

DATA MINING

K-nearest neighbor

- Clasificarea pentru o inregistrare noua se face comparand-o cu inregistrarile similare

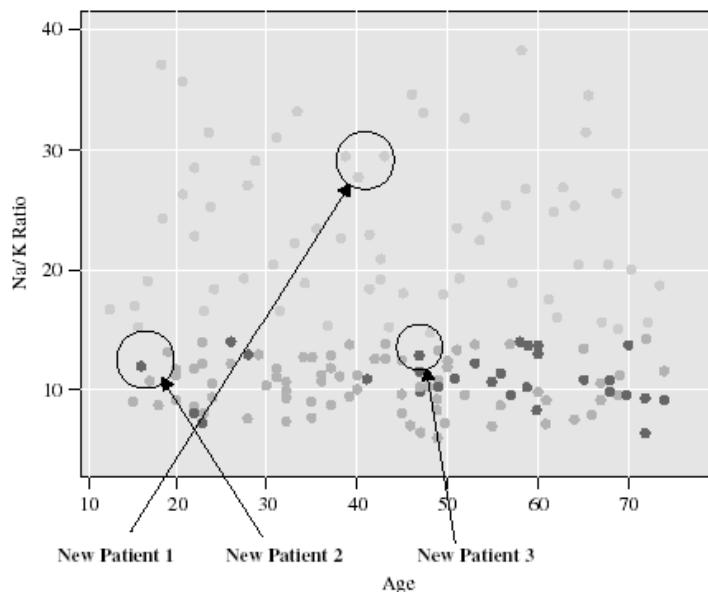


Figure 5.6 Scatter plot of sodium/potassium ratio against age, with drug overlay.

La adaugarea New Patient 1, este usor de determinat la care categorie de medicamente trebuie incadrat

La adaugarea New Patient 2 si New Patient 3, categoria de medicamente la care acestia trebuie incadrati depinde de numarul de indivizi similari cu care se face comparatia

Referinta figura: D. Larose, *Discovering Knowledge in Data. An Introduction to Data Mining*, John Wiley & Sons, 2005.

- Intrebari care se pun atunci cand folosim acest algoritm:
 - Care este valoarea lui k ?
 - Cum masuram similaritatea (distanta)?
 - Cum combinam informatia de la mai multe inregistrari?
 - Ar trebui ca unele inregistrari sa aiba o influenta mai mare decat altele (probabil cele care sunt mai aproape de noua inregistrare)?

- **Functia distanta**
(cum definim similaritatea?)
- Exemplu:
 - Un barbat de 50 ani este mai “aproape” de un barbat de 20 ani sau de o femeie de 50 ani?
- Proprietatile functiei distanta:
 - $d(x,y) \geq 0$, $d(x,y)=0$ dd $x=y$
 - $d(x,y)=d(y,x)$
 - $d(x,z) \geq d(x,y)+d(y,z)$
- Cea mai folosita functie distanta este distația euclidiană

$$d_{\text{Euclidean}}(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_i (x_i - y_i)^2}$$

- Unele atribute cu valori mari (cum ar fi venitul) pot avea o influență mai mare decât alte atribute care sunt măsurate la o scăala mai mică (cum ar fi numărul de ani lucrați)
- Pentru a evita acest lucru se vor face normalizări:
 - Normalizare min-max
 - Standardizare z-score
- Pentru valori categoriale, distanța euclidiană nu este potrivită, putându-se folosi în acest caz funcția “diferit de”

$$\text{different}(x_i, y_i) = \begin{cases} 0 & \text{if } x_i = y_i \\ 1 & \text{otherwise} \end{cases}$$

Exemplu

- Pentru variabila varsta avem
 - $\max(x) - \min(x) = 50$
 - $\min(x) = 10$
 - Mean=45
 - SD=15
- Fie A=barbat in varsta de 50 ani, B=barbat in varsta de 20 ani, C=femeie in varsta de 50 ani
- vom calcula similaritatea (distanta) dintre A si B, respectiv dintre A si C

Patient	Age	Age _{MMN}	Age _{Zscore}	Gender
A	50	$\frac{50 - 10}{50} = 0.8$	$\frac{50 - 45}{15} = 0.33$	Male
B	20	$\frac{20 - 10}{50} = 0.2$	$\frac{20 - 45}{15} = -1.67$	Male
C	50	$\frac{50 - 10}{50} = 0.8$	$\frac{50 - 45}{15} = 0.33$	Female

- a. Fara normalizare
 $d(A,B)=30, d(A,C)=1 \Rightarrow C$ este mai aproape
- b. Cu normalizare min-max
 $d(A,B)=0.6, d(A,C)=1 \Rightarrow B$ este mai aproape
- c. Cu standardizare z-score
 $d(A,B)=2, d(A,C)=1 \Rightarrow C$ este mai aproape

Normalizarea min-max este preferata atunci cand se combina variabile categoriale cu variabile necategoriale

Referinta figura: D. Larose, *Discovering Knowledge in Data. An Introduction to Data Mining*, John Wiley & Sons, 2005.

- Functia de combinare
 - Dupa alegerea celor k vecini, distanta nu mai conteaza – fiecare inregistrare are un vot pentru determinarea clasei careia ii va apartine variabila target
SAU
 - Vecinii care se afla la o distanta mai mica au o pondere mai mare la vot (weighted voting) -> influenta unei anumite inregistrari este invers proportionala cu distanta de la acea inregistrare la cea care trebuie clasificata
- Alegerea valorii lui k
 - Se incearca diferite valori pentru k , pentru cateva seturi de training alese aleator, si se alege acel k care minimizeaza eroarea de clasificare