Класификација - Поређење резултата

Ненад Митић

Математички факултет
nenad@matf.bg.ac.rs

Увод

Сваки пут када треба решити задатак који укључује прављење модела за класификацију, поред корака који не смеју да се прескоче и везани су за припрему података, намећу се следећа питања:

- Који алгоритам за класификацију одабрати за прављење модела, тј. који алгоритам је 'прави' за проблем који се решава?
- Који параметри алгоритма су посебно важни за проблем који се решава, односно на које параметре алгоритма треба посебно обратити пажњу?
- У ком односу поделити податке за тренинг и тест, и да ли ће бити неких предности ако се подаци поделе на три дела: за тренинг, тест и проверу?
- Како проценити резултате класификације које је произвео конструисани модел и како упоредити резултате два или више модела и одабрати најбољи?
- ...

Избор алгоритма

Не постоји алгоритам за који може да се каже да је 'најбољи' у свим случајевима. Избор алгоритма зависи од конкретног проблема, врсте података, од жељеног начина приказа резултата, итд.

Како нису сви алгоритми применљиви на све типове података, избор алгоритма је доста често диктиран типом појединих атрибута података (на пример, да ли су подаци само категорички, само непрекидни или мешани), да ли постоје непознате вредности, да ли је могуће податке претходно трансформисати у жељени облик (дискретизацијом, бинаризацијом, ...), итд.

Избор параметара

За избор параметара алгоритма неопходно је да се детаљно проучи сам алгоритам и значење појединих опција. Препорука је да се пре рада погледају упутства за кориснике и упутство произвођача софтвера који се користи. У избору параметара додатно треба обратити пажњу на избор мере на основу које се рачуна растојање и мере за процену квалитета, ако такве опције постоје. Коректан избор ових параметара у великој мери утиче и на коректност добијених резултата.

Подела података

При формирању класификационог модела могуће су две врсте поделе података:

- на скупове за *тренинг* (енг. *training*) и *тест* (енг. *test*), или
- на скупове за *тренинг*, *тест* и *проверу* (енг. *validation*, *evaluation*)

Модел се формира на основу скупа података за трениниг. Скуп података за тест се користи за проверу формираног модела и подешавање интерних параметара (параметара које не поставља корисник) ради добијања што прецизнијег модела. Скуп података за проверу се користи за добијање информације како се модел понаша када се примени на потпуно непознате податке који нису учествовали нити били познати при прављењу модела.

Подела података

Неки од алгоритама могу да користе унакрсну проверу (енг. *cross validation*). У том случају није неопходно да постоји скуп података за тестирање, али није ни забрањено.

Подела скупа података: у случају да се скуп података дели на два дела, доста чест случај је да се подела врши у односу 70:30 или $2/3 : 1/3$, мада је могућа и подела у односу 50:50, поготову ако је скуп података довољно велики. што се тиче скупа за проверу, најчешће се узима да је он једнак 10% од броја слогова у материјалу.

Напомена 1: Ако је скуп података релативно мали, тада се доводи у питање коректност резултата која се добија његовом поделом на тренинг, тест и скуп за проверу

Напомена 2: У литератури се често значење термина *тренинг скуп* и *скуп за проверу* алтернира.

Процена квалитета модела

Постоји више различитих начина да се процени квалитет модела класификације. Да би се добило коректно поређење услови формирања модела који се пореде треба да буду коректни и прилагођени предусловима за примену тих модела.

На пример, некоректно је поредити различите алгоритме уколико подаци садрже непознате вредности при чему један од алгоритама подржава рад са непознатим вредностима а други не. Такође, скупови података на којима се формирају и тестирају модели треба да буду идентични, итд. Детаљнији опис метода за поређење класификатора дат је у књизи *An Introduction to Data Mining, 2nd ed.* - Tan, Steinbach, Karpatne, Kumar

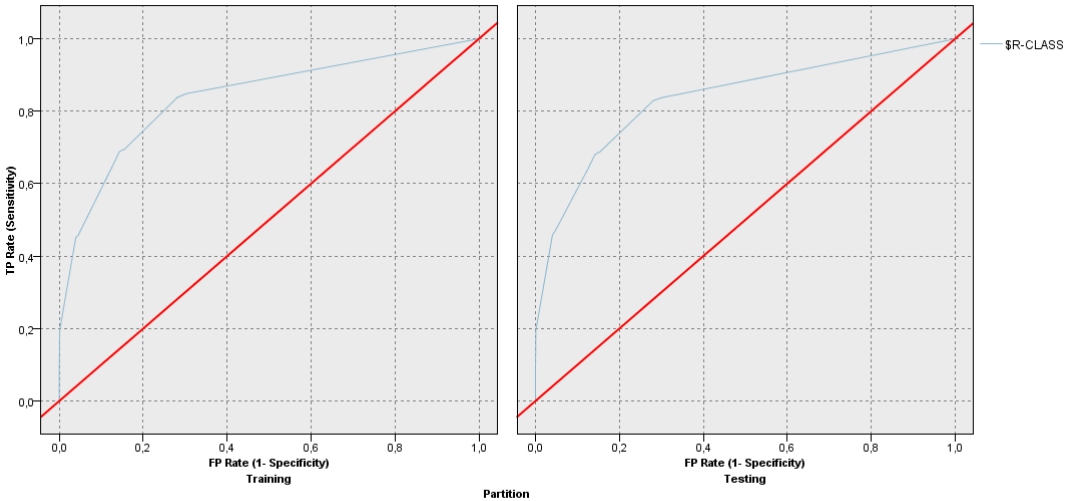
РОЦ крива

Поред мера о којима је до сада било речи, за проверу квалитета модела преко разматрања односа броја позитивних инстанци које су тачно (ТПР) и нетачно (ФПР) класификоване користи се РОЦ крива. РОЦ (енг. *Receiver Operating Characteristic*) представља визуелизацију овог односа при чему се у координатном систему на x оси приказује ФПР а на y оси ТПР.

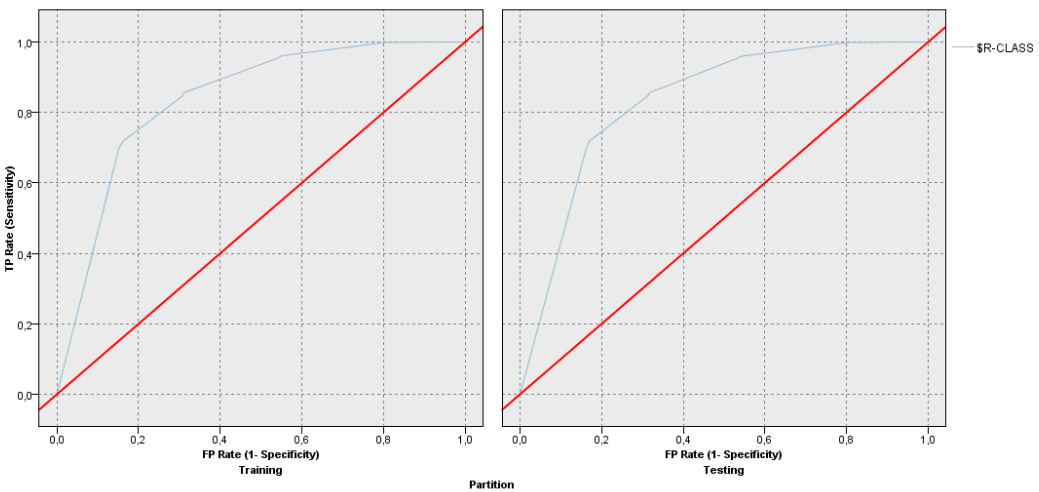
На РОЦ криви постоје карактеристичне тачке

- ТПР=0, ФПР=0 - тачка на правој $y = x$ са координатама (0,0) - модел предвиђа да све инстанце припадају негативној класи
- ТПР=1, ФПР=1 - тачка на правој $y = x$ са координатама (1,1) - модел предвиђа да све инстанце припадају позитивној класи
- ТПР=1, ФПР=0 - тачка на правој $x = 0$ са координатама (0,1) - модел предвиђа позитивну класу са тачношћу 100% - све позитивне инстанце су коректно предвиђене

РОЦ крива



CLASS = ">50K"



CLASS = "<=50K"

РОЦ крива

На основу изгледа РОЦ криве може да се процени квалитет модела. што је крива ближе горњем левом углу дијаграма то је модел прецизнији.

Крива модела који предивђа резултат на основу случајног избора се налази на главној дијагонали (права $y = x$). Нека скуп података садржи n инстанци, од којих је n_+ позитивних и n_- негативних, и нека модел (случајним избором) класификује инстанце као позитивне са вероватноћом p. Тада ће бити коректно класификовано pn_+ инстанци и некоректно класификовано pn_- негативних инстанци. Одатле је ТПР=$(pn_+)/n_+ = p$, а такође ФПР=$(pn_-)/n_- = p$, одакле следи да РОЦ крива у случају класификатора који предвиђа класу методом случајног избора лежи на правој $y = x$

АУЦ

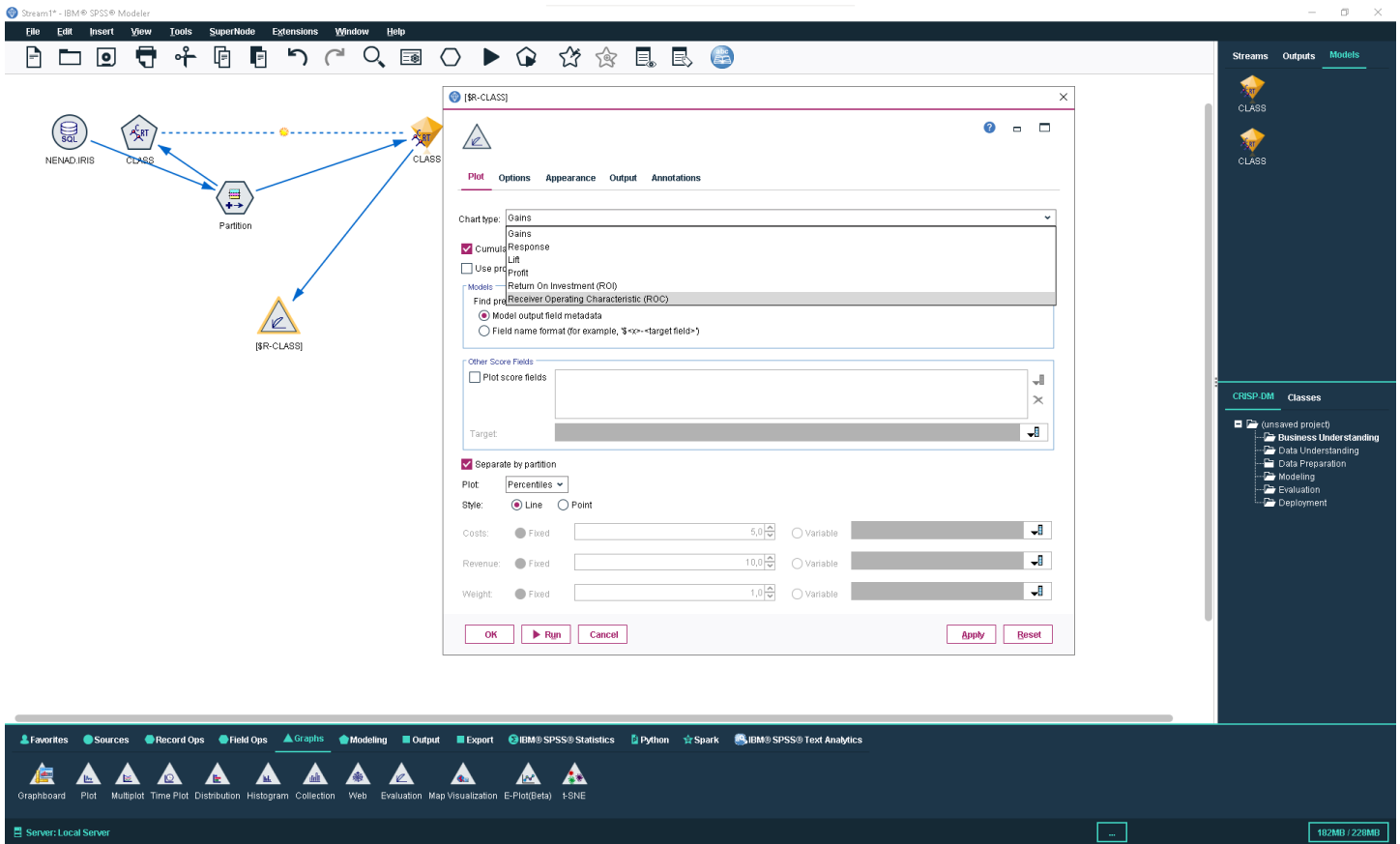
Као додатни критеријум за процену квалитета модела може да послужи и величина коју заузима простор испод РОЦ криве - AUC (енг. *Area Under Curve*). што је површина већа (ближа 1) то модел боље предвиђа одговарајућу класу. Ако је површина испод РОЦ криве једнака 0.5 то значи да модел предвиђа класе методом случајног избора.

Напомена 1: РОЦ крива се може приказати само за бинарне класификаторе, док друге врсте кривих омогућују приказ и за вишекласне случајеве

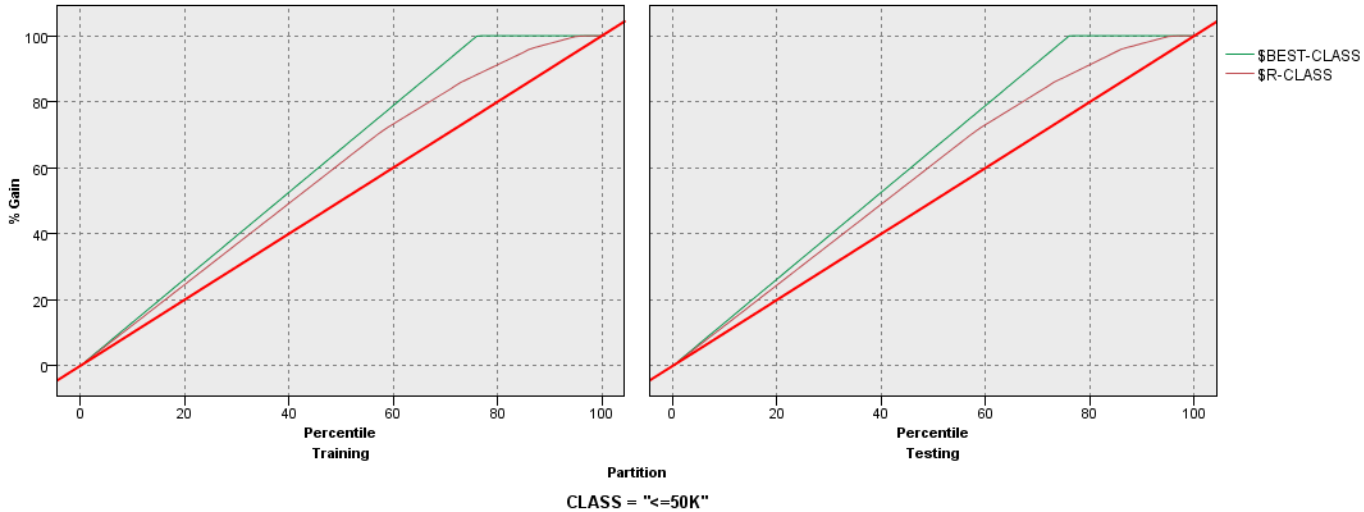
Напомена 2: Да би се добила коректна слика прецизности класификатора потребно је приказати РОЦ криве за сваку класу - модел може да буде такав да се једна класа одлично предвиђа, док за другу резултати нису задовољавајући

Криве - могући критеријуми

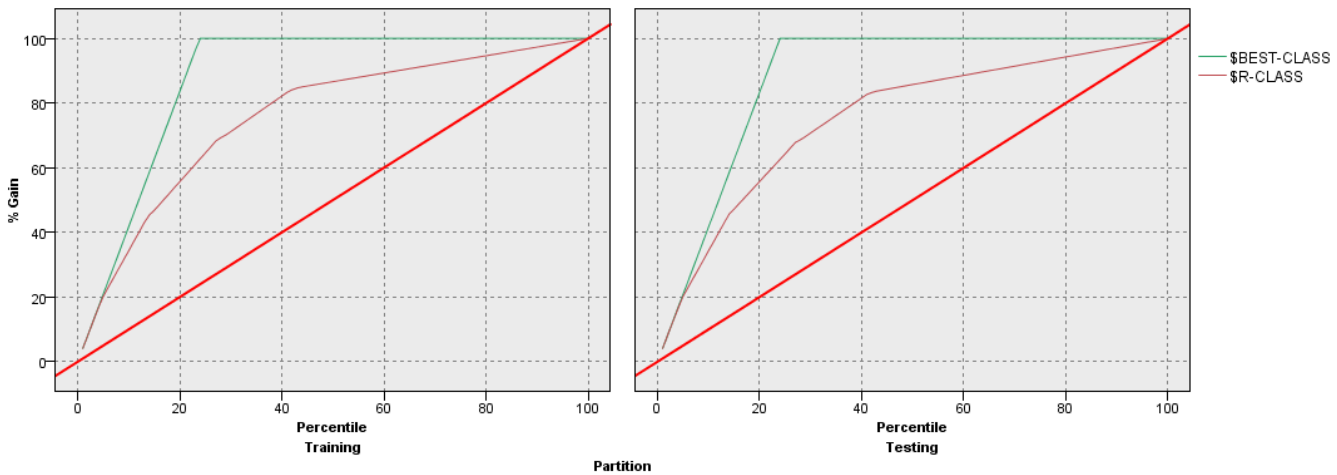
У различитим алатима поред РОЦ односно АУЦ постоје и други критеријуми (са графичком интерпретацијом) који могу да се користе за процену квалитета. На следећим сликама су приказани могући избор типа криве у SPSS modeler V18.3 и добијени графикони за изабрану опцију добит (*gains*)



Крива информационе добити



CLASS = "<=50K"



CLASS = ">50K"

Примери поређења модела

Као илустрација примене различитих алгоритама на исти скуп података и поређења добијених резултата биће дати резултати примене алгоритама који су до сада обрађени на предавањима. Алгоритми ће бити примењени на податке из ADULT скупа, са предвиђањем (бинарног) атрибута *CLASS*. Коришћени пакети и алгоритми су

- У пакету SPSS Modeler: C&RT, C5.0, CHAID, QUEST, Naivni Bajes, KNN, C5.0 Rules
- У пакету Weka: JRIP (RIPPER), OneR (1R), J48, JCHAIDStar, NaiveBayes, IBk (KNN)
- У пакету IBM Intelligent Miner: SPRINT, Naivni Bajes

У скуп примера нису укључени алгоритми из Python алата пошто се они детаљније обрађују на вежбама

Примери поређења модела

- Скуп података ADULT (48842 инстанци) је подељен на део за тренинг (34070 инстанци) и део за тест(14772 инстанце).
- Из материјала је искључен атрибут ID (идентификација особе) пошто је он неважан за класификацију
- При класификацији нису коришћене додатне могућности (нпр. задавање цене класификације по класама)
- Резултати алгоритама су смештени у посебним директоријумима и обухватају прецизност података на тренинг и тест скупу, и где је могуће матрицу конфузије, ROC/GAINS или одговарајућу криву, добијени модел, и важност атрибута у класификацији.
- За моделе рађене у СПСС Моделер пакету је дат модел и придружени поток података одакле могу да се извуку сви наведени резултати (ДОМАЋИ ЗАДАТАК!)

Примери поређења модела

- Сви директоријуми са резултатима се налазе у архиви која је расположива на сајту предмета директно уз овај текст
- Ако се посматра прецизност класификације на тренинг и тест скупу упада у очи велика разлика у случају KHH алгоритма имплементираног у Weки за K=1. Код њега је прецизност на тренинг скупу 99.99%, а на тест скупу 79.48% што говори о очигледној преприлагођености модела.
- Значајнија разлика (око 5.5%) постоји и код KNN алгоритма у Weки за K=5, и код KNN у SPSS Modeleru (нешто мало преко 5%) што говори да алгоритам KHH не даје добре резултате над овим подацима
- Варијација осталих модела је у границама када не може експлицитно да се тврди да је било који од њих преприлагођен.
- Од осталих алгоритама набољи проценат има C5.0, док су најстабилнији (са најмањом разликом између резултата на тренинг и тест скупу) C&RT, QUEST, 1R, ...

Прецизност поређених модела

Paket	Algoritam	Trening	Test
SPSS Modeler	C&RT	83.75	83.87
	C5.0	88.1	86.63
	CHAID	83.23	82.41
	QUEST	80.24	80.31
	Naivni Bajes	77.06	76.46
	KNN	80.99	75.77
	C5.0 Rules	87.83	86.46
Weka	JRIP	85.21	84.49
	1R	81.19	80.82
	Naivni Bajes	83.56	82.84
	IBK (za K=1)	99.99	79.48
	IBK (za K=5)	87.71	82.31
	J48	87.90	85.91
	JCAHIDStar	85.94	84.53
IBM Intelligent miner	SPRINT	85.0	82.6
	Naivni Bajes	79.2	78.7

Други критеријум поређења

У претходим алгоритмима подразумевано је да су подаци обрађени у процесу препроцесирања, али у овом конкретном случају нису обављене све детаљне анализе у фази препроцесирања. Тако нпр. није урађено испитивање корелације између атрибута, а такође је претпостављено да сви атрибути садрже само познате вредности. Ако се ове промене узму у обзир, добијени резултати се могу анализирати посматрајући их са других аспеката.

- због небалансираности класа (76% материјала чине инстанце са класом <=50K и 26% материјала инстанце са класом >50K), уколико је потребно наћи алгоритам који што је могуће тачније препознаје инстанце у мање бројној класи, редослед модела ће бити промењен. Одредити који алгоритам најбоље класификује инстанце у класу >50K.
- Проверити да ли су неки атрибути у корелацији и како они утичу на произведене резултате (проверу вршити преко редоследа важности атрибута за конкретан класификациони модел).
- Проверити који атрибути садрже непознате вредности и као они утичу на резултате појединих алгоритама, односно да ли су ти атрибути укључени у прављењу модела помоћу алгоритама који не подржавају рад са недостајућим вредностима
- Испитати да ли неки од алгоритама подржавају само категоричке податке и да ли то мења претходни редослед квалитета модела
- ...