

MTA Számítástechnikai és Automatizálási Kutató Intézet

Egy számítástechnikai eszköz bonyolult logikai kifejezések leírására orvostatisztikai alkalmazásokban

Ratkó István

Előadásomban egy az ÁSZSZ HwB gépén történő kórházi morbiditásvizsgálat közben felvetődött problémát ismertetek, mivel véleményem szerint alkalmazása egyéb orvostatisztikai feladatoknál is felmerülhet.

A probléma pontos matematikai leírása (1)-ben megtalálható, most csak arra törekszem, hogy egy példán illusztrálva megmutassam, miért érdekes a probléma és hogyan oldható meg.

1. A probléma felvetése

Adott egy fix hosszúságú rekordokból álló adatfile. A szakembereket igen gyakran csak speciális feltételeknek elegettevő adatok érdeklik. Ezek a feltételek logikai kifejezések formájában írhatók fel. A feltételekkel kapcsolatban a következőket tudjuk: a.) sok van belőlük, b.) nem rögzíthető előre minden lehetséges feltétel, amire szükség lesz. A programozási nyelvek szokásos eszközei nem biztosítanak olyan lehetőséget, amellyel ez a két követelményt kielégítő utasítás felírható. Előadásomban mutatok egy módszert, arra a nem lényegtelen szempontokra is ügyelve, hogy a feltételvizsgálat ideje minimális legyen.

2. Egy példa

Az adatfile minden rekordja 40 karaktert tartalmaz. A rekordban a következő adatelemek találhatóak:

<u>A rekordelem jelentése</u>	<u>Neve</u>	<u>Hány karakterből áll</u>
Beteg keresztnévnek kezdőbetűje	BET	2
Anyja keresztnévnek kezdőbetűje	ANY	2
Neme	NEM	1
Születési dátum	SZD	6
Születési hely megyéje	SZH	2
Lakóhely megyéje	LAK	2
Népgazdasági ágazat	NEP	2
Foglalkozási viszony	FOG	1
Kórházkód	KOR	4
Osztálykód	OSZ	2
Ápolási napok száma	NAP	4
Beutaló korforma	BKF	4
Ápolási indokló fő kórisme	AKF	4
Halál oka	HAL	4

Az adatfile több tizezer rekordot tartalmaz. Tételezzük fel, hogy azokra a betegekre akarunk bizonyos feldolgozást végrehajtani, akikre a következők teljesülnek:

- a.) 100 kiválasztott betegséggel ápolat Heves-megyei férfiak, vagy
- b.) 3032 (alkoholizmus) kórformájú betegségben meghalt Szabolcs-megyei vagy budapesti betegek, vagy
- c.) 20 kiválasztott betegséggel, 10 adott megyéből való, legalább 100 napig ápolat betegek, vagy
- d.) 1,5,7,8,9 foglalkozási viszonyu, 40 kiválasztott beutaló kórformájú betegek.

A felirandó logikai kifejezés akkor és csak akkor legyen igaz, ha a rekordot fel akarjuk dolgozni.

Megjegyezzük, hogy a valóságban bonyolultabb és egyszerűbb esetek is előfordulnak.

### 3. A rekordok kiválasztásának módszere

A logikai kifejezés - akár FORTRAN, akár COBOL nyelven írjuk a programot - szerkezetileg így néz ki:

(LAK=10 és (AKF=12 vagy AKF=27 vagy...vagy AKF=8501) és NEM=1)  
vagy (HAL=3032 és (LAK=1 vagy LAK=15)) vagy  
(NAP 100) és (AKF=29 vagy AKF=32 vagy...vagy AKF=9000) és  
(LAK=5 vagy LAK=8 vagy...vagy LAK=20)) vagy  
((FOG=1 vagy FOG=5 vagy...vagy FOG=9) és (BKF=7 vagy...BKF=8001))

Ennek a kifejezésnek 180 tagja van. Esetleg akad vállalkozó szellemű programozó, aki beírná a programba a megfelelő utasítást, de ez semmiképpen sem tekinthető ideális megoldásnak, mert

- a.) sok hibalehetőséget rejt magában
- b.) legtöbbször (FORTRAN esetén) nem is fér el még a 20 folytatósorban sem
- c.) a változó igényekhez nem igazodik, azaz, ha a felhasználónak más feldolgozási igénye van, újabb "hosszu" utasítást kell leírnia.

Mit lehet tenni a nehézségek áthidalására?

Vegyük észre, hogy a lakóhely megyéjére és az ápolást indokló fő kórismére több viszonylatban is szükség van. Új változónevek (tömbök) bevezetésével rövidíthetjük a kifejezés hosszát. Ezek a tömbök a maximális sorszámmal feltüntetve a következők:

ZSLAK1(22), ZSLAK2(22), ZSLAK3(22)  
ZSAKF1(9999), ZSAKF2(9999)  
ZSHAL1(9999), ZSNAP1(9999), ZSNEM1(2), ZSFOG1(99)  
ZSBKF1(9999)

A tömb elemeit az alábbi módon töltjük fel:

ZSLAK1(10)=1,  
ZSAKF1(12)= ZSAKF1 27 =...= ZSAKF1 8501 = 1,  
ZSNEM1(1) = 1, ZSNEM1(2) = 0,  
ZSHAL1(3032) = 1,

ZSLAK2(1) = ZSLAK2(15) = 1,  
ZSNAP1(100) = ZSNAP1(101) = ... = ZSNAP1(9999) = 1,  
ZSAKF2(29) = ZSAKF2(32) = ... = ZSAKF2(9000) = 1,  
ZSLAK3(5) = ZSLAK3(8) = ... = ZSLAK3(20) = 1,  
ZSFOG1(1) = ZSFOG1(5) = ... = ZSFOG1(9) = 1,  
ZSBKF1(7) = ... = ZSBKF1(8001) = 1,

mindegyik több összes többi eleme legyen 0. Ezen változók segítségével a rekord sorsa a következőképpen dönthető el:

1. Beolvasás
2. Ha ZSLAK1(LAK)=1 és ZSAKF1(AKF)=1 és ZSNEM1(NEM)=1  
vagy ZSHAL1(HAL)=1 és ZSLAK2(LAK)=1 vagy  
ZSNAP1(NAP)=1 és ZSAKF2(AKF)=1 és ZSLAK3(LAK)=1  
vagy ZSFOG1(FOG)=1 és ZSBKF1(BKF)=1 menj 4.-re
3. Menj 1.-re, azaz ne dolgozd fel a rekordot.
4. Dolgozd fel a rekordot, majd menj újra 1.-re

Látjuk, hogy ezzel az egyszerű fogással a logikai kifejezés felírása egyszerűsödött. A kifejezés ilyen alakú: (...és...és...) vagy (...és...és...) vagy...vagy (...és...és...), un. diszjunktív normálformában van felírva. A fenti eljárás 2. pontjában szereplő utasítás a következő utasításokra bontható:

- 2.1.a. Ha ZSLAK1(LAK)=1, menj 2.1.b-re, ha nem, 2.2.a.-ra
- 2.1.b. Ha ZSAKF1(AKF)=1, menj 2.1.c-re, ha nem, 2.2.a.-ra
- 2.1.c. Ha ZSNEM1(NEM)=1, menj 4-re, ha nem, 2.2.a.-ra
- 2.2.a. Ha ZSHAL1(HAL)=1, menj 2.2.b-re, ha nem, 2.3.a.-ra
- 2.2.b. Ha ZSLAK2(LAK)=1, menj 4-re, ha nem, 2.3.a.-ra
- 2.3.a. Ha ZSNAP1(NAP)=1, menj 2.3.b-re, ha nem, 2.4.a.-ra
- 2.3.b. Ha ZSAKF2(AKF)=1, menj 2.3.c-re, ha nem, 2.4.a.-ra
- 2.3.c. Ha ZSLAK3(LAK)=1, menj 4-re, ha nem, 2.4.a.-ra
- 2.4.a. Ha ZSFOG1(FOG)=1, menj 2.4.b-re, ha nem, 1-re
- 2.4.b. Ha ZSBKF1(BKF)=1, menj 4-re, ha nem 1-re.

A feldolgozandó rekordok kiválasztását a most leírt módon végezzük. Mi indokolja ennek 2.-vel szembeni használatát?

#### 4. Optimalizációs kérdések

Azzal a megjegyzéssel kezdjük, hogy a logikai kifejezés vizsgálatakor a 2. lépésben mindegyik tag kiértékelődik, míg a 2.1.a., ..., 2.4.b. sorozatban a kiértékelés esetleg hamarabb is befejeződhet. Így nyilván a 2. lépésben szereplő utasítással szemben, a 2.1.a., ..., 2.4.b. utasítássorozatot célszerű használni, ne felejtsük el azt a tényt se, hogy ezt a logikai vizsgálatot több tizezerszer kell elvégezni. Nem mindegy az sem, hogy a 2.1., 2.2., 2.3., ill. azokon belül az a, b, c, utasítások milyen sorrendben követik egymást. Két sorrend közül azt mondjuk jobbnak, amelyiknél a kiértékelés várható értéke, azaz a megvizsgálandó tagok várható száma a kisebb. Az ilyen értelemben legjobb sorrend kialakításához ismerni kell az egyes elemi ítéletek igazságának valószínűségeit.

Egy adott csoporton belül melyik alcsoportot kell előbbrevenni? Amelyik igazságának nagyobb a valószínűsége? Ez szemléletesen így érezhető, de nem így van. A kimondandó 1. tétel bizonyítását illetően (1)-re utalunk.

##### 1. Tétel:

Legyen  $p = P(L_1 \text{ igaz})$ ,  $q = P(L_2 \text{ igaz})$ . Jelölje  $p_1$  annak valószínűségét, hogy az  $(\dots \wedge L_1 \wedge L_2 \wedge \dots)$  diszjunkcióban  $L_1$  előtt minden tag igaz és ezen diszjunkció előtti diszjunkcióban legalább egy konjunkció hamis,  $p_2$  ugyanennek a valószínűségét, csak az  $(\dots \wedge L_2 \wedge L_1 \wedge \dots)$  diszjunkcióra vonatkoztatva. Igaz a következő:

Az  $\dots \vee (\dots \wedge L_1 \wedge L_2 \wedge \dots) \vee \dots$  kifejezés kiértékelése akkor és csak akkor fejeződik be hamarabb, mint a  $\dots \vee (\dots \wedge L_2 \wedge L_1 \wedge \dots) \vee \dots$  kifejezésé, ha

$$p(1-q)p_1 < (1-p)qp_2$$

A csoportok egymás közötti sorrendjére már nem kapunk ilyen szemléletes eredményt. Legyen két egymás melletti csoport:  $L'$  és  $L''$ , vagyis a kifejezés így néz ki:

$$\dots V \dots VL'VL''V \dots \quad /1/$$

Jelölje  $a$  ill.  $b$   $L'$  ill.  $L''$  tagjainak számát. Szeretnénk /1/-et összehasonlítani az

$$\dots V \dots VL''VL'V \dots \quad /2/$$

kifejezéssel. 1-nél jelölje  $p_1$  annak valószínűségét, hogy  $L'$  minden tagja igaz,  $L''$ -ben és az  $L'$  előtti tagok mindegyikében van hamis,  $L''$ -ben az első hamis tag legyen az  $\eta$ -edik,  $p_2$  annak valószínűségét, hogy  $L'$  és  $L''$  minden tagja igaz és  $L'$  előtti tagok mindegyikében van hamis és végül /2/-nél  $p_3$  annak valószínűségét, hogy  $L''$  minden tagja igaz,  $L'$ -ben és  $L''$  előtti tagok mindegyikében van hamis,  $L'$ -ben az első hamis tag legyen  $\xi$ -edik. Ekkor bizonyítható, a következő /ld. (1)/:

## 2. Tétel:

$\dots VL'VL''V \dots$  kiértékelése akkor és csak akkor fejeződik be hamarabb, mint  $\dots VL''VL'V \dots$  kiértékelése, ha:

$$ap_2 + p_3 E \xi < p_1 E \eta + bp_2$$

## 5. Megjegyzések

Ebben a pontban röviden leírjuk a meglévő program lehetőségeit. A lehetőségek pontos ismertetése nem volt célja előadásomnak.

a.) A felhozott példa olyan, hogy az új változónevek bevezetése után a logikai kifejezés

$$(\dots \wedge \dots) \vee (\dots \wedge \dots) \vee \dots \vee (\dots \wedge \dots)$$

alakú lett. Ismeretes, hogy bármilyen logikai kifejezés ilyen alakra hozható. /ld. (2)/. Erre egyszerű módszerek vannak, nem kell hozzá ismerni a Boole-algebra azonosságait.

b.) A konkrét felhasználásban a felhasználó három lehetőség közül választhatott:

1. A feltételt ő írja fel.
2. Ugynevezett szerkesztőprogrammal iratja fel - paraméterek megadásával - a diszjunktív normálformának megfelelő utasításokat.
3. Mint 2., csak konjunktív normálformát használ.

c.) Az új változónevek megadásához a felhasználónak meg kell adnia, hogy a tömbök mely elemei veszik fel az 1 értéket. Ez valahogy így történhet:

12,27,49-92,101

A felhasználónak arra is lehetősége van, hogy a komplementer hal-  
mazzt adja meg.

d.) A konkrét feldolgozásban a felhasználó megtehetette, hogy a rekordnak csak egy részét dolgozza fel.

e.) A Szerkesztő program biztosítja a felhasználó számára azt, hogy tetszőlegesen megválaszthatja logikai kifejezéseit.

f.) Kórházi morbiditási vizsgálattal, s egyéb orvostatisztikai feldolgozással kapcsolatos alaptanulmányként az irodalom (3)-(14) hivatkozásait ajánlhatjuk. Az előadásban elmondott téma leírása (1)-en kívül több-kevesebb részletességgel (4)-ben (7)-ben, (11)-ben (12)-ben és (14)-ben megtalálható.

### Irodalom

- (1) Ratkó István: Bonyolult logikai kifejezések kiértékelésének számítástechnikai és optimalizálási problémái, (MTA SZTAKI, Közlemények, megjelenőben.
- (2) Ruzsa I., Urbán J.: Matematikai logika, Tankönyvkiadó, 1966.
- (3) Greff L., Krámlí A., Ruda M.: Kórházi morbiditási vizsgálattal kapcsolatos statisztikai és számítástechnikai megfontolások, 4. Neumann Kollokvium, Szeged, 1973.

- (4) Csukás A-né, Greff L., Krámlí A., Ruda M.: A kórházi morbiditási vizsgálat számítógépes feldolgozásának tapasztalatai és továbbfejlesztése, 5. Neumann Kollokvium, Szeged, 1974.
- (5) Csukás A-né, Greff L., Krámlí A., Ruda M.: An approach to the hospital morbidity data system development in Hungary. Symposium on Medical Data Processing, Toulouse, 1975.
- (6) Csukás A-né, Greff L., Krámlí A., Ruda M.: Lekérdező rendszer kórházi morbiditási vizsgálat anyagára, 6. Neumann Kollokvium, Szeged, 1975.
- (7) Garádi J., Krámlí A., Ratkó I., Ruda M.: Statisztikai és számítástechnikai módszerek alkalmazása kórházi morbiditási vizsgálatokban, MTA SZTAKI Tanulmányok, 1975.
- (8) Krámlí A., Ruda M.: Izpravocno-informacionnaja szisztyema zaproszov bojnycsnovo morbigyityizma, Sztruktura i organyizacija paketov programm, Nemzetközi Konferencia, Tbiliszi, 1976.
- (9) Krámlí A., Ratkó I., Ruda M., Soltész J.: A statisztikai adatfeldolgozás matematikai és számítástechnikai problémái, MTA SZTAKI Tanulmányok, (megjelenőben).
- (10) Ruda M.: Egy általános információs rendszer kórházi morbiditási adatok feldolgozására, 8. Neumann Kollokvium, Szeged, 1977.
- (11) Matematikai statisztikai és számítástechnikai megfontolások a kórházi morbiditási adatok vizsgálatával kapcsolatban, MTA SZTAKI, esettanulmány, 1977.
- (12) Kórházi morbiditási adatok feldolgozását szolgáló statisztikai programrendszer, MTA SZTAKI, témadokumentáció, 1977.
- (13) Csukás A-né, Greff L.: Az 1972-73. évi Kórházi megbetegedési reprezentatív vizsgálat, Statisztikai Szemle, 52., 11-12 szám.
- (14) Ruda M.: Egy széles körben alkalmazható programprimálizálási módszer, MTA SZTAKI, Közlemények, (megjelenőben).