Klasifikacija

Jure Žabkar
jure.zabkar@fri.uni-lj.si

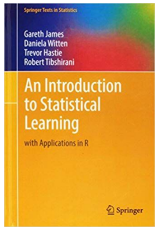
6. 4. 2021

A.I.LAB
Ljubljana

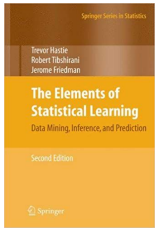
Vsebina

- Nadzorovano učenje
- Odločitvena drevesa
- Ocenjevanje verjetnosti
- Rezanje

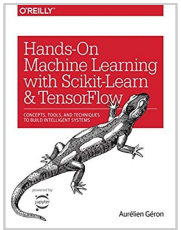
Literatura



Razdelek 8.1



Razdelek 9.2.3

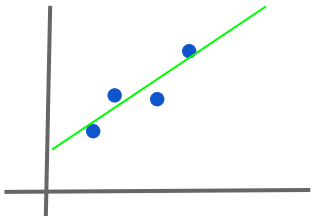


Strani: 162-168

Strojno učenje

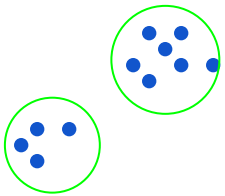
Nadzorovano

Regresija, Klasifikacija

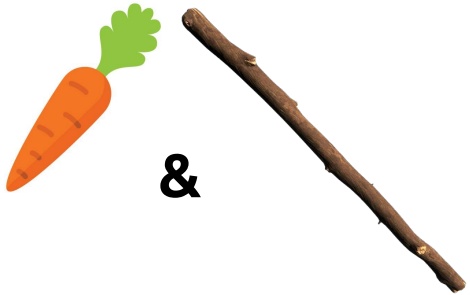


Nenadzorovano

Gručenje, povezovalna pravila



Spodbujevalno učenje



Nadzorovano učenje

- Množica učnih primerov
- Atributi, x_i
- Razred, y
- Hipoteza, h

Atributna predstavitev podatkov

	iris	sepal length	sepal width	petal length	petal width
1	Iris-setosa	5.1	3.5	1.4	0.2
2	Iris-setosa	4.9	3.0	1.4	0.2
3	Iris-setosa	4.7	3.2	1.3	0.2
4	Iris-setosa	4.6	3.1	1.5	0.2
5	Iris-setosa	5.0	3.6	1.4	0.2
6	Iris-setosa	5.4	3.9	1.7	0.4
7	Iris-setosa	4.6	3.4	1.4	0.3
8	Iris-setosa	5.0	3.4	1.5	0.2
9	Iris-setosa	4.4	2.9	1.4	0.2
10	Iris-setosa	4.9	3.1	1.5	0.1

Primeri klasifikacijskih problemov

- napovedovanje vremena (sončno, oblačno, deževno)
- diagnosticiranje pacientov (bolan, zdrav)
- klasifikacija neželene e-pošte

Prostor hipotez

Če imamo binarno klasifikacijo in n binarnih atributov, je možnih največ 2^n učnih primerov in $2^{2\hat{}n}$ hipotez
(recimo, da hipotezo opišemo s tabelo napovedi za vse primere).

- zavedati se moramo **pristranskosti** hipotez
- kako dobiti dobre hipoteze?
- kako dobro ocenjevati hipoteze?

Odločitvena drevesa

- Zelo vsestranska:
  - klasifikacija,
  - regresija,
  - naključni gozdovi
- Močan izrazni jezik
- Razumljivi modeli
- Učinkovita implementacija

Gradnja klasifikacijskih dreves

Cilj:

zgraditi čim manjše drevo,
ki je konsistentno z učnimi podatki.

Kombinatoričen prostor iskanja - vsa možna drevesa; neučinkovito

Gradnja klasifikacijskih dreves

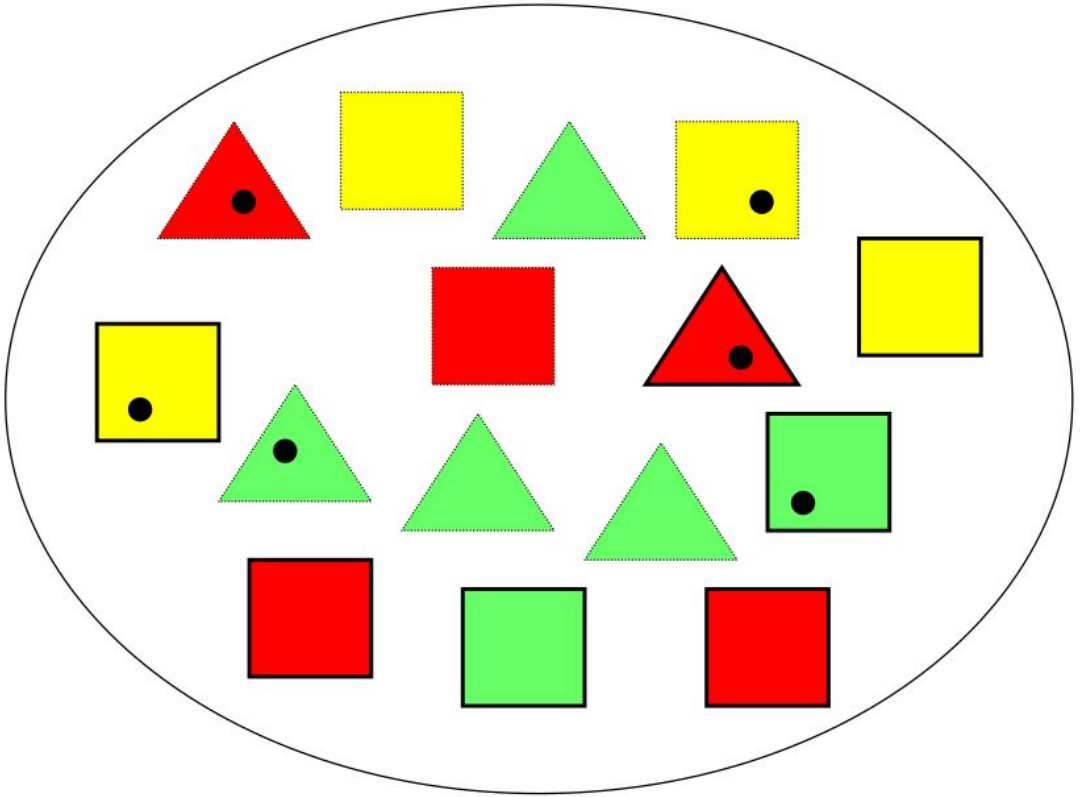
Hevristični požrešni algoritem TDIDT:

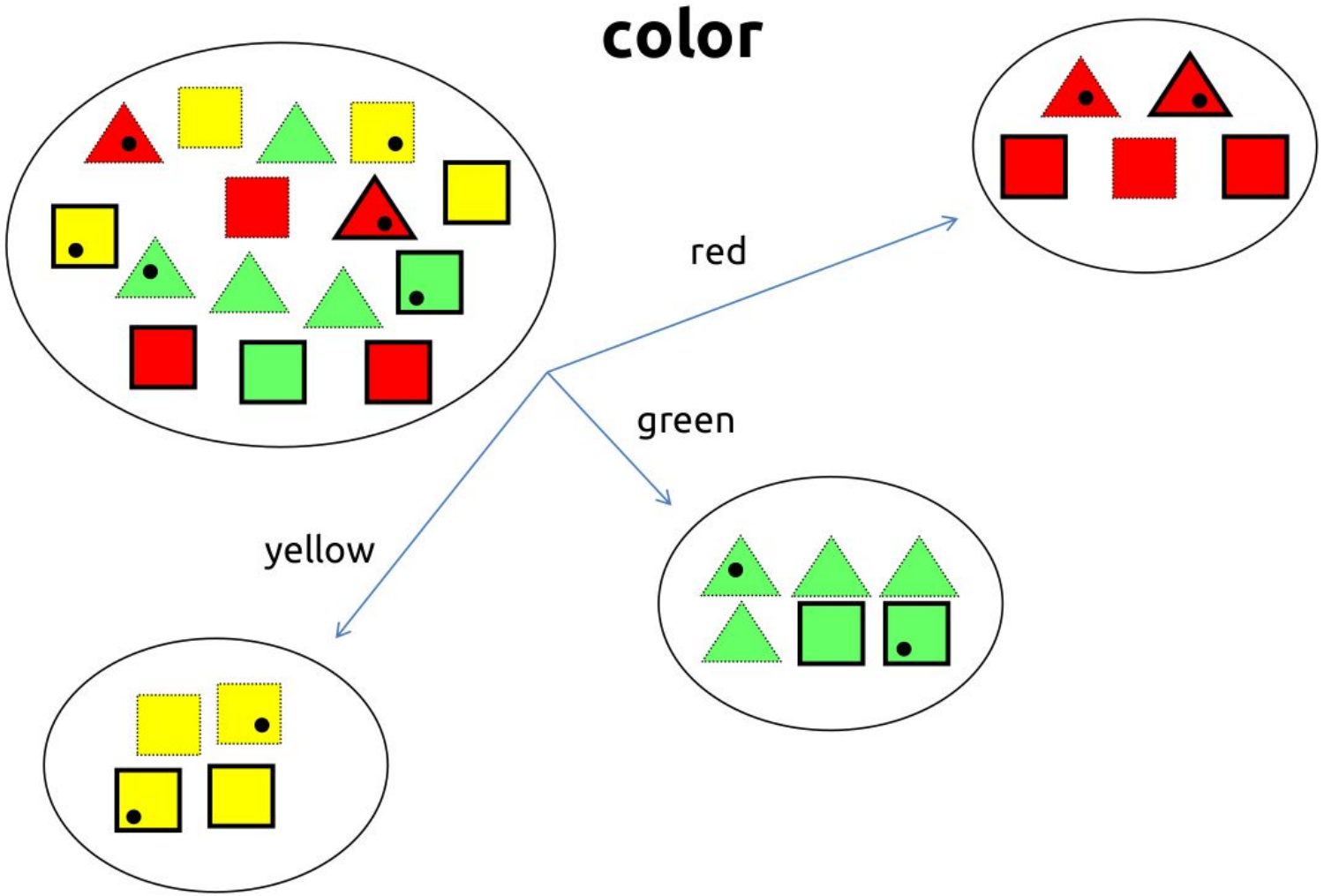
1. izberi najbolj pomemben atribut glede na razred.
2. razdeli primere v poddrevesa
3. ponovi rekurzivno na poddrevesih;
   ustavi gradnjo, ko vozlišča ni možno deliti naprej
   (vsi primeri pripadajo istemu razredu)

Izbor najbolj pomembnega atributa

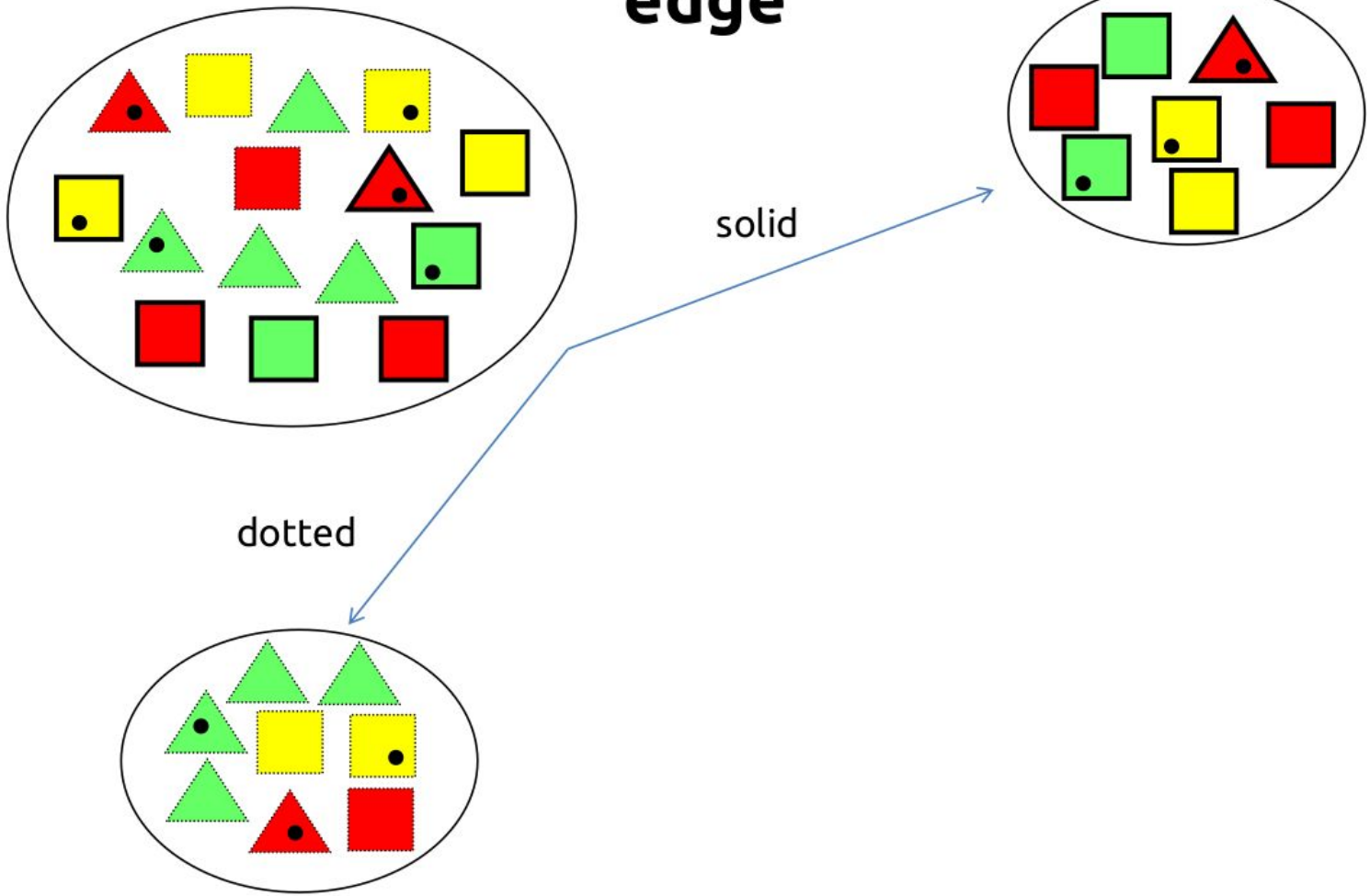
Najboljši atribut je tisti, ki - glede na razred - razdeli množico na najbolj čiste podmnožice.

Oblike likov

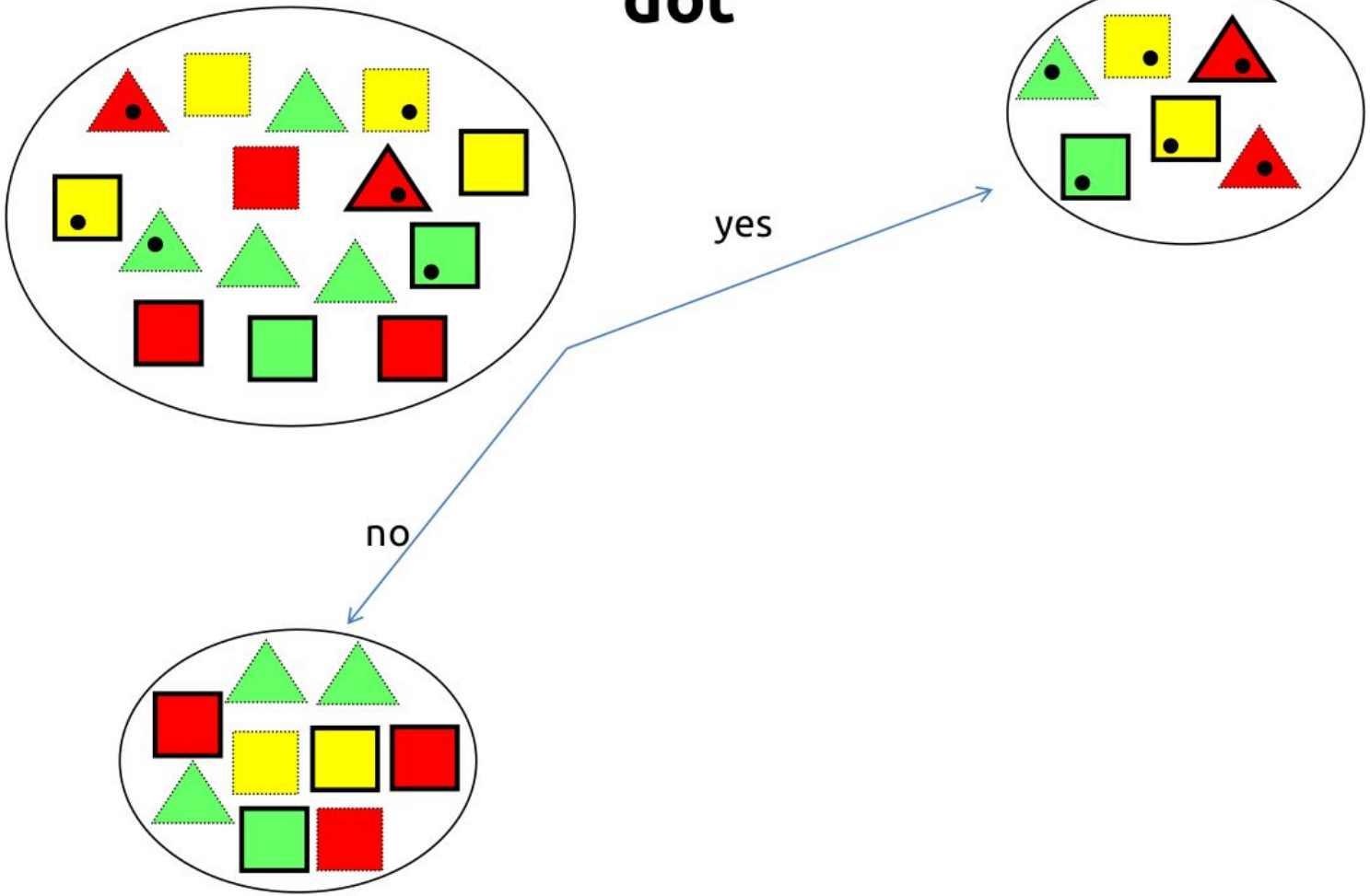


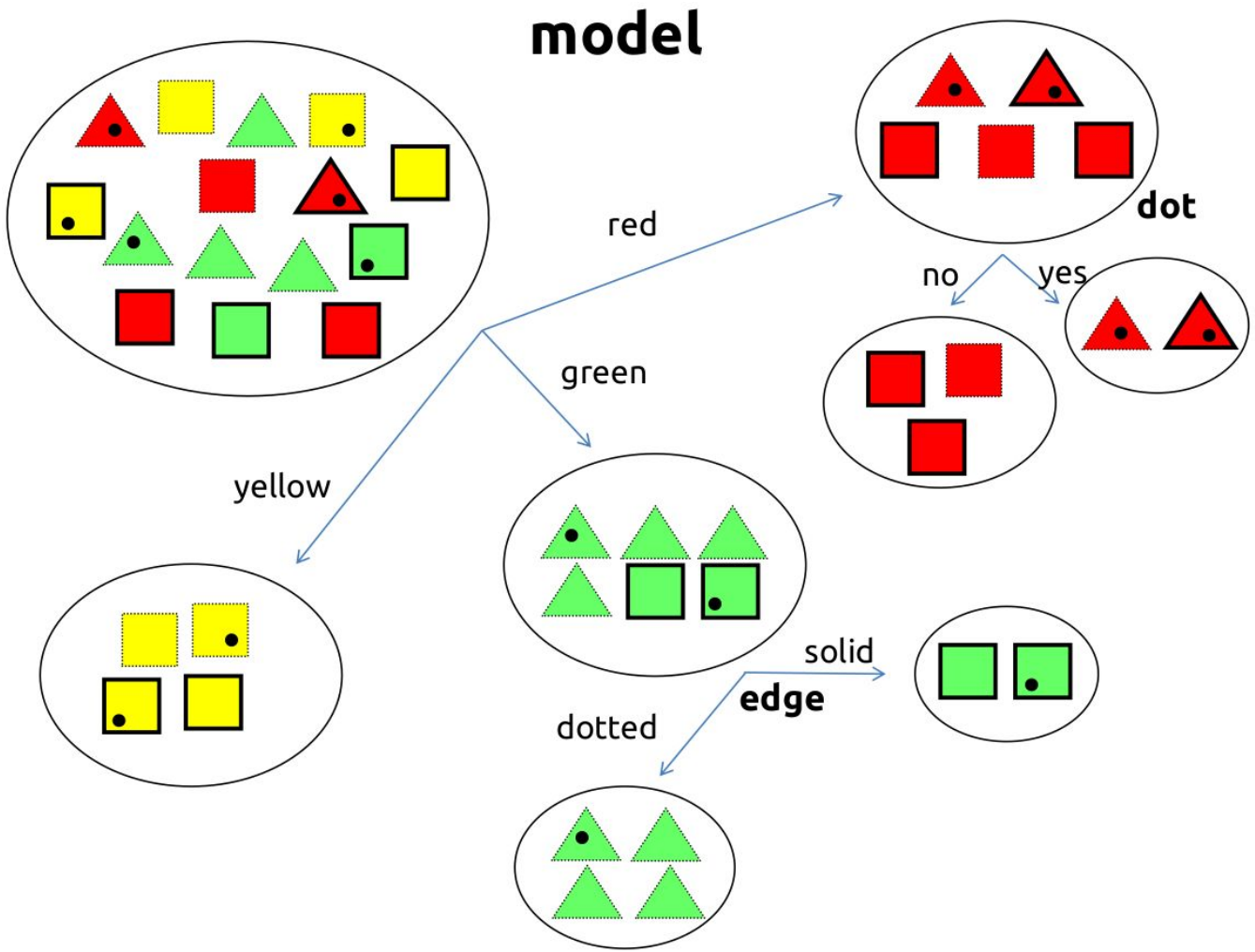

color
red
green
yellow

edge



dot




model
red
dot
no
yes
green
yellow
solid
edge
dotted

Mere nečistoče

Klasifikacijska napaka

Gini indeks

Entropija

Delež primerov z razredom k v vozlišču m

Delež večinskega razreda v vozlišču m

$$\frac{1}{N_m} \sum_{i \in R_m} I(y_i \neq k(m)) = 1 - \hat{p}_{mk(m)}$$

$$\sum_{k \neq k'} \hat{p}_{mk} \hat{p}_{mk'} = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$$

$$-\sum_{k=1}^{K} \hat{p}_{mk} \log \hat{p}_{mk}.$$

Informacijski prispevek

I=H(C) ... Entropija pred delitvijo po vrednostih atributa (v vozlišču n)

I_res=Sum p_vi H(C | vi)

InfoGain(A) = I - I_res(A)

najbolj informativen atribut ima max InfoGain

Informacijski prispevek

precenjuje kakovost večvrednostnih atributov; možne rešitve:

- relativni InfoGain (delimo ga z entropijo atributa)

- binarizacija večvrednostnih atributov

- uporaba alternativnih mer

Težave pri učenju dreves

- manjkajoče vrednosti: v splošnem imputacija (npr. manjkajoče vrednosti nadomestimo s povprečjem prisotnih vrednosti atributa). Lahko vpeljemo vrednost "manjkajoč", ki nam morda pomaga razložiti, kaj se dogaja s primeri, kjer meritev atributa manjka.
- binarna delitev boljša kot večvrednostna, ki preveč drobi na majhne podmnožice
- kratkovidnost požrešnega algoritma (XOR)
- šumni podatki...

Rezanje dreves

- Nepopolni podatki, (merske) napake v podatkih
- Učenje šuma, namesto učenja dejanske funkcije, ki generira podatke
- Slaba razumljivost dreves
- pretirano prilagajanje => nižja klasifikacijska točnost na testnih podatkih

Rezanje naprej

- omejevanje št. primerov v vozlišču
- ustavljanje gradnje pri doseženi želeni točnosti v vozlišču

Rezanje nazaj

Postopek MEP (Minimal Error Pruning)

Cilj: poreži drevo tako, da bo ocenjena klasifikacijska točnost maksimalna

Za vsako vozlišče v izračunamo:

- statično napako
- vzvratno napako

Režemo pod v, če je statična napaka manjša od vzvratne.

Ocenjevanje verjetnosti

Točnost T = verjetnost pravilne klasifikacije.

Napaka = 1 - T

N ... število vseh primerov, n ... število uspešnih poskusov

- relativna frekvenca: p = n/N

- m-ocena: p = (n + pa*m)/(N+m)

    ekspert zaupa v pa => velik m, sicer majhen m (tipično m=2)

- Laplace: p = (n+1)/(N+k)