NASLOVNICA

INFORMACIJE

6 faz (podrobno v e-učilnici)

Programska orodja
- WEKA
- MS Excel / LO Calc
- Python

Ocena:
- Izpit: 50% (lahko s kolokviji, vsak vsaj 40%, avg. 50%)
- Kratke DN: 15% (9-10 nalog a.k.a. kvizov)
- Projekt: 30% → 1. md na GH repo
- Ustni izpit: 5%
- Izračun: $0{,}15 \times DN + 0{,}5 \times PI + 0{,}3 \times PP + 0{,}05 \times UP$

Kolokviji:
1. 21.11.2024
2. 16.01.2025
- Posebne prijave (mimo ŠIS-a)

Prijava na izpit v ŠIS

10.10.24

STROJNO UČENJE, AI, PODATKOVNO RUDARJENJE

Trendi, ki vodijo v podatkovno poplavo
- Generiranje velikega št. podatkov:
  - Poslovne ustanove (npr. banke)
  - Znanstveni podatki (npr. biologija)
  - Splet (npr. socialna omrežja)

Primeri ogromnega št. podatkov
- Evropska VLBI mreža teleskopov generira 1 gigabit podatkov na sekundo za vsak teleskop v 25 dneh.
- AT&T posreduje več kot milijardo klicev na dan

Največje zbirke podatkov leta 2003
- Komercialne DB:
  - France Telecom: 30TB
  - AT&T: 26TB
- Splet:
  - Alexa internetni arhiv: 7 let = 500TB

Kaj pa danes?
- Konec junija 2017 CERN shranjeval 200PB podatkov

Področja uporabe
- Znanost (npr. bioinformatiki)
- Poslovni svet - profiliranje strank
- Splet
- Državna uprava

Prikladni problemi za podatkovno rudarjenje
- Zahtevajo odločitve na podlagi znanj
- Imajo opremljajoče okolje
- Imajo ne-optimalne trenutne rešitve
- Imajo veliko poplačilo za prave odločitve
- Paziti moramo na zasebnost!

Def. „Odkrivanja zakonitosti"
- ... v podatkih je:
  - Netrivialni proces identifikacije veljavnih, novih potencialno uporabnih in razumljivih vzorcev v podatkih

Statistika, str. učenje in PR
- Statistika:
  - Bolj usmerjena v teorijo
  - Večji fokus na testiranju hipotez
- Strojno učenje
  - Uporaba hevristik
  - Usmerjeno v izboljševanje performans učnih agentov
  - Se ukvarja z učenjem v realnem času in robotiko
- Podatkovno rudarjenje
  - Statistika & Strojno učenje
- Razmejitve so „mehke"

Zgodovina
- 1960 - Data fishing, dreding → uporabljeno s strani statistikov
- 1990 - Data mining - uporabljajo DB strokovnjaki
- 1989 - Knowledge Discovery in DB - Uporabljajo AI, ljudje s področja strojnega učenja

Tehnike str. učenja
- Klasifikacija
- Razvrščanje v skupine
- Asociacije
- Vizualizacija
- Povzemanje
- Odkrivanje odstopanj
- Regresija/ocenjevanje
- Analiza povezav

Klasifikacija
- S pomočjo statistike, odločitvenega drevesa, nevronske mreže,...

Razvrščanje v skupine
- Razvrstimo v skupine po razdalji

PR PO CRISP-DM STANDARDU

CRISP-DM
- CRoss-Industry Standard Process for Data Mining

Zakaj standarizirati?
- PR je proces, ki mora biti ponovljiv in zanesljiv
- Nudenje okvira za shranjevanje preteklih izkušenj
  - Projekte lahko repliciramo
- Lažje planiranje in vodenje projektov
- Faktor udobja za nove uporabnike
  - Dokazuje zrelost PR
  - Zmanjšuje odvisnost od tk. super-strokovnjakov

Kako je prišlo do standardizacije
- Iniciativa s strani izkušenejših podjetji
- Razvoj in dopolnitve na workshop-ih
- K procesnemu modelu prispevalo 300+ organizacij
- Leta 1999 izide CRISP-DM 1.0

CRISP-DM je:
- Ne-lastniški
- Neodvisen od aplikacije/problema
- Neodvisen od orodja/programa
- Osredotočen na poslovne probleme
- Ogrodje, ki služi kot vodilo
- Baza iz preteklih izkušenj

CRISP-DM pregled:
- Podatkovno rudarjenje kot metodologija
- Za vsakogar
- S „komplet" navodili
- V ciku iz 6 faz

CRISP-DM faze
1. Razumevanje problema
2. Razumevanje podatkov
3. Priprava podatkov
4. Modeliranje
5. Evaluacija (ocena modelov)
6. Predaja končnemu uporabniku

Razumevanje problema
- Določiti cilj projekta
- Oceniti trenutno situacijo
- Določiti cilje pod. rudarjenja
- Narediti projektni plan

Razumevanje podatkov
- Zbrati začetne podatke
- Opisati podatke
- Raziskati podatke
- Preveriti kakovost podatkov

Priprava podatkov
- Izbrati podatke
- Prečistiti podatke
- Sestaviti podatke

17.10.24

VHODNI PODATKI

Terminologija
- Vhodni podatki glede na:
  - Tip problema: zelo vse se lahko naučimo?
    - ↳ Cilj: razumljiv, uporaben opis koncepta
  - Primere: posamezni, neodvisni primeri koncepta
    - ↳ Pozor: možne so „zapletenejše" oblike VP
  - Atribute: merijo lastnosti posameznih primerov
    - ↳ Osredotočili se bomo predvsem na nominalne in numerične atribute

Kaj je to koncept
- Naloge PR (stil učenja):
  - Klasifikacija
  - Asociacije
  - Razvrščanje v skupine
  - Numerično napovedovanje / regresija
- Koncept: zadeva, ki se je želimo naučiti
- Opis koncepta: izhod algoritma

Klasifikacija
- Primeri:
  ugotavljanje prebegov, uporaba DNA podatkov pri diagnozi, ...
- Klasifikacija je nadzorovano učenje
- Izid imenujemo razred primera
- Uspeh merimo na podatkih, za katere prav tako poznamo izid (testni podatki)
- V praksi se uspeh učenja pogosto meri subjektivno

Asociacije
- Primer
  Analiza nakupovalnih navad
- Lahko uporabimo tudi brez poznavanja razreda
  *

Razvrščanje v skupine
- Primeri
  profiliranje kupcev
- Naloga:
  najti skupine primerov, ki so si med seboj podobni
- Razvrščanje v skupine je nenadzorovano
- Uspeh učenja se pogosto meri subjektivno

Numerično napovedovanje
- Pogosto se imenuje regresija
- Gre za vrsto klasifikacije, le da je razred numeričen
- Učenje je nadzorovano
- Uspeh učenja merimo na testnih podatkih

Kaj je primer/instanca?
- Instanca: točno določen tip primera
- Vhod v shemo učenja: mn. instanc / dataset
- Omejene oblike vhodnih podatkov

Generiranje preproste datoteke
- Proces imenujemo denormalizacija
- Težave: relacije, ki nimajo v naprej določenega št. objektov
- Denormalizacija lahko ustvari „lažne" odvisnosti, ki zgolj odražajo strukturo PB

Kaj je atribut?
- Vsak primer je opisan z vnaprej določenim št. značilk, to so atributi primera
- V praksi lahko št. atributov variira
- So roden problem: obstoj atributa je odvisen od drugega atributa

*

Urejene vrednosti
- Podobno nominalnemu atributu, le da se jih da urediti
  *

Intervalne vrednosti
- Intervalne vrednosti so urejene in jih izražamo v fiksnih in enakih enotah
- Razlika dveh vrednosti ima smisel
- Seštevek in produkt nimata smisla

Razmernostne vrednosti
*

Tipi atributov v praksi
- Večina alg. podpira le nominalne in numerične

Metapodatki
- Info. o podatkih, ki vključujejo predznanje o problemu oz. konceptu
- Lahko jih uporabimo za omejevanje preiskovalnega prostora

Priprava vhodnih podatkov
- Težava: različni viri
- Denormalizacija ni edina težava
- Včasih potrebujemo „zunanje podatke"
- Pomembno: tip in nivo agregacije podatko

PRIPRAVA PODATKOV 24.10.24

Koraki čiščenja podatkov
- Pridobivanje podatkov in metapodatki
- Manjkajoče vrednosti
- Datumski format podatkov
- Pretvorba iz nominalnega v numerično
- Diskretizacija numeričnih podatkov
- Validacija podatkov in statistike

Pridobivanje
- Večinoma shranjeni v DBMS
- Podatki v „preprostih" .txt datotekah
  - Fiksni
  - Razmejeni (.csv, .arff, ...)
- Preveriti št. atrib. pred in po pretvorbi

Metapodatki
- Tipi atributov (binarni, nominalni ...)
- Vloga atributov:
  - Vhodni
  - Ciljni
  - id/pomožni
  - Zanemarljivi
  - Uteži
- Deskriptorji atributov (dodaten opis)

Spremembe formata
- Pretvorba v standardni format
- Manjkajoče vrednosti
  *

Manjkajoče vrednosti
- Različne oznake (< >, "0", ".", "N/A", ...)
- Standardizacija zapisa
- Obravnava manjkajočih vrednosti
  - Izločimo primere, ki jih vsebujejo
  - Izločimo atribute, ki jih vsebujejo
  - Obravnavamo kot ločene vrednosti
  - Nadomestimo s povprečjem, modusom, ...
  - Prepustimo algoritmu strojnega učenja

Datumi
- Želja: predstaviti vse datume na enoten način
- Datumi nastopajo v različnih formatih
- Najpogosteje zadostuje le leto (YYYY)
- Predstavitev datumov kot YYYYMM je ok, a lahko predstavlja probleme

Unificirani datumski formati
- Da ohranimo intervale, lahko uporabimo
  - Unix sistem: št. sec od 1.1.1970 (00:00:00 UTC)
  - SAS sistem: št. dni od 1.1.1960
- Težave:
  - Vrednosti niso intuitivne

KSP format
- Ohranja intervale (skoraj)
- Ožitno v kateri četrtini leta je nek datum
- Konsistentno z dnevi, ki se začnejo ob poldne
- Lahko razširimo, da vsebuje tudi ure...

Y2K: 2-ciferno leto
- 2-ciferna leta v starih podatkih - zapuščina Y2K

Pretvorbe: nominalno → numerično
- Nekateri ML alg. interno podpirajo nominalne atrib.
- Nekateri drugi delujejo le z numeričnimi atributi
- Za uporabo slednjih moramo nominalne pretvoriti v numerične

Pretvorbe: diskretizacije
*

Diskretizacija: enake širine
- Vsi predalčki enako široki
- Lahko povzroči „luknje" v histogramu

Diskretizacija: preciznost metode enakih višin
- Ne generira „lukenj"

Diskretizacija: razredno-odvisna
- Št. vrednosti prilagajamo razredu

Diskretizacija: zaključki
- D. enakih širin je najpreprostejša
- D. enakih frekvenc da boljše rezultate
- Druge metode

Osamelci in napake
- Osamelci so vrednosti izven smiselnih okvirov
- Pristopi:
  - Ne storimo ničesar
  - določimo zg. in sp. meje
  - Uporabimo diskretizacijo

Izbor atributov
- Najprej: odstranimo atribute brez ali z minimalno variabilnostjo vrednosti
- Pregledamo št. vrednosti atributa in le tega odstranimo

Napačni napovedovalci
- Atributi, ki so močno korelirani z razredom, a opisujejo dogodke, ki so se zgodili istočasno ali kasneje kot dogodek opisan z razredom
- Če PB ne beleži časa dogodkov, lahko napačni napovedovalec izpade kot dober napovedovalec

Napačni napovedovalci: iskanje sumljivih atributov
- Zgradimo odločitveno drevo
- Obravnavamo atribute z veliko napovedno močjo kot „sumljive"
- Preverimo „sumljive"
- Odstranimo napačni napovedovalec in postopek ponovimo

Izbor najrelevantnejših atributov
*

Izpeljani atributi
- Boje
*

Neuravnotežena porazdelitev razreda
- Včasih so frekvence vrednosti razreda neenakomerne
- Podobno tudi pri več-vrednostnih razredih
- Večinski klasifikator je lahko 97% točen
*

Učenje iz neuravnoteženih podatkov
*

ALGORITMI ZA KLASIFIKACIJO

Klasifikacija
- Naloga: zgraditi model / klasifikator z uporabo že klasificiranih primerov.
- Nadzorovano učenje: vrednost razreda primerov, ki so del learning dataset-a so poznani.
- Klasifikator je lahko: mn. pravil, odločitveno drevo, nevronska mreža ...
- Tipične aplikacije: odobritve kreditov, neposredni marketing, diagnoze v medicini, ...

Enostavni algoritmi
- Enostavni algoritmi pogosto zelo dobro delujejo
- Mnogo primerov enostavnih struktur:
  - Klasifikator „večinskega razreda
  - En atribut „sprav vse delo"
  - Vsi atributi prispevajo v enaki meri
  - Utežena lin. kombinacija
  - Na podlagi razdalje
  - Preprosta logična pravila
- Uspeh odvisen od podatkov

Obravnava numeričnih atributov
- Z uporabo „razredne" diskretizacije
- Razpon vrednosti atributa razdelimo na intervale
  - Uredimo vrednosti atributa po velikosti
  - Postavimo meje intervalov, kjer se spremeni vrednost razreda
  - Na ta način minimiziramo napako

Problem prekomernega prilagajanja
- Ta način diskr. zelo občutljiv na šum:
  - Primer z napačno vrednostjo razreda bo zelo verjetno povzročil tvorbo novega intervala

*

Bayes-ovo (statistično) modeliranje
- Upoštevajo se vsi atributi
- Dve predpostavke - atributi naj bodo:
  - Enako pomembni
  - Statistično neodvisni
- Predpostavka „neodvisnosti" skoraj nikoli ne drži
- V praksi deluje dobro

Klasifikacija z naivnim Bayesom
- Klasifikacijsko učenje:
  - Dejstva E = vrednosti atributov
  - Hipoteza H = vrednost razreda
- Naivna predpostavka: dejstva razdelimo na dele, ki so neodvisni

$$Pr[H|E] \frac{Pr[E_1|H]Pr[E_2|H] \dots Pr[E_n|H]Pr[H]}{Pr[E]}$$

Problem „frekvenca = nič"
- Dodamo 1 vsem frekvencam v tabeli

Popravljene ocene
*

Manjkajoče vrednosti
- Pri učenju: ne upoštevamo pri izračunu
- Pri klasifikaciji: ne upoštevamo atributa pri izračunu

Numerični atributi
- Običajna predpostavka: atributi imajo Gaussovo porazdelitev
- Fja gostote je def. z 2 parametroma
  - Povprečje:

$$\mu = \frac{1}{n} \sum_{i=1}^{n} x_i$$

  - Standardna deviacija

$$\sigma = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \mu)^2$$

- Fja gostote vrjetnosti je:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

ODLOČITVENA DREVESA

Odločitveno drevo
- Notranje vozlišče predstavlja test po atributu
- Veja predstavlja rezultat tega testa
- List predstavlja oznako lista
- V vsakem vozlišču: en atribut je izbran, po katerem delimo učne primere v kar se da „čiste" podmnožice
- Nove primere klasificiramo tako, da sledimo ustreznim potem

Gradnja odločitvenega drevesa
- Od zgoraj navzdol
  - Na začetku vsi primeri v korenu
  - Rekurzivno delimo primere v podmnožice

Izbor atributa za deliter primerov
- Ocenimo razpoložljive atrib. glede na njihovo sposobnost delitve
- Tipične fje primernosti
  - Informacijski prispevek
  - Razmerje info. prispevka
  - Gini indeks

Kriterij za izbiro „najboljšega" atributa
*

Računanje informacije
- Info. prispevek merimo v bitih
  - Informacijo, da napovemo dogodek glede na verjetnostno porazdelitev imenujemo ENTROPIJA
- Formula:
  $entropija(p_1, \dots, p_n) = -p_1 \log p_1 - \dots - p_n \log p_n$

Računanje info. prispevka
- Info. prispevek = (info. pred razbitjem) - (info. po razbitju)

Končno odločitveno drevo
- Pozor: listi niso vedno čisti

Atributi z veliko vrednostmi
- Težava: atributi z veliko vrednostmi
*

Razmerje info. prispevka
- Gain Ratio
*

Intrinzična informacija
- Entropija porazdelitve posameznega atributa

*

Izračun razmerja info. prispevka
- Pomembnost atributa se zmanjšuje z naraščanjem intrinzične informacije

Gini index
- Če podatki (T) vsebujejo primere iz n razredov, je def. kot:

$$gini(T) = 1 - \sum_{j=1}^{n} p_j^2$$

Vaje pred 1. kolokvijem 14.11.24

(10 t) 1. Datum v KSP (na 3 decimalke)

Jan 0 + dan
Feb 31 + dan + 1
Mar 59 + dan + 1
Apr 90 + dan + 1
Maj 120 + dan + 1
Junij 151 + dan + 1

(3t) $KSP(200) = 2017 + \frac{59+19-0,5}{365} = 2017,212$

(3t) $KSP(201) = 2019 + \frac{90+1-0,5}{365} = 2019,248$

(4t) $KSP(202) = 2012 + \frac{152+13-0,5}{366} = 2012,449$

(10t) 2.

Vrednosti po velikosti:

2, 4, 4, 7, 9, 10, 11, 12, 13, 15, 18, 19

1. 2. 3. 4. 5. 6. 7. 8. 9. 10. 11. 12.

min ↑ max ↑

(1t) min = 2 $R(Q_1) = (n+1) \cdot 0{,}25 =$

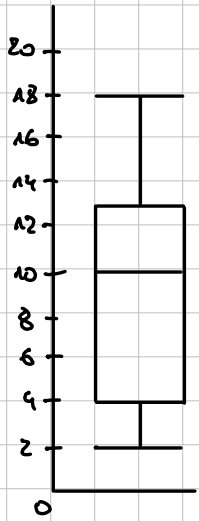
(1t) max = 18 $= 12 \cdot 0{,}25 = 3$

(2t) med. = 10 $R(Q_3) = (n+1) \cdot 0{,}75 =$

(2t) $Q_1 = 4$ $= 12 \cdot 0{,}75 = 9$

(2t) $Q_3 = 13$

(2t)



min = 2, max = 19

$med = \frac{10+11}{2} = 10{,}5$

$R(Q_1) = 13 \cdot 0{,}25 = 3{,}25 \Rightarrow 4{,}75$

$R(Q_3) = 13 \cdot 0{,}75 = 9{,}75 \Rightarrow 14{,}5$

(10 t) 3. Diskretizacija (G, 3 intervali, enake višine)

0 — 2

4 — 4 — G1 — $\left\lfloor \frac{n}{3} \right\rfloor = \left\lfloor \frac{11}{3} \right\rfloor = 3$ ost. 2

4

8 — 7

G2 — 9

10 — G2

11

12.5 — 12

G3 — 13

15 — G3

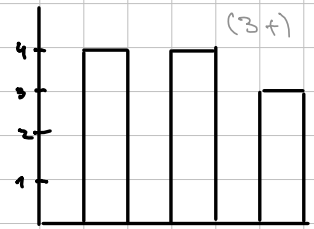
G4 — 18

20

(4t)

(3t) diskretizacija v tabeli

(3t)



Diskretizacija (enake širine, G, 4 in)

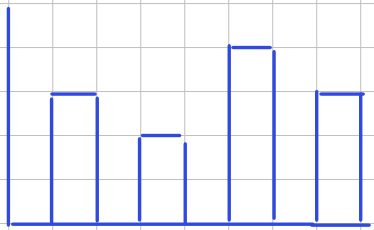
R[0,20]

G1: [0,5)

G2: [5,10)

G3: [10,15)

G4: [15,20]



(15 t) 4. OneR (E, F)

(5 t)

E	+	-	∘	Napaka
L	3	2	0	2
H	2	2	2	4

Skupna napaka je 6

Izberemo

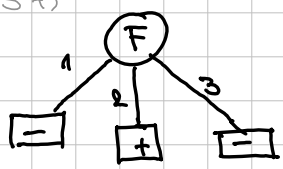
(5 t)

F	+	-	∘	Napaka
1	1	2	1	2
2	3	0	0	0
3	1	2	1	2

Skupna napaka je 4

(2 t) za pravilno ugotovitev manjše napake

(3 t)



← Enonivojsko odločitveno drevo

(25 t) 5. Naivni Bayes (E, F)

(3 t)

E	+	-	∘
L	4	3	1
H	3	3	3

F	+	-	∘
1	2	3	2
2	4	1	1
3	2	3	2

C	+	-	∘
	6	5	3

200: $l(+) = \frac{4}{7} \cdot \frac{2}{8} \cdot \frac{6}{14} = \frac{3}{49} \doteq 0{,}06122$ (4 t)

$l(-) = \frac{3}{6} \cdot \frac{3}{7} \cdot \frac{5}{14} = \frac{15}{196} \doteq 0{,}07653$

$l(\circ) = \frac{1}{12} \cdot \frac{2}{5} \cdot \frac{3}{14} = \frac{3}{140} = 0{,}02143$

Class v tabeli = -

(5 t)

E	+	-	∘
L	$\frac{4}{7}$	$\frac{3}{6}$	$\frac{1}{4}$
H	$\frac{3}{7}$	$\frac{3}{6}$	$\frac{3}{4}$

F	+	-	∘
1	$\frac{2}{8}$	$\frac{3}{6}$	$\frac{2}{5}$
2	$\frac{4}{8}$	$\frac{1}{6}$	$\frac{1}{5}$
3	$\frac{2}{8}$	$\frac{3}{6}$	$\frac{2}{5}$

C	+	-	∘
	$\frac{6}{14}$	$\frac{5}{14}$	$\frac{3}{14}$

201: $l(+) = \frac{3}{7} \cdot \frac{2}{8} \cdot \frac{6}{14} = \frac{9}{196} \doteq 0{,}04591$ (4 t)

$l(-) = \frac{3}{6} \cdot \frac{3}{7} \cdot \frac{5}{14} = \frac{15}{196} \doteq 0{,}07653$

$l(\circ) = \frac{3}{4} \cdot \frac{2}{5} \cdot \frac{3}{14} = \frac{9}{140} = 0{,}02143$

Class v tabeli = -

202: $l(+) = \frac{3}{7} \cdot \frac{4}{8} \cdot \frac{6}{14} = \frac{9}{98} \doteq 0{,}09184$ (4 t)

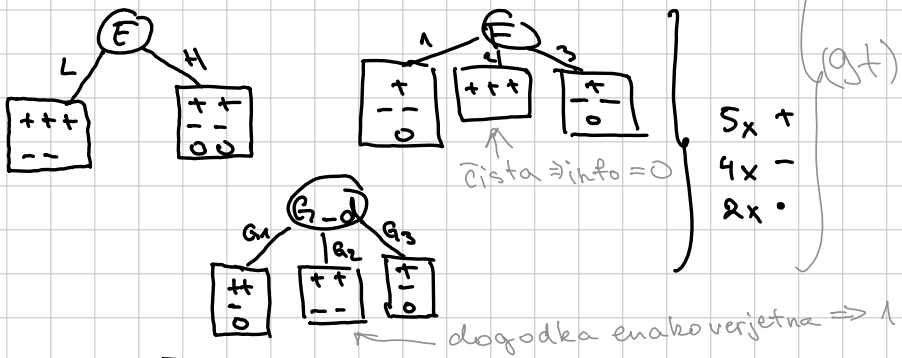
$l(-) = \frac{3}{6} \cdot \frac{1}{7} \cdot \frac{5}{14} = \frac{5}{196} \doteq 0{,}02551$

$l(\circ) = \frac{3}{4} \cdot \frac{1}{5} \cdot \frac{3}{14} = \frac{9}{280} \doteq 0{,}03214$

Class v tabeli = +

(3 t) po 1 točka za pravilno izbiro klasifikacije

(30t) 6. Odločitveno drevo (Info Gain, E, F, G_d)



$$IBS = Info\left(\frac{5}{11}, \frac{4}{11}, \frac{2}{11}\right) = -\frac{5}{11}\cdot\log_2\frac{5}{11} - \frac{4}{11}\cdot\log_2\frac{4}{11} - \frac{2}{11}\cdot\log_2\frac{2}{11} \doteq 0{,}4500$$

Ni pomembno ali je $\log_2$, $\log_{10}$ ... dokler je za vse izračune uporabljen isti $\log_x$.

(5t)

$$Info(E) = \frac{5}{11}\left[Info\left(\frac{3}{5}, \frac{2}{5}\right)\right] + \frac{6}{11}\left[Info\left(\frac{2}{6}, \frac{2}{6}, \frac{2}{6}\right)\right] =$$

$$= \frac{5}{11}\left[-\frac{3}{5}\log\frac{3}{5} - \frac{2}{5}\left(\log\frac{2}{5}\right)\right] + \frac{6}{11}\left[-\frac{2}{6}\left(\log\frac{2}{6}\right)\right]\cdot 3 \doteq 0{,}3931$$

$$Gain(E) = IBS - Info(E) \doteq 0{,}0569$$

(5t)

$$Info(F) = 2\cdot\frac{4}{11}\left(-\frac{1}{4}\log\frac{1}{4} - \frac{1}{2}\log\frac{1}{2} - \frac{1}{4}\log\frac{1}{4}\right) \doteq 0{,}3284$$

$$Gain(F) = IBS - Info(F) = 0{,}1216 \leftarrow \text{Best}$$

(5t)

$$Info(G\_d) = \frac{4}{11}\left(-\frac{1}{4}\log\frac{1}{4} - \frac{1}{2}\log\frac{1}{2} - \frac{1}{4}\log\frac{1}{4}\right) + \frac{4}{11}\cdot 1 + \frac{3}{11}\left(-\frac{1}{3}\log\frac{1}{3}\right)\cdot 3$$
$$\doteq 0{,}65786$$

$Gain(G\_d) =$? Preveri s kalkulatorjem doma!

(6t) Uporabimo OD od F. Class v tabeli: 200 = −
201 = −
202 = +

6. Gini index

$$Gini(E) = \frac{5}{11}\left[1-\left(\left(\frac{3}{5}\right)^2+\left(\frac{2}{5}\right)^2\right)\right] + \frac{6}{11}\left[1-\left(\left(\frac{1}{3}\right)^2-3\right)\right]$$

$$= \frac{5}{11}\cdot\frac{12}{25} + \frac{6}{11}\cdot\frac{2}{3} = \frac{12}{55} + \frac{4}{11} = \frac{12+20}{55} = \frac{32}{55} \doteq 0{,}582$$

Izračunamo še Gini(F) in Gini(G_d) in izberemo najmanjšega

STROJNO UČENJE NA REALNIH PODATKIH: C4.5

Industrijski algoritmi
- Morajo omogočati realne podatke

C4.5
- Razširi ID3

Numerični atributi
- Izberemo mejno vrednost in razdelimo v 2 podmn. glede na pogoj
- Bolj zahtevni za računanje

Sortiranje
- Uredimo po velikosti $O(n\log n)$
  - Da se ne ponavljamo rekurzivno ponavljamo v otrokih $O(n)$
- Entropijo lahko računamo le med točkami z različnimi razredi

Razbitja: binarno vs večkratno
- Čas zahtevnost se poveča če razbitje večkratno

Manjkajoče vrednosti
- Če predpostavimo, da manjkajoča vrednost svoja vrednost, to lahko uporabimo le pri nominalnih podatkih
- Manjkajoče vrednosti pošiljamo po drevesu navzdol

Rezanje
- Cilj: preprečiti prekomerno prileganje podatkov
- 2 vrsti:
  - Predhodno → ustavimo, ko informacije postanejo neuporabne ✗
  - Naknadno → po zaključku porežemo veje z neupo. info ✓
    - zamenjava poddreves: od spodaj navzgor, poskušamo zamenjati podrevo z listom
    - povzdigovanje poddreves: počasnejše od zamenjav
- Rešeno le, če je napaka po rezanju manjša

Zahtevnost gradnje drevesa
- $O(m \cdot n \log n)$ ⇒ gradnja, kjer m atributov, n instanc ter višina $O(\log n)$
- Zamenjava $O(n)$
- Povzdigovanje $O(n(\log n)^2)$

KLASIFIKACIJSKI ALG. - ODLOČITVENA PRAVILA

Gradnja odločitvenih pravil
- Vsaka pot od korena do lista je svoje pravilo

Prekrivni algoritem
- Pokrivamo pravila
- V množici pravil pokrivamo dele dokler ne dobimo željenega rezultata

Pravila vs Drevo
- Vsako drevo lahko prepišemo v pravila, obratno pa ne.

Izbor pogoja
- Za čimvečjo natančnost
  - $t \Rightarrow$ št. instanc
  - $n \Rightarrow$ št. poz. instanc
  - $t \cdot p \Rightarrow$ št. napak
- Zaključimo, če $p/t = 1$ ali ne moremo razdeliti več
- Če več pogojev $p/t = 1$ vzamemo tistega, kjer sta p in t večja
- Tak alg. imenujemo prism alg.

Pravila in odložitveni seznam
- Zgenerira se odložitveni seznam - tretirana po vrsti
- Potrebujemo default rule, ker se lahko pravila prekrivajo

„Osami in vladaj"

(SimpleEducationalSchemes v weki za prism alg in ostale)

EVALUACIJA 12. 12. 24

Evaluacijski problemi
- Možna merila
  - Klasifikacijska točnost
  - Cena napak

Classifier Error Rate
- Naravna mera performansa

Evaluacija na velikih podatkih
- Lahko naključno razbijemo na $\frac{2}{3}$ učna, $\frac{1}{3}$ testna mn.

Neuravnoteženi podatki
- Če imajo razredi zelo različne frekvence
- Rokovanje:
  - Uravnotežimo 50/50 +/- v test. mn. in učni mn.

Napovedovanje performanca
- Predpostavimo, da je napaka 25%. Kako blizu je realni napaki, je odvisno od velikosti podatkov

Evalvacija na malo podatkih
- Večkratno naključno razbitje
  - Še vedno ni optimalno ⇒ Prečno preverjanje

Prečno preverjanje
- Mn. razbijemo na več podmn.
- Največkrat 10x preverimo
- Leave-One-Out preverjanje - ko je $k = n$, k = št. preverjanj, n = št. podatkov
  - Onemogoča stratifikacijo

RAZVRŠČANJE V SKUPINE IN PCA 19.11.24

Razvrščanje v skupine
- Nenadzorovano učenje
- Najde "naravne" skupine neoznačenih primerov
- Več metod:
  - Glede na tip podatkov
  - Glede na verjetnost pripadanja posameznega primera skupini
  - Glede na prekrivanje skupin
  - Glede na organizacijo skupin
  - Glede na smer razvrščanja

Evaluacija metod
- Ročni pregled
- Benchmarking na obstoječe oznake
- Mere kakovosti skupin
  - Mere razdalje/verjetnosti

Fja razdalje
- Preprosto: $d(e_1, e_2) = |A(e_1) - A(e_2)|$
- Bolj zapleteno: $d(e_1, e_2)$ = evklidska razdalja
- Nominalni atributi: $d(e_1, e_2) = 0$, če $A(e_1) = A(e_2)$, drugače 1
- Možne fje:
  - Evklidska: $\sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$, diskr. atrib.: $p_i = q_i \Rightarrow 1$ sicer 0
  - Hammingova: šteje koliko atrib. ima različne vrednosti
  - Manhattanska: $\sum_{i=1}^{n} (|p_i - q_i|)$
  - Maksimalna: $\max(|p_i - q_i|)$

Metoda k-tih povprečji (k-means)
- Deluje predvsem na numeričnih podatkih
- Postopek:
  1. Naključno izberemo k središč skupin

  Ponavljamo:

  2. Vsak primer označimo s pripadajočim najbližjim središčem
  3. Središča skupin premaknemo v težišča označenih primerov

  ... dokler ne skonvergira (središča se nehajo spremin.)
- Težave:
  - Končna težišča odvisna od začetne naključne postavitve
  - Osamelci lahko zelo "povlečejo" težišča proti sebi.
- k-medoid: spremenjena metoda k-tih median
  - Neobčutljiva na osamelce

Hierarhično razvrščanje v skupine
- 2 pristopa:
  - Odspodaj navzgor (na kolokviju): razdelimo primere vsakega v svojo skupino, nato jih združujemo dokler niso vsi primeri v eni skupini
  - Odzgoraj navzdol

REGRESIJA

Linearni modeli
- Deluje najbolj naravno z numeričnimi atrib.
- Stand. tehnika za napovedovanje s št.
  - Rezultat: lin. kombinacija atributov
- Utrži izračunane iz učnih podatkov

Zmanjševanje kvadratne napake
- Izberemo k+1 koeficientov
- Napaka: $\sum_{i=n}^{n} \left(x^{(i)} - \sum_{j=0}^{k} w_j a_j^{(i)} \right)^2$
- Izvedljiva če več primerov kot atributov
- Zmanjšati absolutno napako je težje

Instance-based representation
- Najpreprostejši način učenja
  - Učni primeri so uporabljeni za klasifikacijo novega

*

Fja razdalje
*

Normalizacija
- Različni atributi merjeni na različnih skalah ⇒ normaliziramo
- Nominalni atributi: razdalja 0 ali 1
- Manjkajoče vrednosti: predpostavimo max razdaljo

1-NN
- Pogosto natančen
- Počasen:
  - Preproste verzije pregledajo vse učne podatke za napoved
- Predpostavi enako pomembnost vseh atributov
- Občutljiv na šum
- Statistiki uporabljajo k-NN že od 1950ih
  - n-NN = zeroR