Katedra Informatiky
Fakulta Matematiky, Fyziky a Informatiky
Univerzita Komenského, Bratislava

Datamining, princípy a metódy

(Bakalárska práca)

Marek Mardiak

9.2.1 Informatika

Vedúci: Mgr. Tibor Hegedüs

Bratislava, 2008

Datamining, princípy a metódy

Bakalárska práca

Marek Mardiak

UNIVERZITA KOMENSKÉHO, BRATISLAVA
FAKULTA MATEMATIKY, FYZIKY a INFORMATIKY
KATEDRA INFORMATIKY

Študijný odbor: 9.2.1 Informatika

Vedúci záverečnej práce
Mgr. Tibor Hegedüs

Bratislava, 2008

Abstrakt

Autor:	Marek Mardiak
Názov bakalárskej práce:	Datamining, princípy a metódy
Škola:	Univerzita Komenského v Bratislave
Fakulta:	Fakulta matematiky, fyziky a informatiky
Katedra:	Katedra informatiky
Vedúci bakalárskej práce:	Mgr. Tibor Hegedüs
Rozsah práce:	43 strán

Bratislava, jún 2008

Bakalárska práca popisuje základné metódy a techniky dolovania dát(Dataminingu). Poskytuje prehľad procesu dolovania dát, od motivácie vedúcej k rozvoju tejto multidisciplinárnej oblasti, stručného popisu najpoužívanejších metód ku detailnejšiemu popisu algoritmov a princípov segmentácie a klasifikácie.

Formou experimentu na reálnych dátach následne približuje metódu klasifikácie v praxi, od predprípravy dát, transformácie dát až ku zhodnoteniu úspešnosti klasifikácie, kde porovnáva vlastnosti, výsledky a parametre dvoch metód pri nej použitej.

Kľúčové slová: datamining, objavovanie znalostí, dolovanie z dát, klasifikácia

Čestne prehlasujem, že som túto bakalársku prácu vypracoval samostatne s použitím citovaných zdrojov.

. . . . . .

Poďakovanie

Ďakujem môjmu školiteľovi Mgr. Tiborovi Hegedüsovi za cenné rady a pripomienky pri tvorbe tejto práce.

Predhovor

Informácie a znalosti, získané a použité v správny čas sú dnes veľmi cenné. Vysoký stupeň informatizácie skoro všetkých oblastí nášho života prináša so sebou obrovské množstvá dát, ktoré často obsahujú informácie a znalosti vyplývajúce z reálnych procesov a vzťahov, avšak nie explicitne uložené v dátových štruktúrach.

Datamining ako oblasť informatiky sa zaoberá práve extrahovaním týchto znalostí a informácií. Keďže ide o oblasť často neznámu študentovi pregraduálneho štúdia, motiváciou k tvorbe tejto práce bolo vytvoriť dielo, poskytujúce základný rámcový prehľad danej problematiky, spolu s detailnejším popisom najpoužívanejších metód a ich experimentálnym porovnaním, ktoré umožní čitateľovi sa zorientovať v danej problematike a vhodne voliť ďalšie materiály k štúdiu.

Obsah

Zoznam obrázkov

Zoznam tabuliek

Kapitola 1

Úvod

Dnešný svet sa mení veľmi rýchlo. Pokrok v oblasti výpočtovej techniky, telekomunikácií, globalizácia obchodu a ekonomík tomu len napomáhajú. Sú potrebné rýchle, a pritom strategické a rozumné rozhodnutia v každej oblasti nášho života, či už ide o hospodárstvo, vedu a výskum alebo súkromný sektor. V kontexte týchto rozhodnutí hrájú kľúčovú rolu správne informácie a znalosti ako ich efektívne použiť.

Datamining, alebo tiež dolovanie z dát, objavovanie znalostí, je multidisciplinárna oblasť informatiky, využívajúca poznatky zo štatistiky, umelej inteligencie a strojového učenia, teórie informácie a databáz. Vysoký stupeň informatizácie spoločnosti dnes poskytuje také enormné množstvo údajov o každodennej stránke života, že nie je v ľudských silách analyzovať možné vzťahy a štruktúry v týchto údajoch.

Datamining je teda proces objavovania informácií a vzťahov medzi údajmi, ktoré nie sú explicitne v dátových štruktúrach uložené, ale vyplývajú z podstaty reflektovania reálnych procesov v prírode, spoločnosti.

Cieľom tejto práce je podať základný pohľad na datamining, objasniť ho ako proces s jeho jednotlivými krokmi a podať prehľad metód pri ňom používaných. Práca neskôr podrobnejšie rozoberá základné metódy klasifikácie a segmentácie, oboznamuje s ich teoretickým pozadím, algoritmami a vhodnými prípadmi použitia.

V závere práce sa čitateľovi cez experiment na reálnych dátach(predikciu magnitúdy zemetrasenia na základe jeho iných nameraných atribútov) podá pohľad na klasifikáciu ako metódu dataminingu v praxi. Experimentálne sa tu porovnajú dva klasifikačné algoritmy. Od procesu predprípravy dát, ich nutných transformácií na vhodný tvar pre skúmané klasifikačné algoritmy

práca postupne prejde k analýze úspešnosti/neúspešnosti predikcie pri jednotlivých algoritmoch a porovná ich výsledky vzhľadom na ich parametre a skúmané dáta.

Kapitola 2

Datamining ako proces, motivácia, využitie

2.1 Pojem Datamining

Pojem Datamining(Dolovanie z dát) sa dá preložiť ako *extrahovanie alebo dolovanie znalostí, informácií z veľkého množstva dát.* Rozmach tejto oblasti priamo súvisí s rýchlym vývojom informačných technológií, vývojom databázových systémov, ich zavádzaním do každodenného života, získavaním a ukladaním čoraz väčšieho a väčšieho objemu dát, ktoré popisujú skoro celé dianie v bežnom živote. Banky, poisťovne, úrady archivujú a pracujú s obrovskými množstvami údajov, ktoré detailne popisujú ich klientov a ich správanie. Obchodné reťazce ukladajú informácie o nákupoch, mobilní operátori majú k dispozícii detailné informácie o hovoroch svojich klientov. V podstate čoraz väčšia a väčšía časť procesov z bežného života sa denne ukladá v elektronickej forme do databáz.

Datamining teda môže byť chápaný ako prirodzený krok v evolúcii informačných technológií. Najskôr poskytol vývoj schopnosť ukladať dáta do databázových systémov, čo neskôr slúžilo ako podklad pre dátový management(ukladanie a získavanie dát na základe dotazov, transakčné spracovanie dát), tieto systémy, keďže už reflektovali procesy z bežného života(napr transakcie z obchodov), poskytli podklad pre vývoj dátovej analýzy a datamingu.

Dnes, keď máme k dispozícii dáta o skoro každej oblasti života, sa nachádzame v situácii kedy sme síce bohatí na dáta, ale chudobní na znalosti, informácie, ktoré tieto dáta v sebe skrývajú. Ide o informácie, ktoré nie sú v

dátach explicitne uložené, ale vyplývajú z procesov v realite a teda túto realitu dáta v istom zmysle zachycujú. Datamining je objavovanie práve týchto informácií, závislostí, analógií medzi dátami. Tieto sa neskôr snažíme využiť v rozhodovacích procesoch obchodu, výskumu, na predikciu rôznych procesov(burza, poistné podvody).

2.2 Datamining ako proces

Datamining ako proces pozostáva z viacerých krokov:

- **Čistenie dát**, krok kedy sa zo skúmaných dát odstraňujú irelevantné, prípadne chybné vzorky.

- **Integrácia dát**, dáta môžeme získavať z rôznych zdrojov, kde sa môže líšiť ich reprezentácia, tento krok integruje rôzne vstupné formáty dát do jedného kompaktného formátu.

- **Selekcia dát**, kedy zo vzoriek vyberáme dáta najviac relevantné pre náš výskum, tento krok často potrebuje nutnú znalosť domény skúmaných dát.

- **Transformácia dát**, atribúty dát je niekedy potrebné transformovať do podoby akej vyžaduje samotná metóda dolovania, prípadne sa transformácia robí za účelom zjednodušenia nadbytočných detailov(napr agregácia atribútov).

- **Dolovanie**, krok kedy aplikujeme algoritmy dataminingu na získanie znalostí(*patterns*).

- **Ohodnotenie získaných znalostí**. Získané znalosti môžu reflektovať rôznu mieru praktickej použiteľnosti, či už na základe ich aplikovateľnosti do praxe, presnosti predikcie, prípadne interpretovateľnosti človekom.

- **Prezentácia získaných znalostí**, kedy najhodnotnejšie znalosti prezentujeme vo forme, ktorá umožňuje efektívne skúmať a pochopiť čo nám prinášajú a čo nám o skúmanej množine hodnôt hovoria.

2.3 Čo vlastne hľadáme?

Výstupom procesu dataminingu by mali byť znalosti, informácie, nie explicitne uložené v dátach. Tieto môžme reprezentovať rôznymi spôsobmi, na základe cieľa, ktorý chceme dosiahnuť, a metódy ktorú použijeme. V základe rozlišujeme dva prístupy.

- **Deskriptívny**, kedy je cieľom popísať, charakterizovať skúmané dáta, prípadne nájsť medzi nimy závislosti. Výstupom často býva skupina atribútov ktoré sú najviac charakteristické pre skúmanú množinu dát, *asociačné pravidlá*, ktoré popisujú závislosť výskytu hodnôt istých atribútov od iných atribútov, prípadne vizuálna reprezentácia skúmanej množiny a skupín vzoriek dát ktoré majú "podobné" hodnoty atribútov.

- **Prediktívny**, kedy je cieľom na základe známej množiny dát, obsahujúcej kompletné atribúty, predpovedať výskyt hodnôt niektorých atribútov na dátach, ktorým budú tieto atribúty chýbať. Výstupom je najčastejšie tzv. *model*, čo môže byť súbor pravidiel, ktoré na základe hodnôt známych atribútov určujú hodnoty hľadaných atribútov(*klasifikačné pravidlá, rozhodovacie stromy*), prípadne dátová štruktúra naučená robiť predikciu týchto atribútov(*neurónová sieť*).

Kapitola 3

Prehľad metód

V tejto kapitole podáme stručný prehľad metód využívaných v procese dataminingu, ich hlavné použitie a formy výstupu, teda reprezentáciu znalostí získaných počas procesu dolovania z dát.

3.1 Charakterizácia tried

Charakterizácia tried patrí medzi deskriptívnu metódu. Dáta sú často priradené istým skupinám, triedam. Typickým príkladom sú dáta o študentoch univerzity, kde môžme definovať triedy ako študenti bakalárskeho, magisterského, doktorandského typu štúdia. Iným príkladom zadelenia do tried je rozdelenie podľa fakulty, kde študent študuje. Cieľom metódy charakterizácie tried je poskytnúť sumarizovaný, jednotný ale dostatočne presný popis skúmanej triedy vzoriek, na základe atribútov dát, zachytávajúci vlastnosti dát, ktoré sú najtypickejšie pre danú triedu.

Proces charakterizácie má viacero prístupov, jedným z najrozšírenejších je *charakterizácia na základe generalizácie a sumarizácie*. Zjednodušene ide o proces postupného vynechávania atribútov čo najmenej popisujúcich danú triedu(napr. na základe výskytu veľkého množstva rôznych hodnôt týchto atribútov), generalizáciu atribútov na vyššie úrovne abstrakcie(napr. zovšeobecnenie atribútu ulica na atribút mesto, okres) a agregácie numerických atribútov pridružených týmto dátam.

Výstupom tejto metódy bývajú rôzne grafy, vizualizácie(napr. koláčové diagramy), koré prehľadne reprezentujú sumárne charakteristiky jednotlivých tried dát.

Príklad: Trieda študentov FMFI UK môže byť charakterizovaná na základe dát v (v tomto prípade fiktívnom)celouniverzitnom systéme ako ľudia vo veku 19-25 rokov, s druhom strednej školy ako gymnázium(80% z celkového počtu), majúcich študijný priemer 1, 56, pochádzjúcich s Bratislavského kraja(70% z celkového počtu).

3.2 Klasifikácia a predikcia

Klasifikácia a predikcia patria medzi prediktívne metódy. Klasifikácia je proces, kedy priraďujeme skúmaným vzorkám tzv. cieľový atribút(má diskrétne, nominálne hodnoty), na základe analýzy trénovacej množiny vzoriek, ktorá mala tento atribút priradený. Predikciou rozumieme obdobný proces, len cieľový atribút má spojitý charakter hodnôt. Typickým príkladom predikcie je regresná analýza.

Výstupom metód je tzv. model(v prípade klasifikácie), čo je objekt reprezentujúci informácie a znalosti získané z analýzy trénovacej množiny, ktorý nám umožňuje určovať cieľový atribút. V prípade predikcie je výstupom najčastejšie funkcia, predpovedajúca hodnoty skúmaného atribútu.

Klasifikáciou a jej algoritmami sa detailne zaoberáme v kapitole 4.

3.3 Asociačná analýza

Asociačná analýza je proces objavovania *asociačných pravidiel*, vzťahov a závislostí medzi atribútmi a ich hodnotami. Analýza sa robí na výskyte týchto atribútov a ich hodnôt v transakciách.

Klasickou ukážkou je tzv. analýza nákupného košíka, kde nás zaujíma s akou pravdepodobnosťou zákazník nakúpi isté druhy výrobkov, v závislosti od predchádzajúcich nákupov. Získaná informácia typu ak zákazník nakúpil mlieko, s 80% pravdepodobnosťou nakúpi aj chlieb môže viesť k rozhodnutiam o priestorovom rozložení tovaru v obchode, a tým k zvýšeniu zisku.

Výstupom sú asociačné pravidlá majúce najčastejšie formu implikácií $A_1 \bigwedge \ldots \bigwedge A_m \Rightarrow B_1 \bigwedge \ldots \bigwedge B_n$ kde A_i pre $i \in \{1, \ldots, m\}$ a B_j pre $j \in \{1, \ldots, n\}$ sú dvojice atribút-hodnota.

Príklad: vek(X,"20-29") $\bigwedge$ príjem(X,"30000-40000") $\Rightarrow$ nakúpil(X,"LCD monitor") [výskyt(support)=2%,dôvera(confidence)=50%]

Interpretujeme nasledovne: 2% analyzovaných vzoriek(výskyt) spĺňajú

dané pravidlo, teda 2% všetkých vzoriek sú zákazníci s príjmom v intervale 30-40 tisíc vo veku 20-29 rokov, ktorý nakúpili LCD monitor. Dôvera 50% hovorí, že 50% vzoriek spĺňajúcich hlavu pravidla, teda zákazníci s príjmom v intervale 30-40 tisíc vo veku 20-29 rokov, spĺňa aj dôsledok, teda že nakúpili LCD monitor.

3.4 Segmentácia

Metóda segmentácie sa používa pri hľadaní tried, skupín objektov s podobnými vlastnosťami. Pri klasifikácii sme v trénovacej množine mali k dispozícii cieľový atribút, teda rozdelenie do tried bolo určené vopred. Pri segmentácii sa naopak snažíme tieto triedy nájsť často bez informácie koľko ich má byť, a aké majú mať objekty patriace do nich vlastnosti.

Proces segmentácie pracuje na princípe merania vzdialeností v prietore atribútov skúmaných vzoriek. Každý atribút je jeden rozmer, jednotlivé skúmané vzorky sú body v n-rozmernom priestore(za predpokladu, že každá vzorka v množine súmaných dát má n atribútov). Proces segmentácie sa snaží na týchto bodoch nájsť segmenty, obasti, kde sú body viac pri sebe, čo reprezentuje podobnosť v hodnotách atribútov. Na nájdených segmetoch sa často robia ďalšie analýzy, snažiace sa popísať ich typické vlastnosti(napr charakterizácia tried).

Výstup procesu segmentácie môže mať rôzne formy, či už grafickú reprezentáciu segmentov, popis atribútov segmentov(napríklad jeho centra v priestore atribútov) a samozrejme je výstupom zadelenie skúmaných vzoriek do segmentov, pre potreby ďalšej analýzy.

Segmentáciou a jej algoritmami sa budeme detailnejšie zaoberať v kapitole 5.

Kapitola 4

Klasifikácia

Databázy, ktoré hlavne v posledných desaťročiach obsahujú obrovské množstvá údajov, týkajúcich sa všetkých stránok nášho života, sú taktiež plné skrytých informácií, ktoré môžu byť využité na plánovanie a inteligentné rozhodnutia v oblastiach obchodu. Klasifikácia je forma dátovej analýzy, ktorá nám môže pomôcť extrahovať modely popisujúce dôležité triedy dát. Klasifikácia na rozdiel od iných predikčných techník(napr regresia) sa používa na predpoveď diskrétnych(nominálnych) hodnôt.

V tejto kapitole popíšeme základné techniky klasifikácie dát ako sú rozhodovacie stromy, bayesovská klasifikácia, klasifikácia metódou k-najbližších susedov. Metódy klasifikácie cez rozhodovacie stromy a k-najbližších susedov v neskorších kapitolách experimentálne porovnáme.

4.1 Pojem klasifikácia

Klasifikácia dát je 2-krokový proces. V prvom kroku sa na základe trénovacej množiny dát vytvorí **model**. Jeho konštrukcia je na základe analýzy vzoriek v trénovacej množine. Každá vzorka(n-tica atribútov) v tejto množine prislúcha istej preddefinovanej triede, určenej jedným z jej atribútov-nazvyme ho **Cieľovým** atribútom. Keďže cieľový atribút každej vzorky je vopred daný, nazýva sa tento princíp tiež ako učenie s učiteľom, na rozdiel od iného princípu **Segmentácie**, kde cieľový atribút nie je známy a nie je známy teda ani počet cieľových tried, určených týmto atribútom. Segmentácii sa bližšie budeme venovať v nasledujúcej kapitole.

Výstupom z tohto prvého kroku je teda model, ktorý istým spôsobom

v sebe nesie informáciu, postup, ako na základe atribútov vzorky určiť jej cieľový atribút, čo sa "naučil" na základe analýzy vzoriek trénovacej množiny, kde bol cieľový atribút vopred známy.

V druhom kroku sa model použije na klasifikáciu (určenie cieľového atribútu) množiny dát, kde nás zaujíma príslušnosť vzoriek k cieľovej triede určenej cieľovým atribútom. Podstatné je tiež určiť **presnosť** s akou daný model klasifikuje vzorky, napr. použitím testovacej množiny dát(obsahujúcej cieľový atribút), nezávislej na trénovacej množine, a porovnáva sa úspešnosť určenia cieľového atribútu daným modelom. Tento proces se nerobí na trénovacej množine z dôvodu možného pretrénovania(overfitting) modelu, čo je situácia, kedy model ku klasifikácii začne používať informácie špecifické pre trénovaciu množinu, nemajúce všeobecný význam pre určenie cieľového atribútu. Pretrénovanie je teda nežiaduci stav, kedy je model príliš naviazaný na testovaciu množinu, čím stráca univerzálnosť použitia na iných súboroch vzoriek.

4.1.1 Predpríprava dát

Predpríprava dát je dôležitý proces, ktorého účelom je zabezpečiť také vstupné dáta pre učiacu fázu(krok 1), ktoré obsahujú informácie dôležité pre určenie cieľových atribútov, a očistené od nadbytočných informácií nemajúcich v tomto kontexte podstatný význam. Z podstaty tohto procesu je jasné, že pri rozhodovaní, ktoré atribúty odstrániť, ktoré ponechať, hrá dôležitú úlohu človek, majúci predstavu o danej doméne, z ktorej dáta pochádzajú. Proces predprípravy dát pozostáva z nasledujúcich krokov:

1. **Dátové čistenie**-odstránenie nepresností ako šum pri spojitých atribútoch(použitím vyhladzovacích techník), nahradenie chýbajúcich hodnôt(napr priemernou prípadne najpravdepodobnejšou hodnotou)

2. **Relevančná analýza**-odstránenie redundantných atribútov

3. **Dátová tranformácia**- generalizácia atribútov na hrubšie členenie, normalizácia dát(napr aby vyhovovali vstupným intervalom hodnôt pre isté algoritmy), priradenie váh atribútom(pre vyznačenie signifikantných atribútov)

4.1.2 Kritéria klasifikačných metód

Klasifikačné metódy môžme hodnotiť nasledujúcimi kritériami:

1. **Presnosť predpovede**-schopnosť modelu korektne určiť cieľový atribút na neznámych vzorkách

2. **Rýchlosť**-schopnosť efektívne vytvárať a používať modely aj pre veľké množstvá dát

3. **Robustnosť**-schopnosť modelu robiť korektné klasifikácie a predikcie aj pri "znečistených" dátach

4. **Interpretovateľnosť**-snáď jedno z najdôležitejších kritérií, výpovedná hodnota modelu. Schopnosť poskytnúť nové poznatky, informácie, pochopenie skrytých vzťahov medzi skúmanými dátami, na základe analýzy reprezentácie daného modelu človekom.

4.2 Klasifikácia pomocou rozhodovacích stromov

Klasifikácia pomocou metódy rozhodovacích stromov sa najčastejšie používa na určenie cieľového atribútu majúceho diskrétne hodnoty. Táto metóda patrí medzi induktívne odvodzovacie metódy učenia a bola úspešne aplikovaná v mnohých oblastiach ako klasifikácia pacientov podľa diagnózy, prípadne predpoveď úverových, poistných podvodov(klasifikácia podľa rizika úverového, poistného podvodu).

Naučený model ako výstup tejto metódy je reprezentovaný *rozhodovacím stromom*. Rozhodovací strom je strom s nasledujúcimi vlastňosťami:

- Uzol, ktorý nie je listom reprezentuje test na atribúte z danej množiny vzoriek, uzlu tiež prislúcha podmnožina vzoriek, spĺňajúca testy na atribútoch pri ceste od koreňa ku danému uzlu

- Vetva reprezentuje výsledok testu na atribút

- Listy stromu reprezentujú triedy cieľového atribútu(hodnoty cieľového atribútu)

Rozhodovacie stromy klasifikujú vzorky postupne od koreňa na základe testov na hodnoty testovacích atribútov v uzloch stromu, čím sa klasifikovaná vzorka postupne prepracuje k listu, ktorý určí hodnotu jej cieľového atribútu.

Testovacie atribúty môžu mať diskrétne hodnoty, v tomto prípade je výstupom z daného uzla počet vetiev pre každú hodnotu testovacieho atribútu a daný atribút sa v nižších úrovniach už netestuje. Ak majú testovacie atribúty spojité hodnoty, test v uzle je reprezentovaný rozdelím intervalu hodnôt na 2 vetvy a v nižších úrovniach sa znovu testujú len prislúchajúce podintervaly.

Rozhodovací strom vieme ľahko konvertovať do *klasifikačných pravidiel*, každá cesta od vrcholu stromu k listu reprezentuje konjukciu podmienok podľa výsledkov testov na jednotliých uzloch.

4.2.1 Konštrukcia rozhodovacieho stromu

Základný algoritmus konštrukcie rozhodovacieho stromu- ID3(Quinlan 1986), ktorý tu prezentujeme, vytvára strom princípom zhora nadol, s počiatočnou otázkou: Ktorý atribút sa má testovať v koreni stromu?

Každý atribút je ohodnotený istým kritériom(prezentované nekôr), ktoré určuje ako dobre daný atribút sám o sebe klasifikuje trénovaciu množinu. Atribút s najlepšou hodnotou tohto kritéria je zvolený ako testovací atribút pre koreň, pre každú jeho hodnotu(ak ma diskrétne hodnoty) je vytvorená vetva, a množina vzoriek je rozdelená do patričných uzlov pod vetvami podľa hodnoty testovacieho atribútu.

Proces sa znova opakuje pre každý uzol nižšej úrovne(syna) a množinu vzoriek jemu prislúchajúcu, kde sa znova zvolí testovací atribút a jemnejšie rozdelí množina vzoriek tohto uzla, až kým sa nenaplnia terminačné podmienky(napr. počet prvkov v listoch(leaf-size)), kedy považujeme rozdelenie na množiny vzoriek dostatočne presné na určenie hodnoty cieľového atribútu.

Ide o greedy prístup, kedy sa vždy vyberá lokálne najlepšia voľba a algoritmus sa už nevracia aby prehodnotil skoršie voľby.

Po skonštruovaní stromu na základe trénovacej množiny sa tento model použije na klasifikáciu skúmaných vzoriek. Vzorka sa postupne od koreňa na základe testov na hodnoty svojich atribútov dopracuje ku listom stromu, ktoré reprezentujú hodnotu cieľového atribútu.

4.2.2 Kritérium voľby testovacieho atribútu

Najdôležitejším krokom pri konštrukcii rozhodovacieho stromu je voľba testovacieho atribútu pre jeho uzly. Tento atribút by mal čo najlepšie rozdelovať danú množinu vzoriek na danej úrovni stromu, v zmysle cieľovej klasifikácie. Kritériom pre jeho voľbu bude funkcia *informačný zisk, IZ*, pracujúca s *mierou*, ktorú nazývame *entropia, E*.

Entropia ako miera homogenity vzoriek

Entropia charakterizuje (ne)čistotu,(ne)homogenitu istej množiny vzoriek, vo vzťahu k nejakému cieľovému konceptu(vlastnosti prvkov danej množiny). Majme množinu S, obsahujúcu pozitívne a negatívne vzorky nejakého cieľového konceptu(cieľového atribútu). Entropia množiny S vo vzťahu k tejto booleovskej klasifikácii je

$$E(S) = -p_{\oplus} \log_2 p_{\oplus} - p_{\ominus} \log_2 p_{\ominus}$$

kde $p_{\oplus}$ je pomer počtu pozitývnych vzoriek a $p_{\ominus}$ pomer počtu negatívnych vzoriek v S voči všetkým vzorkám v S.

Explicitne definujeme situáciu $0 \log_2 0$ ako entropiu 0.

Príklad: Nech S má 14 prvkov, 9 pozitývnych, 5 negatívnych. Entropia množiny S vo vzťahu k tejto booloevskej klasifikácii je

$$E(S) = -(9/14) \log_2(9/14) - (5/14) \log_2(5/14) = 0.940$$

Entropia pri booleovskej klasifikácii má teda rozsah [0,1], s hodnotou 0 ak všetky prvky majú rovnaký cieľový koncept(v našom príklade buď všetky pozitívne alebo negatívne), hodnotou 1 pri rovnakom počet pozitívnych a negatívnych vzoriek.

Entropiu teda môžme interpretovať(v zmysle teórie kódovania) ako minimálny počet bitov potrebných na zakódovanie klasifikácie náhodne zvolenej(pri rovnomernom rozdelení pravdepodobnosti výberu) vzorky z množiny S, vo vzťahu k cieľovému konceptu.

Ak v našom príklade je $p_{\oplus} = 1$, prijímateľ vie, že akýkoľvek výber vzorky bude mať hodnotu cieľového konceptu=pozitívny, nie je potrebné posielať žiadnu správu, entropia je 0(množina S je homogénna k cieľovému konceptu). Ak je $p_{\oplus} = 0.5$, teda je rovnaká pravdepodobnosť voľby pozitívnej či negatívnej vzorky, potrebujejeme 1 bit na zakódovanie informácie či je vybraná

vzorka pozitívna alebo negatívna, entropia je 1. V situácii, že $p_{\oplus} = 0.8$ vytvoríme množinu správ, pričom pozitívnej vzorke sa priradí kratší bitový reťazec(správa) a negatívnej dlhší, čím sa pri prenose v priemere dosiahne menej ako 1 bit na správu, entropia je 0.721 .

Doteraz sme prezentovali entropiu v prípade, že klasifikácia podľa cieľového konceptu je booleovská. Nech cieľový koncept(cieľový atribút vzoriek v S) môže nadobúdať m rôznych hodnôt, potom entropia množiny S vo vzťahu k tejto m-hodnotovej klasifikácii je

$$E(S) = -\sum_{i=1}^{m} p_i \log_2 p_i$$

kde p_i je pomer počtu vzoriek patriacich cieľovej triede i (ich cieľový atribút=i), voči počtu všetkých prvkov v S.

Logaritmus je pri základe 2, pretože entropiu tu interpretujeme ako očakávanú dĺžku kódovania v bitoch.

Entropia v tomto prípade nadobúda rozsah hodnôt $[0, \log_2 m]$.

Informačný zisk ako miera redukcie entropie

Majúc entropiu ako mieru homogenity množiny trénovacích vzoriek, môžme definovať mieru efektívnosti atribútu v klasifikovaní trénovacej množiny.

Informačný zisk(IZ) vzhľadom na množinu S a atribút A, je očakávaná redukcia entropie, pri rozdelení vzoriek podľa atribútu A. Formálnejšie

$$IZ(S, A) = E(S) - \sum_{v \in Hodnoty(A)} \frac{|S_v|}{|S|} E(S_v)$$

kde $Hodnoty(A)$ je množina možných hodnôt atribútu A, S_v je podmnožina S, kde vzorky majú hodnotu atribútu $A = v$.

Prvý člen je entropia množiny S, kým druhý člen je suma entropií každej podmnožiny S_v, ohodnotených pomerom počtu prvkov v S_v ku mohutnosti S, ide o hodnotu entropie množiny S pri rozdelelení podľa atribútu A.

Informačný zisk je teda očakávaná miera redukcie entropie množiny S, pri znalosti hodnoty atribútu A. Inak povedané reprezentuje informáciu o hodnote cieľového atribútu, pri znalosti hodnoty atribútu A.

Kritérium pre výber testovacieho atribútu pri konštrukcii rozhodovacieho stromu, je teda maximalizovať funckiu $IZ(S, A)$.

4.2.3 Algoritmus ID3

Algorimtus ID3 používa ako vstup množinu vzoriek M, prislúchajúcu danému uzlu(*Root*) v strome, cieľový atribút *Ciel_atr*, rozdeľujúci trénovaciu množinu na triedy, podľa ktorého chceme neskôr klasifikovať skúmané vzorky a množinu atribútov *Attributy*, z ktorých vyberáme testovací atribút pre daný uzol. Listy stromu sa označujú cez *label*, čo reprezentuje hodnotu cieľovéo atribútu.

```
ID3(M,Ciel_atr,Attributy)
  vytvor nový koreň Root pre strom;
  if všetky prvky v M patria rovnakej triede C
      Root= jedno-uzlový strom, label=C;
  else if Attributy je prázdna
      Root= jedno-uzlový strom, label=najčastejšia
        hodnota Ciel_atr v M;
  else
      A= prvok z Attributy, maximalizujúci IZ(M,A);
      A je testovací atribút pre Root;
      for každú hodnotu x atribútu A
        pridaj novú vetvu pod Root, testujúcu A = x;
        M_x= podmnožina M, kde jej prvky majú A = x;
        if M_x je prázdna
            pod danú vetvu pridaj list, label=
              najčastejšia hodnota Ciel_atr v M;
        else
            pod danú vetvu pridaj podstrom
              s koreňom ID3(M_x,Ciel_atr,Attributy - {A});
return Root;
```

4.2.4 Výpočet informačného zisku, IZ

Informačný zisk,IZ je kritériom na výber atribútu pre ďalšie delenie vzoriek. Výber atribútu s najväčším informačným ziskom reprezentuje minimalizáciu potreby ďalej klasifikovať vzorky patriace pod uzol s testom na daný atribút. Nech S je množina obsahujúca s vzoriek. Nech cieľový atribút má m rôznych hodnôt, určujúcich m cieľových tried $C_i (i = 1, \ldots, m)$. Nech s_i je počet vzoriek množiny S v triede C_i. Potom položme entropiu E danej množiny S

ako

$$E(s_1, s_2, \ldots, s_m) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

kde p_i je pravdepodobnosť, že vzorka patrí do triedy C_i, $p_i = \frac{s_i}{s}$. Nech atribút A má v rôznych hodnôt $\{a_1, a_2, \ldots, a_v\}$. Atribút A môže rozdeliť S na v podmnožín $\{S_1, S_2, \ldots, S_v\}$, kde S_j obsahuje tie vzorky z S, majúce $A = a_j$. Ak by A bol zvolený ako testovací atribút, tak tieto podmožiny by boli priradené synom uzla(na základe testu na A) ktorému prislúcha množina S. Nech s_{ij} je počet vzoriek triedy C_i v podmožine S_j. Entropia E, na základe rozdelenia podľa atribútu A je:

$$E(s_1, s_2, \ldots, s_m, A) = \sum_{j=1}^{v} \frac{s_{1j} + \cdots + s_{mj}}{s} E(s_{1j}, \ldots, s_{mj})$$

Člen $\sum_{j=1}^{v} \frac{s_{1j}+\cdots+s_{mj}}{s}$ reprezentuje "váhu" j-tej podmnožiny S_j, ako pomer počtu tých vzoriek z S_j majúcich $A = a_j$ delených počtom vzoriek v S. Informačný zisk pri delení na atribúte A je

$$IZ(S, A) = E(s_1, s_2, \ldots, s_m) - E(s_1, s_2, \ldots, s_m, A)$$

Algoritmus ID3 spočíta IZ každého atribútu v množine S prislúchajúcej danému uzlu u, atribút s maximálnym IZ je zvolený ako testovací, množina S podľa toho rozdelená do synov uzla u.

4.3 Bayesovská klasifikácia

Bayesovské klasifikátory sú štatistické klasifikátory, dokážu predikovať pravdepodobnosť príslušnosti danej vzorky k cieľovej triede. Sú založené na *Bayesovej theoréme*, predpokladajú, že efekt atribútu na príslušnosť vzorky k istej cieľovej triede je nezávislý na ostatných atribútoch. Často sa kvôli tomuto predpokladu používa termín *Naivná Bayesovská klasifikácia*.

4.3.1 Bayesova theoréma

Nech X je vzorka, ktorej cieľová trieda je neznáma, nech H je hypotéza, že X patrí cieľovej triede C. Pre našu potrebu klasifikácie potrebujeme určiť $P(H|X)$, ide o tzv. posteriórnu pravdepodobnosť hypotézy H podmienej na X. Ak X má atribúty červený a guľatý a H je hypotéza, že X je jablko, potom

$P(H|X)$ reprezentuje našu istotu že X je jablko, na základe toho, že vieme že X je guľatý a červený objekt. $P(H)$ je tzv apriórna pravdepodobnosť, istota, že daný objekt je jablko bez uvažovania jeho vlastností. Bayesova theoréma nám umožní na základe znalosti $P(H)$, $P(X)$, $P(X|H)$ určiť $P(H|X)$:

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)}$$

4.3.2 Naivná Bayesovská klasifikácia

Naivný Bayesov klasifikátor pracuje nasledovne:

1. Každá vzorka je reprezentovaná n-dimenzionálnym vektorom vlastností $X = (x_1, x_2, \ldots, x_n)$, patriacim n-atribútom $A_1, A_2, \ldots, A_n$.

2. Majme m cieľových tried $C_1, C_2, \ldots, C_m$. Pre neznámu vzorku X(hodnota jej cieľového atribútu je neznáma), klasifikátor priradí triedu s najväčšou posteriórnou pravdepodobnosťou, podmieneonou na X. Teda klasifikátor priradí vzorke X cieľovú triedu C_i IFF:

   $$P(C_i|X) > P(C_j|X) \text{ pre } 1 \leq j \leq m, j \neq i$$

   Hľadáme teda maximálne $P(C_i|X)$. Podľa Bayesovej theorémy $P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)}$

3. $P(X)$ je konštanta pre všetky triedy, teda len $P(X|C_i)P(C_i)$ potrebujeme maximalizovať. Ak nie sú vopred určené apriórne pravdepodobnosti tried $C_1, C_2, \ldots, C_m$, platí, že $P(C_i) = \frac{s_i}{s}$, kde s_i je počet vzoriek v trénovacej množine, prislúchajúcich triede C_i, s je počet všetkých vzoriek v trénovacej množine. Maximalizujeme teda len člen $P(X|C_i)$.

4. Kedže pri veľkom počte atribútov by bolo výpočtovo náročné určovať $P(X|C_i)$, je urobený práve predpoklad nezávislosti atribútov pri príslušnosti k danej cieľovej triede, teda:

   $$P(X|C_i) = \prod_{k=1}^{n} P(x_k|C_i)$$

   Pravdepodobnosti $P(x_1|C_i), P(x_2|C_i), \ldots, P(x_n|C_i)$ sa dajú určiť z trénovacej množiny, kde: Ak A_k je diskrétny potom $P(x_k|C_i) = \frac{s_{ik}}{s_i}$, kde

s_{ik} je počet vzoriek z trénovacej množiny, patriacich triede C_i, pre ktoré $A_k = x_k$. s_i je počet vzoriek z trénovacej množiny patriacich triede C_i.

Ak A_k je spojitý, tak sa predpokladá Gausovo normálne rozdelenie hodnôt, teda:

$$P(x_k|C_i) = g(x_k, \mu_{C_i}, \sigma_{C_i}) = \frac{1}{\sqrt{2\pi}\sigma_{C_i}} e^{-\frac{(x-\mu_{C_i})^2}{2\sigma_{C_i}^2}}$$

kde μ_{C_i} je stredná hodnota a σ_{C_i} rozptyl hodnôt atribútu A_k trénovacích vzoriek z triedy C_i.

5. Na klasifikáciu neznámej vzorky X je vypočítaný člen $P(X|C_i)P(C_i)$ pre každú triedu C_i. Vzorke X je priradená trieda C_i IFF:

$$P(X|C_i)P(C_i) > P(X|C_j)P(C_j) \text{ pre } 1 \leq j \leq m, j \neq i$$

teda, vzorka je priradená tej triede C_i, pre ktorú $P(X|C_i)P(C_i)$ je maximálny.

4.4 Klasifikácia pomocou metódy K-najbližších susedov

Klasifikácia metódou K-najbližších susedov je klasifikácia na princípe učenia sa pomocou analógie. Vzorky z trénovacej množiny majú n číselných atribútov, každá vzorka teda reprezentuje bod v N-rozmernom priestore. Keď chce klasifikátor určiť cieľový atribút neznámej vzorky, hľadá v tomto priestore k vzoriek z trénovacej množiny, ktoré sú najbližšie našej neznámej vzorke, na základe miery, pomocou ktorej určuje vzdialenosť(napr. Euklidovská, Manhatanovská).

Majme vzorku X, určenú vektorom hodnôt jej n-atribútov $X = (x_1, x_2, \ldots, x_n)$, ktorú potrebujeme klasifikovať(pomocou euklidovskej miery). Euklidovská miera pre vzorku X a vzorky $Y = (y_1, y_2, \ldots, y_n)$ patriace trénovacej množine je definovaná ako

$$d(X, Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

Proces klasifikácie určí zo vzoriek(bodov v n-rozmernom priestore) Y, k najbližších a neznámej vzorke X je priradená hodnota cieľového atribútu(cieľová trieda), ktorá je najčastejšia medzi k-najbližšími susedmi. Ak je cieľový atribút spojitý, priradí sa neznámej vzorke priemerná hodnota cieľového aribútu medzi jej k-najbližšími susedmi.

Klasifikátory na princípe najbližších susedov sú tzv *inštančné*, pretože ako model ukladajú celú trénovaciu množinu, pre potreby hľadania susedov k neznámym vzorkám. Dôležitú úlohu tu preto zohráva rozumná implementácia a indexovacie techniky, keďže porovnávania v hustom priestore trénovacích vzoriek sú výpočtovo náročné. Na rozdiel od rozhodovacích stromov, tieto klasifikátory defaultne uvažujú všetky atribúty s rovnakou váhou(dôležitosťou pre proces klasifikácie), čo môže viesť k chybným klasifikáciam, ak sa v trénovacej množine vyskytujú *outliers*, vzorky majúce chybne prípadne nesprávne určené hodnoty aribútov, čím negatívne ovplyvňujú proces klasifikácie nových vzoriek.

Ďalším problémom môže byť tzv. kliatba rozmerov(curse of dimensionality). Ide o situáciu, kedy majú vzorky veľký počet atribútov a proces klasifikácie určuje vzdialenosť na základe všetkých atribútov. Atribúty signifikantné pre cieľovú klasifikáciu môžu byť takto "prebité" ostatnými atribútmi, keďže vzdialenosť dvoch vzoriek majúcich rovnaké hodnoty týchto signifikantných atribútov môže byť stále veľká kôli hodnotám ostatných atribútov.

Problém tzv. outliers aj kliatby rozmerov je možné riešiť priradením rôznych váh jednotlivým atribútom, signifikantné atribúty(rozmery, osi v priestore atribútov) sa prenásobia faktorom $x > 1$ a menej podstatné atribúty faktorom $x < 1$, faktor $x = 0$ plne eliminuje daný atribút(rozmer) z procesu klasifikácie.

Kapitola 5

Segmentácia

Proces zoskupovania skupiny fyzických alebo abstraktných objektov do tried *podobných* objektov nazývame **segmentáciou**(zhlukovaním). **Segment** je skupina objektov, ktoré sú si v istých kritériách *podobné* voči sebe navzájom a sú *rozdielne* voči objektom mimo segmentu.

Tento proces je človeku prirodzený, už od detstva sa učíme rozlišovať medzi psami a mačkami, rastlinami a zvieratami, pozorovaním, hľadaním spoločných znakov a všímaním si rozdielov sa postupne učíme zaraďovať objekty, ktoré nás obklopujú do skupín(segmentov).

Segmentácia sa široko využíva v rôznych oblastiach ako rozpoznávanie vzoriek, spracovanie obrazu, analýza obchodu, pomocou segmentácie napríklad vieme identifikovať skupiny zákazníkov správajúcich sa podobne v stratégii nákupu a môžeme na týchto skupinách(segmentoch) prevádzať ďalšie analýzy, ktoré nám bližšie charakterizujú ich spoločné vlastnosti, čo vieme využiť v obchodných rozhodnutiach. V biológii napr. dokážeme pomocou segmentácie určit segmenty génov, a ďalej analyzovať ich vlastnosti.

Segmentácia sa v dataminingu využíva teda ako prvý krok, hrubá analýza a zatriedenie objektov do skupín(segmentov), na ktoré môžme aplikovať jemnejšie spôsoby analýzy. Segmentácia je často predchádzajúci krok pred inými metódami dataminingu ako **klasifikácia**, **charakterizácia tried**, prípadne **asociačná analýza**.

5.1 Pojem segmentácia

Segmentácia sa často nazýva tiež **učenie bez učiteľa**. Na rozdiel od klasifikácie, segmentácia nie je závislá na prítomnosti cieľového atribútu pri trénovacích vzorkách, je to skôr forma *učenia sa pozorovaním* ako učenia sa na základe príkladov(v zmysle prítomnosti cieľ.atribútu pri trénovacích vzorkách). Bežný proces segmentácie zhlukuje objekty do segmentov na základe *miery podobnosti*, často určovanej podľa vzdialeností v priestore vzoriek(obdobne ako pri klasifikácii cez k-najbližších susedov).

5.1.1 Kritériá segmentačných metód

1. **Schopnosť pracovať s rozdielnymi typmi atribútov**, mnohé algoritmy segmentácie zvládnu len číselné atribúty vzoriek, je potrebná schopnosť pracovať aj na nominálnych typoch atribútov.

2. **Schopnosť objavovať segmenty rôznych tvarov**, mnohé algoritmy pracujú na princípe merania Euklidovskej, prípadne inej vzdialenosti v priestore atribútov vzoriek, čím nachádzaju segmenty *guľového* tvaru, s podobnou veľkostou a hustotou vzoriek, ale segmenty v dátach môžu byť akýchkoľvek tvarov, reflektujúcich podobnosť vzoriek v nich obsiahnutých.

3. **Minimalizácia potreby znalosti skúmanej domény dát, pre určenie vstupných parametrov**, mnohé algoritmy potrebujú citlivé nastavenia vstupných parametrov, čo nie je vždy možné napr pri skúmaní domény s veľkým počtom atribútov.

4. **Schopnosť pracovať zo znečistenými dátami**, reálne dáta obsahujú nepresnosti, na ktoré sú algoritmy segmentácie zvášť citlivé, keďže často pracujú na princípe merania vzdialeností.

5. **Zvládanie veľkého množstva dimenzií(atribútov)**, požiadavka vyplývajúca z rozmachu *dátových skladov*, kde máme pre dáta k dispozícii obrovské množstvo atribútov.

6. **Interpretovateľnosť**, často je potrebné charakterizovať nájdené segmenty, schopnosť priradiť segmentom istú sémantiku v zmysle skúmanej domény dát je dôležitým kritériom.

5.2 Segmentácia metódou k-means

Algoritmus metódy k-means(MacQueen 1967) vezme vstupný parameter k a rozdelí n objektov do k segmentov tak, že maximalizuje podobnosť objektov v rámci segmentu a minimalizuje podobnosť medzi segmentami navzájom.

Algoritmus najskôr náhodne zvolí k objektov, z ktorých každý reprezentuje *mean*-stred, centrum segmentu. Ostatné objekty sú priradené ku segmentom(reprezentovanými centrami segmentov) na základe podobnosti, určenej cez vzdialenosť medzi objektami a centrami segmentov. Na základe rozdelenia objektov do segmentov sa nanovo vypočítajú ich centrá(body rovnako vzdialené od objektov v segmente). Proces sa opakuje až kým nezkonverguje funkcia:

$$E = \sum_{i=1}^{k} \sum_{x \in C_i} |x - m_i|^2$$

kde x je bod v priestore atribútov, reprezentujúci daný objekt, m_i sú centrá jednotlivých segmentov C_i. Čiže pri prepočte nových centier segmentov sa neudeje zmena.

Táto metóda pracuje dobre, keď sú segmenty v dátach kompaktné zhluky, dobre navzájom oddelené. Tiež dobre pracuje na veľkých súboroch dát, pretože jeho výpočtová zložitosť je $O(nkt)$, kde n je počet objektov, k počet segmentov, t počet iterácií, obvykle platí $k \ll n$ a $t \ll n$.

Slabinou metódy je počiatočný náhodný výber segmentov, čo v konečnom dôsledku nemusí znamenať termináciu pri rozdelení do najlepších segmentov, kedy funkcia E nadobúda minimum. Toto minimum môže byť len lokálne(z hľadiska rozdelenia do segmentov). Typickým príkladom je segmentácia 4 bodov v rovine A,B,C,D, reprezentujúcich obdĺžnik so stranami a, b $b > a$, na 2 segmenty. Ak sa ako počiatočné centrá segmentov určia stredy najdlhších strán b, ide o situáciu kedy funkcia E nadobudne minimum pri rozdelení do 2 segmentov reprezentovaných najdlhšími stranami obdĺžnika, kedže nový prepočet centier nájde rovnaké centrá segmentov. Optimálne rozdelenie sú však segmenty reprezentované ako najkratšie strany obdĺžnika.

Metóda sa dá ale aplikovať len v prípade, keď vieme definovať centrá segmentov, čo býva problém ak sa vo vzorkách vyskytujú nominálne atribúty. Taktiež nieje táto metóda vhodná na objavovanie segmentov nekonvexných tvarov, prípadne veľmi rozdielnych veľkostí. Je tiež citlivá na znečistené dáta, pretože takéto vzorky znateľne ovplyvňujú výpočet centier segmentov. Nutnosť určiť vstupný parameter k, je tiež problémom pri analyzovaní dát z

domény, kde užívateľ nemá dostatok znalostí na predpokladanie počtu segmentov, ktoré chce nájsť.

5.3 Segmentácia metódou k-medoids

Algoritmus k-means je citlivý na *outliers*, pretože takýto objekt, môže podstatne ovplyvniť výpočet centier segmentov. Namiesto výpočtu centra segmentov ako referenčného bodu, môžeme vziať jeden z objektov v segmente(*medoid*, reprezentant), čo bude objekt, ktorý spomedzi ostatných môžme najviac uvažovať, ako centrum segmentu.

PAM(partition around medoids) je typ segmentačnej metódy,(pracujúcej obdobne ako k-means), ktorá nájde k segmentov v n objektoch, cez nájdenie reprezentatívneho objektu(medoid) každého segmentu. Po prvotnom náhodnom určení k reprezentantov, sa algoritmus snaží opakovane určiť lepší výber reprezentantov cez porovnanie všetkých dvojíc typu reprezentant-nereprezentant, a hodnoty funkcie určujúcej *mieru podobnosti* všetkých objektov,(napr funkcie E z metódy k-means). Najlepší výber nových reprezentantov v tejto iterácií bude vstupom do ďalšej iterácie. Proces končí ak žiadny výber nových reprezentantov nezlepší mieru podobnosti.

5.4 Segmentácia hierarchickými metódami

Segmentácia cez hierarchické metódy pracuje na princípe zoskupovania objektov do stromu, kde uzly reprezentujú segmenty. Tieto metódy môžme rozdeliť na zoskupovacie(aglomerative) a rozdeľovacie(divisive) podľa toho či sa hierarchia formuje princípom zdola nahor resp zhora nadol.

1. **Zoskupovacie hierarchické segmentovanie** začína položením každého objektu do vlastného segmentu, a tieto segmenty neskôr spája kým všetky objekty nie sú v jednom segmente alebo sa splní podmienka ukončenia. Metódy implementujúce tento princíp sa líšia len v spôsobe výberu segmentov pre zlúčenie, napr metóda AGNES(Aglomerative Nesting), určuje podobnosť 2 segmentov na základe podobnosti najbližšej dvojice objektov z ktorých každý patrí do iného segmetu. Podobnosť segmentov sa často učuje cez vzdialenosti medzi nimy. Bežne používané miery:

$$d_{min}(C_i, C_j) = min_{p \in C_i, \acute{p} \in C_j} |p - \acute{p}|$$

$$d_{stred}(C_i, C_j) = |m_i - m_j|$$
$$d_{max} = max_{p \in C_i, \acute{p} \in C_j} |p - \acute{p}|$$

kde m_i je *stred,centrum* segmentu C_i, n_i je počet bodov v C_i a $|p - \acute{p}|$ je vzdialenosť bodov p a $\acute{p}$

2. **Rozdeľovacie hierarchické segmentovanie** začína so všetkými objektami v jednom segmente, ktorý postupne rozdeľuje na menšie a menšie segmenty, pokým sa nenaplní ukončovacia podmienka, ako napr želaný počet segmentov alebo vzdialenosť dvoch najbližších segmentov prekročila istú limitnú hodnotu. Rozdeľovacie metódy sa zriedka využívajú, kôli problému zvoliť správne delenie na vyšších úrovniach.

5.5 Segmentácia cez meranie hustoty, DBSCAN

DBSCAN(Density-Based Spatial Clustering of Application with Noise)(Estrel, Kriegel, Sander, Xu 1996) je segmentačný algorimus, pracujúci na princípe hustoty vzoriek v priesore. Princíp je nasledovný:

Pre každý objekt(vzorku), je určené okolie polomeru $\epsilon(\epsilon - okolie)$, ktoré musí obsahovať minimálny počet objektov ($MinPts$). Objekt, ktorý má vo svojom ϵ-okolí aspoň ($MinPts$) objektov, nazývame **Core objekt**.

Objekt p je **priamo dosiahnuteľný** z objektu q v množine D v súlade s polomerom ϵ a minimálnym počtom objektov ($MinPts$), ak p je vnútri ϵ-okolia bodu q, ktoré obsahuje aspoň ($MinPts$) objektov.

Objekt p **dosiahnuteľný** z objektu q v množine D v súlade s polomerom ϵ a minimálnym počtom objektov ($MinPts$), ak existuje reťaz objektov $p_1, \ldots, p_n$, $p_1 = q$ a $p_n = p$ takých, že $1 \leq i \leq n$, $p_i \in D$ a platí, že p_{i+1} je priamo dosiahnuteľný z p_i.

Objekt p je **spojený** s objektom q v množine D v súlade s polomerom ϵ a minimálnym počtom objektov ($MinPts$), ak existuje objekt $o \in D$, taký že p aj q sú priamo dosiahnuteľné z objektu o.

Segment založený na hustote je množina *spojených* objektov, ktorá je maximálna v zmysle *dosiahnunteľnosti*, a každý objekt nepatriaci žiadnemu segmentu je považovaný za šum(neželanú vzorku).

DBSCAN pracuje na základe kritéria *dosiahnunteľnosti*. Preskúma ϵ-okolie každého objektu, ak ϵ-okolie objektu p obsahuje aspoň ($MinPts$) objektov, je vytvorený nový segment s p ako jeho *core objektom*. Následne algoritmus iteratívne zhlukuje objekty *priamo dosiahnuteľné* z týchto *core-objektov*, čo

môže zahŕňať aj spojenie segmentov. Proces skončí ak sa nemôžu pridať nové objekty k žiadnemu zo segmentov.

Kapitola 6

Experiment na reálnych dátach

Cieľom tejto kapitoly, je prezentovať v hrubom rámci proces klasifikácie na dátach, získaných z procesov a javov v reálnom svete, priblížiť proces dolovania z dát ako sa s ním stretávame v praxi.

Experimtálne tu porovnáme dve klasifikačné metódy,

1. **Klasifikácia rozhodovacím stromom, algoritmom ID3**
2. **Klasifikácia metódou k-najbližších susedov**

6.1 Výber programového nástroja

Nástrojov pre datamining je k dispozícii dosť, či už open-source alebo proprietárnych riešení. Do užšieho výberu sa dostali nástroje:

1. **RapidMiner**- http://rapid-i.com/content/blogcategory/10/69/
2. **WEKA**- http://www.cs.waikato.ac.nz/ml/weka/
3. **TANAGRA**- http://eric.univ-lyon2.fr/ ricco/tanagra/en/tanagra.html

Každý poskytuje bohaté sady algoritmov, WEKA má k dispozícii kompletné API rozhranie k svojim algoritmom, TANAGRA je produkt s veľkou flexibilnosťou pri zostavovaní samotných datamingových procesov, zvolil som si však open-source nástroj RapidMiner, keďže splňoval hlavné kritériá pre náš experiment:

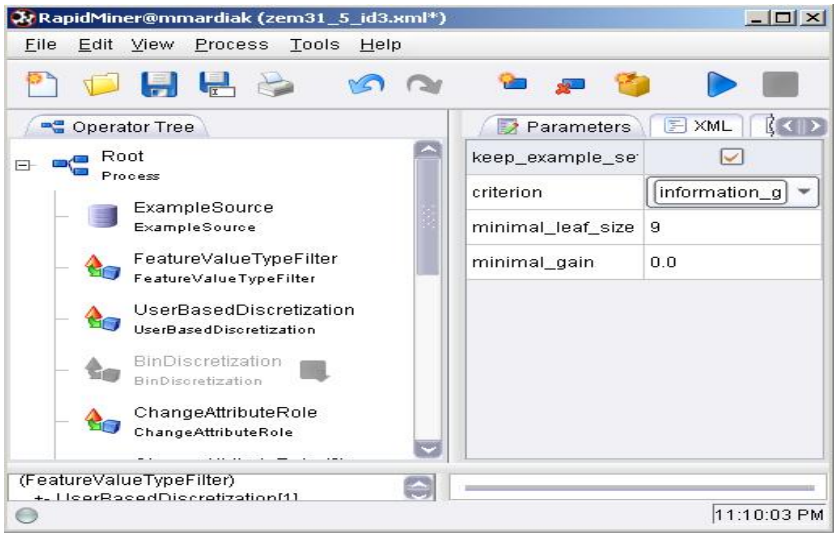
- **Robustnosť**, experimentoval som na datasetoch rôznej veľkosti, hlavne nástroj WEKA mal problém pri datasetoch väčších objemov, RapidMiner sa javil najstabilnejší

- **Prehľadné užívateľské rozhranie pri dostatočnej komplexnosti tvorby procesov**

- **Vizualizácia**, keďže náš primárny cieľ je prezentovať proces a princípy dataminingových metód, bohaté možnosti vizualizácie výsledkov boli veľmi dôležité

6.1.1 Prostredie RapidMiner

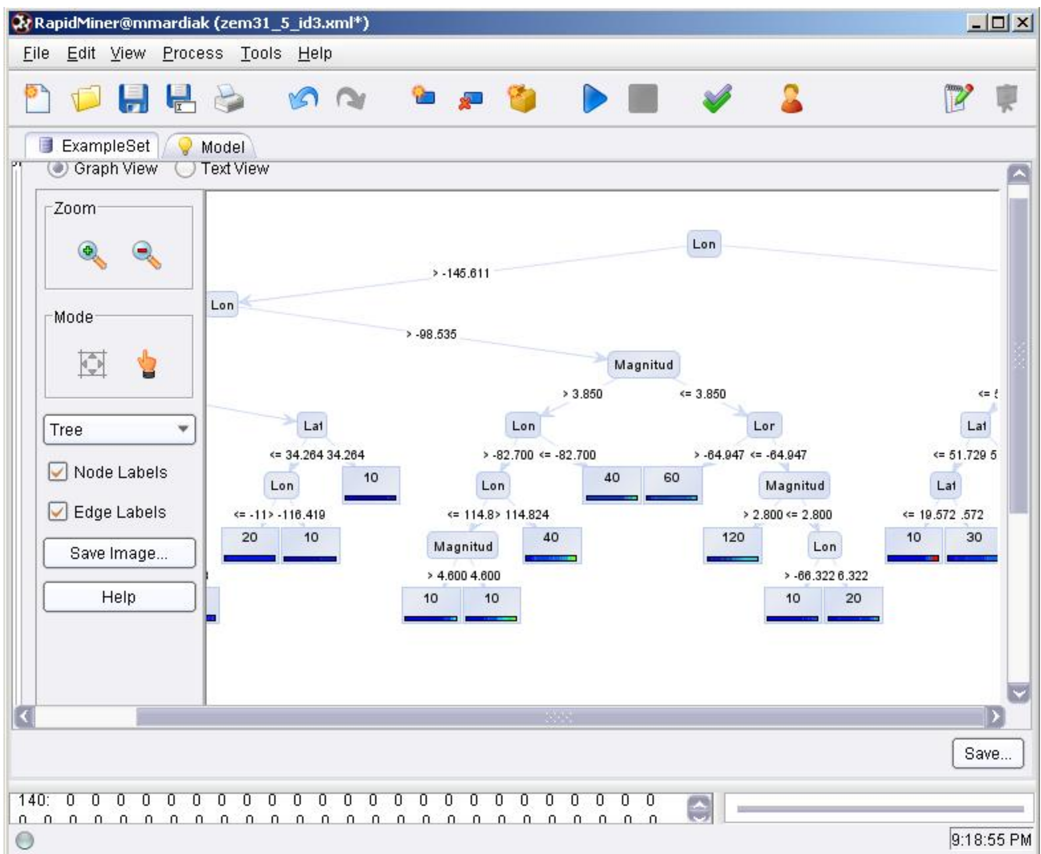
RapidMiner je veľmi užívateľsky príjemný, jeho filozofia je skladanie samotného procesu dataminingu ako skladanie prvkov stromu, ktorý sa zobrazuje v ľavej časti prostredia, každý prvok ma definovaný vstup a výstup, jednotlivé prvky tvoria rôzne procesy či už načítanie zdroja dát, predprípravu dát, potrebné transformácie atribútov, aplikáciu algoritmov dataminingu, uloženie naučených modelov, ohodnotenie výsledkov.



Obr. 6.1: RapidMiner

6.1.2 Vizualizácia v RapidMiner

RapidMiner umožňuje zobrazenie výsledkov v mnohých vizuálnych formátoch. Po dokončení procesu je možné vizualizovať pôvodný dataset aj s atribútmi určenými modelom, model v textovom, prípadne grafickom formáte(napr rozhodovacie stromy), metadáta a štatistiky o datasete a výsledky hodnotení úspešnosti klasifikácie.



Obr. 6.2: RapidMiner vizualizácia rozhodovacieho stromu

6.2 Výber a popis datasetu pre experiment

Dáta pre náš experiment pochádzajú z Earthquake Hazards Program, sekcie Earthquake center, informačnom zdroji prístupnom na sieti World Wide Web http://earthquake.usgs.gov/eqcenter/catalogs/index.php#csv

Ide o pravidelne aktualizovanú databázu zemetrasení. K dispozícii sú datasety za poslednú hodinu, deň, a sedem dní. Zemetrasenia sú špecifikované súborom atribútov:

- **Dátum výskytu**- Date
- **Lokácia**- Region
- **Magnitúda**- Magnitude, je funkciou dekadického logaritmu amplitúdy vlny (zjednodušene)
- **Hĺbka epicentra pod zemským povrchom v km**- Depth
- **Počet staníc, ktoré zemetrasenie zachytili**- NST
- **Zemepisná šírka**- Latitude
- **Zemepisná dĺžka**- Longitude

okrem nich obsahujú datasety aj syntetické identifikátory ako Src, ID, version, podstatné pre identifikáciu záznamu v danom informačnom portáli, ale pre nás tieto údaje z hľadiska reálneho procesu zemetrasenia a jeho atribútov význam nemajú.

Naším cieľom bude klasifikácia vzoriek podľa ich magnitúdy, na základe ostatných atribútov. Trénovacou množinou bude súbor vzoriek zemetrasení za obdobie *24.5.2008* až *31.5.2008 eqs7day-M1_31_5_08*, klasifikovať budeme súbor za obdobie *1.6.2008* až *8.6.2008 eqs7day-M1_8_6_08*, obidva získané z vyššie spomenutého zdroja.

6.2.1 Predpríprava dát

Predpríprava dát zahŕňa väčšinu času pri akomkoľvek procese dolovania z dát. V našom prípade odpadá potreba dátového čistenia, keďže máme priamo od zdroja k dispozícii množinu vzoriek bez chýbajúcich hodnôt atribútov. Je však potrebná relevančná analýza: Potrebujeme nutne všetky atribúty pre našu klasifikáciu? Aké sú vzťahy medzi nimy? Keďže skúmame oblasť(doménu),

o ktorej máme k dispozícii údaje- vieme zistiť napr do akej hĺbky sa vyskytujú najčastejšie zemetrasenia(wikipedia, odborná literatúra), vieme, že amplitúda zemetrasenia súvisí z jeho hĺbkou atď. dokážeme na základe týchto explicitných znalostí určiť najrelevantnejšie atribúty pre našu klasifikáciu.

Vezmime si atribút *Region*. Priamo súvisí s atribútmi *Longitude* a *Latitude*, dokonca tieto dva bližšie špecifikujú informáciu, ktorú poskytuje atribút Region, preto ho v predpríprave dát vypustíme z trénovacej množiny, a taktiež ho nezahrnieme do množiny vzoriek, ktorú budeme klasifikovať.

Atribút *Date* taktiež nemá veľký význam, keďže ide o vzorky v rozsahu 7 dní, a tento atribút naberá toľko unikátnych hodnôt(je zahrnutý aj čas), že jeho informačný zisk je veľmi vysoký, umelo, bez priamej väzby na cieľový atribút(v našej vzorke za obdobie len 7 dní dátum a čas výskytu zemetrasenia nemá významný vplyv na jeho charakteristiky). Tento atribút taktiež vypustíme.

6.2.2 Problémy v predpríprave dát

Počas fázy transformácie dát(algoritmus ID3 potrebuje nominálne hodnoty cieľového atribútu, avšak atribút *Magnitude* má reálne hodnoty), sa vyskytol problém. Prostredie RapidMinera síce poskytuje diskretizáciu spojitých atribútov, nedovoľuje však špecifikovať ktoré(diskretizuje všetky okrem tých, ktoré nemajú špeciálnu úlohu napr. cieľový atribút), naviac jeho metóda rovnomennej diskretizácie(rozsah sa rozdelí na intervaly rovnakej veľkosti) viedla k problému s kompatibilitou tohto rozdelenia na rôznych datasetoch. Trénovacia množina mala iný rozsah hodnôt cieľového atribútu ako množina určená na klasifikáciu, čo je neprípustná situácia, keďže intervaly(nominálne hodnoty) atribútu *Magnitude* po diskretizácii, neboli pre trénovaciu množinu a množinu určenú na klasifikáciu rovnaké.

Problém sa vyriešil manuálnou diskretizáciou atribútu *Magnitude* jednotnou pre trénovaciu aj klasifikačnú množinu, v intervaloch od 0 po krokoch 0.5 až do hodnoty 10, keďže magnitúda sa určuje celosvetovo v *Richterovej škále* s rozsahom $< 0, 10 >$.

6.3 Experimentálne porovnanie klasifikácie metód Rozhodovacieho stromu a K-najbližších susedov

Na trénovacej množine vytvoríme 2 modely, jeden ako výstup algoritmu ID3numerical(algoritmus ID3, podporujúci spojité hodnoty necieľových atribútov), a druhý ako výstup algoritmu K-nearestNeighbours, oba implementované v prostredí RapidMiner. Budeme sledovať presnosť predikcie atribútu *Magnitude* pomocou týchto modelov aplikovaných na množinu vzoriek, ktorú chceme klasifikovať- *eqs7day-M1_8_6_08*.

6.3.1 Parametre porovnávaných metód

Presnosť predikcie nás bude zaujímať vo vzťahu k nasledovným vstupným parametrom porovnávaných metód:

- **Leaf size**(ID3numerical)-minimálny počet vzoriek trénovacej množiny pre utvorenie listu v rozhodovacom strome, silne ovplyvňuje hĺbku rozhodovacieho stromu
- **K**(K-nearestNeighbours)-počet najbližších vzoriek v priestore trénovacej množiny na základe ktorých algorimtus K-najbližších susedov zvolí cieľový atribút pre skúmanú vzorku
- **Váhy atribútov**(K-nearestNeighbours)-koeficienty, ktorými sa prenásobia rozmery(atribúty), čím sa ovplyvňuje ich dôležitosť pri klasifikácii inštančnou metódou K-najbližších susedov, rozmer prenásobený koeficientom $koef > 1$ bude signifikantnejší pri klasifikácii

6.3.2 Výsledky experimentu klasifikácie cez ID3

Zdá sa, že parameter leaf-size nehrá dôležitú úlohu pri úspešnosti klasifikácie v našom experimente. Hoci celková úspešnosť klasifikácie**(tabuľka 6.1)** sa drží na vyrovnaných hodnotách, s maximami pri parametre leaf-size=16, hlbšia analýza ukázala zaujímavú skutočnosť, že presnosť predikcie uvažovaná aj v rámci presnosti v jednotlivých triedach cieľového atribútu *Magnitude*, silno závisí od parametra leaf-size. Pri nižších hodnotách tohto parametra, je rozhodovací strom hlbší, a je schopný presnejšie klasifikovať cieľový atribút

leaf size	úspešnosť v %
1	40.4
2	41.06
4	42.1
8	44.56
12	45.6
16	45.88
20	45.22
32	44.28
64	45.03

Tabuľka 6.1: ID3-úspešnosť klasifikácie

pre našu 20 hodnotovú škálu. Zdanlivo vyrovnaná celková presnosť klasifikácie**(tabuľka 6.1)** pri vyšších hodnotách tohto parametra je spôsobená nerovnomerným zastúpením hodnôt cieľového atribútu *Magnitude* v našej 20 hodnotovej škále.

Porovnanie presnosti predikcie v % pre jednotlivé triedy cieľového atribútu *Magnitude***(tabuľka 6.2)**, pri parametroch leaf-size(ls)=4, ls=12, ls=16, ls=32 :

magnitude	zastúpenie v %	ls=4	ls=12	ls=16	ls=32
0.5	0	-	-	-	-
1.0	4.26	14.29	0.0	0.0	0.0
1.5	36.42	58.77	61.52	59.62	52.19
2.0	27.34	37.84	39.07	38.89	37.02
2.5	14.29	31.76	34.88	34.31	7.14
3.0	6.62	19.23	37.5	0.0	0.0
3.5	2.27	14.29	33.33	33.33	0.0
4.0	1.61	62.5	0.0	0.0	0.0
4.5	1.51	31.25	0.0	0.0	0.0
5.0	3.22	50.0	33.33	31.33	41.82
5.5	1.51	27.27	0.0	0.0	0.0
6.0	0.19	25.0	25.0	0.0	0.0
6.5	0.19	0.0	0.0	0.0	0.0

Tabuľka 6.2: ID3-klasifikácia v triedach

6.3.3 Výsledky experimentu klasifikácie cez K-najbližších susedov

	1	2	4	8	16	32	64
F	39.07	42.01	43.61	45.13	43.71	44.28	42.19
OOOO	24.98	29.52	32.45	34.53	37.75	38.32	32.73
OOOX	29.04	32.92	32.73	33.4	34.44	35.76	36.33
OOXO	25.26	28.57	28.19	32.54	33.59	31.5	33.02
OXOO	38.51	42.19	40.49	43.42	42.86	41.53	40.68
XOOO	17.12	21.1	25.35	33.02	32.07	30.65	29.99
OOXX	24.03	28.1	30.09	34.72	36.9	35.95	35.48
OXOX	39.55	39.07	41.53	45.13	44.47	44.18	42.29
XOOX	16.56	25.54	28.38	32.45	33.87	34.06	32.26
OXXO	31.88	34.82	37.75	36.23	37.65	40.11	41.06
XOXO	20.53	23.65	28.95	31.88	28.48	30.27	28.1
XXOO	24.5	24.69	32.36	35.0	31.41	34.06	30.46
OXXX	37.37	36.9	40.11	44.56	44.94	43.8	42.67
XOXX	21.48	25.64	29.33	31.98	33.96	33.77	33.21
XXOX	23.56	34.15	35.48	39.07	39.17	39.45	32.45
XXXO	21.19	28.57	33.3	33.02	33.11	32.92	30.37

Tabuľka 6.3: K-sused-úspešnosť klasifikácie

Vo vyššie uvedenej **tabuľke(6.3)** sú zhrnuté výsledky celkovej úspešnosti klasifikácie v % cez K-najbližších susedov, v závislosti od váh jednotlivých atribútov(riadky), a parametra počet susedov(k) (stĺpce). **Atribúty sú uvedené v poradí Latitude, Longitude, Depth, NST**. Pre potreby úpravy váh sa tieto číselné atribúty normalizovali do rozsahu $< 0, 1 >$, aby sa dali pozorovať zmeny v úspešnosti klasifikácie pri zmene ich váh. Symbol X značí koeficient váhy atribútu koef = 100, symbol O koef = 1.0 (váha bez zmeny). Symbol F značí atribúty v pôvodnom nameranom rozsahu, bez normalizácie. Riadok $OXOX$ značí teda nasledovné váhy(koeficienty) atribútov:Lat(1.0),Lon(100),Depth(1.0),NST(100).

V našom experimente je možné pozorovať významnú úlohu atribútu Longitude pre úspešnosť klasifikácie. Taktiež atribút NST sa ako ďalší podstatne podiela na presnosti klasifikácie.

Na druhej strane je badať menej významný(až rušivý) vplyv atribútu Latitude na presnosť našej predikcie, čo je vidieť z poslednej časti **tabuľky(6.3)**,

prípadne poslednej skupiny zobrazených dát v nasledujúcom **grafe(6.3)**, kedy po priradení tomuto atribútu ako jedinému najmenšej váhy($koef = 1$), stúpla presnosť klasifikácie.

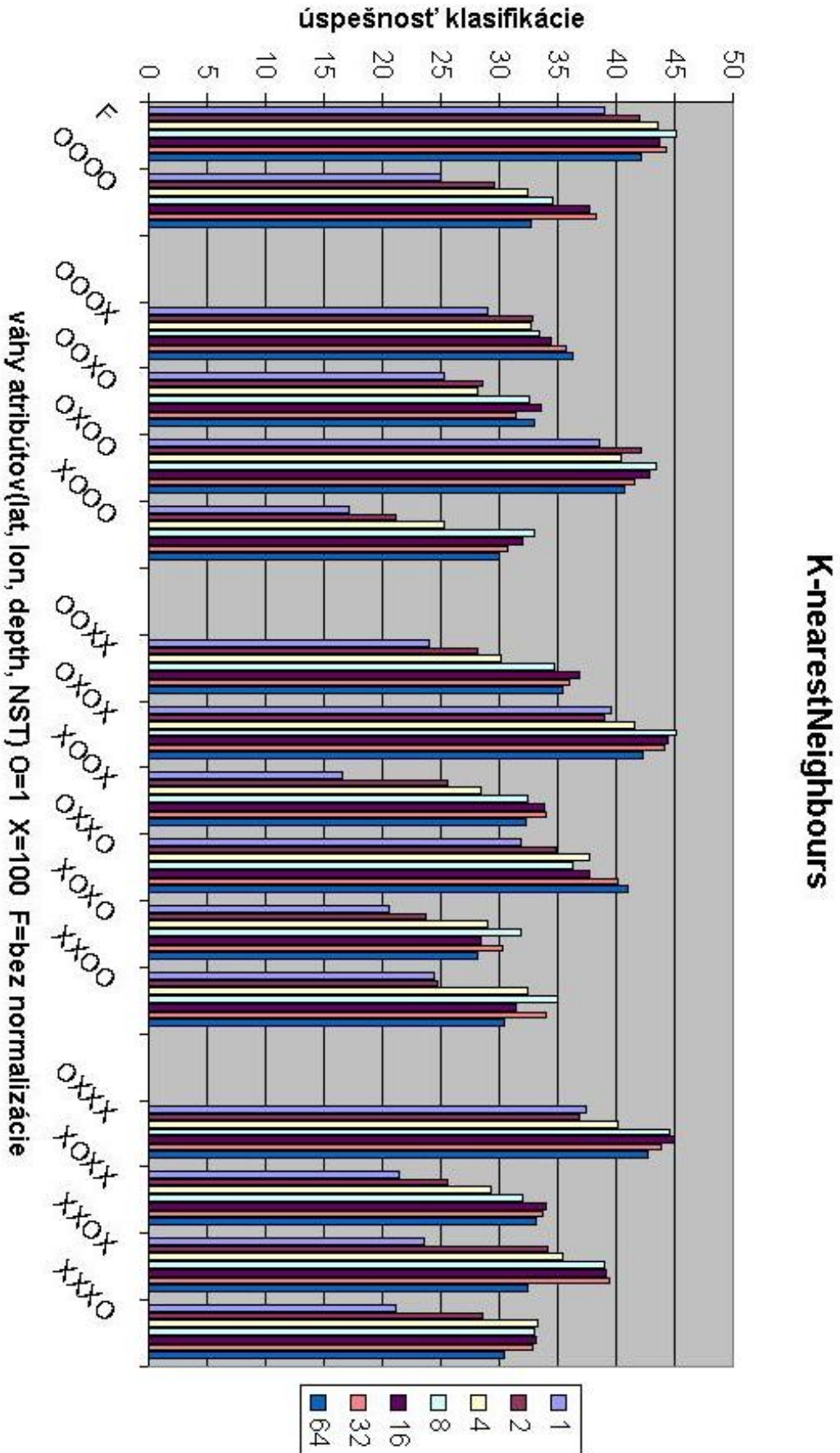
Najlepšie výsledky bez normalizácie sa dosiahli pri počte susedov $k = 8$, pri normalizácii s počtom susedov $k = 32$.

Na rozdiel však ako pri klasifikácii cez metódu ID3, faktor nerovnomerného zastúpenia hodnôt cieľového atribútu *Magnitude* nespôsobuje taký výrazný pokles presnosti predikcie tried s menším zastúpením hodnôt vo vstupnej množine vzoriek. Pokles je badateľný až od hodnoty $k = 32$.

Porovnanie prináša nasledujúca **tabuľka(6.4)** úspešnosti klasifikácie v % pre jednotlivé triedy cieľového atribútu, v závislosti od parametra počtu susedov k. Atribúty sú bez normalizácie:

magnitude	zastúpenie v %	k=4	k=8	k=16	k=32
0.5	0	-	-	-	-
1.0	4.26	0.0	0.0	0.0	0.0
1.5	36.42	47.43	46.1	46.92	46.25
2.0	27.34	28.77	29.58	29.28	29.2
2.5	14.29	14.5	18.79	17.24	11.11
3.0	6.62	10.71	13.79	9.52	20.0
3.5	2.27	50.0	44.44	25.0	0.0
4.0	1.61	100.0	0.0	0.0	0.0
4.5	1.51	26.67	27.27	33.33	0.0
5.0	3.22	51.22	40.91	47.27	40.91
5.5	1.51	42.86	26.67	100.0	0.0
6.0	0.19	50.0	66.67	66.67	0.0
6.5	0.19	0.0	0.0	0.0	0.0

Tabuľka 6.4: K-sused-klasifikácia v triedach


úspešnosť klasifikácie
0
5
10
15
20
25
30
35
40
45
50
K-nearestNeighbours
váhy atribútov(lat, lon, depth, NST) O=1 X=100 F=bez normalizácie
F
OOOO
OOOX
OOXO
OXOO
XOOO
OOXX
OXOX
XOOX
OXXO
XOXO
XXOO
OXXX
XOXX
XXOX
XXXO
1
2
4
8
16
32
64

Kapitola 7

Záver

Cieľom práce bolo popísať základné techniky a metódy dataminingu, podať pohľad na túto oblasť informatiky a cez experimentálne porovnanie dvoch klasifikačných algoritmov uviesť čitateľa aspoň orientačne do problematiky dolovania dát v praxi.

Práca podáva prehľad základných metód dataminingu, oboznamuje detailnejšie s princípmi, potrebnou teóriou a algoritmami základných klasifikačných a segmentačných metód.

Z experimentu predikcie magnitúdy zemetrasenia na základe jej zemepisnej šírky, dĺžky, hĺbky pod zemským povrchom a počtu staníc, ktoré zemetrasenie zachytili, vyšli obe porovnávané metódy klasifikácie- *klasifikácia cez rozhodovací strom* a *klasifikácia pomocou k-najbližších susedov* porovnateľne úspešne.

Metóda klasifikácie cez rozhodovací strom sa ukázala presnejšia pri predikcii jednotlivých tried cieľového atribútu(magnitúdy)**(tabuľka 6.2)**, avšak viac citlivá na parameter určujúci minimálny počet prvkov v listoch(leaf-size). Pre presnejšiu predikciu jednotlivých tried cieľového atribútu bol potrebný hlboký rozhodovací strom(malá hodnota leaf-size).

Metóda klasifikácie cez k-najbližších susedov spolu s rôznymi konfiguráciami váh vstupných atribútov zase priniesla zaujímavý pohľad na signifikantnosť jednotlivých atribútov pre predikciu magnitúdy v našom experimente **tabuľka(6.3)**. Atribút zemepisná dĺžka(Longitude) spolu s atribútom počtu reportovacích staníc(NST) sa ukázali ako signifikantné pre našu klasifikáciu. Atribút zemepisná šírka(Latitude) vykazoval zase rušivý vplyv pre presnosť klasifikácie.

Tieto zistenia samozrejme nemusia platiť všeobecne, môžu byť časovo a

priestorovo špecifické pre našu skúmanú vzorku dát. Avšak aj to je jeden z reálnych problémov v oblasti dolovania dát, s ktorým sa v praxi čitateľ stretne.

Pevne verím, že práca splnila svoje vytýčené ciele a čitateľovi neznalému problematiky dolovania dát objasnila a poskytla prehľad v tejto zaujímavej oblasti informatiky.

Literatúra

[HK00] Jiawei Han and Micheline Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, 2000.

[Mit97] Tom M. Mitchell. *Machine learning*. McGraw-Hill, 1997.

[Par03] Ján Paralič. *Objavovanie znalostí v databázach*. Technická univerzita v Košiciach, 2003.

[WF05] Ian H. Witten and Eibe Frank. *DATA MINING, Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers, 2005.

Príloha

K bakalárskej práci prikladám CD, na ktorom sa nachádzajú datasety použité pri experimente. Ide o dataset trénovacej množiny vzoriek *eqs7day-M1_31_5_08*, a dataset skúmanej množiny vzoriek *eqs7day-M1_8_6_08*, kde robíme predikciu magnitúdy.

Zároveň sa na CD nachádza aj voľne šíriteľná verzia prostredia RapidMiner, v zmysle licencie *GNU GENERAL PUBLIC LICENSE*. Prostredie RadidMiner je voľne k dispozícii v community edition aj na web stránke *RapidMiner*- http://rapid-i.com/content/blogcategory/10/69/