STABLA ODLUČIVANJA

Jelena Jovanovic

Email: jeljov@gmail.com

Web: http://jelenajovanovic.net

Zahvalnica:

Ovi slajdovi su bazirani na materijalima pripremljenim za kurs “Applied Modern Statistical Learning Techniques” (link), kao i na poglavlju 8 knjige “Introduction to Statistical Learning” (link)

Primer: Klasifikacija igrača bejzbola

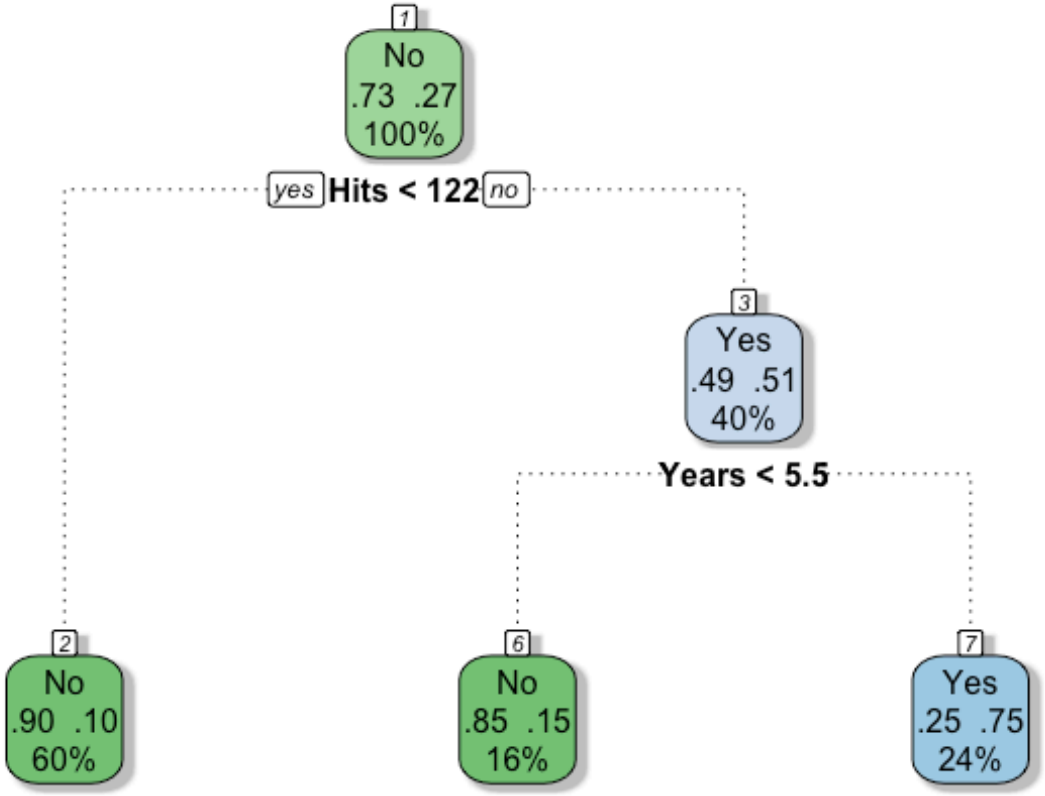
Potrebno je klasifikovati igrače bejzbola na one koji su jako dobro plaćeni i one koji to nisu (WellPaid), na osnovu

- broja ostvarenih poena u prethodnoj godini (Hits) i
- broja godina koje je igrač proveo u glavnoj ligi (Years)

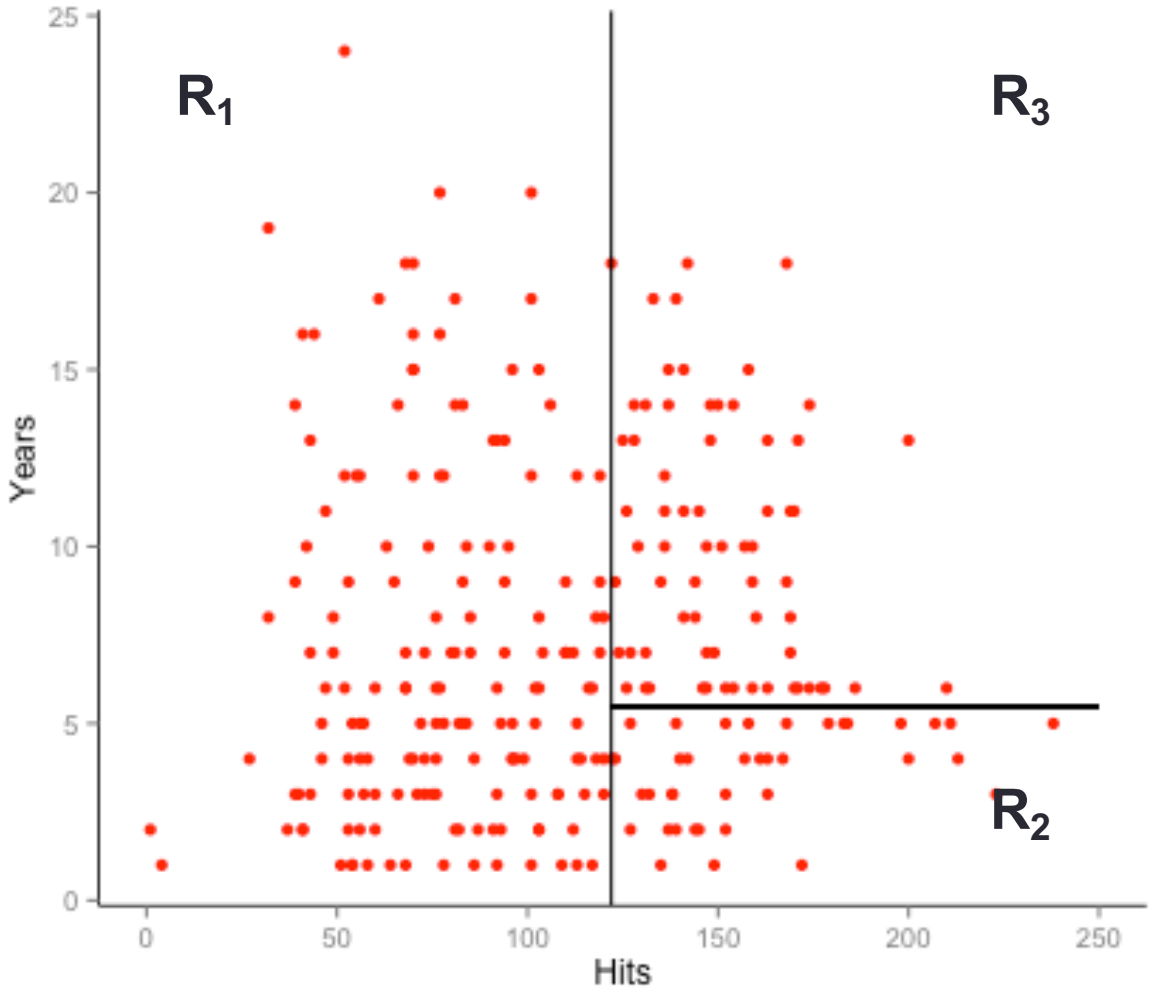
```
Console ~/R Studio Projects/Intelligent Systems Fall 2015/
> str(hitters.subset)
'data.frame':   263 obs. of  3 variables:
 $ Hits     : int  81 130 141 87 169 37 73 81 92 159 ...
 $ Years    : int  14 3 11 2 11 2 3 2 13 10 ...
 $ WellPaid: Factor w/ 2 levels "No","Yes": 1 1 1 1 2 1 1 1 2 1 ...
> head(hitters.subset)
                  Hits Years WellPaid
-Alan Ashby         81    14       No
-Alvin Davis       130     3       No
-Andre Dawson      141    11       No
-Andres Galarraga   87     2       No
-Alfredo Griffin   169    11      Yes
-Al Newman          37     2       No
>
```

Primer: Klasifikacija igrača bejzbola

- Stablo odlučivanja ukazuje da su dobro plaćeni oni igrači koji su ostvarili bar 122 pogotka u prethodnoj godini i koji bar 5.5 godina igraju u glavnoj ligi
- Verovatnoća da je igrač sa opisanim karakteristikama dobro plaćen je 0.75
- Ti igrači čine 24% svih igrača za koje su nam raspoloživi podaci (skup za trening)



Drugi način za vizuelizaciju stabla odlučivanja…



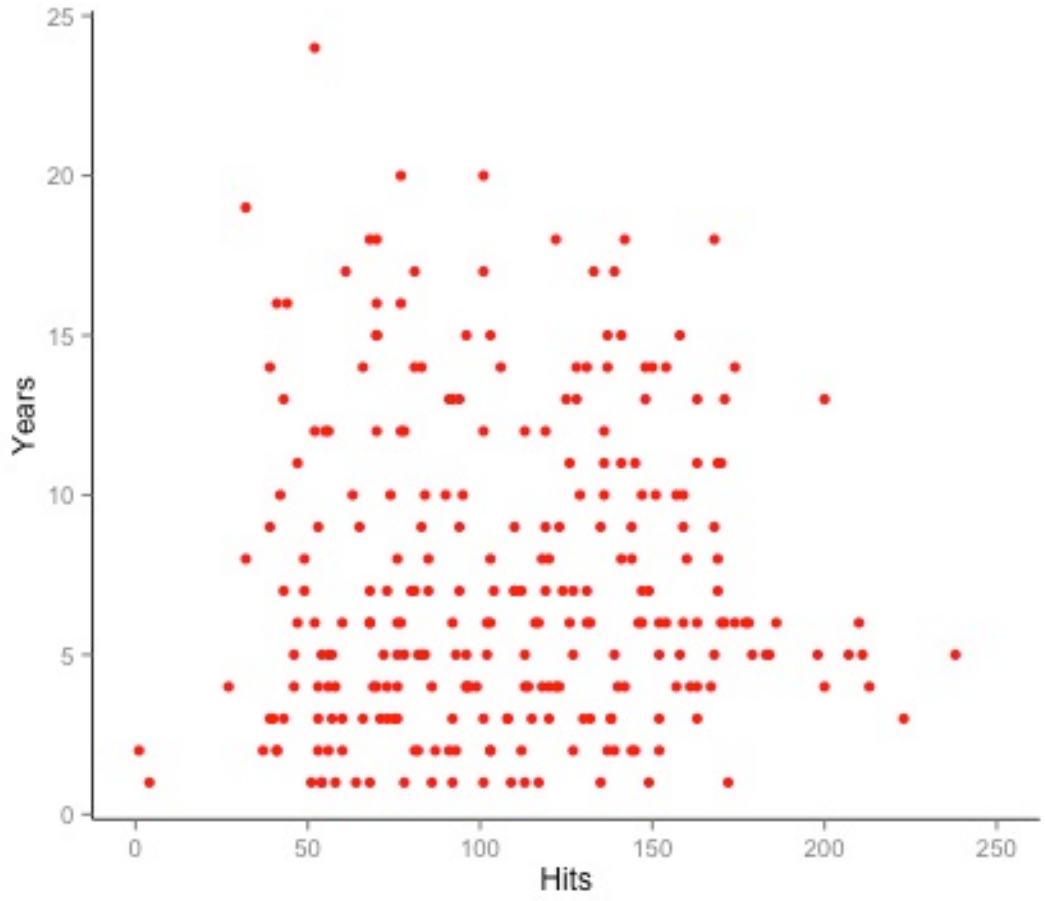
Osnovna ideja klasifikacionih stabala

- Podeliti prostora atributa kojima su objekti opisani u više različitih i međusobno neprekloplјenih regiona R_1, R_2, …, R_n
  - prostor atributa je *p*-dimenzionalni prostor koga čine moguće vrednosti *p* atributa $(x_1,x_2,\ldots,x_p)$ kojima su dati objekti opisani
- Za novi objekat X, određuje se pripadnost jednom od regiona $R_1 \ldots R_n$ na osnovu vrednosti atributa $(x_1,x_2,\ldots,x_p)$ kojima je X opisan
- Klasa novog objekta će biti ona klasa koja dominira (*majority class*) u regionu R_j u koji je X svrstan

Podela prostora atributa

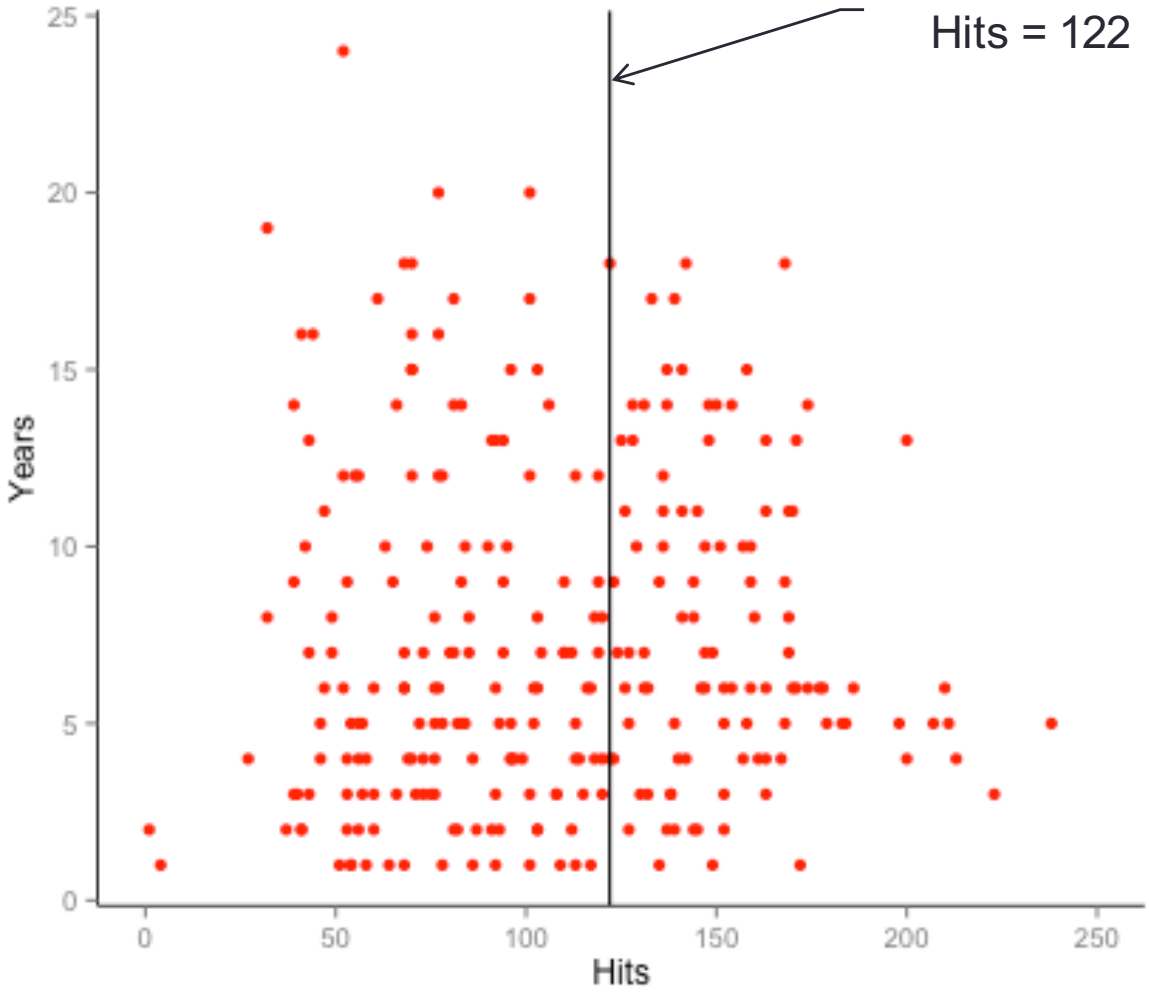
Podela prostora atributa na regione R_j je iterativni proces koji se sastoji od:

- izbora atributa x_i koji će biti osnova za podelu
- izbora vrednosti atributa x_i koja će poslužiti kao ‘granična’ vrednost



Podela prostora atributa

Za prvu podelu, u datom primeru, izabran je atribut Hits, i vrednost 122

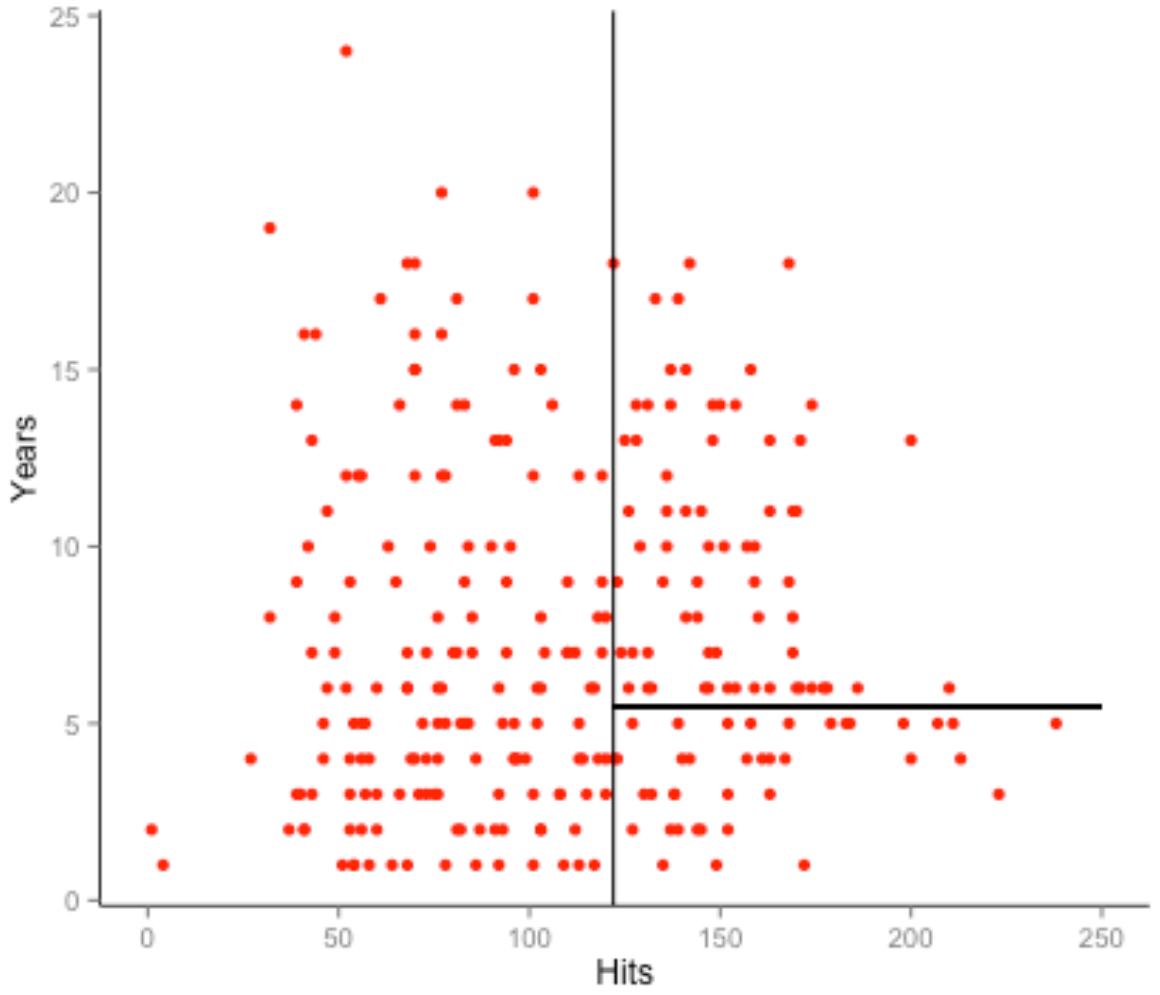


Podela prostora atributa

Prva podela: Hits = 122

Ukoliko je Hits > 122, sledeća podela je na atributu Years:

Years= 5.5



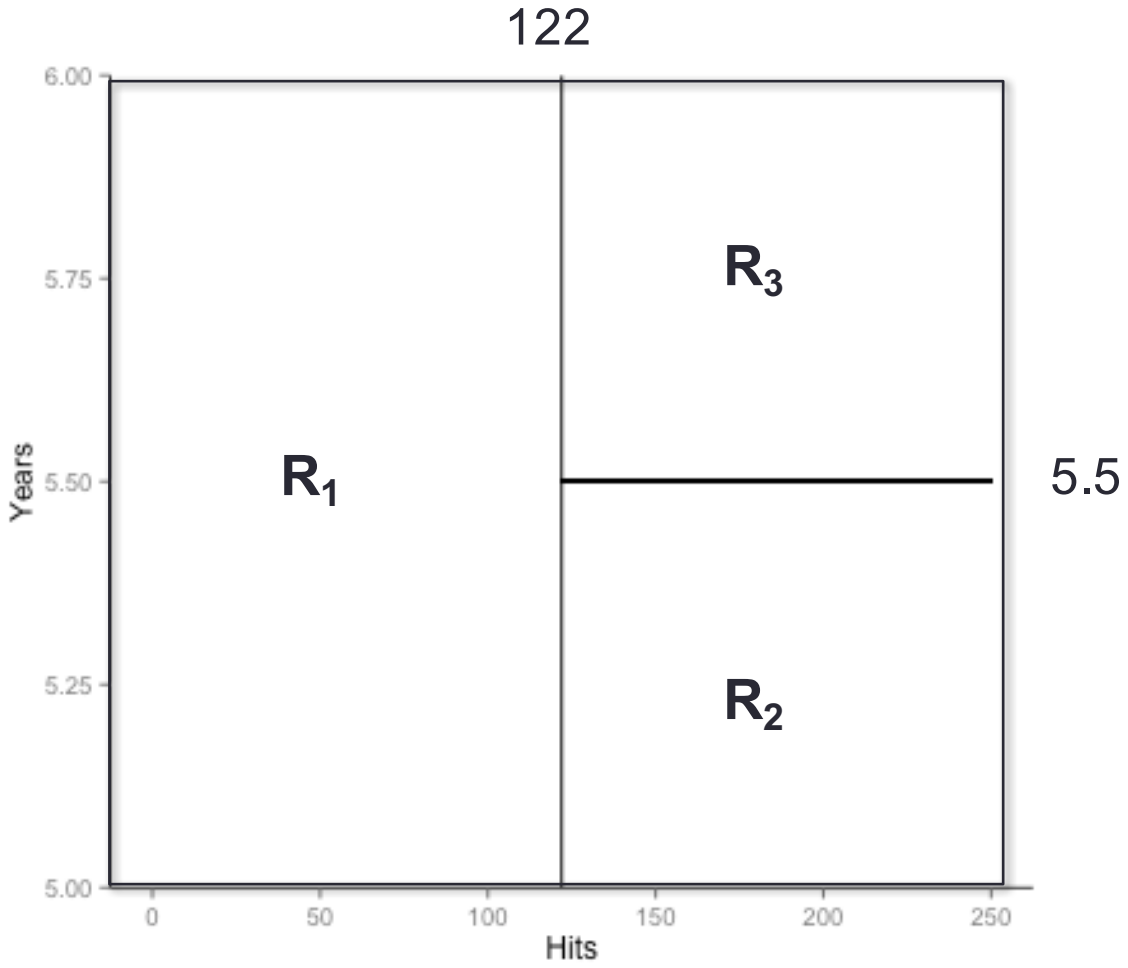
Podela prostora atributa

Prva podela:

Hits = 122

Ako je Hits > 122, sledeća podela:

Years = 5.5



Podela prostora atributa

Pitanja koja se prirodno nameću:

- Kako i gde izvršiti podelu?
  - drugim rečima, kako kreiramo regione R_1, R_2,…,R_n?

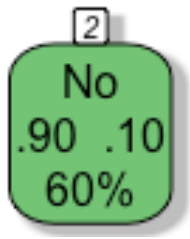
- Kako odrediti klasu instanci u svakom od regiona R_1,..,R_n?

Kako odrediti klasu instanci u regionima $R_1 \ldots R_k$?

Jednostavno, koristeći princip većinske klase (*majority class*):

svakom regionu R_j, pridružiti klasu kojoj pripada većina instanci iz skupa za trening koja je svrstana u region R_j

U datom primeru, u regionu R1, 90% instanci čine igrači koji nisu visoko plaćeni => svaki novi igrač koji bude svrstan u region R1 biće klasifikovan kao igrač koji nije vrhunski plaćen



Kako i gde izvršiti podelu?

Cilj je pronaći regione R_1, R_2,…,R_n tako da se minimizuje greška pri klasifikaciji - *Classification Error Rate* (CER)

CER predstavlja proporciju instanci (iz skupa za trening) u datom regionu koje ne pripadaju dominantnoj klasi tog regiona

$$CER = 1 - \max_k \hat{p}_{ik}$$

$\hat{p}_{ik}$ je proporcija (trening) instanci u regionu *i* koje pripadaju klasi *k*

Kako i gde izvršiti podelu?

- Pristup koji se primenjuje da bi se identifikovali regioni koji minimizuju grešku pri klasifikaciji zasniva se na ***rekurzivnoj, binarnoj podeli*** (*recursive binary splitting*) prostora atributa
- Osnovne karakteristike ovog pristupa:
  - *top-down* pristup
  - *greedy* pristup

Rekurzivna, binarna podela prostora atributa

- *Top-down* pristup
  - kreće od vrha stabla, gde sve (trening) instance pripadaju jednoj (zajedničkoj) regiji, a zatim sukcesivno deli prostor atributa na regione
- *Greedy* pristup
  - pri svakom koraku, najbolja podela se određuje na osnovu stanja u tom koraku, odnosno, ne uzima se u obzir šta će biti u narednim koracima, tj koja bi to podela mogla dovesti do boljih rezultata u nekom narednom koraku

Rekurzivna, binarna podela

Algoritam razmatra svaki atribut x_j (j=1,p) i svaku tačku podele s_j za taj atribut, i

bira onu kombinaciju koja će podeliti prostor atributa u dva regiona $\{X|x_j > s_j\}$ i $\{X|x_j < s_j\}$ tako da se minimizuje greška klasifikacije

Kako i gde izvršiti podelu?

Osim greške pri klasifikaciji (*Classification Error Rate*), kao kriterijumi za podelu prostora atributa, često se koriste i:

- Gini index
- Cross-entropy

Gini index

- Definiše se na sledeći način:

$$G = \sum_{k=1}^{K} \hat{p}_{ik} (1 - \hat{p}_{ik})$$

$\hat{p}_{ik}$ predstavlja proporciju trening instanci u regiji *i* koje pripadaju klasi *k*

- Često se opisuje kao mera ‘čistoće’ čvora (*node purity)*
  - ‘čisti’ čvorovi su oni u kojima visok procenat instanci pripada istoj klasi
  - mala vrednost za Gini indeks ukazuje na ‘čiste’ čvorove

Cross-entropy

- Definiše se na sledeći način:

$$D = -\sum_{k=1}^{K} \hat{p}_{ik} \, log \hat{p}_{ik}$$

- Kao i Gini indeks, cross-entropy predstavlja meru ‘čistoće’ čvora (što je vrednost manja, to je čvor ‘čistiji’)

Orezivanje stabla (*Tree pruning*)

- Velika klasifikaciona stabla, tj. stabla sa velikim brojem terminalnih čvorova (listova), imaju tendenciju over-fitting-a (tj. prevelikog uklapanja sa trening podacima)

- Ovaj problem se može rešiti ‘orezivanjem’ stabla, odnosno odsecanjem nekih terminalnih čvorova

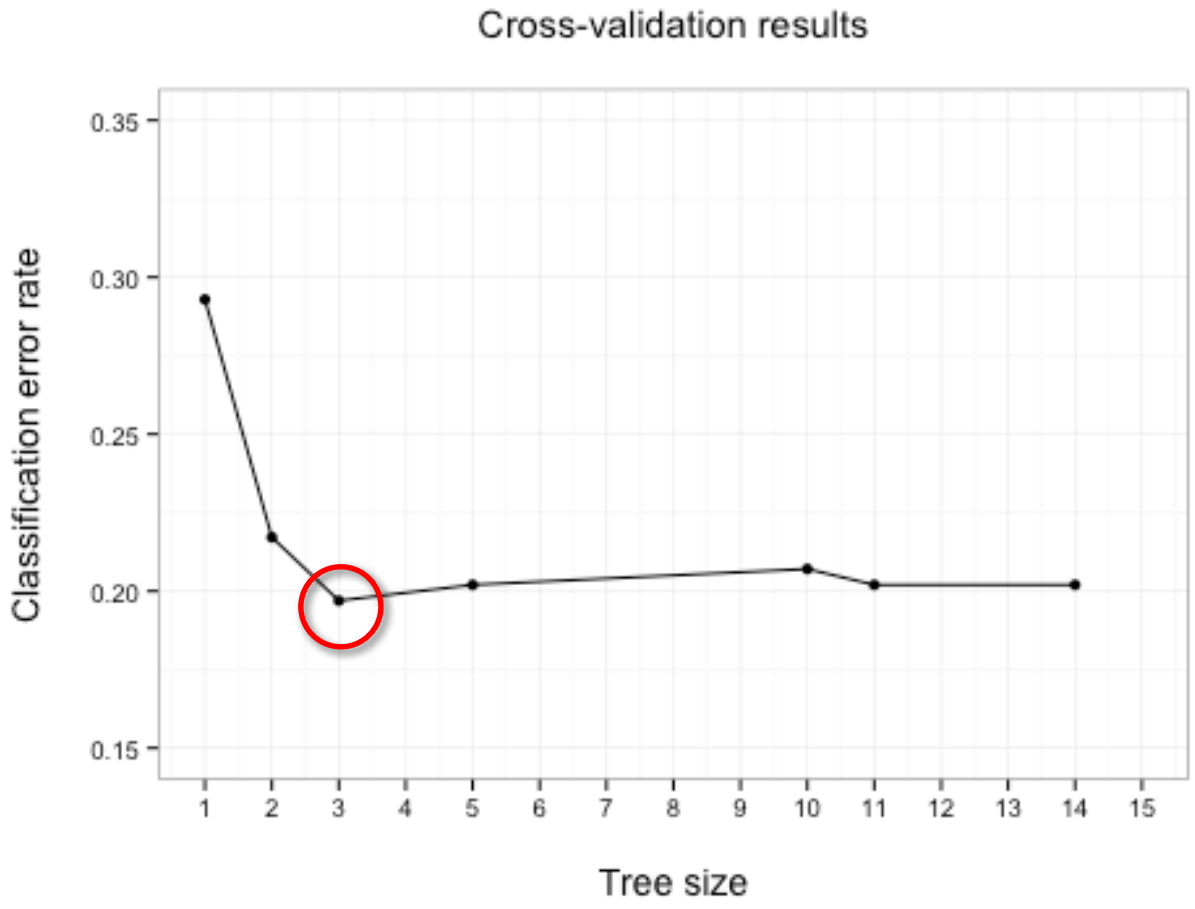
Orezivanje stabla (*Tree pruning*)

- Kako ćemo znati na koji način i u kojoj meri treba da ‘orežemo’ stablo?

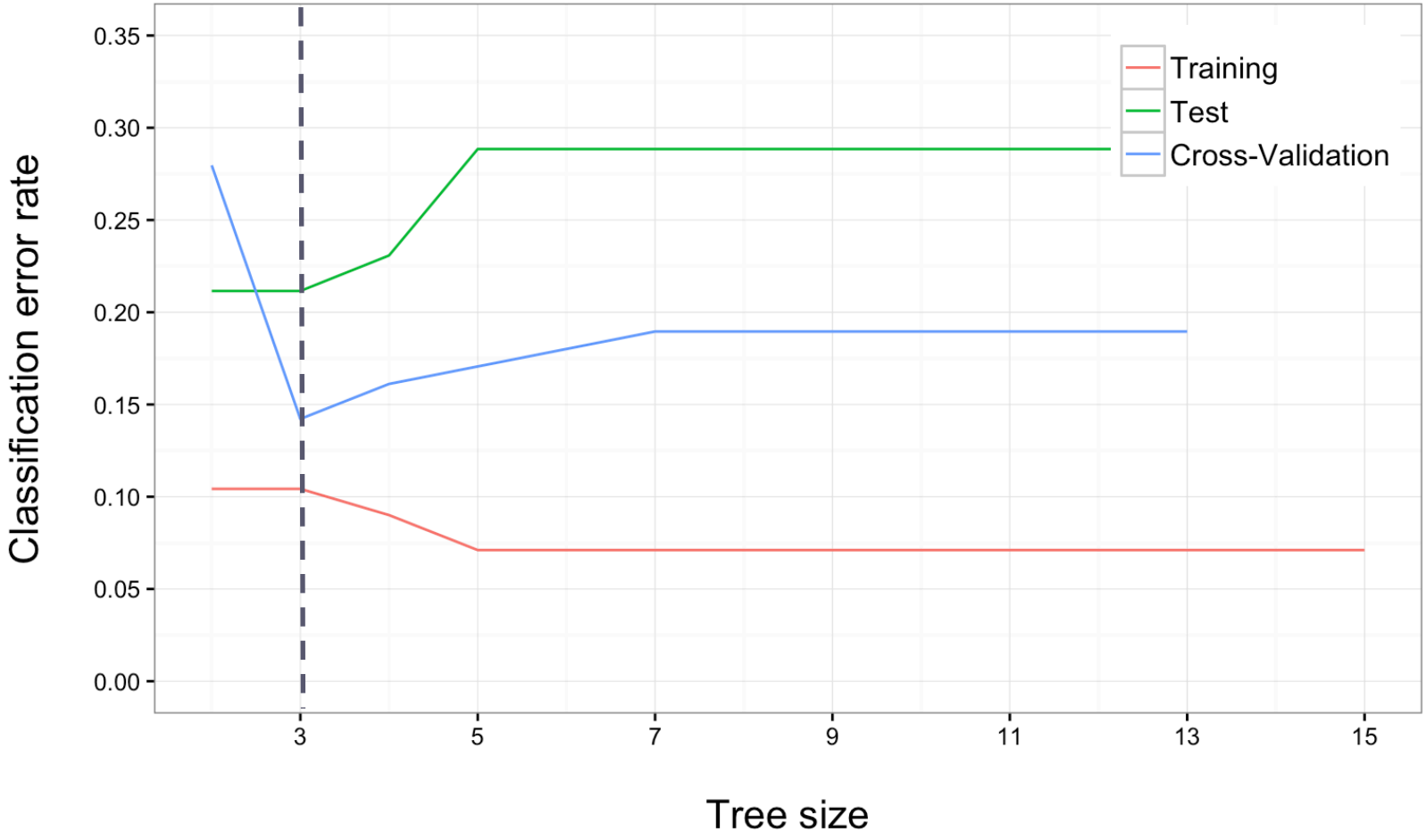
  Preporuka je primenom kros validacije (*cross validation)* utvrditi grešku pri klasifikaciji za podstabla različite veličine (tj broja terminalnih čvorova) i izabrati podstablo koje daje najmanju grešku

Orezivanje stabla kroz kros validaciju

U primeru klasifikacije igrača bejzbola, kros validacija pokazuje da se najmanja greška klasifikacije postiže u slučaju stabla veličine 3 (tj. stabla sa 3 terminalna čvora)



Orezivanje stabla kroz kros validaciju



Grafikon potvrđuje da veličina stabla utvrđena kros validacijom (n=3), vodi minimalnoj grešci na test setu

Prednosti i nedostaci stabala odlučivanja

- Prednosti:
  - Mogu se grafički predstaviti i jednostavno interpretirati
  - Mogu se primeniti kako na klasifikacione, tako i regresione probleme
  - Mogu se primeniti i u slučaju da atributi imaju nedostajuće vrednosti
- Nedostaci:
  - Daju slabije rezultate (manje tačne predikcije) nego drugi pristupi nadgledanog m. učenja