

Припрема података

Ненад Митић

Математички факултет
nenad@matf.bg.ac.rs

Увод

- Различити извори и формати података
- Недостајући и неконзистентни подаци
- Неопходна је припрема (препроцесирање)
 - Издвајање карактеристика
 - Преносивост типова података
 - Чишћење података
 - Избор и трансформација
 - Редукција података

Преносивост типова података

- Неке карактеристике онемогућају примену готових алата
- Поједини алгоритми раде само са одређеним типовима података (промена типа?)
- Могуће губљење информација
- Најчешћа трансформација (категоричких) у нумеричке податке

Дискретизација

Трансформација непрекидних у категоричке атрибуте

- Обично се примењује на атрибуте у класификацији или правилима придруживања
- Кораци у трансформацији
 - одабрати број категорија n
 - интервал бројева се дели на n подинтервала
 - све вредности из једног подинтервала се пресликавају у исту категоричку вредност
- Између добијених вредности (ознака) не постоји уређење (категоричке вредности!)

Текстуални у нумеричке податке

- Представљање текстуалних података преко ретких нумеричких вектора није погодно за највећи број ИП метода
- Ограничен број мера (нпр. косинусна мера, али не и Еуклидско растојање)
- Латентна семантичка анализа (LSA) - текст се преводи у не-ретку репрезентацију мање димензије
- После трансформације документ $\bar{X} = (x_1, x_2, \dots, x_d)$ се скалира функцијом $\frac{1}{\sqrt{\sum_{i=1}^d x_i^2}}(x_1, x_2, \dots, x_d)$
- На овако добијене податке може да се примени Еуклидско растојање
- У пракси - ИП алгоритми се примењују директно на податке добијене са LSA док се даља трансформација не ради

Подаци из временских серија у дискретне ниске

Конверзија се врши *SAX* алгоритмом (симболичка апроксимација агрегата, енг. *symbolic aggregate approximation*)

- 1 У првом кораку се серија дели у прозоре величине w за које се рачуна просечна вредност атрибута (временске серије) за коју је прозор одређен
- 2 У другом кораку се средње вредности временских серија дискретизују помоћу технике са интервалима који имају исти број елемената

Дискретне ниске у нумеричке податке

Трансформација се врши у два корака

- 1 Дискретне ниске се конвертују у скуп бинарних временских серија чији је број једнак броју различитих симбола
- 2 Свака серија се конвертује у мултидимензиони вектор помоћу трансформације таласићима. Особине из ових вектора се комбинују и формира се мултидимензионални слог

Пример: ДНК секвенца

```
ACACACTGTGACTG
10101000001000
01010100000100
00000010100010
00000001010001
```

Чишћење података

Аспекти

- 1 Рад са недостајућим подацима
- 2 Рад са некоректним подацима
- 3 Рад са дуплираним подацима
- 4 Скалирање и нормализација

Рад са недостајућим подацима

Разлози за појаву

- Информације нису прикупљене (нпр. људи одбијају да прикажу своју тежину, старост, величину плате, ...)
- Атрибути нису применљиви у свим случајевима (нпр. плата није применљива на децу)
- Шта радити у таквим случајевима?

Руковање недостајућим вредностима

- 1 Комплетни слогови (цео објекат) који садрже такав податак се бришу
- 2 Недостајућа вредност се процењује и уноси (импутација)
- 3 Неки алгоритми могу да обрађују слогове/атрибуте са недостајућим подацима
- 4 Замена могућим вредностима (зависи од алгоритма)

Рад са некоректним подацима

Аспекти

- 1 Откривање некозистентности (нпр. подаци из више извора који се односе на исту ствар су различити)
- 2 Доменско знање
- 3 Метода оријентисана ка подацима

Рад са дуплираним подацима

Најчешће се јављају код спајања података из хетерогених извора

- Пример: иста особа са више електронских адреса
- Најчешће се елиминишу из материјала
- Када дуплиране податке не треба брисати?

Скалирање и нормализација

- Потреба за нормализацијом - више атрибута који су различито скалирани
- Стандардизација: нека j -ти атрибут има средњу вредност μ_j и стандардну девијацију σ_j . Тада се вредност x_i^j j -тог атрибута слога \bar{X}_i нормализује применом изрази $z_i^j = \frac{x_i^j - \mu_j}{\sigma_j}$
- За нормалну расподелу добијене вредности најчешће се налазе у интервалу $[-3, 3]$
- За свођење у интервал $[0, 1]$ се примељује *min-maxs* скалирање $y_i^j = \frac{x_i^j - \min_j}{\max_j - \min_j}$
- Пример - *Transform, Derive, Filler* чворови у СПСС Моделеру

Редукција и трансформација података

Мања количина података - ефикаснија примена алгоритма

- 1 Агрегација
- 2 Узимање узорака
- 3 Избор карактеристика
- 4 Редукција помоћу ротације оса
- 5 Остале методе димензионе редиукције

Агрегација

Комбиновање два или више атрибута (или објекта) у један

Сврха

- 1 Редукција података (смањивање броја атрибута/објеката)
- 2 Промена скале (нпр. уместо 365 дана добија се 12 месеци)
- 3 'Стабилнији' подаци (агрегирани подаци имају тенденцију ка мањим одступањима)
- 4 ...

Узимање узорака

- Избор узорака је главна техника која се користи у истраживању података
- Често се користи како за прелиминарна истраживања тако и за коначне резултате анализе података
- Статистичари бирају узорке јер је добијање комплетног скупа података који су од интереса јако скупо и временски захтевно
- Избор узорака се користи у ИП јер је обрада комплетног скупа података који је од интереса такође јако скупа или временски захтевна

Узимање узорака

Кључни принципи за ефективан избор узорака су

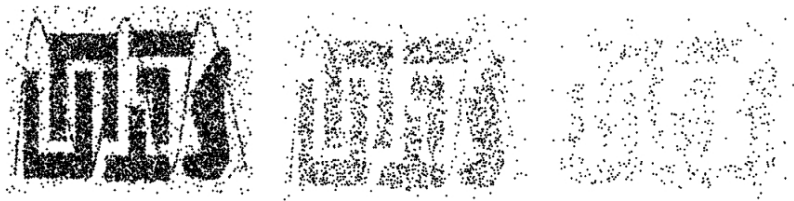
- Коришћењем узорака који су репрезентативни добија се ефекат скоро исти као да је рађено на комплетном скупу података
- Узорак је репрезентативан ако има апроксимативно исте особине као и оригинални скуп података

Типови узорака

- Једноставан случајни узорак (једнака вероватноћа за избор било које случајне ставке)
- Са и без враћања (дупликата из оригиналног скупа)
- Пристрасно узорковање (неки подаци су важнији од других)
- Стратификовано узорковање (узорковање са раслојавањем)
 - Подаци се деле у више делова, а затим се бира случајни узорак из сваког од тих делова

Величина узорка

Величина узорка треба да буде довољно велика да се не наруши структура објекта или уклоне интересанте особине



Величине узорка су редом 8000, 2000 и 500 тачака

Избор карактеристика

- 1 Један од начина за смањење димензионалности
- 2 Елиминација редундантних карактеристика
- 3 Елиминација ирелевантних карактеристика
- 4 Развијен је велики број техника, поготову за класификацију
- 5 Често се формирају нови атрибути који укључују важне карактеристике због ефикасније обраде
- 6 Пресликавање у нови простор (нпр. Фуријеова анализа, таласићи)

Редукција помоћу ротације оса

- Аутоматско уклањање координатних оса помоћу ротације?
- *PCA* (Principal Component Analysis)
- *SVD* (Singular Value Decomposition)

Principal Component Analysis

- Смањење броја димензија података
- Налажење образаца у подацима велике димензионалности
- Визуелизација података велике димензионалности

Principal Component Analysis (наставак)

- Основна идеја: ротација података у систем са осама где је највећи број варијанси покривен најмањим бројем димензија
- Нови систем са осама зависи од корелације између атрибута
- *PCA* се (најчешће) примењује после одузимања средње вредности од сваке тачке

Principal Component Analysis (наставак)

- За матрицу података D реда $m \times n$ може да се формира матрица коваријанси C са елементима $c_{ij} = cov(d_{*i}, d_{*j})$ (c_{ij} је коваријанса i -те и j -те колоне података)
- Коваријанса је мера како се атрибути мењају у пару. Ако је $i = j$ тада је коваријанса једнака варијанси атрибута
- Ако се матрица D претходно припреми тако да је средња вредност сваког од атрибута једнака 0, тада је $C = D^T D$

Principal Component Analysis (наставак)

Циљ *PCA* је налажење трансформације података за коју важи

- 1 Сваки пар новодобијених атрибута има коваријансу 0
- 2 Атрибути су уређени (у опадајућем редоследу) у односу на величину варијансе која је покривена од стране атрибута
- 3 Захтева се ортогоналност између атрибута, тако да сваки наредни атрибут покрива што је могуће већи број преосталих варијанси

Principal Component Analysis (наставак)

Трансформација се врши употребом сопствених вредности матрице коваријанси. Нека су

- 1 λ_i (ненегативне) сопствене вредности од C уређене у редоследу $\lambda_1 \geq \lambda_2 \geq \dots \lambda_{m-1} \geq \lambda_m$
- 2 $U = [u_1, \dots, u_n]$ матрица сопствених вектора од C уређена тако да i -ти вектор одговара i -тој највећој сопственој вредности
- 3 Нека је матрица D претходно припремљена тако да је средња вредност сваког од атрибута једнака 0

