке из других класа

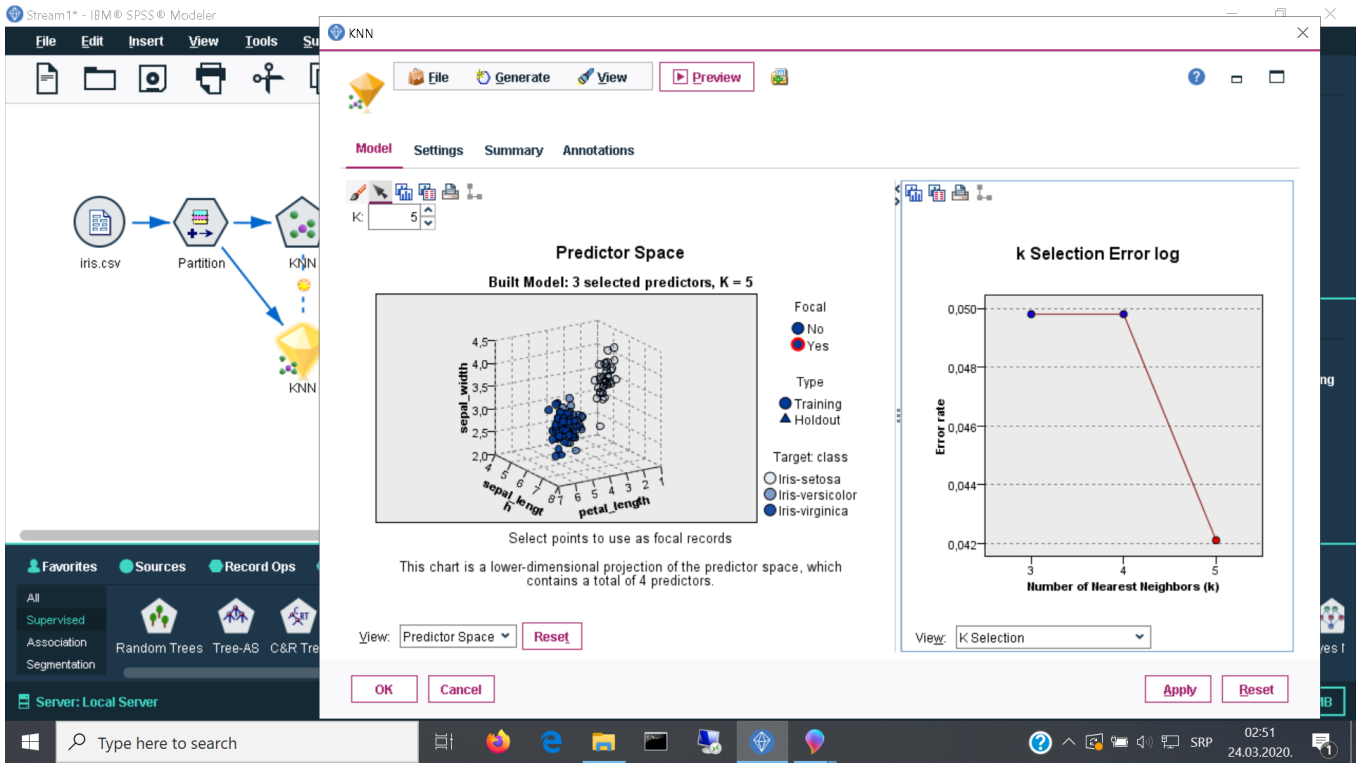
1 најближи сусед

Ако се узме $k = 1$ тада се подела простора за класификацију представња преко дијаграма Voronoi-a

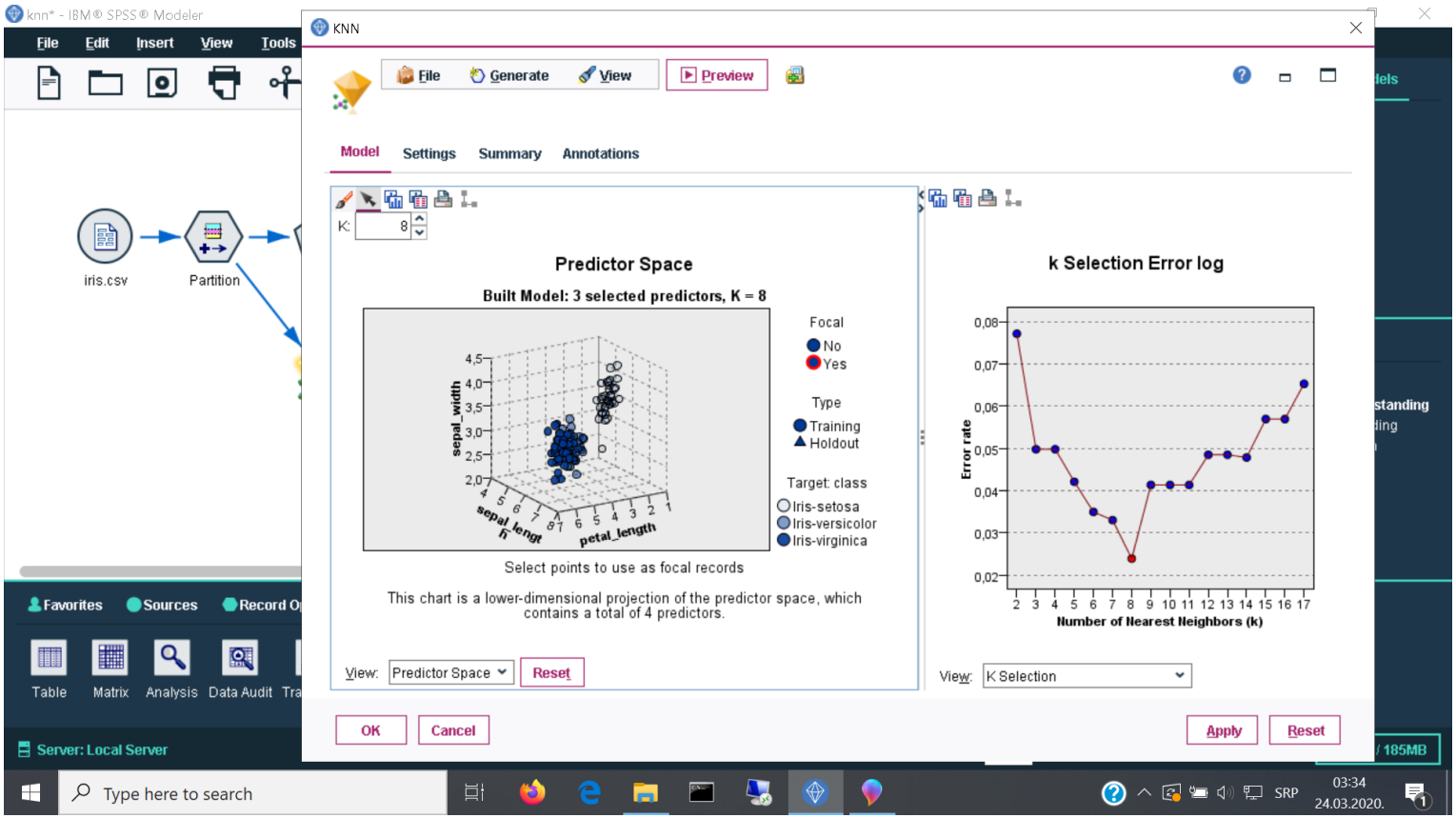


Пример

Пример на сајту: ИРИС скуп; предвиђање класе перунике за различите вредности k



Утицај изабране вредности k на величину грешке



Класификација помоћу најближег суседа

- Могу да се користе различите метрике
  - Еуклидско растојање није увек погодно. Нпр. за парове асиметричних атрибута са вредностима

    1 1 1 1 1 1 1 1 1 1 1 0 и 1 0 0 0 0 0 0 0 0 0 0 0

    0 1 1 1 1 1 1 1 1 1 1 1 и 0 0 0 0 0 0 0 0 0 0 0 1

    добија се исто растојање
- Одређивање ознаке класе
  - ознака класе већине
  - растојања могу да добију одређене тежине, нпр. $w = 1/d^2$
- Атрибути могу бити скалирани ради спречавања да у мери растојања доминира један атрибут