OŠETŘENÍ CHYBĚJÍCÍCH HODNOT V LISP-MINER

Specifikace implementace SW

Předmět

4IZ460 – Pokročilé přístupy k DZD

Daniil Bladyko, Michael Drdlíček

xblad11@vse.cz, qdrdm00@vse.cz

CÍL SOFTWARE

Umožnit uživateli získat kompletní soubor dat z tabulky, která původně obsahovala chybějící hodnoty (prázdné buňky).

ZDŮVODNĚNÍ PŘÍNOSU

Databáze dat, kterou uživatel dostane na výstupu, může být použita pro další analýzu (pokud analytik chce aplikovat metody DZD, které nemohou fungovat efektivně při existence chybějících hodnot).

Zároveň, výstup může být užitečný také v případě, kdy naléhavě potřebujeme mít kompletní data, ale není vyžadováno, aby všechny hodnoty byly na 100% pravdivé[1].

POPIS ŘEŠENÉHO PROBLÉMU

Náš tým navrhne software, který bude hledat chybějící hodnoty v databázi a nabízet možné varianty jejich doplnění. Je to obecnější problém, který se může objevit v jakémkoliv souboru dat.

VSTUPY

Databáze s chybějícími hodnotami (soubor CSV), nastavení (např. soubor JSON). Analytikovi bude k dispozici výběr algoritmu pro doplnění chybějících hodnot a popř. i jeho parametrů.

VÝSTUPY

Databáze bez chybějících hodnot (soubor CSV), zpětně použitelný jako soubor externích dat pro aplikace technik DZD.

PŘEDPOKLÁDANÉ ALGORITMY

Jakýkoliv algoritmus začne následovně:

- Importujeme data z CSV souboru,
- Vybíráme sloupce s chybějícími hodnotami, získáváme informace o počtu těchto hodnot, řadíme podle počtu a ukládáme do pole,
- Postupně aplikujeme níže popsané algoritmy na všechny tyto sloupce.

Jednodušší algoritmy (Metody imputace nezaložené na modelu)

První algoritmus spočívá v dosazení reprezentativní hodnoty do prázdných buněk v daném sloupci, např. nepodmíněný průměr, medián anebo modus (pro kategoriální proměnné)[2]:

- Zjišťujeme typ proměnné,
- Počítáme průměr, medián anebo modus (podle nastavení a podle typu proměnné),
- Dosazujeme do všech prázdných buněk v tabulce.

Pro spojité proměnné můžeme alternativně použit Buckovu metodu a dosadit podmíněný průměr. Budeme postupovat podle následujícího algoritmu:

---

[1] Např. máme tabulku s 50% podílem chybějících hodnot ve sloupci „Pohlaví“. Management od nás vyžaduje, abychom vypočítali relativní počet žen mezi klienty. Půlka hodnot pravděpodobně nebude postačující pro odvození situace v základním souboru hodnot v tomto sloupci, takže je vhodné použit techniky DZD pro vyplnění mezer.

[2] Při větším počtu chybějících hodnot může docházet ke zkreslení a k deformaci odhadu parametru rozdělení hodnot.

- Volíme sloupec s nejvyšším počtem chybějících hodnot,
- Provádíme dechotomizaci sloupců kategoriálního charakteru (jen sloupce bez chybějících hodnot),
- Počítáme hodnoty regresních koeficientů,
- Pomocí lineární regrese dosazujeme hodnoty do prázdných buněk,
- Opakujeme postup pro další sloupec se spojitými proměnnými.

Sofistikovanější algoritmy (Metody imputace založené na modelu)

Shluková analýza

Pomocí standardních nástrojů LISp-Miner vytvoříme shluky, podle metody k-NN, pak na základě přiřazení do konkrétního shluku, doplňujeme chybějící hodnotu. V případě, že pozorování bude patřit do více shluků nebo bude ležet na hranici, náhodně přiřadíme do shluku s pravděpodobnosti, která je rovna relativnímu počtu sousedů v každém shluku vzhledem k sumárnímu počtu pozorování v obou shlucích[3].

Regresní analýza

Druhý algoritmus, který jsme zařadili mezi jednodušší, je aplikace regresní analýzy. Pravděpodobně by ovšem mohl vést ke snížení volatility hodnot ve sloupci, takže bychom ho mohli modifikovat. Po aplikaci algoritmu a před dosazením hodnot můžeme spočítat rezidua, jejich průměr a rozptyl, a následně přičítat k výstupu lineární regrese náhodnou složku s normálním rozdělením, s průměrem a směrodatnou odchylkou jako u reziduí.

Rozhodovací strom

Tento algoritmus budeme aplikovat jenom na kategoriální data. Pomocí standardních nástrojů LISp-Miner najdeme hypotézu s nejvyšší kvalitou, která je vyšší, než požadována mez (např. 0.6). Pak aplikujeme tuto hypotézu na pozorování s chybějícími hodnotami a zjistíme, jaká hodnota se hodí nejvíce.

Naivní Bayes

Tento algoritmus zajišťuje stabilní výsledky pří různém počtu pozorování, což může být v praxi důležitým faktorem. Metoda je založena na analýze vztahů mezi každou nezávisle proměnnou a závisle proměnnou s cílem stanovit podmíněnou pravděpodobnost pro každý vztah. Stačí nám pouze jeden průchod daty.

POUŽITÉ KNIHOVNY A PŘEVZATÉ KÓDY

Knihovny:

1. lm
2. lm.data
3. lm.metabase
4. lm.prepro
5. lm.task
6. lm.task.results
7. lm.task.settings
8. popř. další

Převzaté kódy:

1. DEMO kódy ve složce Exec (včetné EverMinerSimple).
2. Popř. části kódu z ukázek studentských prací.

[3] Např. v prvním shluku 60 entit, ve druhém 40, takže hraniční pozorování přiřadíme do prvního shluku s pravděpodobnosti 0.6 a do druhého s pravděpodobnosti 0.4

OČEKÁVANÁ OMEZENÍ NAVRŽENÉHO ŘEŠENÍ

1. Samozřejmě je zbytečné algoritmy aplikovat na kompletní tabulky.
2. Pravděpodobně budou nalezené vztahy spíše vypovídat o způsobu doplnění, než o samotných datech (pokud na výstupních datech budou aplikovány metody DZD).