ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

PROVOZNĚ EKONOMICKÁ FAKULTA

Využití vybraných statistických metod při zpracování dat technikami Data mining

disertační práce

Autor: **Ing. Dagmar Bínová**

Školitel: **Doc. RNDr. Bohumil Kába, CSc., katedra statistiky**

Praha 2006

OBSAH

1 ÚVOD 3

2 CÍLE DISERTAČNÍ PRÁCE 5

3 PŘEHLED O SOUČASNÉM STAVU PROBLEMATIKY 6

3.1 Data mining 6

3.1.1 Charakteristiky technik Data mining 6
3.1.2 Technologie dolování dat 11
3.1.3 Typy úloh a metody pro jejich řešení 16
3.1.4 Aplikace Data mining 18

3.2 Data pro použití technik Data mining 26

3.2.1 Typy dat 26
3.2.2 Zdroje dat 26
3.2.3 Velikost datového souboru 29

3.3 Příprava dat pro Data mining 30

3.3.1 Získání dat 31
3.3.2 Vytvoření sady dat 31
3.3.3 Kontrola dat 31
3.3.4 Výběr a transformace proměnných 34
3.3.5 Rozdělení datového souboru 37

3.4 Vybrané metody 38

3.4.1 Metody shlukové analýzy 38
3.4.2 Další vícerozměrné statistické metody 61
3.4.3 Nestatistické metody 68

4 ZVOLENÉ METODY ZPRACOVÁNÍ 72

4.1 Příprava dat pomocí programování v systému SAS 72

4.2 Shlukování pomocí programování 76

4.2.1 Procedura CLUSTER 76
4.2.2 Procedura FASTCLUS 77

4.3 Shlukování v Enterprise Miner 81

4.3.1 Metodologie SEMMA 81
4.3.2 Uzly pro výběr - SAMPLE 82
4.3.3 Uzly pro průzkum – EXPLORE 83
4.3.4 Uzly pro úpravu - MODIFY 85
4.3.5 Uzly pro modelování - MODEL 86
4.3.6 Uzly pro vyhodnocování - ASSESS 88
4.3.7 Ostatní typy uzlů 89
4.3.8 Nové uzly v SAS Enterprise Miner 5.1 90

4.4 Uzel Clustering 91

4.4.1 Okno Data (Data) 92
4.4.2 Okno Variables (Proměnné) 94
4.4.3 Okno Clusters (Shluky) 95

4.4.4 Okno Seeds (Středy) 98
4.4.5 Okno Missing Values (Chybějící hodnoty) 101
4.4.6 Okno Output (Výstup) 102
4.4.7 Prohlížeč výsledků uzlu Clustering 103

5 VÝSLEDKY DISERTAČNÍ PRÁCE 107

5.1 Příprava dat 107
5.1.1 První hodnocení kvality dat 112
5.1.2 Druhé hodnocení kvality dat 116
5.2 Shlukování pomocí programování v systému SAS 122
5.2.1 Shrnutí údajů o jednom zákazníkovi 122
5.2.2 Shluková analýza průměrných údajů za zákazníky 123
5.2.3 Shluková analýza sumarizovaných údajů za zákazníky 132
5.2.4 Shluková analýza neagregovaných záznamů 138
5.3 Shlukování v SAS Enterprise Miner 142
5.3.1 Segmentace zákazníků dle průměrného objemu přenesených dat 142
5.3.2 Shluková analýza agregovaných údajů dle kategorizovaných proměnných 146

6 DISKUSE 152

6.1 Hodnocení přípravy dat 152
6.2 Hodnocení shlukování 153
6.2.1 Hodnocení shlukování pomocí programování 153
6.2.2 Hodnocení shlukování v SAS Enterprise Miner 154
6.3 Klíčové momenty realizace úlohy Data mining 154

7 ZÁVĚR 156

8 POUŽITÁ LITERATURA A ZDROJE 158

9 SEZNAM OBRÁZKŮ 164

10 SEZNAM TABULEK 166

11 SEZNAM GRAFŮ 167

12 PŘÍLOHY 168

12.1 Výsledek shlukování průměrů pomocí programování 168
12.1.1 Shlukování průměrů úplných záznamů 168
12.1.2 Shlukování průměrů nenulových záznamů 170
12.2 Výsledek shlukování v SAS Enterprise Miner 173
12.2.1 Po filtrování a standardizaci směrodatnou odchylkou 173
12.2.2 Při použití kategorizovaných proměnných 176

1 ÚVOD

Uspokojovat potřeby zákazníků rychleji a lépe než konkurence, to je dnes snahou firem, které chtějí získat perspektivní a ziskový segment trhu. Za tímto účelem lze využít moderní metodologii Data mining, která pro obchodní využití odkrývá dříve neznámé souvislosti a vztahy mezi daty. Jedná se tedy o nevšední vytěžování implicitních, dříve neznámých a potenciálně užitečných informací z datových údajů.

Dolování dat umožňuje pomocí speciálních algoritmů automaticky objevovat v datech strategické informace. Je to analytická technika pevně spjatá s datovými sklady jako s velmi kvalitním datovým zdrojem pro tyto speciální analýzy.

Dolování dat lze charakterizovat jako proces extrakce relevantních, předem neznámých nebo nedefinovaných informací z velmi rozsáhlých databází. Důležitou vlastností dolování dat je, že se jedná o analýzy odvozované z obsahu dat, nikoli předem specifikované uživatelem nebo implementátorem, a jedná se především o odvozování prediktivních informací, nikoliv pouze deskriptivních. Dolování dat slouží manažerům k objevování nových skutečností, čímž pomáhají zaměřit jejich pozornost na podstatné faktory podnikání, umožňují testovat hypotézy, odhalují ve stále se zrychlujícím a složitějším obchodním prostředí skryté korelace mezi ekonomickými proměnnými apod. Data mining je orientován na praktickou využitelnost výsledků.

Data mining je relativně nová disciplína, která byla vyvinuta hlavně na základě studií prováděných v jiných disciplínách jako jsou informatika, marketing a statistika. Mnoho z metodologií, jež jsou v metodologii Data mining používány, pochází ze dvou odvětví výzkumu, jednoho rozvinutého ve společenství strojového učení a druhého rozvinutého ve statistickém společenství.

První náznaky aktivit, které se dnes označují jako Data mining, se objevily v 60. letech 20. století s rozvojem počítačové techniky. Šlo například o využívání regresní analýzy s automatickým výběrem proměnných a prvních rozhodovacích stromů. Většinou však šlo jen o ojedinělé nebo akademické záležitosti.

Rozvoj statistických metod, databázových aplikací a umělé inteligence spolu s rychlým růstem rychlosti a paměti počítačů byly předpoklady, které umožnily v sedmdesátých a osmdesátých letech první systematická využití dataminingové metodologie v praxi. Slovní spojení Data mining označovalo „vyzobávání rozinek" z dat, hledání korelací ve velkých datových souborech, které je vystaveno obrovskému nebezpečí, že „objeví" pouze nahodilé fluktuace v datech bez možnosti zobecnění a praktického využití.

Od osmdesátých let 20. století je za základ statistické analýzy považována rostoucí důležitost výpočetních technik. Současně s ní probíhal i vývoj statistických metod pro analýzu vícerozměrných aplikací. Obrat přišel počátkem devadesátých let. V té době začali statistici prokazovat zájem i o strojové učení, což vedlo k důležitému metodickému vývoji. Byly vybudovány metody, umožňující vyhnout se zmíněnému nebezpečí falešných korelací. Navíc rostla poptávka ze strany komerčních organizací, disponujících již velkými objemy dat a neschopných z nich pomocí klasických tabelačních metod získat potřebné podklady pro rozhodování. To napomohlo k rychlému etablování Data miningu jako svébytného oboru aplikované vědy a k jeho širokému použití v komerční praxi.

Časté aplikace jsou především v oblastech přímého marketingu (výběr klientů pro oslovení), v bankovnictví a finančnictví (např. odhadování rizika, hledání podvodů,

k analýze trendů), maloobchodního prodeje (zjišťování asociací, analýza nákupních košíků aj.), telekomunikací (segmentace klientů, prodej programů aj.) a internetového prodeje (analýza přechodů mezi stránkami, efektivity reklamy apod.). Z těchto oblastí využití lze tedy vyvodit široký rozsah přístupů a řešení v různých odvětvích. Dá se očekávat, že oblast vyhledávání znalostí v databázích (KDD) do budoucna poroste s tím, jak porostou požadavky firem na zpracování již shromážděných dat.

Existují různé druhy nástrojů pro dolování dat. Některé z nich jsou určeny specialistům se znalostmi statistiky, některé řídícím pracovníkům. Cílové určení úloh dolování dat je poskytovat strategické informace širokému spektru manažerů v organizaci. To, co odlišuje dolování dat od jiných statistických nástrojů, je právě zaměření na odlišné uživatele. Statistické úlohy dolování dat jsou prováděny automaticky podle určených algoritmů, a tak jejich cílovým uživatelem může být i manažer bez speciálních znalostí statistiky, nikoliv pouze specialista, který návazně zhotovuje reporty pro manažera.

2 CÍLE DISERTAČNÍ PRÁCE

Segmentační analýza transakčních dat je problém z hlediska teoretického, časového a technického velmi náročný. V uplynulých letech mnohé podniky, instituce a organizace shromáždily velmi rozsáhlé databáze a datové sklady. Proces akumulace dat má explozivní charakter a do popředí tedy stále naléhavěji vystupuje otázka, jak se v těchto velkých datových souborech orientovat a jak z nich extrahovat relevantní informace. Přes intenzivní teoretický výzkum zůstává stále nevyřešená a otevřená řada otázek, souvisejících s problematikou shlukování dat, jež jsou v odborné statistické literatuře zmiňovány velmi neúplně.

Cílem této práce je navrhnout postup zpracování transakčních dat pro přípravu datového souboru a vyhodnotit možnosti shlukování pomocí programování a pomocí modulu pro Data mining.

Daná práce se zabývá:

1) popisem a návrhem postupu přípravy dat, navržením originálního programu pro přípravu dat pocházejících z txt souboru,

2) výběrem vhodných statistických postupů a ověřením jejich použitelnosti pro segmentaci velkého objemu dat. Hlavní cíle jsou definovány takto:

a) identifikace a efektivní odstraňování odlehlých pozorování,

b) posouzení iterativního shlukování,

c) chování kritérií se zvyšováním počtu shluků,

d) užití standardizace proměnných transakčních dat stejného typu,

e) sestavování procesů pro shlukování a zadávání jejich parametrů.

Pro získávání vyčerpávajících odpovědí na otázky, zformulované ve výše uvedených cílech, je nezbytné uskutečnit podrobnou empirickou analýzu, zahrnující rozsáhlý soubor reálných statistických dat a mít k dispozici odpovídající výpočetní prostředí. Pro podporu technologií Data mining existuje řada softwarových produktů, které využívají různé statistické respektive výpočetní algoritmy, lišící se efektivitou i robustností.

Z metodologického hlediska mohou být postupy Data mining v širším slova smyslu chápány jako průzkumová analýza rozsáhlých datových souborů. Z toho vyplývá, že daná práce bude metodicky vycházet z modelového paradigmatu, které může být symbolicky prezentováno následující posloupností: problém $\rightarrow$ data $\rightarrow$ analýza $\rightarrow$ model $\rightarrow$ závěry. Cílem takovéto analýzy je zejména umožnit orientaci ve zpracovávaných datech, odhalit jejich zvláštnosti, skryté struktury, případně extrahovat klíčové proměnné.

Praktických řešení se nabízí celá řada, neboť různí výrobci programového vybavení navrhují různá pojetí, jsou vyvíjeny nové algoritmy poskytující rozdílné výsledky. Disertační práce se zaměří na empirické hodnocení provedené na základě postupů používaných v systému SAS – metodologii SEMMA, která zahrnuje moderní postupy řešení.

3 PŘEHLED O SOUČASNÉM STAVU PROBLEMATIKY

Kapitola obsahuje přehled o současném stavu zkoumané problematiky a odborné literatury zaměřené na techniky Data mining, datové zdroje, přípravu dat a vybrané metody. Uvádí možnosti praktického využití technik Data mining s důrazem na vymezení základních vlastností a charakteristik této metodologie. Věnuje se vymezení, pro jaká data jsou tyto techniky vhodné, komu jsou určené, jak prezentují výsledky a jaké používají metody. Z metod se věnuje především metodám shlukové analýzy.

3.1 Data mining

Data mining je vědní disciplína, která vznikla teprve zcela nedávno na rozhraní statistiky, umělé inteligence a databázových systémů [77]. Jedná se o proces extrakce relevantních, předem neznámých nebo nedefinovaných informací z velmi rozsáhlých databází [20]. Data mining (DM) je proces výběru, průzkumu a modelování velkého rozsahu údajů [70]. Českých adekvátních názvů termínu „Data mining" je celá řada: „Analýza dat", „Vyhledávání v datech", „Dolování dat", „Vytěžování dat" [77].

Data mining vychází z předpokladu, že ve velkých databázích jsou ukryty zajímavé a důležité poznatky, které lze vyjádřit jednoduchými tvrzeními, vyjadřujícími příčinné závislosti, klasifikace a jiné vztahy. Některé takové poznatky mohou vést k novým odhalením a objevům. V této souvislosti se proto hovoří o „Vyhledávání znalostí z databází", popř. „Objevování znalostí v databázích" (Knowledge Discovery from Database - KDD) [26]. Jádrem celého procesu dobývání znalostí z databází je použití analytických metod. Tento krok bývá v anglické literatuře nazýván Data mining, modeling nebo analysis [3].

Vzrůstající dostupnost dat v dnešní informační společnosti vedla k potřebě dostatečných nástrojů pro modelování a analýzu. Data mining a aplikované statistické metody jsou vhodnými nástroji pro dobývání znalostí z takových dat. Data mining může být definován jako proces výběru, průzkumu a modelování rozsáhlých databází, aby se objevily modely a schémata, jež jsou apriori neznámé. Tím se především odlišuje od aplikované statistiky; ta se týká aplikací statistických metod na data, kdežto data mining je celým procesem extrakce dat a analýzy zaměřené na tvorbu rozhodovacích pravidel pro dané obchodní cíle. Jinými slovy je Data mining procesem Business intelligence [24].

Business intelligence a Data mining jsou důležitými nástroji v procesu rozhodování zejména pro střední a vyšší management. Stávají se ovšem nepostradatelnými i v dalších odděleních firem, kam si postupně prorážejí cestu [50].

3.1.1 Charakteristiky technik Data mining

Metody Data mining objevují neočekávané zákonitosti implicitně obsažené v datech, projevující se v anomáliích a neobvyklém chování dat z hlediska jejich kvality, kvantity nebo časové změny [51]. Účelem technik Data mining je najít skryté závislosti, které lze využít při obchodním rozhodování. Smyslem je tedy objevit v datech vzory pro poznání jejich významu a pro řešení problémů [89]. Data mining je analytický proces transformace podnikových dat do obchodní informace, která je využita pro zvýšení efektivity a ziskovosti společnosti [57].

Data mining nenahrazuje, ale vhodně doplňuje dosud užívané postupy vyhodnocování hromadných dat. Stále se uplatňují klasické programy pro statistickou analýzu (SAS a SPSS), systémy pro podporu rozhodování a manažerské informační systémy (DSS a MIS, resp. EIS), vícerozměrné tabulkové procesory i neuronové sítě [51]. Ze statistických metod se používají tzv. data driven metody, kam patří shluková analýza, exploratorní analýza (EDA), regresní a jiné stromy [77]. Metody početní EDA zahrnují jak základní jednoduché statistiky, tak i pokročilejší, specifické vícerozměrné vyšetřovací techniky navržené pro vyhledávání závislostí ve vícerozměrných datových souborech [80].

Zvláště v souvislosti s některými vybranými metodami, jako je třeba shluková analýza nebo procedury hledání asociačních pravidel, lze techniky Data mining vnímat spíše jako deduktivní proces v porovnání s běžným postupem, kdy úsudky mají převážně induktivní charakter [26].

Metody pro vyhledávání znalostí z dat přistupují k analýze dat odlišně v porovnání s klasickými metodami statistiky jak v získávání dat, tak ve filozofii přístupu k vytváření modelů. Klasická statistika zpravidla předpokládá, že data jsou vybrána podle známých nebo zvolených principů, lze na ně pohlížet jako na pozorování, která podléhají modelovým zákonitostem a jsou vzorkem dané reality, ze které byla data vybrána. Při použití technik Data mining je nezbytné předchozí očištění zdrojových dat od možných rušivých vlivů, použití analytických nástrojů je často formalizováno, je nutné uplatňovat mechanizmy pro rozpoznání nahodilých modelů nebo nahodilých zákonitostí v datech, způsobených například předem neodhalenou kontaminací [72].

V technikách Data mining nejde v prvé řadě o nalezení přesného modelu, testování významnosti, validaci, interpretaci parametrů. Za úspěch se považuje už nalezení alespoň nějakého netriviálního modelu, relace, pravidla či použitelného predikčního nástroje, které vysvětlí byť jen malou část variability v datech a predikují lépe, než házení mincí. To totiž může znamenat nepatrný, ale rozhodující náskok v konkurenčním prostředí [38].

V současné době přispívá k vývoji nových postupů ve vyhodnocování dat sílící konkurence. Nestačí už jen sbírat informace v datech zřetelně obsažené, ale je nutné objevovat i souvislosti, které nejsou snadno patrné. Klíčovou technologií pro zvýšení kvality a účinnosti rozhodovacích procesů se tedy stává Data mining. Tento postup lze stručně charakterizovat přechodem *Data* → *Informace* → *Znalosti* [51]. Data jsou nejcennější surovinou, ovšem nemají význam, dokud se účelně nezpracují v informace. Pod pojmem informace si lze představit poznatky, které uspokojují konkrétní informační potřebu manažera. Informace jsou totiž základem pro znalosti manažerů i jejich podřízených pracovníků a nelze si bez nich jakoukoliv práci ani představit [89]. Důležitým krokem v celém procesu dobývání znalostí je interpretace a ocenění nalezených znalostí. V případě deskriptivních úloh je hlavním kritériem novost, zajímavost, užitečnost a srozumitelnost. Tyto charakteristiky úzce souvisejí s danou aplikační oblastí, s tím, co přinášejí expertům a koncovým uživatelům. Z tohoto pohledu lze hovořit o:

- zřejmých znalostech, které jsou ve shodě se „zdravým selským rozumem". Odborníkovi na KDD potvrzují, že použitý algoritmus funguje tak, jak má.
- zřejmých znalostech, které jsou ve shodě se znalostmi experta z dané oblasti. Takovéto znalosti nepřinášejí nic nového, ale ukazují expertovi, že použitá metoda je schopna objevovat v datech znalosti.

- nových zajímavých znalostech, které přinášejí nový pohled. Jsou to ideální znalosti, které expert hledá.
- znalostech, které musí expert podrobit analýze, neboť není zcela jasné, co znamenají.
- „znalostech", které jsou v rozporu se znalostmi experta [3].

Imperativem dneška jsou strategické znalosti, založené na efektivním využívání firemních dat, na základě kterých lze dělat účinná rozhodnutí pro progresivní růst stanovených cílů organizace [91].

Primárně jde o to uvědomit si rozdíl mezi operativními daty přicházejícími z transakčních systémů, jako ERP, CRM nebo SCM, a jejich nutnou přeměnou v analytická data, která dávají těmto datům rozměr informací a přidanou hodnotu pro strategické rozhodování společnosti. Jinými slovy, je potřeba připravit organizaci na implementaci Business intelligence jako řešení, které přidává organizaci skutečnou konkurenční výhodu [61].

Metody získávání znalostí z databází mají zpřístupnit nové, dosud neznámé znalosti uživatelům na základě dosavadních znalostí a nových informací z dat. Analyzovaná data by měla být relevantní danému problému. Při analýze je nutné uvažovat proměnlivé prostředí a výsledky analýzy musí být prakticky využitelné. Tyto požadavky zvyšují nároky na integraci znalostí o dané problematice s možnostmi technologických prostředků analýzy, což zpravidla vyžaduje spolupráci odborníků různého zaměření. Výsledkem analýzy mohou být postupy, využitelné jako standardy pro extrakci znalostí, nebo analytické modely [72].

Tabulka 1: Některé rozdíly mezi statistikou a procesy DM - KDD [72]

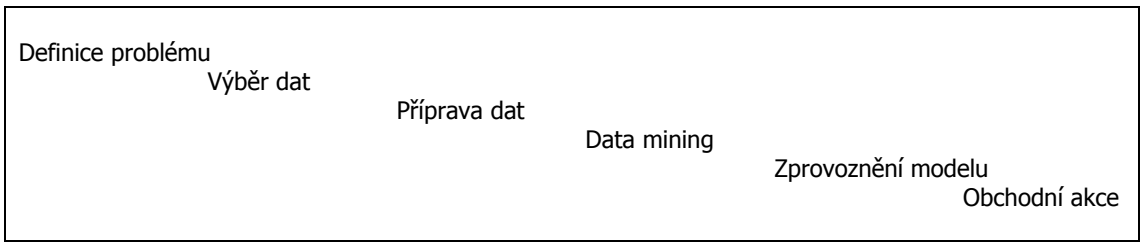
Statistika	**DM – KDD**
Data získána cílené v definovaných podmínkách tak, aby odpověděla na dané cíle (hypotézy).	Data získána z datových skladů, operativních záznamů o dějích, retrospektivně, nemusí obsahovat požadované informace (nejsou-li průkazná, je nutno rozpoznat), nemusí být zjišťována jednotným způsobem, před analýzou je nutné očištění.
Použité statistické metody závisí na hypotézách a na tom, zda data mají nebo nemají očekávané vlastnosti. Jejich volba bývá součástí plánu zjišťování.	Použití analytických nástrojů je formalizováno, součástí jsou mechanismy nebo znalosti uživatele, umožňující identifikaci nesprávně zvolených dat nebo nevhodně použitých analytických metod.
Data nejčastěji rozsahu řádově 10^2 -10^5.	Nejsou neobvyklá data rozsahu až 10^9 a více.
Data jsou vzorkem dané reality v daném čase, ve kterém byla získána.	Je možný vývoj vztahů v datech v průběhu času nebo v důsledku geografických rozdílů (změna modelu, změna parametrů), cílem může být identifikace nebo popis změn.
V rámci současných technologií je většina algoritmů řešitelná statistickými algoritmy pro data daných rozsahů.	Řešení problému „analyzovatelnosti" celých dat (scalability), pokud jsou velmi rozsáhlá. Problém se řeší analýzou „po částech", může mít nepříznivé důsledky.
Data ve tvaru dvourozměrného pole v jednotném formátu.	Data distribuovaná, různé formáty při přenosu mezi aplikacemi, další technické problémy.
Měření hodnot sledovaných proměnných sjednoceno výzkumným plánem.	Data nemusí být sbírána ani zaznamenávána jednotným způsobem. Příprava vhodných dat je součástí procesu (zpravidla 70 – 80 % celkové doby pro DM).
Odlehlých pozorování je početně málo: identifikace, často vyloučení z analýz.	Malé relativní zastoupení odlehlých hodnot může být nezanedbatelné v absolutním počtu, nutno je zvlášť analyzovat.
Rozpoznání náhodných vztahů je možné v rámci metod klasické statistiky.	Pro rozpoznání nahodilých modelů a nahodilých zákonitostí jsou nutné specifické metody a multioborový přístup k řešení problému.
Teoretický rozvoj často předchází	Priority vědeckého bádání jsou často dány požadavky praxe (vývoj

Statistika	DM – KDD
praktickému užití. Výsledky jsou podloženy teoretickými principy, platnými za daných předpokladů.	efektivnějších algoritmů apod.) a často pouze empiricky ověřeny. Převaha empirických výsledků nad teoretickými. Je nebezpečí přeceňování významu nových metod z komerčních důvodů.

3.1.1.1 Proces Data mining

Jednotlivé kroky procesu dobývání znalostí jsou různě časově náročné a mají i různou důležitost pro úspěšné vyřešení dané úlohy. Praktici v oboru uvádějí, že nejdůležitější je fáze porozumění problému (80 % významu, 20 % času) a časově nejnáročnější je fáze přípravy dat (80 % času, 20 % významu) [2]. Většina nákladů na projekty Data mining jsou investice do přípravy a integrace dat [50]. Poměrně málo práce zaberou vlastní analýzy [2].

Následující schéma znázorňuje **základní kroky procesu Data mining**:



Obrázek 1: Schéma procesu Data mining [57]

1. Definice problému

Prvním krokem v procesu je definice obchodního problému nebo příležitosti, na kterou se máme zaměřit.

Úspěšná iniciativa Data mining je vždy zahájena dobře definovaným projektem. Pro ověření, že bude vytvořena určitá nová hodnota, by mělo být zahrnuto vyhodnocení status quo v dané oblasti. V této fázi lze také shrnout přehled o technologiích, organizačních a obchodních procesech, což umožní navrhnout zvýšení hodnoty vůči stávajícím postupům.

2. Výběr dat

Poté, co je definován problém, musí být definovány zdroje dat. Avšak ne každý zjištěný datový zdroj je pro řešení vhodný. Data jsou obvykle extrahována ze zdrojových systémů nebo datových skladů na zvláštní server, kde je realizován Data mining.

3. Příprava dat

Příprava dat je časově nejnáročnější částí každého projektu dolování dat, vyžaduje až 80 % celkových zdrojů. Data mining vyžaduje, aby data, která budou analyzována, byla připravena do podoby jednoduché tabulky (každý záznam, který bude modelován, obsahuje mnoho sloupců). Tato metodologie umožňuje vytvoření stovek a občas i tisíců proměnných, které budou vstupovat do modelování.

Tato projektová fáze je nejkritičtější - výsledné modely jsou tak dobré, jak dobrá jsou data, která jsou použita pro jejich vytvoření. Expertiza v oblasti Data mining spočívá nejvíce v tom, aby reprezentace podrobných dat měla formu odpovídající všem aspektům řešeného obchodního problému.

Významné zlepšení výsledků může být dosaženo zlepšením metodologie přípravy dat.

4. Dataminingové analýzy

Tato fáze zahrnuje využití statistických a nestatistických nástrojů pro vytvoření matematických modelů. Tato fáze je typicky nejkratší a nejjednodušší částí jakéhokoli Data mining projektu. Většina organizací, která zaměstnává analytiky, je schopna si v tomto směru postupně vystačit i sama.

Data mining se typicky realizuje na serveru, který je oddělený od datového skladu nebo jiných informačních systémů společnosti. Některé společnosti dokonce vytvářejí modely na počítačích PC s využitím vzorkování dat.

5. Zprovoznění modelu (Deployment)

Zprovoznění modelu je proces, kdy se matematické modely implementují do operačního systému, aby mohly být využity ke zlepšení obchodních výsledků.

6. Obchodní akce

Tato fáze zahrnuje využití zprovozněných modelů pro zajištění zlepšených výsledků v rámci identifikovaného obchodního problému nebo příležitosti [57].

3.1.1.2 Text mining a web mining

Při dobývání znalostí se v poslední době objevují i nové oblasti aplikací. Mezi dnes velmi populární oblasti patří dobývání znalostí z textu tzv. *text mining* a dobývání znalostí z webu tzv. *web mining*. Do budoucna se očekává i rozvoj oboru *multimedia mining*, tedy dobývání znalostí z multimediálních dat, kombinujících texty, obrázky, zvuky, videosekvence apod.

Dobývání znalostí z textů lze chápat jako speciální typ úlohy dobývání znalostí z databází. Zatímco u databází se pracuje s údaji uloženými v pevné struktuře, zde se pozornost věnuje nestrukturovanému textu. Hlavním problémem je, jak vhodně reprezentovat textový dokument, aby bylo možné použít některý z algoritmů.

Dobývání znalostí z webu soustřeďuje svoji pozornost na službu world wide web. Rozlišuje se:

- dobývání znalostí na základě obsahu webu (web content mining),
- dobývání znalostí na základě struktury webu (web structure mining),
- dobývání znalostí na základě používání webu (web usage mining) [3].

3.1.2 Technologie dolování dat

Úlohy dolování dat mohou být realizovány rozmanitými technologiemi, často i kombinací různých technologií. Část procesu Data mining je typicky realizována nástroji využívajícími statistickou a nestatistickou analýzu dat. Dělení nástrojů na statistické a nástroje Data mining není samoúčelné. Statistické produkty poskytují většinu potřebné funkcionality se zaměřením na „klasičtější" statistické metody a mají příznivější cenu, ovšem je náročnější s nimi pracovat. Produkty Data mining mají velmi intuitivní uživatelské rozhraní, díky kterému je práce s nimi velmi efektivní, obsahují některé nestatistické metody, které nevyžadují hluboké statistické znalosti pro svou parametrizaci, a umožňují automatizaci výsledných modelů oběma výše popsanými postupy [45].

3.1.2.1 Programové vybavení

Aplikace pokročilých statistických metod je vázána na kvalitní statistické programové vybavení. Systémy pro dobývání znalostí nabízejí jak malé firmy vzešlé z akademického prostředí, tak význační producenti statistického softwaru [2]. Ve světě se k nejrozšířenějším řadí systém SAS s produktem Enterprise Miner, systém SPSS s produktem Clementine (a AnswerTree), statistický balík Statistica (Data Miner) a S-plus (Insightful Miner) [3].

Kromě toho existuje řada dalších produktů, například DBMiner kanadské firmy DBMiner Technology, DB2 Intelligent Miner firmy IBM, KnowledgeSTUDIO kanadské firmy ANGOSS, či GhostMiner firmy FQS Poland. Existuje i přídavný program k tabulkovému kalkulátoru Microsoft Excel, který má název XLMiner a jehož tvůrcem je firma Cytel [29].

Do sféry produktů pro analýzu dat postupně pronikají i výrobci databází jako Microsoft, Oracle či NCR, zatím ovšem nedosahují šíře metod dostupných ve specializovaných aplikacích a jsou spíše příslibem do budoucnosti [45].

Tabulka 2 uvádí některé systémy pro dobývání znalostí z dat.

Tabulka 2: Systémy pro Data mining [upraveno podle 3]

Systém	Výrobce	URL
CART	Salford Systems	http://www.salford-systems.com
Clementine	SPSS	http://www.spss.com
DataEngine	Management Intelligenter Technologien GmbH	http://www.dataengine.de/
Enterprise Miner	SAS Institute	http://www.sas.com
Intelligent Miner	IBM	http://www.ibm.com/
KnowledgeSTUDIO	Angoss	http://www.angoss.com
LISP Miner	VŠE	http://lispminer.vse.cz
MineSet	Purple Insight (dříve Silicon Graphics)	http://www.purpleinsight.com/
PolyAnalyst	Megaputer Intelligence Inc.	http://www.megaputer.com/
See5	RuleQuest Research	http://www.rulequest.com/see5-info.html
Statistica Data Miner	StatSoft	http://www.statsoft.com
Weka	University of Waikato	http://www.cs.waikato.ac.nz/~ml/weka
WizWhy	WizSoft	http://www.wizsoft.com/
GhostMiner	Fujitsu	http://www.fqs.pl/

Uvedený přehled nezahrnuje všechny systémy pro dobývání znalostí. Je jisté, že v současné době totiž nelze hovořit o nějakém standardním, všeobecně používaném systému. Problémem tedy často je, který systém vybrat. Svou roli hraje jak univerzálnost a specifičnost systému, tak cena [3].

Kromě stanovení cíle analýzy je třeba rozhodnout, která metoda bude pro dosažení tohoto cíle využita, případně ve kterém programovém systému. Nejvíce možností má uživatel v oblasti klasifikace a predikce, méně jsou zastoupeny metody pro shlukování. Statistické postupy pro shlukování ve velkých datových souborech jsou zařazovány zatím zřídka, řešením je tedy použití neuronových sítí, případně genetických algoritmů [29].

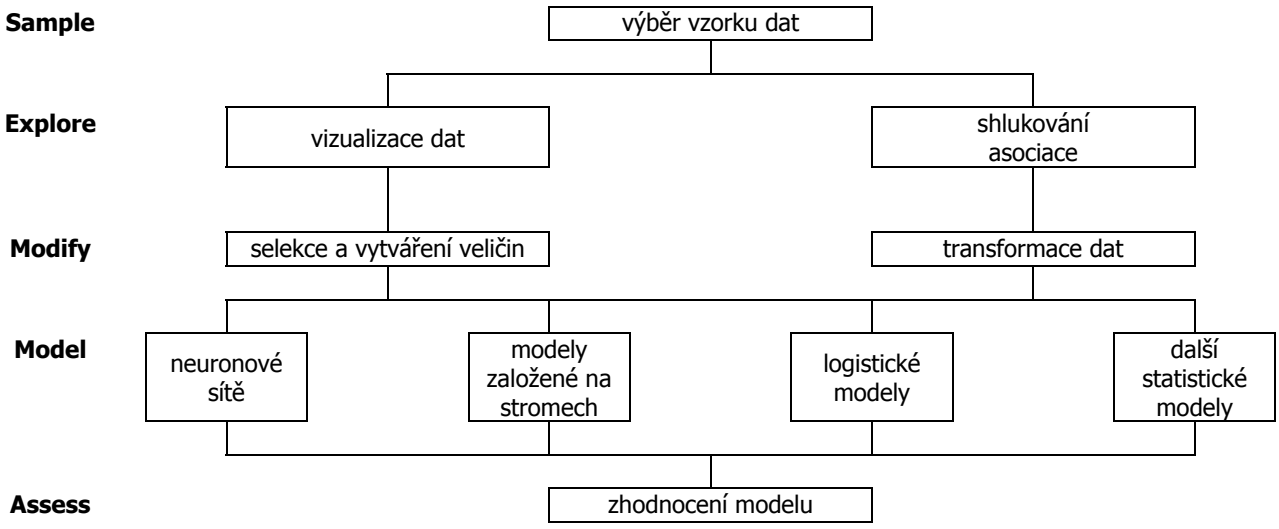
3.1.2.2 Systém SAS

Základ systému SAS tvoří části s názvem SAS/BASE a SAS/STAT. Jejich funkčnost může být uživateli zpřístupněná pomocí aplikací s interaktivním rozhraním SAS/Assist, Analyst, Market Research či grafickým uživatelským rozhraním SAS/Enterprise Guide. Mezi nástroje na realizaci vícerozměrné analýzy s dodatečnou podporou pro grafickou prezentaci dat a výsledků patří modul SAS/Insight. Pro účely analýzy časových řad a prognózování jejich budoucího vývoje se používá modul SAS/ETS s uživatelským rozhraním Time Series Forecasting System a Time Series Viewer. Pro grafickou vizualizaci se využívá SAS/GRAPH. Systém SAS používá programovací jazyk, který je dostupný v SAS/IML.

Mezi pokročilejší aplikace dalších modulů a řešení systému SAS se řadí SAS/Enterprise Miner, SAS/Text Miner, SAS High-Performance Forecasting Software, SAS/AF, SAS/EIS, SAS/OR, SAS/QC, OROS ABC/M software. Pro Business inteligenci a budování datových skladů je možné pořídit moduly SAS Marketing Automation Solution, SAS Risk Dimension, SAS Financial Management Solutions, SAS/Warehouse Administrator, SAS/IntrNet, SAS/Access atd [76].

SAS Enterprise Miner

Enterprise Miner je produkt firmy SAS Institute. K jeho nejpropracovanějším postupům patří statistické metody, které využívají již implementované procedury.

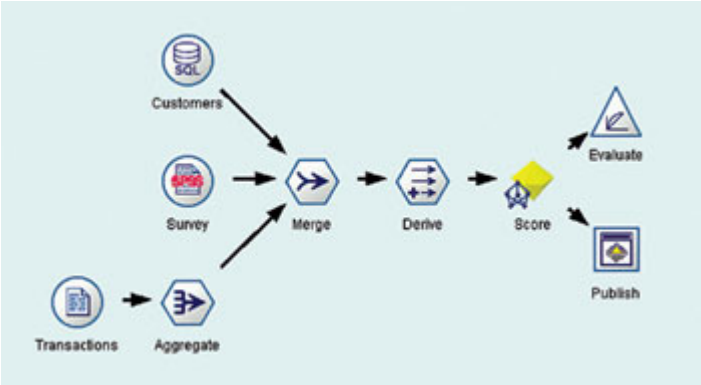


Obrázek 2: Metodika SEMMA [3]

Enterprise Miner použité metody integruje a nabízí uživatelsky příjemnější prostředí, než je příkazový jazyk (kód SAS) [3]. Jedná se o nástroj pro dolování dat implementující SEMMA metodologii (Sample, Explore, Modify, Model, Assess). Jednotlivé kroky zahrnují výběr statisticky reprezentativních souborů z dat, aplikaci exploratorních statistických a vizualizačních technik, výběr a transformaci nejdůležitějších proměnných, tvorbu modelu a potvrzení správnosti modelu [65].

3.1.2.3 SPSS Clementine

Systém Clementine od SPSS má velice propracovaný způsob ovládání, tzv. *vizuální programování* (vizual programming). Z nástrojů v jednotlivých paletách se na pracovní ploše poskládá sekvence řešení úlohy (stream). Clementine nabízí analytikům modul pro přidávání vlastních algoritmů a koncovým uživatelům modul pro přenesení provedené analýzy [3]. Clementine provádí dataminingové analýzy za použití metodologie CRISP-DM (CRoss-Industry Standard Process for Data Mining), která uvádí tyto dílčí kroky procesu dobývání znalostí: porozumění problematice (Business understanding), porozumění datům (Data understanding), příprava dat (Data preparation), modelování (Modeling), vyhodnocení výsledků (Evaluation) a využití výsledků (Deployment) [2].

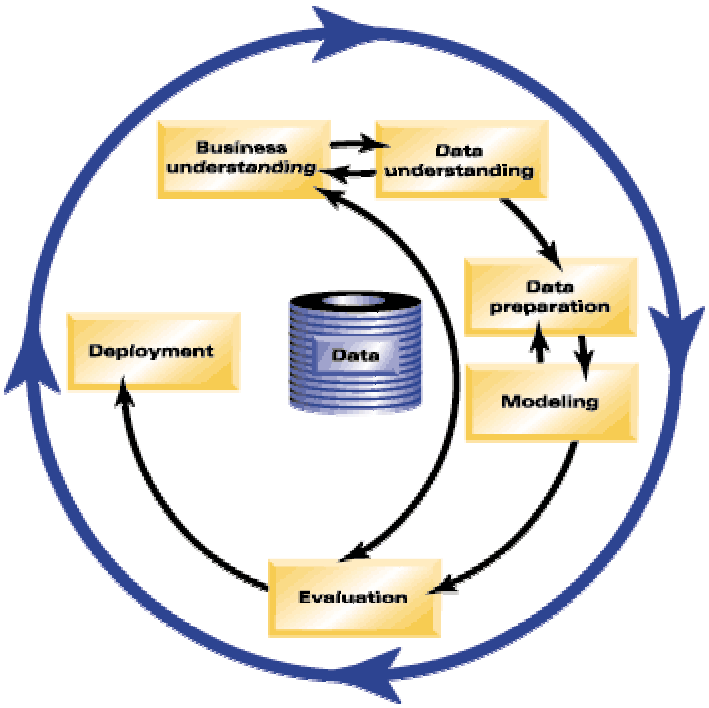


Obrázek 3: Vizuální programování v produktu Clementine od SPSS

Jednotlivé fáze zahrnují definování problému v dané oblasti, náhled do struktury dostupných dat, přípravu dat, modelování, ověření kvality modelu a jeho sdílení. Celý proces znázorňuje následující obrázek. Jednotlivé fáze zachycené v diagramu jsou dále detailněji rozpracovány [75].

Porozumění problematice je úvodní fáze zaměřená na pochopení cílů projektu a požadavků na řešení formulovaných z uživatelského hlediska. Tato uživatelská formulace musí být převedena do zadání úlohy pro dobývání znalostí z databází [2].

Fáze *porozumění datům* začíná prvotním sběrem dat. Následují činnosti, které umožní získat základní představu o datech, která jsou k dispozici (posouzení kvality dat, první „vhled" do dat, vytipování zajímavých podmnožin záznamů v databázi...). Obvykle se zjišťují různé deskriptivní charakteristiky dat (četnosti hodnot různých atributů, průměrné hodnoty, minima, maxima apod.), s výhodou se využívají i různé vizualizační techniky [2].



Obrázek 4: Metodologie CRISP-DM

Příprava dat zahrnuje činnosti, které vedou k vytvoření datového souboru, který bude zpracováván jednotlivými analytickými metodami. Tato data by tedy měla obsahovat údaje relevantní k dané úloze a mít podobu, která je vyžadována vlastními analytickými algoritmy [2].

Analytické metody použité ve fázi *modelování* zahrnují algoritmy pro dobývání znalostí. Obvykle existuje řada různých metod pro řešení dané úlohy, je tedy třeba vybrat ty nejvhodnější (doporučuje se použít více různých metod a jejich výsledky kombinovat) a vhodně nastavit jejich parametry. Jde tedy opět o iterativní činnost (opakovaná aplikace algoritmů s různými parametry), navíc použití analytických algoritmů může vést k potřebě modifikovat data, a tedy k návratu k datovým transformacím z předcházející fáze [2].

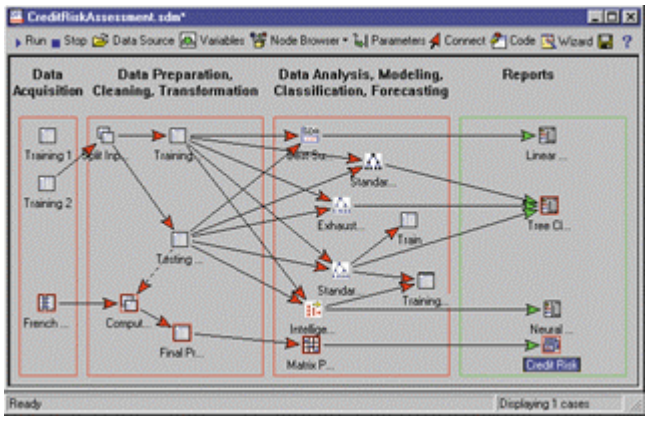
Ve fázi *interpretace* se dosažené výsledky vyhodnocují z pohledu uživatelů, tedy z pohledu, zda byly splněny cíle formulované na počátku projektu [2].

Vytvořením vhodného modelu celý projekt obecně nekončí. Dokonce i v případě, že řešenou úlohou byl „pouze" popis dat, je třeba získané znalosti upravit do podoby použitelné pro podporu rozhodování. Podle typu úlohy může tedy *využití* (nasazení) výsledků na jedné straně znamenat prosté sepsání závěrečné zprávy, na straně druhé pak zavedení (hardwarové, softwarové, organizační) systému pro automatickou klasifikaci nových případů [2].

3.1.2.4 STATISTICA Data Miner

Statistica Data Miner je dalším příkladem systému pro dobývání znalostí, který vyvinula firma specializovaná na statistické programy. Systém opět pokrývá proces dobývání znalostí počínaje přípravou dat a využitím výsledků konče. Sekvence zpracování dat se tvoří na pracovní ploše z jednotlivých nástrojů – uzlů. Uživatel pracuje s uživatelským

rozhraním „drag-and-drop“, které se vyznačuje přizpůsobivostí, upravitelností dle požadavků zákazníka a poskytuje jednoduchý přístup k základním skriptům.



Obrázek 5: Data Miner - sekvence kroků

Techniky vytěžování dat jsou v Data Mineru založeny na výkonných nástrojích obsažených v pěti modulech, které lze používat interaktivně nebo pro výstavbu, testování a zavedení nových nástrojů řešení.

STATISTICA Data Miner obsahuje pestrý výběr metod Data mining např. výběr shlukovacích technik, architekturu neuronových sítí, klasifikační/regresní stromy, vícerozměrné modelování a mnoho dalších prediktivních technik; velký výběr grafických a vizualizačních procedur. SW je optimalizován pro zpracování extrémně velkých datových souborů (přes milion proměnných, stratifikované či prosté výběry záznamů).

Z výsledných modelů lze vygenerovat spustitelný kód v jazycích Visual Basic, C++, C#, Java, atd. Visual Basic je přímo součástí systému. Výsledky lze uspořádat do výstupních sestav (report), jako spreadsheety, grafy, atd. nebo je uveřejnit jako web [79], [3].

3.1.3 Typy úloh a metody pro jejich řešení

V různých odborných publikacích a v praxi existuje mnoho členění typů úloh dolování dat. Novotný a spol. používají dělení uvedené v publikaci „Principles of Data Mining". V ní se úlohy v dolování dat člení na:

- **Explorační analýzy dat** - podstatou je prozkoumat data bez předcházející znalosti, která by určitým způsobem hledání usměrňovala. Využívají se zde různé grafické metody či speciální techniky.
- **Deskriptivní úlohy** - podstatou je určitým způsobem popsat celou datovou množinu. Z hlediska dolování dat je například takovou metodou shlukování, při kterém dochází k vytvoření skupin, do kterých se dají projevy v datech rozdělit.
- **Prediktivní úlohy** - cílem je předpovědět hodnotu určité veličiny na základě znalosti hodnot ostatních veličin. Z hlediska statistiky je takovou metodou regresní analýza. Predikci v dolování dat lze provádět zejména klasifikací příkladů do tříd.
- **Hledání vzorů a pravidel** (hledání nuggetů) - podstatou je hledání určitých vztahů a vzorů chování v datech. Klasickou úlohou je zde analýza nákupního košíku, která má rozkrýt, které druhy zboží jsou zákazníky kupovány současně. Dalším takovým příkladem může být úloha z oblasti bankovnictví, spočívající v detekci vzorů implikujících provádění operací praní špinavých peněz.
- **Hledání podle vzorů** - před prováděním hledání znalostí podle vzorů má analytik k dispozici určitý vzor a cílem je nalézt v datech vzory, shodující se nebo podobné s touto předlohou. Jedná se tedy o rozpoznávání vzorů v datech na základě předem definované šablony. Tyto typy úloh se realizují v oblasti rozpoznávání obrázků a textů. Například při rozpoznávání textů je k dispozici vzorový informační vektor vyjadřující daný text. Při aplikaci tohoto typu úloh se potom porovnávají ostatní informační vektory s reprezentantem a vyhodnocuje se jejich podobnost, například na základě metod podobnosti vektorů [45].

Řezanková uvádí, že „existují jednak různé typy úloh, které je možno řešit, jednak různé postupy, které lze při řešení použít". Základní klasifikace je uvedena v následující tabulce.

Tabulka 3: Přehled úloh a metod při technikách Data mining [55]

Úloha	Metoda
Klasifikace	Diskriminační analýza Logistická regresní analýza Klasifikační (rozhodovací) stromy Neuronové sítě (algoritmus "back propagation")
Odhady hodnot vysvětlované proměnné	Lineární regresní analýza Nelineární regresní analýza Neuronové sítě (RBF - "radial basis function")
Segmentace (shlukování)	Shluková analýza Genetické algoritmy Neuronové shlukování (Kohonenovy mapy)
Analýza vztahů	Asociační algoritmus pro odvozování pravidel typu If X, then Y
Predikce v časových řadách	Boxova-Jenkinsova metodologie Neuronové sítě ("recurrent back propagation")
Detekce odchylek	Vizualizace Statistické postupy

Kupka uvádí, že při porovnání různých metod se ukazuje, že v efektivitě analýzy mírně převažují statistické metody (jsou obvykle stabilnější) a že automaticky se učící algoritmy představují obecně spíše nevýhodu a větší nebezpečí chybné interpretace [38].

3.1.3.1 Techniky dolování dat

Úlohy dolování dat je možno řešit s použitím celé řady technik. Mezi nejdůležitější techniky dolování dat patří:

- **Analýza nákupního košíku** (Market Basket Analysis) - je speciální formou clusteringu (detekce shluků) používanou k vyhledání skupin a prvků, které mají tendenci vyskytovat se pospolu (v jedné transakci). Analýza nákupního košíku hledá opakující se nákupní košíky a popisuje je prostřednictvím implikačních pravidel.
- **Dedukce** (Memory-Based Reasoning) - technika, která využívá známé skutečnosti jako model k predikci neznámých skutečností. Dedukce sleduje nejbližší okolí známých instancí a kombinuje jejich hodnoty za účelem odhadu predikovaných hodnot.
- **Detekce shluků** (Cluster Detection) - vytváří modely identifikující datové záznamy, které jsou si navzájem podobné. Detekce shluků nevychází z předem definovaných skupin charakteristiky shluků, i jejich počet vyhledává na základě podobnosti zkoumaných dat.
- **Analýza závislostí** (Link Analysis) - oproti výše uvedeným technikám analýza závislostí nezkoumá prvky na základě jejich vlastností, ale zaměřuje se na vztahy mezi prvky. Jedná se o aplikaci teorie grafů.
- **Rozhodovací stromy a indukce** (Decision Trees and Rule Induction) - představují výkonné modely, které jsou výstupem statistických a nestatistických metod, např. klasifikační a regresní stromy (CART), chí-kvadrát automatická indukce (CHAID), kritéria informační entropie apod. Rozdělují záznamy v tréninkových sadách dat do disjunktních skupin, kde každá skupina může být popsána pomocí jednoduché množiny pravidel.
- **Neuronové sítě** (Artifical Neural Networks) - jsou v podstatě zjednodušeným modelem neuronových propojení v lidském mozku modelovatelným výpočetní technikou. Jejich principem je nastavení parametrů jednotlivých „neuronů" v procesu učení se z tréninkových vzorků dat, aby výsledná konfigurace co nejlépe vyhovovala následné klasifikaci a predikci. Neuronové sítě jsou příkladem aplikace jedné z vývojových linií dolování dat - umělé inteligence.
- **Genetické algoritmy** (Genetic Algorithms) - aplikují mechaniku genetiky a přirozeného výběru pro vyhledání optimální množiny parametrů, například pro použití v predikci. Genetické algoritmy neslouží k predikci určitých hodnot zkoumaných prvků (jako všechny výše popsané techniky), ale slouží k vývoji, resp. k parametrizaci dalších modelů pro predikci hodnot těchto prvků [45].

3.1.4 Aplikace Data mining

Existuje řada úloh pro dataminigové aplikace a jejich počet stále roste. Data mining se dnes používá především v bankovní sféře, telekomunikacích, plánování, medicíně, marketingu, při analýzách internetových přístupů apod. [75].

Reálné aplikace, v nichž se Data mining uplatňuje, je možné rozdělit do několika skupin. Jedná se zejména o kreditní skóring klientů, který funguje v každé bance již velmi dlouho, o specializované aplikace pro detekci podvodů v pojišťovnách i bankách, o prodejně marketingové aplikace nebo se jedná o řízení kvality ve výrobních procesech [45].

3.1.4.1 Kreditní skóring

Kreditní skóring předpovídá, kteří klienti nebudou splácet úvěr nebo platit za poskytnuté služby. V bankách se používá *aplikační skóring*, který rozhoduje o tom, kterým klientům banka na základě jejich žádosti poskytne úvěr. Jeden ze zásadních problémů aplikačního skóringu je skutečnost, že množina hodnocených klientů se liší od množiny klientů, kterým byl poskytnut úvěr a na kterých je vytvářen model pro aplikační skóring. V bankách i telekomunikačních firmách se používá *behaviorální kreditní skóring*, který pro všechny klienty na základě údajů o jejich chování předpovídá, kteří z nich nebudou splácet úvěry či platit za služby. Behaviorální kreditní skóring má oproti aplikačnímu tu výhodu, že skóruje všechny klienty, nikoliv jen ty, kteří požádali o úvěr. Proto je často využíván k rozhodování, kterému klientovi bude zaslána marketingová nabídka, zvýšen úvěrový limit na jeho kartě nebo kontokorent na běžném účtu.

Myšlenkou skórování je přiřadit a periodicky - např. měsíčně - aktualizovat individuálně pro každého zákazníka jedno nebo více skóre, jako jsou „Pravděpodobnost odchodu v nejbližším období" nebo „Marketingový segment zákazníka". Jinými příklady skóre zákazníka jsou například vyčíslení indikativní nebo dlouhodobé očekávané hodnoty zákazníka (Customer Value, Lifetime Value).

Koncovým výstupem skórování pak může být výpis zákazníků s největší pravděpodobností odchodu či souborný pokyn call centru nabídnout určitý produkt zákazníkům z určitého marketingového segmentu, nebo mohou mít obchodní manažeři či pracovníci call centra skóre k dispozici on-line, např. v průběhu každého telefonického kontaktu se zákazníkem [45].

3.1.4.2 Detekce – odhalování podvodů (fraud)

Perspektivní aplikací Data mining je odhalování podvodů **(fraud)**. Tato aplikace má uplatnění v pojišťovnách (odhaduje se, že cca 15 % pojistných událostí jsou podvody), v bankách (posuzování přidělení úvěrů, podvody s platebními kartami, případy praní špinavých peněz), v telekomunikacích pro odhalení špatného placení účtů. Podobné technologie se používají také ve státní sféře, např. identifikace podvodně získaných sociálních dávek, detekce celních či daňových podvodů. Dataminingové predikční modely mohou vystihnout komplikované vazby mezi vlastnostmi klientů a velmi dobře předpovídat potenciální podvodníky [75], [45].

3.1.4.3 Segmentace

Segmentace je obchodní úlohou pro jejíž realizaci se používá technika Data mining shlukování (clustering). Shlukovací techniky umožňují po zadání i většího počtu segmentačních proměnných najít shluky (clusters), které odpovídají „nejlepším možným" segmentům. Často se ale tyto dvě skupiny pojmů (segmentace, segmenty – shlukování, shluky) překrývají [86]. K segmentaci se využívá několik algoritmů Data mining. Jako příklad lze uvést algoritmus K-Means, Two-Step clustering nebo Kohonenovu neuronovou síť [49].

Segmentace nejčastěji představuje segmentaci zákazníků. Segmentovat lze ale i telefonní hovory podle jejich typů, stroje podle druhů údržby atd. Segmentace zákazníků znamená rozčlenění zákazníků na podskupiny, které jsou s ohledem na kritéria segmentace vnitřně relativně homogenní a mezi sebou poměrně heterogenní. Segmentaci zákazníků dnes využívá každá významnější společnost pro roztřídění zákazníků do podskupin, pro které se sjednocují obchodní a marketingové postupy [86].

Marketingové aplikace jsou v zásadě rozděleny na tři typy.

Propensity to buy

První a nejpřínosnější je cílení produktových marketingových kampaní na klienty, kteří mají zájem si daný produkt pořídit. Jedná se o takzvané *„propensity to buy*" nebo také afinitní modely, které předpovídají budoucí nákup tohoto produktu. Tyto modely typicky vznikají pro každý významný produkt [45].

Ztráta zákazníka, zachování zákazníka

Druhou typickou aplikací je ztráta zákazníka (churn) neboli předpověď odchodu zákazníků, která umožňuje tomuto nepříznivému vývoji včas předejít. Získání nového zákazníka bývá v praxi finančně daleko náročnější než udržení už existujícího. Proto je výhodné vytipovat klienty, kteří mají sklony k přechodu ke konkurenci, a udělat pro ně speciální akce nebo nabídky. Takovéto modely najdou uplatnění např. v telekomunikacích, kde je velká migrace zákazníků mezi jednotlivými společnostmi [75], [45]. Pro analýzu a predikci odchodu zákazníků se dnes využívají nejčastěji logistická regrese, rozhodovací stromy a umělé neuronové sítě. Při segmentaci je možné se setkat ještě s úlohou označovanou jako retence – zachování zákazníka (retention), tedy pokračování ve využívání produktů, služeb nebo pokračující nákupy určitého zákazníka [84].

Hodnotová a behaviorální segmentace

Třetí aplikací je segmentace zákazníků, která rozdělí zákazníky do homogenních skupin podle jejich hodnoty nebo podle jejich chování [45].

Hodnotová segmentace je účelným prvním krokem pro iniciální rozčlenění portfolia zákazníků na hlavní skupiny (popřípadě ověření takového již existujícího rozdělení) a bývá doplňována dalšími analýzami [86]. Hodnotová segmentace je typicky založena na několika málo proměnných, k nimž patří např. současná hodnota zákazníka, potenciál zákazníka, riziko odchodu zákazníka, další rizika zákazníka. Hodnotová segmentace používá obvykle 6 až 8 segmentů a slouží k zodpovězení otázky „Co chceme s těmito zákazníky udělat?" [36].

Behaviorální segmentace se pokouší primárně odhlédnout od hodnoty zákazníka a zaměřuje se na jeho „chování" [86]. Většinou se používá pro návrh produktů, volbu

komunikačního kanálu, způsob komunikace i vlastní sdělení [45]. Behaviorální segmentace je typicky založena na mnoha (desítkách až stovkách) proměnných. Behaviorální segmentace obvykle používá 12 až 20 segmentů a slouží k zodpovězení otázky „Jak dosáhnout u těchto zákazníků stanoveného cíle?". Behaviorální segmentace se často využívá k cílení marketingových kampaní. Její nejsilnější aplikací je možnost nabízet tentýž produkt různým segmentům různým způsobem, zdůrazňovat jeho různé vlastnosti a použít různé optimální komunikační kanály [36].

Jak již bylo uvedeno, hodnotová segmentace je založena na několika málo proměnných, a proto se pro ni používají tradiční metody clusteringu (shlukování), založené na vzdálenosti, například algoritmus k-means. Naproti tomu behaviorální segmentace je založena na mnoha proměnných a je nutné použít metody clusteringu založené na pravděpodobnostním modelu, například algoritmus EM pro naivní bayesovský model.

V projektu segmentace se často vytvářejí obě tyto segmentace a pak je behaviorální segmentace zjemněním segmentace hodnotové nebo hodnotová segmentace je přehlednou agregací behaviorální segmentace. Potom je třeba kombinovat obě uvedené metody tak, aby byla zachována tato hierarchie obou segmentací [36].

Výběr segmentačních proměnných

Volba segmentačních proměnných není jednoznačnou záležitostí. Obecně lze říci, že musí vyhovovat několika kritériím:

„Z pohledu zdravého rozumu" musí být segmentační proměnné vybrány tak, aby předjímaly účel segmentace. Například, je-li cílem striktně behaviorální segmentace, neměla by se mezi segmentačními proměnnými vyskytnout proměnná vyjadřující objem/množství.

Segmentační proměnné musí být vybrány či upraveny s ohledem na použitou shlukovací metodu a konkrétní nástroj. Některé nástroje umožňují použít spojité proměnné, jiné kategoriální a jiné i oba typy.

Hodnoty segmentačních proměnných mají mít určité statistické předpoklady. Mezi nejdůležitější patří:

Nezávislost. Není vhodné, aby segmentační proměnné byly vzájemně závislé (korelované). To se ověřuje předem použitím korelační analýzy, faktorové analýzy či testů závislosti kategoriálních proměnných. Při provádění těchto testů je nutné dbát na předpoklady použitých metod. Například faktorová analýza realizovaná pomocí tabulky korelací vypočítaných Pearsonovým korelačním koeficientem předpokládá normální rozdělení hodnot analyzovaných proměnných.

Rozdělení hodnot. Nejlepším vhodným rozdělením hodnot spojitých segmentačních proměnných je rozdělení normální (jeho grafické znázornění je dáno symetrickou jednovrcholovou hustotou, která je zvonovitého tvaru a nikde neprotíná vodorovnou osu), u kategoriálních proměnných pak zachování pravidla, že počet hodnot jedné kategorie nemá přesahovat 85 % a klesnout pod 15 % všech hodnot. Shlukovací metody bývají natolik robustní, že se dokážou vyrovnat i s odchylkami od popsaného ideálu, nicméně je dobré se neodchylovat příliš. Jednou z cest, jak zlepšovat výsledky segmentace, je realizovat segmentaci na podskupiny zákazníků: Například u zákazníků banky se nejprve realizuje celková hodnotová segmentace a pak pouze pro zákazníky využívající i účet v cizí

měně behaviorální segmentace, zaměřená na chování ve vztahu k využití účtu v cizí měně.

Odlehlé hodnoty. Odlehlé hodnoty (outliers a extremes) mohou výrazně vychýlit středy hledaných shluků a před realizací shlukování je vhodné tuto situaci řešit (viz dále).

Počet. Počet segmentačních proměnných je dobré udržovat rozumně malý. Obecným doporučením je nezvyšovat jejich počet nad 7 až 10. Dobré zkušenosti z některých segmentací jsou ale i s počtem segmentačních proměnných, například 14 [86].

Při realizaci segmentace s využitím shlukování je ideální kupříkladu použít dvě různé metody a výsledky porovnat. V případě, že se příliš liší, je dobré ověřit, zda segmentační proměnné jsou vybrány a připraveny podle uvedených kritérií. Při segmentaci může analytika i překvapit, že shlukovací algoritmy jsou vesměs citlivé na pořadí případů. Požaduje-li se při opakované segmentaci dosažení konzistentních výsledků, nesmí se v jejím průběhu pořadí případů změnit [86].

CRM - Consumer Relationship Management

Zavedení těžby dat by měl cítit jako potřebu vrcholový management firmy s tím, že využívá stávajících informací, které má k dispozici, ale zároveň ví, že to ještě není ono. Že například potřebuje doplnit informaci o zákazníkovi historickým vývojem jeho chování za několik let dozadu [16]. Potřeby marketingu značně urychlily rozvoj postupů Data mining [51]. Čábela uvádí, že „za tímto účelem se v poslední době stále více využívají možnosti výpočetní techniky a nástroje CRM".

CRM je módní obchodní strategie, jejímž cílem je získat a udržet si profitabilní zákazníky. CRM umožňuje jednak řídit nabídku produktů a služeb podle potřeb jednotlivých zákazníků, jednak řídit přístup ke klientům a vynaložené náklady podle jejich významu pro podnik [16].

Ale to, co je v pozadí a co firmě přináší skutečné hodnoty, je CVM (Consumer Value Management), jenž je pro úspěch implementace CRM klíčový. Pod pojmem CVM se rozumí koncept diferenciovaného přístupu ke klientům dle jejich hodnoty a aktivní řízení a budování této hodnoty optimalizací souvisejících nákladů a vhodně cílených nabídek dalších produktů a služeb [16].

Třemi hlavními prvky CRM jsou lidé (lidský kapitál, zákazníci), procesy a technologie. Existuje mezi nimi bezprostřední souvislost a doplňuje je čtvrtý prvek: data. Význam a účel těchto čtyř prvků spočívá v komplexním pohledu na CRM, nikoli v detailním zaměření na význam jednotlivých prvků [90].

Úspěšné zavedení CRM vyžaduje velice dobrou znalost zákazníka. To znamená mít k dispozici dostatek údajů o jeho chování v minulosti a současnosti. K optimalizaci dialogu se zákazníkem je nutná rovněž vybudovaná technická infrastruktura. CRM je tedy sada procesů a postupů, jimiž firma musí disponovat. Nejsou-li zavedené, je třeba je implementovat. Kromě toho je nutné podpořit tyto procesy technologickou infrastrukturou. Technologická a procesní část CRM jsou vzájemně neoddělitelné [1].

Prvním krokem k porozumění zákazníkovi je vědomí o všech produktech a službách, které využívá, a o všech uskutečněných kontaktech mezi ním a podnikem. To se řeší produkty tzv. operativního CRM, jejichž cílem je především "pamatovat si zákazníka" a historii jeho kontaktů s firmou. Tyto produkty pomáhají společnosti při interakci s klientem a zvyšují

konzistenci vzájemné komunikace, ale samy o sobě nevedou k dokonalému porozumění zákazníkovi a pochopení jeho potřeb.

Dalším důležitým krokem je proto solidní analýza dat o klientech, identifikace typických vzorů chování určitých skupin zákazníků a predikce jejich chování a potřeb do budoucna na základě těchto analýz. K tomu slouží produkty analytického CRM vystavěné kolem centrálního zákaznického datového skladu (Data Warehouse). Ten je klíčovou složkou úspěšné implementace komplexního CRM a v kombinaci s analytickými nástroji umožňuje skutečnou strategickou změnu v řízení přístupu ke klientům, nikoli jen optimalizaci běžných kontaktů s nimi [16].

Data v datovém skladu musí být organizována podle analyzovaných subjektů (zákazník, jeho transakce a události netransakčního charakteru, adresní údaje, produkty, finanční údaje, data o marketingové kampani a jejích výsledcích) a nikoli podle údajů v hlavní účetní knize nebo podle toho, z jakého provozního systému se data do datového skladu dostávají [1].

Mezi důležité přínosy analytického CRM patří schopnost modelování a predikce chování konkrétních zákazníků v různých situacích. Prediktivní a náklonnostní modely mohou být použity pro výběr vhodné nabídky produktu, pro správné a efektivní cílení marketingových kampaní na jednotlivé segmenty zákazníků, či dokonce na konkrétní klienty, ale také například k předpovědi poptávky po jednotlivých druzích zboží [16].

Hlavním problémem CRM je, že nashromážděné informace o zákaznících se nacházejí kdesi v podniku, ale nejsou k dispozici tam, kde je to nutně zapotřebí. Tím dochází k přerušení sledu interakcí a nemůže se dostatečně využít toho, že zákazník zanechává v jednotlivých kontaktních místech důležité informace, které jsou nezbytné pro udržení a rozvíjení vztahů [90].

Nasazení a používání komplexního CRM systému pro účely CVM je obvykle iterativní proces, který zpočátku rozdělí zákazníky do určitých segmentů podle poměrně jednoduchých kritérií. CRM systém cílí na tyto segmenty různé marketingové aktivity, řídí jejich průběh a po jejich skončení analyzuje jejich výsledky. Na základě této analýzy se upraví a zjemní segmentace zákazníků pro další kampaně a celý proces se opakuje. Souhrn současné a potenciální ziskovosti klienta v průběhu celého jeho životního cyklu s organizací se nazývá celoživotní hodnota klienta (*Customer Lifetime Value*). K jejímu zjišťování se používají zejména nástroje Data mining v kombinaci s prediktivními a afinitními metodami modelování v zákaznickém datovém skladu. Stanovení CLV je jednou z nejtěžších úloh analytického CRM. Je však mimořádně důležitá pro umožnění skutečně optimálního přístupu k jednotlivým zákazníkům [16].

Ne každé podnikání ale CRM potřebuje. Říká se dokonce, že až 50 % investic do CRM bylo utraceno zbytečně. Firmy, kde se zákazníci často mění nebo kde chybí přímý kontakt mezi firmou a koncovým zákazníkem, nebo výrobci produktů, které si koupíte jednou za život, jej příliš nevyužijí. Databázový marketing je drahý, vyžaduje obrovskou investici do sběru informací o jednotlivých zákaznících a neustálou aktualizaci těchto dat, nemalou investici do hardwaru a softwaru i do školení pracovníků a také specialisty na dolování dat. Naopak nejvýhodnější je investice do CRM jednoznačně pro firmy, které sbírají hodně dat o klientech, jako jsou banky, pojišťovny nebo telekomunikační společnosti. Systém je výhodný také pro společnosti, které mohou využít křížový prodej. O CRM by měly uvažovat firmy, jejichž zákazníci mají velmi individuální potřeby a také jejich hodnota pro

firmu se výrazně liší. Někdy může 20 % nejlepších zákazníků vytvářet 80 % zisku společnosti – o ty je potřeba se proaktivně starat a motivovat je k loajalitě [44].

Firmami v první linii s obrovským množstvím údajů o zákaznících jsou zcela jistě banky, telekomunikační společnosti a supermarkety, ale i nemocnice, pojišťovny, meteorologické ústavy a státní sektor všeobecně. Aplikačních oblastí se určitě najde v budoucnu mnoho, jde jen o to uvědomit si dobře význam uložených dat a možnost jejich využití při podpoře rozhodování [40].

3.1.4.4 Stanovení diagnózy

Pro stanovení správné diagnózy a podání správného léčiva na základě známých příznaků může být s výhodou použit dataminingový predikční model, který dovede zahrnout i různé anomálie a skryté závislosti [75].

3.1.4.5 Analýza časových řad

Existují časové řady, které lze velmi obtížně popsat standardními matematicko-statistickými modely. Pomocí dataminingových metod lze v takových řadách detekovat různé interakce vyšších řádů, modelovat nelineární závislosti apod. Své uplatnění tu Data mining najde ve všech oblastech, kde je třeba provádět předpovědi na základě historických dat, tedy např. ekonomika, meteorologie, kontrola kvality apod. [75].

3.1.4.6 Analýza prohlížení stránek na Internetu (web mining)

Soubory z www serverů se záznamy o prohlížení stránek představují velmi objemná a dynamicky se rozrůstající data, která obsahují množství skrytých vazeb. Díky technikám Data mining lze z těchto dat získat informace o nejčastějších vzorech v prohlížení či zákazníky segmentovat podle jejich chování na internetu. Uvedené postupy uplatňují především společnosti, které po internetu prezentují nebo prodávají své produkty [75].

Současné studie, zabývající se využitím dat z webu, používají metody, k nimž patří např. *asociační pravidla* (association rules), *shlukování* (clustering), *prediktivní modelování* (predictive modeling), *analýza cest* (path analysis) či *analýza časových sekvencí* (temporal sequences). Ačkoliv většina metod používaných pro zpracování dat získaných z webu má původ v databázovém marketingu, metodách pro získávání informací či zpracování dat, metoda nazvaná analýza cest byla navržena jen pro zpracování webových dat. Lze předpokládat, že s pokračujícím nárůstem využívání webu budou vyvinuty další metody pro zpracování webových dat, umožňující integraci dat různého typu.

Firma může využít ve svém obchodním plánování každé kliknutí myší na webové stránce, jež se kombinuje s předcházejícími a vytváří určitý vzor chování návštěvníka daného webu. Správně nakonfigurovaný webový server dokáže zaznamenat každé kliknutí, které návštěvník na prohlížené stránce provede. Každé kliknutí v této posloupnosti (clickstream) je zapsáno do protokolu daného serveru, přičemž příslušný záznam obvykle obsahuje identitu návštěvníka, stránku, na které byla interakce zaznamenána, a údaj o čase. Lze říci, že i při minimální konfiguraci všechny webové servery zaznamenávají protokoly přístupů a chybové protokoly. Webové servery lze nakonfigurovat také tak, aby ke standardním protokolům přidaly ještě protokol odkazů. Popis některých důležitých zdrojů dat na webu následuje níže:

Protokoly serveru

Protokol přístupů. Pokaždé, když nějaký návštěvník požaduje načtení nějakého souboru z webového serveru, je do speciálního ASCII textového souboru, nazývaného protokol přístupů (log), přidán další nový záznam. Protokol přístupů zaznamenává nejenom požadavky na načtení těchto souborů, ale současně také zapisuje úspěšnost či neúspěšnost každého takového požadavku. Každá transakce provedená v průběhu dané relace je chronologicky zaznamenána do protokolu přístupů. Z toho vyplývá, že protokol přístupů je hlavním zdrojem informací o návštěvnících webu a o stránkách, které daný návštěvník prohlížel.

Protokol odkazů. Protokol odkazů je dalším protokolem vytvářeným webovým serverem a obsahujícím záznamy o adresách, ze kterých se daný návštěvník dostal na web vaší firmy. Současně může protokol odkazů obsahovat i záznamy o klíčových slovech, která jej na váš web přivedla. URL adresa, ze které se návštěvník dostal na váš web, může být jen odkazem z jiné stránky téhož webu nebo výsledkem hledání nějakého vyhledávacího serveru. V případě, že se návštěvník dostal na váš web v důsledku hledání na takovém serveru, jsou do protokolu odkazů zapsána i klíčová slova, na jejichž základě byl váš webový server nalezen. Kromě toho je samozřejmě zapsána i informace o webové adrese vyhledávacího serveru.

Cookie

Cookie je v podstatě malým množstvím informace, odeslaným z webového serveru na počítač návštěvníka ve chvíli, kdy tento návštěvník vstoupí na daný server. Cookie obsahují informace o tom, na které stránky webu návštěvník přechází. Jakmile se pak daný návštěvník vrátí na stránku, kterou již předtím navštívil, cookie (uložené na pevném disku počítače daného návštěvníka) umožní serveru zjištění identity návštěvníka a umožní serveru změnu nastavení tak, aby odpovídala požadavkům tohoto návštěvníka. Ačkoliv používání cookie vyprovokovalo diskusi o ochraně osobních údajů na Intemetu, mnohé webové servery a - zejména - servery pro elektronické obchodování je využívají jako jednu z klíčových marketingových komponent, umožňujících personalizaci webových stránek a nabídek různých produktů.

Registrační formuláře a jiné způsoby registrace návštěvníků

Webové servery mohou sloužit ke sběru dat o návštěvnících webu, a to tak, že na počátku nové relace server požádá návštěvníka o vyplnění registračního formuláře, Ten může vyžadovat vyplnění jména a adresy, data narození, pohlaví, povolání apod. Tyto informace jsou poté načítány do databází, které se tak stávají cenným zdrojem informací o návštěvnících a jsou základem pro získávání dodatečných demografických dat, používaných při vytváření podrobných profilů návštěvníků.

Požadavek na zadání adresy elektronické pošty

Mnohé webové adresy umožňují vyplnění pole, obvykle nazývaného *poslat poštu na*. Díky tomu získá firma provozující daný webový server možnost v podstatě okamžitě informovat konkrétní potenciální zákazníky o nových produktech a službách, či je může informovat o nabídce na firemním serveru pro elektronické obchodování. Cenné informace o obchodních trendech a analýzy ziskovosti pak mohou být získány analýzou celkové databáze poštovních adres, spojené s daty o reakcích zákazníků. Výsledné údaje lze využít na webovém serveru k automatizaci odesílání e-mailových informací o novinkách konkrétním návštěvníkům či celým skupinám návštěvníků. Stejně tak mohou být těmto

návštěvníkům automaticky odesílány informace o speciálních nabídkách výrobků, o které se tito lidé zajímali.

Data o nákupech na webu

Data, nashromážděná ze záznamů o nákupech na webu (jako např. jméno zákazníka, jeho adresa, PSČ, demografické údaje, vybrané zboží a prodejní ceny), mohou být doplněna a rozšířena údaji získanými z newebových zdrojů, mezi které patří např. různé účetnické systémy a databáze automatizovaného prodeje. Výsledná data lze pak podrobně analyzovat a získat tak podrobnou představu o současných i možných budoucích marketingových strategiích. Kromě výše uvedených údajů, které může každá firma snadno získat ze svého webového serveru, lze též u poskytovatelů připojení k Intemetu zakoupit celé databáze informací o chování jednotlivců či celých skupin na jiných webových serverech [54].

3.2 Data pro použití technik Data mining

Při aplikaci technik Data mining se využívá různých typů dat z různých datových zdrojů.

3.2.1 Typy dat

Ať už data pocházejí odkudkoli, spadají do tří základních typů: demografický, behaviorální a psychografický.

Demografická data obecně popisují charakteristiky osob či domácností. Mezi tento typ dat patří pohlaví, věk, rodinný stav, příjem, vlastnictví domu, typ obydlí, úroveň vzdělání, národnost a počet dětí. Demografická data jsou velmi stabilní, což je činí výborně použitelnými v prediktivních modelech. Charakteristiky jako rodinný stav, vlastnictví domu, úroveň vzdělání a typ obydlí se nemění tak často. Demografická data jsou obvykle levnější než názorová či behaviorální, zvláště jsou-li zakoupena dohromady. Jednou z nevýhod demografických dat je fakt, že je poměrně obtížné je získat.

Behaviorální data vyjadřují míru akce nebo chování. Behaviorální data jsou obecně typem dat poskytujících nejlepší prediktivní sílu. V závislosti na odvětví mohou být jejich součástí prvky jako prodaná množství, typy a data nákupů, data a výše plateb, činnost zákaznických služeb, pojišťovací nároky, chování při krachu a podobně. Jiným typem behaviorálních dat jsou aktivity na webových serverech (prodeje, jednotlivá klepnutí uživatele nebo přesná cesta procházení každého návštěvníka webem). Behaviorální data obvykle plní úlohu předpovědi budoucího vývoje lépe než jiné typy dat. Je však obvykle také složitější a dražší taková data z vnějšího zdroje získat.

Psychografická neboli **attitudiální data** jsou charakterizována názory, životním stylem či osobními hodnotami. Tento typ dat je tradičně spojován s výzkumem trhu a získává se hlavně prostřednictvím šetření, výzkumů mínění a zájmových skupin. Lze je také odvodit z nákupního chování. Vyjadřují zamýšlené chování, které může vysoce, částečně nebo jen okrajově korelovat se skutečným chováním. Tyto data lze aplikovat na větší skupiny lidí na základě segmentace nebo jiné statistické metody [54].

Novotný a spol. doplňují tyto tři základní typy dat ještě o data produktová a kontaktní. **Produktová data** jsou údaje o tom, které produkty zákazník ve sledovaném období využíval, jak často atd. **Kontaktní data** představují údaje, jež se zaznamenávají o reakcích na určité marketingové kampaně, o počtu a druhu dotazů zákazníka na „horkou linku" atd. [45].

3.2.2 Zdroje dat

Podle způsobu pořízení dat lze rozlišovat mezi primárními a sekundárními daty. Primární data jsou taková, která jsou získána pro potřeby řešení konkrétního úkolu. Sekundární data byla získána za jiným účelem, někým jiným, přičemž málokdy jsou taková data pro analýzu zcela vhodná. V kontextu technik Data mining se jedná o data, která se zaznamenávají například v souvislosti s určitou hospodářskou nebo výzkumnou činností.

Data pro modelování lze získat z mnoha zdrojů, jež se podle způsobu pořízení rozdělují na interní a externí.

3.2.2.1 Interní zdroje dat

Interní zdroje poskytují data s nejvyšší vypovídací schopností pro modelování. Jedná se o data, která vznikají prostřednictvím aktivit firmy jako záznamy o zákaznících, firemní web, záznamy z poštovních či telefonních kampaní nebo databáze či datové sklady, které jsou přímo určeny k uchovávání firemních dat. Typickými zdroji interních dat jsou databáze zákazníků, databáze provedených transakcí, databáze historie nabídek, databáze pro kampaň a datové sklady.

Zákaznická databáze je zpravidla tvořena jedním záznamem na zákazníka. V některých organizacích jsou toto jediné databáze. V takových případech mohou obsahovat všechny záznamy o prodejích a aktivitách pro každého zákazníka. Je však obvyklejší, že zákaznická databáze obsahuje identifikační údaje, které lze propojit s jinými databázemi, například s databázemi transakcí, a získat tak aktuální snímek aktivity zákazníka.

Transakční databáze obsahují záznamy o aktivitě zákazníků. Jde často o nejbohatší informace s nejvyšší schopností predikce, ale může být náročné ji zužitkovat. Ve většině případů představuje každý řádek jedinou transakci, takže databáze může u každého zákazníka obsahovat více záznamů. Transakční databáze mohou nabývat různých podob v závislosti na druhu podnikání. Aby byla tato data využitelná pro modelování, musí být sumarizována a agregována na úrovni zákazníka. Počet záznamů na jednoho zákazníka se může lišit.

Databáze historie nabídek obsahuje podrobnosti o nabídkách učiněných potenciálním či stávajícím zákazníkům. Vhodným formátem je jedinečný záznam pro každého zákazníka. Proměnné vytvořené z této databáze mají často nejvyšší schopnost predikce v cílených modelech odpovědí a aktivace.

Z databází stávajících či potenciálních zákazníků se vybírají **data potřebná pro kampaň**. Využívají se pro generování individualizovaných dopisů nebo pro nahrávky telefonní nabídky telemarketingových firem [54].

Pro přenos dat mezi dvěma (či více) libovolnými systémy lze použít tzv. ETL (Extraction, Transformation and Loading). Běžným označením pro prostředky ETL je rovněž **datová pumpa**. Jejím úkolem je data ze zdrojových systémů získat a vybrat (Extraction), upravit do požadované formy a vyčistit (Transformation) a nahrát je do specifických datových struktur, resp. datových schémat, datového skladu (Loading). ETL systémy získaly na důležitosti s rozvojem analytických systémů, tedy s explicitní potřebou pro zajištění přenosu dat mezi různými aplikačními systémy v rámci různorodého databázového prostředí. Nástroje ETL pracují v dávkovém (batch) režimu, data jsou tedy přenášena v určitých časových intervalech. Většinou se jedná o denní, týdenní a měsíční intervaly. Nástroje pracující v reálném čase se označují EAI (Enterprise Application Integration) a většinou pouze doplňují dávkový přenos, což umožňuje vznik nové generace datových skladů, tzv. Real-Time Data Warehouse [45].

Datový sklad (Data Warehouse) je integrovaný, subjektově orientovaný, stálý a časově rozlišený souhrn dat, uspořádaný pro podporu potřeb managementu [45]. Datový sklad je struktura, která spojuje informace ze dvou či více databází [54]. Toto sjednocování zahrnuje zajištění shody názvů stejných ukazatelů, sjednocení měřítek, sjednocení kódování (např. pohlaví kódované v jedné databázi hodnotami „M" a „Z", v jiné databázi „0" a „1") apod. [3]. Za pomoci výše zmíněných datových zdrojů dává datový sklad data

dohromady do jednoho centrálního úložiště, provádí jejich určitou integraci, vyčištění a sumarizaci a distribuuje informace do datových martů [54].

Do **datového martu** (datového tržiště) se z datového skladu přesouvají data relevantní pro určitý typ analýz [3]. Princip datových tržišť je obdobný jako v případě datových skladů. Rozdíl je v tom, že datová tržiště - Data Marts jsou určena pro omezený okruh uživatelů (oddělení, divize, pobočka, závod apod.). Podstatou jsou tak decentralizované datové sklady, které se budou postupně integrovat do celopodnikového řešení. V některých případech slouží dále Data Marts, i po vytvoření celopodnikového datového skladu, jako mezistupeň při transformacích dat z produkčních databází [45]. Datové marty slouží k uchovávání podmnožin dat z centrálního úložiště, která byla vybrána a připravena pro určité koncové uživatele. (Často se také označují za oborové datové sklady.) Analytik, jenž chce získat data pro určitý cílený model, pak přistupuje k odpovídajícímu datovému martu [54].

Data Mart je tak problémově orientovaný datový sklad, určený pro pokrytí konkrétní problematiky daného okruhu uživatelů a umožňující flexibilní „ad hoc" analýzu. Výsledkem vytváření datových tržišf je zkrácení doby návratnosti investic, snížení nákladů a podstatné zmenšení rizika při jejich zavádění [45].

Data ve formě Data Warehouse je tedy nutno filtrovat, čistit, doplnit, normalizovat, agregovat, převést na společnou strukturu, doplnit potřebnými externími údaji. Data Warehouse lze označit za kompletní soubor problémově orientovaných, integrovaných podnikových dat, získaných z transakčních systémů. Data v Data Warehouse jsou nejen detailní, ale i souhrnná, zachycují i historický vývoj a využívají se analyticky [51]. Pro práci s daty uloženými v datovém skladu slouží analytické manažerské nástroje EIS (Executive Information System), MIS (Management Information System) a DSS (Decision Support System). EIS a MIS slouží pro různý stupeň agregace a prezentace výsledků procesů ve firmě, kdežto systémy DSS hledají závislosti mezi daty, popisují firemní procesy a sledují jejich vzájemné závislosti. Právě pro DSS je použita technologie Data mining [89].

3.2.2.2 Externí zdroje dat

Mezi externí zdroje se řadí obvykle prodejci a kompilátoři seznamů. Prodejci seznamů jsou firmy, které prodávají seznamy osob. Jen málo z těchto firem však má prodej seznamů jako svůj výhradní předmět činnosti. Mnohé z nich se zabývají v první řadě prodejem prostřednictvím časopisů nebo katalogů a prodej seznamů osob bývá jejich vedlejší činností. Podle druhu činnosti obvykle shromažďují a prodávají jména, adresy a telefonní čísla společně s demografickými, behaviorálními či psychografickými údaji. Někdy také provádějí „očistu" seznamů nebo jejich pročištění, aby zvýšili jejich hodnotu. Mnohé z těchto firem prodávají své seznamy prostřednictvím kompilátorů a brokerů seznamů.

Kompilátoři seznamů jsou firmy, které prodávají různé seznamy, z nichž některé jsou založeny na jediném seznamu a jiné jsou kompilovány z několika různých databází. Některé firmy vycházejí z podkladů, jako je telefonní seznam nebo registrační data z řidičských průkazů. Pak nakupují seznamy, vzájemně je slučují a doplňují chybějící údaje. Mnohé z těchto firem provádějí vlastní výzkumy, aby zdokonalily přesnost svých seznamů [54].

3.2.3 Velikost datového souboru

Většina výzkumných a analytických úloh se opírá o primární data. Vzhledem k předmětu zájmu a charakteru úlohy se primární data získají způsoby, jako je navrhování experimentů, dotazování a někdy i skutečným pozorováním daného jevu či procesu. Cílem je získání odpovědí na otázky, které byly na samém začátku definovány. V porovnání s tímto přístupem bývá Data mining spojován s analýzou rozsáhlých sekundárních dat - procesem hledání zajímavých vztahů a struktur v rozsáhlých databázích historických dat. Pro analýzu primárních dat jsou typické datové soubory relativně malého rozsahu, předpokládá se i určitá kvalita dat, jejich stacionarita, technika použitá k jejich získání i alternativní metody, které se používají pro jejich analýzu. V porovnání s těmito závěry je situace v technikách Data mining odlišná [26].

Analyzují-li se sekundární data z podnikových databází, pracuje se se soubory velkého rozsahu. Označení souboru jako malého nebo velkého závisí na různých faktorech. Dané označování se vztahuje především na počet objektů. U shlukové analýzy je však nutné přihlížet i k počtu proměnných. Jako příklad velkého souboru uvádí Hebák a kol. případ, kdy má soubor více než 250 objektů. Je-li objekt charakterizován více než 16 proměnnými, jsou již obtížně identifikovatelné rozdíly ve vzdálenostech mezi objekty. Již v takovém případě se hovoří o vysoké dimenzionalitě. Praxe však ukazuje, že databázové zdroje dat obsahují záznamy, jež se dají počítat v miliónech (a vyšších řádech) pozorování.

Rozsah pozorování v řádech statisíců či milionů či počet proměnných v řádu stovek není ničím neobvyklým. Také není překvapením u takto rozsáhlých souborů, že ani dnešní výpočetní prostředky nemusí být pro efektivní analýzu dostačující [26]. Malá implementace projektu Data mining se pohybuje v řádech gigabajtů objemu dat, střední a větší pak pokračuje přes desítky, stovky až k terabajtům objemu dat [50].

Proto se pro většinu úloh datový soubor redukuje ve fázi předzpracování. Redukce se může týkat jak počtu objektů, tak počtu proměnných, ale i počtu kategorií, pokud datová matice obsahuje kategoriální proměnné. Zatímco snížení počtu proměnných a počtu kategorií se využívá i při analýze primárních dat, redukce počtu objektů se týká převážně analýzy sekundárních dat. Existují i metody, které řeší současně problém velkého počtu objektů i problém velkého počtu proměnných, a to pomocí shlukování podprostorů. Při shlukování je však potřeba analyzovat všechny objekty, proto jsou pro velké datové soubory vyvíjeny nové metody [29].

I přes neustálý technický rozvoj a s ním související růst kapacit paměťových médií, zůstávají běžné statistické metody z důvodu kupříkladu nedostačující operační paměti mimo oblast použití. V takovýchto situacích je zapotřebí aplikace jiných přístupů k analýze dat, mezi něž patří některé adaptivní či sekvenční metody a techniky a jim společná snaha o nalezení různých optimalizačních algoritmů. Jejich hlavním účelem je v prvé řadě výpočetní dosažitelnost hledaných řešení, ve srovnání s jinými používanými algoritmy dosažení v zásadě stejných či alespoň podobných výsledků v relativně kratším čase a konečně zajištění opakovatelnosti provádění, jejich automatizaci [26].

S velkým rozsahem dat úzce souvisí i kvalita dat. Hanyš uvádí, že „úsudky o kvalitě dat v souborech dat obrovských rozměrů" mohou být velmi složité a obtížné. Způsob, jak přistupovat například k pozorováním extrémním, odlehlým nebo k pozorováním chybějícím či chybným, není zcela zřejmý [26].

3.3 Příprava dat pro Data mining

Data mining nelze provádět bez kvalitní přípravy dat. A právě příprava dat je další oblast, kde se ve značné míře uplatňují statistické metody. Zjišťování odlehlých a extrémních hodnot, nahrazování chybějících údajů, výpočty s datovými soubory obsahujícími chybějící údaje (různé způsoby vynechávání údajů) - to jsou oblasti, které se bez statistických postupů neobejdou [56].

Dasu a Johnson [18] se věnují problematice průzkumových technik Data mining a čištění dat. Upozorňují, že „data jsou obvykle velmi znečištěná, složená z mnoha tabulek a mají neznámé vlastnosti". Předtím, než se začne tvořit jakýkoli výsledek dataminingových analýz, tak se data musí očistit a prozkoumat, což bývá často tou nejnáročnější činností jak na čas, tak obtížnost. Dasu a Johnson uvádí, že „mezi opravdové výzvy v dataminingové úloze patří:

- tvorba datového souboru, který by obsahoval relevantní a bezchybné informace, a
- určení vhodné techniky pro analýzu".

Proto je nutné věnovat velkou pozornost systematickému procesu průzkumu dat a řízení kvality dat. Průzkumná fáze jakéhokoli projektu datové analýzy nevyhnutelně zahrnuje vyřešení problému s kvalitou dat a zároveň jakékoli zlepšování kvality dat nevyhnutelně zahrnuje průzkum dat. Problém s kvalitou dat se dá řešit pomocí metod mnoha disciplín: statistiky, průzkumových technik Data mining (Exploratory Data Mining), databází, managementem a pomocí metadat. Pro exploratorní analýzu komplexního, neznámého souboru je tedy nutné použít několik nesourodých technik, aby se získaly dodatečné informace [18].

Průzkumový Data mining se skládá z jednoduchých a přehledných souhrnů a analýz, které odhalují charakteristiky dat, stejně jako typické hodnoty (průměry, mediány), kolísavost (rozptyl, variační rozpětí), převládání rozdílných hodnot (kvantily) a mezivazební vztahy (korelace). Je důležité poznamenat, že při řešení kvality dat je nutné konzultovat danou problematiku s odborníky a začlenit jejich znalosti do dalších etap průzkumové analýzy. Nesmí se zapomínat ani na to, že datové soubory generované automaticky mohou obsahovat chyby způsobené nespolehlivostí softwaru, hardwaru a zpracováním [18].

Příprava dat je v procesu vytváření modelu jedním z nejdůležitějších kroků. Kvalita vstupních dat je klíčem k úspěchu projektu [54].

Nástroje pro zajištění datové kvality

Nástroje pro zajištění datové kvality zažívají svůj prudký rozvoj s růstem nasazení analytických aplikací, zejména díky faktu, že pro úspěch nasazení řešení je, kromě již funkcionální a technické znalosti, třeba korektní obsah. Vzhledem k povaze řešení - podpoře analytické práce - je důležité, aby tato práce probíhala nad korektními daty, dokumentujícími reálnou situaci podniku.

Nástroje pro zajištění datové kvality se proto zabývají zpracováním dat s cílem zajistit jejich:

- *Úplnost* - jsou identifikována a ošetřena data, která chybí nebo jsou nepoužitelná (z různých důvodů)
- *Soulad* - jsou identifikována a ošetřena data, která nejsou uložena ve standardním formátu.

- *Konzistenci* - jsou identifikována a ošetřena data, jejichž hodnoty reprezentují konfliktní informace.
- *Přesnost* - jsou identifikována a ošetřena data, která nejsou přesná nebo jsou zastaralá.
- *Unikátnost* - jsou identifikovány a ošetřeny záznamy, které jsou duplicitní.
- *Integrita* - jsou identifikována a ošetřena data, která postrádají důležité vztahy vůči ostatním datům.

Implementace datové kvality je jedním z horkých témat současnosti [45].

3.3.1 Získání dat

Prvním krokem v procesu přípravy dat je získání dat v použitelném formátu. Zpravidla se data vyžadují ve tvaru ASCII. Pro soubor typu ASCII se též vžilo označení plochý soubor (flat file) nebo textový soubor. Řádky reprezentují jednotlivé záznamy nebo pozorování, sloupce neboli pole pak představují vlastnosti neboli proměnné týkající se záznamů. ASCII soubor se vyskytuje, pokud jde o délku záznamů, ve dvou základních formátech: o pevné a proměnné délce. Formát *pevné délky* se snadněji čte, protože pro každou vlastnost používá pevně vyhrazený prostor. Každý řádek dat má stejnou délku. Nevýhodou pevné délky záznamů je, že spotřebovává pro data prázdný prostor. Chybí-li v polích větší množství hodnot, může být tedy neúsporný. Formát *proměnné délky* má v každém řádku stejnou strukturu. Rozdíl spočívá v hodnotách sloupců (poli). Jestliže hodnota v některém sloupci chybí, není vyplněna mezerami, ale pro oddělení od sousedních hodnot je použit oddělovač. Mezi možné oddělovače patří čárky, lomítka a mezery. Velká výhoda tohoto formátu spočívá v tom, že pokud v něm chybí hodně hodnot, zabírají data méně místa.

Je důležité vyžádat si k datům potřebnou dokumentaci, například rozvržení souboru a datový slovník. Rozvržení souborů prozradí názvy proměnných, počáteční pozici dat a délku pole a typ všech proměnných. Datový slovník vám poskytne informaci o typu a podrobný popis významu každé proměnné. Doporučuje se rovněž získat „výpis dat" nebo výtisk prvních 25-100 záznamů [54].

3.3.2 Vytvoření sady dat

Pro vytvoření sady dat je v mnoha případech nutné zkombinovat data z několika zdrojů. Předtím, než je vytvořen datový soubor pro modelování, by se však mělo zvážit, zda nebude vhodné zredukovat jeho velikost pomocí vzorkování. Ačkoli během posledních let nesmírně vzrostla výkonnost počítačů, má stále dostatečný smysl. Urychluje celý proces a produkuje v podstatě stejné výsledky [54].

3.3.3 Kontrola dat

Dalším krokem je kontrola dat, zda v nich nejsou chyby, hodnoty mimo přijatelný rozsah a chybějící hodnoty. Jde o časově nejnáročnější, nejméně zajímavý, avšak nejdůležitější krok v celé přípravě dat. Kontrolu dat lze provádět jak na základě grafických, tak i výpočetních postupů [54].

3.3.3.1 Vizualizace vyhodnocovaných dat

Těžbu dat lze provádět od jednoduchého dotazu do databáze přes tvorbu tabulky z uložených dat až po vizuální zobrazení analýz z dat pocházejících z několika databází.

První stupeň znamenají jednoduché dotazy, krátké výpisy, malé tabulky nebo nepříliš složité analýzy. O stupeň výše je "typicky počítačové" zobrazení, např. ve formě tabulky nebo 3D grafu, spolu s jednoduchou analýzou. Nejvyšší stupeň je 2D nebo 3D vizualizace uložených dat [40]. Kvalitní vizualizace vyhodnocovaných dat má pro odhalování anomálií neobyčejný význam [51].

Vizualizace vznikla proto, že grafická podoba dat je pro člověka intuitivní, více přijatelná, rychleji se chápe a lépe se pamatuje. Lidské smysly odhalí anomálie a podobnosti v datech, která jsou zobrazena v grafické podobě, mnohem dříve, než když jsou data získána ve formě tabulky. Ve 3D člověk umí efektivně analyzovat i velmi složité vztahy. Pro usnadnění analýzy trendů bývá trojrozměrný prostor rozšířen o rozměr čtvrtý formou animace objektů v čase. Vizualizační metody proto používají lidé, kteří se potřebují rychle a kvalitně rozhodovat [40].

Doporučit vhodné grafické prezentace pro vícerozměrné statistické výstupy bývá o něco složitější, protože v nabídkách statistických a specializovaných programů převládají z pochopitelných důvodů dvourozměrné grafy. Hebák a kol. ve své knize věnují celou jednu kapitolu vícerozměrným grafům. Zmiňují se o zajímavých grafických možnostech s cílem naznačit krásu, názornost a užitečnost vizualizace různých vícerozměrných metod a úloh [29].

Velmi dobré možnosti vizualizace nabízí též grafické rozhraní nástroje OLAP (On-Line Analytical Processing), které umožňuje uživatelům nahlížet na data jak v numerické podobě, tak v podobě nejrůznějších grafů. Základem OLAP je pohled na data jako na mnohorozměrnou tabulku nazývanou datová krychle (*data cube*). Tento způsob uložení umožňuje různé pohledy na data: natáčení krychle (*pivot*), provádění řezů (*slice*), výběr určitých částí (*dice*) a zobrazování různých agregovaných hodnot [3].

3.3.3.2 Odlehlé hodnoty a chyby dat

Odlehlá hodnota (*outlier*) je případ, kdy se hodnota proměnné vyskytuje jednou nebo při nízké frekvenci daleko od střední hodnoty i od většiny ostatních hodnot této proměnné. Rozhodnutí, zda je určitá hodnota odlehlou hodnotou nebo chybou dat, je věda sama pro sebe. Nejlepší zbraní je v tomto směru důkladná znalost zpracovávaných dat. Nejlepším způsobem, jak zkontrolovat diskrétní hodnoty, je výpočet četnosti [54].

Jednou z cest při řešení odlehlých hodnot je použití diskretizace, kdy se případy nabývající odlehlých hodnot dostanou do krajních „košů" (viz dále).

Druhou možností je vypustit z analytické tabulky pro analýzu věty s odlehlými hodnotami a modely trénovat pouze na řádcích, které takové odlehlé hodnoty neobsahují. Příklady, kdy jsou v daném řádku považovány hodnoty jedné z jeho proměnných za odlehlé:

- Absolutní hodnota se liší od průměru o více než sedmínásobek (pětinásobek, trojnásobek) standardní odchylky. To je vhodné pro proměnné, jejichž průběh hodnot se příliš neliší od normálního.
- Hodnota patří do prvního nebo posledního decilu všech hodnot.
- Hodnota je vyšší nebo nižší než předem stanovené pevné meze zjištěné předběžnou analýzou dat či znalostí doplňujících informací.

Při řešení odlehlých hodnot je třeba být opatrný a zvolit správnou rovnováhu mezi čistotou modelu (k níž vede odstranění většího počtu případů s odlehlými hodnotami) a potřebou

neodfiltrovat z analytické tabulky pro analýzu skupinu případů, které spolu vytvoří významný shlukovací segment nebo predikční typologii [85].

Pokud se narazí na chybu, jejíž náprava není zřejmá, je možné ji považovat za chybějící hodnotu. Jinou metodou pro zpracování hodnot mimo rozsah je vytvořit pro ně *překrývací pravidlo* (tj. pravidlo, které umožní nahrazení extrémní hodnoty nějakou jinou hodnotou, nacházející se v akceptovatelných mezích). Lze jej užít např. tak, že se zjistí, zda je směrodatná odchylka větší než dvojnásobek hodnoty 99. percentilu. Pokud ano, pak je daná extrémní hodnota nahrazena čtyřnásobkem hodnoty 99. percentilu. Tím se získá poměrně široké rozdělení, ovšem už bez evidentně extrémních hodnot. Toto konkrétní pravidlo funguje jen u proměnných s kladnými hodnotami. Podle dat lze toto pravidlo upravovat tak, aby vyhovovalo daným potřebám [54].

Výhodu je, pokud jsou k dispozici algoritmy, které řešení odlehlých hodnot nabízejí v průběhu analýzy (pro případ segmentace je to např. SPSS Two step cluster). Pak může nalezený segment „outliers" často představovat signifikantní a zajímavou skupinu [85].

U kategoriálních proměnných se používá slučování hodnot kategorií majících malý počet případů s jinými kategoriemi, popřípadě se tyto případy vyloučí [86].

3.3.3.3 Chybějící hodnoty

Při sbírání a kombinování dat se vyskytují chybějící hodnoty téměř v každé sadě dat. I skutečnost, že hodnota chybí, totiž může mít prediktivní vlastnosti. Tyto informace je nutno zachytit [54]. Typická jsou tato řešení:

- Řádky s chybějící hodnotou segmentační nebo predikční proměnné jsou při analýze ignorovány. Takový případ je indikován např. v prostředí relační databáze ponecháním nastavení hodnoty na NULL s tím, že analytický algoritmus takovou hodnotu jako chybějící vyhodnotí [85].
- Chybějící hodnota se nahradí vhodnou zvolenou hodnotou. Může zde posloužit aritmetický průměr nebo medián – je však třeba velké opatrnosti [85]. Cíl při nahrazování chybějících hodnot je dvojí: zaplnit prázdná místa nejpravděpodobnějšími hodnotami a zachovat celkové rozdělení hodnot proměnné.
    - *Substituce jedné hodnoty* je nejjednodušší metodou nahrazování chybějících hodnot. Na výběr jsou tři obvyklé možnosti: střední hodnota, medián a mód. Střední hodnota je založena na statistickém výpočtu nejmenší chyby čtverců. Tím se do rozdělení hodnot proměnné zavádí nejmenší možná variance. Je-li rozdělení velmi špičaté (nesouměrné), může lépe posloužit medián.
    - Při *substituci střední hodnotou* třídy se využívají střední hodnoty podskupin jiných proměnných nebo kombinací proměnných. Tato metoda zachovává lépe původní rozdělení hodnot.
    - Podobně jako u substituce střední hodnotou třídy využívá *regresní substituce* střední hodnoty skupin jiných proměnných. Výhodou regrese je schopnost pracovat se spojitými proměnnými stejně jako hledat ve více proměnných přesnější míru. Výsledné hodnocení regrese slouží k dopočtení náhradních hodnot. Jednou z výhod regresní substituce je její schopnost zachovat celkové rozdělení dat [54]. Např. SPSS nabízí modul „Missing values", obsahující algoritmus, který se pokouší nahradit chybějící hodnotu spojité

proměnné hodnotou získanou prostřednictvím posloupnosti regresí z jiných proměnných.

- Chybějící hodnota se nahradí hodnotou získanou sofistikovanějším způsobem. Příkladem práce s chybějícími hodnotami jsou některé algoritmy rozhodovacích stromů, které pro rozhodnutí o zařazení daného případu do určitého uzlu pro případ, že hodnota určité proměnné chybí, použijí hodnotu náhradní proměnné [85].

Při modelování s nečíselnými (diskrétními) proměnnými je nejlepším způsobem, jak naložit s chybějícími hodnotami, považovat je jako další kategorii [54].

3.3.4 Výběr a transformace proměnných

Jakmile lze data považovat za správná a chybějící hodnoty za ošetřené, je dalším krokem vyhledání možných nových (odvozených) proměnných. Kombinacemi a permutacemi již vytvořených proměnných lze mnoha způsoby vytvořit další proměnné. Proto je tak nutné znát jak data, tak i obor.

3.3.4.1 Sumarizace

Sumarizace je jedním ze způsobů kombinace proměnných. Provádí se v určitých případech při generování velkých objemů dat. Mezi obvykle používané metody patří sčítání, odčítání a průměrování.

3.3.4.2 Segmentace

Někteří analytici a tvůrci modelů rozčleňují spojité proměnné do segmentů, s nimiž pracují jako s kategorickými proměnnými. Hlavní nevýhodou je fakt, že tím ztrácejí přínos z vazby mezi body na křivce, která může být v průběhu i dlouhé doby velmi robustní. Jiný přístup je vytvořit segmenty pro evidentně oddělené skupiny. Pak tyto segmenty otestovat s transformovanými spojitými hodnotami a vybrat vítěze [54].

3.3.4.3 Diskretizace

Diskretizací se rozumí převedení hodnot spojité proměnné do „košů" (bins) a další práci nikoli s původními hodnotami, ale s kódy košů. Výsledné kódy košů jsou primárně diskrétními hodnotami ordinálního typu (lze je seřadit podle pořadí, např. kód 3 je 'více' než 2). Jedním z triků, které lze po ověření použít, je však i to, že se proměnná obsahující hodnoty kódu diskretizované proměnné použije jako proměnná spojitá. Tímto způsobem se elegantně vyřeší také problém odlehlých hodnot [85].

Diskretizaci spojité proměnné často provádíme do stanoveného počtu binů na základě kvantilů, například 10 %, 25 %, 50 %, 75 %, 90 % [36].

Zda pro daný účel použít proměnné spojité (normalizované), nebo diskretizované, nebo jejich kombinaci, to je jedna z voleb, které se při aplikaci dataminingových technik řeší. Řešením může být např. použití spojité normalizované proměnné a jedné nebo dvou diskretizovaných hodnot, které se na základě výchozích předpokladů jeví pro analýzu jako užitečné [85].

3.3.4.4 Transformace dat, standardizace, škálování

Často se používá globální úprava pomocí transformací. Těmito transformacemi se často linearizují jinak nelineární vztahy nebo se upravuje tvar rozdělení dat, aby se více podobalo rozdělení popsanému Gaussovou křivkou. Jedná se o

- Přičítání nebo odečítání konstanty

Jednoduchou akce, kdy se ke všem datům přičte kladná nebo záporná konstanta. Nejobvyklejší taktikou je tzv. centrování - odečtení aritmetického průměru od všech získaných skórů dané proměnné. Dostáváme tzv. centrovaná data nebo odchylky od průměru ($\bar{x}$) a novým centrem stupnice znaku je nula:

$$x'_{ij} = x_{ij} - \bar{x} \qquad (1)$$

Získá se přehled, jak jsou jednotlivé údaje vzdálené od průměru. Použít lze také místo průměru medián ($\tilde{x}$) nebo jinou míru polohy (míru centrální tendence). Průměr, medián a modus takto transformovaných dat se změní stejně jako původní údaje.

- Násobení nebo dělení konstantou

Tato operace se často nazývá škálování (toto slovo však má i jiné významy). Používá se např. při přechodu mezi použitými jednotkami měření (mezi kilogramy a gramy, metry a centimetry apod.). Také pro tuto transformaci platí, že průměr, medián a modus transformovaných dat se změní stejně jako původní údaje.

- Standardizace

K nejpoužívanějším transformacím patří standardizace. Standardizace kombinuje odečítání a násobení. Standardizace se provádí podle předpisu

$$x'_{ij} = \frac{x_{ij} - \bar{x}}{s} \qquad (2)$$

K transformaci lze také použít kvartilové charakteristiky

$$x'_{ij} = \frac{x_{ij} - \tilde{x}}{IQR} \qquad (3)$$

Standardizace znamená, že průměr (nebo medián) standardizovaných dat je 0 a jejich směrodatná odchylka (nebo interkvartilové rozpětí IQR) je 1. Rozdělení, která jsou takto standardizována, se mnohem snadněji srovnávají a někdy i kombinují. Standardizovaná data se často nazývají též standardizované skóry.

Data se symetrickým rozdělením standardizovaná průměrem a směrodatnou odchylkou jsou symetricky rozdělená kolem nuly a jejich hodnoty se pohybují přibližně v rozmezí od –3 do 3. Hodnoty mimo tyto meze se prověřují, zda nemají charakter odlehlých hodnot.

- Převod hodnot na pořadové hodnoty a percentily

V těchto dvou transformacích se přiřazují naměřené hodnotě její pořadí nebo percentilová hladina. Nová hodnota udává relativní pozici původní hodnoty v celé množině dat vzhledem k relaci řazení podle velikosti.

Transformace do pořadí – označuje se někdy R – převádí daný údaj do intervalu 1 až n. Jsou-li všechny údaje různé, hledá se nejmenší údaj x_1 a přiřadí se mu číslo $R_i = 1$, a tak se postupuje, dokud se nepřiřadí všem prvkům jejich pořadová čísla. Obecně lze říci, že se přiřazuje údaji x_j číslo R_j, což je počet x_i, jež jsou menší nebo rovny údaji x_j. Pokud jsou některé x_j stejné, pak se jim přiřazuje průměrné pořadí, které odpovídá této skupince shodných hodnot.

Percentilová transformace převádí údaje do intervalu 0-100. Každému údaji je přiřazena percentilová hladina, jež odpovídá relativnímu počtu údajů (vynásobenému číslem 100), které jsou menší než tento údaj nebo stejné. Percentilová hodnota 50 odpovídá mediánu a hodnota 100 maximu původních dat u všech proměnných [30].

- Logaritmická transformace

Logaritmická transformace eliminuje pozitivní zešikmení dat.

Obecný termín škálování vystihuje, že operace transformace se týká jak jednotek veličin, tak i počátku stupnice. Škálování může být použito na znaky, na objekty nebo na obojí. Škálování by mělo zahrnout:

1. posun centra souřadného systému,
2. protažení nebo zkrácení měřítka na osách. Po posunu centra do nuly se vzálenost mezi dvěma objekty nezmění. To však neplatí při změně měřítka. Znaky před škálováním v prostoru objektů dobře oddělené mohou být po škálování totožné [43].

Zdaleka ne vždy se podaří transformace do podoby blízké normálnímu rozdělení, což ale díky robustnosti analytických algoritmů technik Data mining nemusí být vždy na závadu. Je však dobré zajistit, aby průběh hodnot dané proměnné měl jeden vrchol. Pokud to zajistit nelze, je lepší proměnnou diskretizovat [85].

3.3.4.5 Interakční proměnné

Dalším základním druhem transformací je vytváření „interakčních proměnných". Pro segmentační úlohu je typickou interakční proměnnou faktor jako výstup faktorové analýzy. V reálných segmentacích se často vyskytuje snaha najít faktory „silné" (tj. silně korelované na své určující členy a málo korelované na ostatní proměnné) a v takových případech je možné místo výběru faktoru jako segmentační proměnné zvolit proměnnou, která faktor zastupuje. Interpretace i prezentace takových modelů je pak přehlednější [85].

3.3.4.6 Užití statistických vah

Škálování eliminuje nestejný pořádek a měřítko u znaků (méně často u objektů) a tvoří znaky stejné důležitosti. Použitím statistických vah lze však zvýšit důležitost některých znaků. Užití vah je obecně potřebné v následujících případech:

- existují-li rozličné nejistoty v měřených znacích,
- pokud již máme zkušenosti o důležitosti znaků,
- jestliže existují rozličné důležitosti znaků dle účelu analýzy dat [43].

3.3.5 Rozdělení datového souboru

Při induktivním získávání znalostí se obvykle použitá data rozdělují na část trénovací a část testovací. Trénovací data se použijí ve fázi učení, testovací data pak představují příklady, které slouží k prověření získaných znalostí. V některých případech se používají dokonce tři soubory dat: data trénovací, data validační (používaná pro eventuální modifikaci znalostí získaných na základě trénovacích dat) a data testovací [2].

Např. velikost trénovací množiny pro klasifikační úlohy může být stanovena tak, že se provádí výpočty pro náhodné výběry různých rozsahů, a to tím způsobem, že se začne od náhodného výběru malého rozsahu, který se postupně zvětšuje (například se začne od 10 % a pokračuje pro 20 %, 30 % apod.). Pokud se již výsledky téměř nemění (rozdíly mezi po sobě získanými výsledky jsou menší než předem stanovená hodnota), další zvětšení souboru by nepřispělo ke zlepšení řešení [29].

V případě shlukové analýzy k rozdělení souboru na množiny nedochází a pak nastávají problémy s velkým počtem objektů. Ty jsou řešeny jednak modifikací klasických metod, jednak vývojem metod nových, přičemž k modifikaci klasických metod existuje několik přístupů.

3.4 Vybrané metody

V technikách Data mining se využívá nejen metod statistických (především vícerozměrných), které v souvislosti s výkonnější výpočetní technikou prožívají svoji renesanci, ale i metody nestatistické. Tato podkapitola je zaměřena především na první skupinu, přičemž těžiště této práce spočívá v nalezení, popsání a použití metod shlukové analýzy.

Vícerozměrné vyšetřovací techniky slouží k identifikaci vztahů ve vícerozměrných vzorcích dat. Mezi vícerozměrné techniky patří: shluková analýza, faktorová analýza, diskriminační analýza, vícerozměrné škálování, log-lineární analýza, kanonická korelační analýza, lineární a nelineární (např. Logit) regrese, analýza shody, analýza časových řad a klasifikační stromy [80]. Následující podkapitola je věnována shlukové analýze, dále budou následovat krátké charakteristiky ostatních metod.

3.4.1 Metody shlukové analýzy

Shluková analýza (Cluster Analysis) patří mezi metody, které se zabývají vyšetřováním podobnosti vícerozměrných objektů a jejich klasifikací do tříd. Hodí se zejména tam, kde objekty projevují přirozenou tendenci se seskupovat [42]. Cílem shlukové analýzy je použitím vhodných algoritmů odhalit strukturu studované množiny objektů a jednotlivé objekty klasifikovat, tzn. dosáhnout stavu, kdy objekty uvnitř shluku jsou si podobné co nejvíce a s objekty různých shluků co nejméně [83]. Pojem shluková analýza zahrnuje celou řadu metod a přístupů, jejichž cílem je nalézt skupiny podobných objektů (kromě shlukové analýzy lze ke stejnému účelu použít i metody patřící k jiným typům analýz, například k vícerozměrnému škálování) [29].

Dnešní doba dává metodám shlukové analýzy nedozírné možnosti rozvoje a aplikace [39]. Výsledky shlukování mohou sloužit, stejně jako v dalších induktivních postupech, ke dvou různým účelům, a to: predikci a deskripci. Deskripce má tendenci být tou důležitější úlohou, protože hlavní pozornost se v oblasti technik Data mining věnuje nalezení vysvětlitelných vzorů [88]. Shluková analýza může sloužit též pouze jako pomocný postup pro výběr objektů při analýze velkých datových souborů. Je-li vytvořen potřebný počet shluků objektů, pak lze analyzovat pouze data zjištěná u zástupců těchto shluků [29].

Shluková analýza používá původní data z celého měřeného souboru statistických jednotek bez jakékoli úpravy [83]. Shluková analýza nevyžaduje předem podmínky týkající se rozdělení proměnných tvořících vícerozměrnou veličinu a kvalita výsledku tudíž především závisí na tom, zda hodnocený soubor statistických jednotek má přirozenou tendenci vytvořit zřetelně odlišné podsoubory (shluky). Zprostředkovaně umožňuje shluková analýza identifikovat extrémní odchylku vícerozměrné veličiny - jednotka s hodnotami extrémně vybočující vytvoří při výpočtu „samostatný shluk" [15].

Postup při shlukování objektů lze popsat formálněji následujícím způsobem. Stejně jako v mnoha ostatních úlohách z vícerozměrné analýzy je k dispozici datová matice **X** typu n × p, kde n je počet objektů a p je počet proměnných. Uvažují se různé rozklady $S^{(k)}$ množiny n objektů do k shluků. Hledá se takový rozklad, který by byl z určitého hlediska nejvýhodnější. Připouští se pouze rozklady s disjunktními shluky (v širším slova smyslu chápaná shluková analýza řeší i úlohy spojené s pokrytím množiny objektů překrývajícími se shluky). Cílem je v podstatě dosáhnout stavu, kdy objekty uvnitř shluku jsou si podobné co nejvíce a s objekty z různých shluků co nejméně [29].

3.4.1.1 Míry vzdálenosti a podobnosti

Po provedení výběru proměnných, které budou charakterizovat vlastnosti shlukovaných objektů, a po zjištění jejich hodnot se rozhodne o způsobu hodnocení vzdálenosti či podobnosti objektů. Velmi často je první etapou realizace shlukovacího algoritmu právě výpočet příslušných měr pro všechny páry objektů. Vzniká tak symetrická čtvercová matice typu n × n, která má na diagonále nuly, jde-li o matici měr vzdálenosti **D**, nebo jedničky, jde-li o matici měr podobnosti **A**. Uložení matice v paměti počítače může být při velkém počtu objektů problémem, který ovlivní volbu algoritmu.

V úlohách, v nichž jsou jednotlivé proměnné zhruba na stejné úrovni nebo jsou alespoň vyjádřeny ve stejných měřicích jednotkách, lze použít *Hemmingovu vzdálenost* D_H (v současné literatuře a v programových systémech je uváděna pod názvem *Manhattan* nebo *city-block*) nebo *euklidovskou vzdálenost* D_E nebo *Čebyševovu vzdálenost* D_C.

$$D_H\ (x_i, x_{i'}) = \sum_{j=1}^{p} \left| x_{ij} - x_{i'j} \right| \tag{4}$$

$$D_E\ (x_i, x_{i'}) = \sqrt{\sum_{j=1}^{p} (x_{ij} - x_{i'j})^2} \tag{5}$$

$$D_C\ (x_i, x_{i'}) = \max_j \left| x_{ij} - x_{i'j} \right|. \tag{6}$$

Všechny uvedené míry mají stejné nevýhody; jde o již zmíněnou závislost na použitých měřicích jednotkách, která někdy brání smysluplnému pořízení jakéhokoli součtu pro různé proměnné, ale také o to, že jsou-li proměnné uvažovány v součtu se stejnými vahami, silně korelované proměnné mají nepřiměřeně velký vliv na výsledek. V odborné literatuře jsou popsány další míry vzdálenosti a podobnosti objektů, z nichž lze pro kvantitativní proměnné uvést ještě *Lanceyovu-Williamsovu vzdálenost* $D_{LW}\ (x_i - x_{i'})$. Z měr podobností lze zmínit *Jaccardův koeficient* $A_J\ (x_i - x_{i'})$ [29].

Všem znakům je tedy většinou potřeba dát předem stejnou váhu. Výběr množiny proměnných rozhoduje o úspěchu analýzy a je nutné mu věnovat náležitou pozornost. Pokud některé použité jednotky měření způsobují, že se určité znaky jejich vlivem jeví jako dominující a jiné jen velmi málo ovlivňují průběh shlukování, pak je třeba upravit data tak, aby všechny znaky byly souměřitelné. Jedním ze způsobů, jak docílit této souměřitelnosti znaků, je *standardizace dat*. Rozhodnutí o tom, jakým způsobem mají být data transformována, závisí především na zkušenostech nebo záměrech uživatele a na shlukovacím postupu [39].

Shlukovat lze nejen objekty, ale také proměnné. Existují i metody, které umožňují shlukovat současně objekty i proměnné, případně současně kategorie dvou proměnných. Podobnost proměnných se nejčastěji zjišťuje pomocí výběrového *korelačního koeficientu* $A_r\ (x_j, x_{j'})$. Obvykle je třeba převést získanou matici na matici nepodobností. Existují dva přístupy, podle interpretace hodnoty -1. V případě, kdy hodnota -1 reprezentuje maximální nesouhlas, platí vztah D = 1 - A. Pokud jsou ovšem hodnoty -1 a 1 uvažovány jako maximální souhlas mezi proměnnými, pak lze použit buď D = 1 - A^2, nebo D = 1 -

|A|. Pro některé aplikace se doporučuje použít jako míru podobnosti kosinus úhlu mezi příslušnými dvěma vektory. Tato *kosinová míra* $A_c(x_j, x_{j'})$ je speciálním případem výběrového korelačního koeficientu, kdy jsou výběrové průměry u obou sledovaných proměnných rovny hodnotě 0.

Z nejužívanějších měr vzdálenosti, popř. podobnosti pro alternativní data lze uvést *koeficient prosté shody*, *Russehiv a Raoův koeficient*, *Jaccardův koeficient*. Používají se též některé z měr uvedených pro kvantitativní proměnné. Pokud jsou vlastnosti objektů popsány nominálními nebo ordinálními proměnnými, lze je převést na skupinu alternativních proměnných a použít některou z výše uvedených měr. Existují také speciální přístupy, které vytváření pomocných proměnných nevyžadují. Záleží ovšem na konkrétním programovém systému, zda takové možnosti poskytuje, či nikoli. Pokud je cílem shlukové analýzy nalézt skupiny podobných kategorií nominální proměnné, pak lze použít speciální koeficienty založené na chí-kvadrát statistice. Kromě *základní chí-kvadrát míry nepodobnosti* je možné použít koeficient ϕ [29].

3.4.1.2 Optimalizační kritéria

Základním cílem shlukové analýzy je vytvořit kompaktní a dobře separované shluky. Tento cíl lze konkretizovat pomocí objektivního kriteria kvalitního shlukování. Tzv. *funkcionály kvality rozkladu* umožňují posoudit kvalitu výsledku. Obecně se tedy požaduje, aby pro daný počet shluků *k* bylo dosaženo extrému některého funkcionálu. Hebák s Hustopeckým uvádí, že obvykle to bývá minimum *Wardova kritéria* $\mathbf{G_1}$, jež představuje minimum součtu čtverců odchylek všech hodnot od příslušných shlukových průměrů (tedy součet čtverců v nově vznikajícím shluku, zmenšený o součty čtverců v obou zanikajících shlucích) [27].

$$G_1 = st\ E = \sum_{h=1}^{k}\sum_{i=1}^{n_h}\sum_{j=1}^{p}(x_{hij} - \bar{x}_{hj})^2 . \qquad (7)$$

Při požadavku dosažení nezávislosti na použitých měřicích jednotkách lze doporučit minimalizaci determinantu matice vnitroshlukové variability $\mathbf{G_2}$ = |E| nebo maximalizaci stopového kritéria $\mathbf{G_3}$ = st $|BE^{-1}|$ popř. $\mathbf{G_4}$ = st $|BT^{-1}|$. Matice vnitroshlukové variability se značí **E** a matice mezishlukové variability se značí **B**, v součtu dávají matici celkové variability **T**.

$$E = \sum_{h=1}^{k}\sum_{i=1}^{n_h}(x_{hi} - \bar{x}_h)(x_{hi} - \bar{x}_h)^T \qquad (8)$$

$$B = \sum_{h=1}^{k} n_h\ (\bar{x}_h - \bar{x})(\bar{x}_h - \bar{x})^T \qquad (9)$$

$$T = \sum_{h=1}^{k}\sum_{i=1}^{n_h}(x_{hi} - \bar{x})(x_{hi} - \bar{x})^T \qquad (10)$$

Uvedená kritéria se nepoužívají jen retrospektivně k vyhodnocení kvality provedeného rozkladu, ale změny jejich hodnoty mohou být vodítkem pro tvorbu shluků [29]. Ne vždy

je vhodné volit počet shluků „náhodně", ale v některých případech je nutné využít i dalších statistických metod, které zjistí optimální počet *k*. Mezi tyto metody se řadí např. metoda hlavních komponent (a to buď formou scattergramů komponentních skóre, nebo formou proměnných v Andrewsových grafech) [17].

Za kvalitní výsledek lze označit řešení, při kterém jsou si všechny jednotky ze stejného shluku navzájem podobnější (bližší) než kterékoli dvě jednotky z rozdílných shluků [14]. Rozklad, který by tuto podmínku splnil, lze považovat za optimální [27].

Spolehlivou cestou k nalezení optimálního rozkladu by bylo probrání všech možných variant rozkladu s výpočtem hodnoty G_1 pro každou z nich. V reálných úlohách je však možných variant rozkladu příliš mnoho. Počet způsobů $S^{(k)}$, kterými lze n objektů rozdělit do k shluků, udává Stirlingovo číslo 2. druhu

$$S^{(k)} = \frac{1}{k!}\sum_{h=0}^{k}(-1)^{k-1}\binom{k}{h}h^n \qquad (11)$$

Při nepředepsaném k lze určit počet způsobů G, kterými lze n objektů rozdělit do shluků podle

$$S = \sum_{k=1}^{n} S^{(k)} \qquad (12)$$

Proto se v praxi využívají algoritmy, který zaručují nalezení alespoň lokálního extrému zvoleného funkcionálu kvality rozkladu [29].

3.4.1.3 Stanovení optimálního počtu shluků

Jedním z nejobtížnějších úkolů ve shlukové analýze je nalézt vhodný počet shluků. Pro určení počtu shluků pro jakýkoli typ shlukové analýzy neexistuje žádná vyhovující metoda [68], [59]. V literatuře se objevuje celá řada různých kritérií. Nejčastěji používané optimalizační kritérium pro rozdělení pozorování do shluků je známé jako vnitroshlukový součet čtverců odchylek, chyba součtu čtverců odchylek, reziduální součet čtverců, metoda nejmenších čtverců, (minimální) čtvercová chyba, (minimální) rozptyl, součet čtverců euklidovských vzdáleností, stopa (E), (podíl) vysvětlovaného rozptylu nebo R^2. Mnoho algoritmů bylo navrženo pro maximalizaci R^2 nebo podobných kritérií [59].

Index R^2 - RSQ

Nejjednodušším příkladem zvolení počtu shluků je na základě dendrogramu, v němž mohou být v některých případech znázorněny výrazné shluky [29]. Prvním intuitivním kritériem dobré kvality shlukování je vzdálenost sloučených shluků v každém kroku procesu shlukování. Proces může být zastaven, když vzdálenost neočekávaně vzroste. Nejčastěji používané kritérium R^2 je založeno na rozkladu celkového rozptylu p proměnných, jako ve Wardově metodě. Myšlenkou je mít nízkou variabilitu ve shluku (E) a vysokou variabilitu mezi shluky (B). Pro rozdělení do k shluků se index vyjadřuje vztahem

$$R^2 = 1 - \frac{E}{T} = \frac{B}{T}, \tag{13}$$

kde T = E + B a index $R^2 \in \langle 0, 1 \rangle$. Jestliže se hodnota R^2 blíží 1, znamená to, že odpovídající rozdělení je optimální, protože pozorování patřící do stejného shluku si jsou velmi podobná (nízké E) a shluky jsou dobře separovány (vysoké B). Adekvátně tomu kvalita shlukování klesá jak se R^2 blíží 0.

Pozn. $R^2 = 0$, když se vytvoří pouze jeden shluk; $R^2 = 1$, když je shluků stejně jako počet pozorování. Jak počet shluků roste, tak stoupá homogenita v rámci shluku (každý shluk obsahuje méně objektů) a též R^2. To ale vede ke snížení úspornosti při shlukování. Tudíž maximalizace R^2 nemůže být považována za jediné vhodné kritérium pro stanovení počtu shluků. To by nakonec vedlo ke shlukování do n shluků majících po jednom pozorování (kde $R^2 = 1$) [24].

Kritérium Pseudo-F statistika

Obvyklou mírou jež doplňuje R^2 je kritérium Pseudo-F statistika. To je definováno vztahem

$$F_k = \frac{B/(k-1)}{E/(n-k)}. \tag{14}$$

Obecně F_k klesá s počtem shluků *k* vzhledem k tomu, že variabilita mezi shluky by měla klesat a variabilita ve shlucích by měla růst. Náhlý pokles značí, že jsou spojovány velmi odlišné shluky. Výhodou kritéra Pseudo-F je, že je možné objasnit způsob tvorby rozhodovacího pravidla, které umožňuje přijmout tj. nezamítnout nulovou hypotézu o sloučení shluků nebo upřednostnit zastavení procesu (tj. alternativní hypotézu). Toto rozhodovací pravidlo je stanoveno intervalem spolehlivosti na základě F rozdělení, s (k - 1) a (n - k) stupni volnosti. Při použití tohoto rozhodovacího pravidla se ale předpokládá, že pozorování mají (přibližně) normální rozdělení, což trochu snižuje jeho výhody [24].

Střední kvadratická směrodatná odchylka – RMSSTD

Alternativou k R^2 je střední kvadratická směrodatná odchylka (root mean square standard deviation – RMSSTD. Ta bere v úvahu pouze část variability v dalších shlucích vytvořených v každém kroku hierarchického shlukování. RMSSTD je definována

$$RMSSTD = \sqrt{\frac{E_h}{p\,(n_h - 1)}}, \tag{15}$$

kde h vyjadřuje h-tý krok shukování (h = 2, …, n-1) a E_h vyjadřuje variabilitu ve shluku vytvořeném v h-tém kroku procesu, p je počet proměnných. Prudký růst RMSSTD z jednoho kroku na další znázorňuje, že dva shluky, které byly spojeny, jsou silně heterogenní a tudíž by bylo vhodné zastavit proces na dřívějším kroku [24].

Semiparciální R^2 – SPRSQ

Jiný index, podobný RMSSTD, měří „další" přínos h-tého kroku shlukování a je označován jako semiparciální R^2 – SPRSQ. Je definován vztahem

$$SPRSQ = \frac{E_h - E_r - E_s}{T}, \tag{16}$$

Kde h je nový shluk získaný v kroku h sloučením shluků r a s. Index SPRSQ měří růst vnitroshlukové variability E získané sloučením shluků r a s. Náhlý vzrůst SPRSQ indikuje, že jsou spojeny heterogenní shluky, a tak je vhodné zastavit shlukování v předchozím kroku [24].

Kubické shlukovací kritérium (Cubic Clustering Criterion)

V systému SAS se používají na základě výsledků studií autorů Milligen a Cooper (1984, 1985) tři kritéria: pseudo F statistika, pseudo t^2 statistika a kubické shlukovací kritérium (Cubic Clustering Criterion – CCC). Tato kritéria jsou vhodná pouze pro kompaktní nebo mírně protáhlé shluky, nejlépe shluky, jež jsou zhruba vícerozměrně normální [68].

Kritérium CCC se počítá porovnáním zjištěné hodnoty R^2 s přibližnou očekávanou hodnotou R^2 použitím přibližného rozptylu stabilizovaného transformací. Kladné hodnoty kubického shlukovacího kritéria znamenají, že zjištěné R^2 je větší než byly očekávané, jestliže výběr pocházel z rovnoměrného rozdělení, a tudíž indikují možnou přítomnost shluků. Vyjádření CCC jako standardního běžného testového kritéria poskytuje hrubý test hypotéz:

- H_0: data byla vybrána z rovnoměrného rozdělení o jednom hyperboxu
- H_A: data byla vybrána ze směsi kulatých vícerozměrných normálních rozdělení se shodnými rozptyly a stejnými pravděpodobnostmi výběrů.

Na základě této alternativní hypotézy je R^2 ekvivalentní k maximálněpravdě-podobnostnímu kritériu.

Kritérium CCC je založeno na předpokladu, že shluky získané z rovnoměrného rozdělení o jednom hyperboxu jsou hyperkrychle stejné velikosti. Předpoklad hyperkrychle je evidentně chybný ve většině případů, ale je všeobecně konzervativní, když ve dvou a více dimenzích není počet shluků velmi rozsáhlý. Bylo dokázáno, že tvar shluku má sklon být hexagonálním (pro mnoho shluků ve dvou dimenzích z rovnoměrného rozdělení).

Kubické shlukovací kritérium CCC se spočítá ze zjištěné hodnoty R^2 jako

$$CCC = \ln\left[\frac{1 - E\left(R^2\right)}{1 - R^2}\right] \frac{\sqrt{\frac{np^k}{2}}}{(0{,}001 + E\,(R^2))^{1,2}}. \tag{17}$$

Výše uvedený vzorec byl odvozen na základě zkušeností (empiricky) z pokusu stabilizovat rozptyl přes počty pozorování, proměnných a shluků [59].

Nejlepší použití kritéria CCC je zobrazit jeho hodnoty ve vztahu k počtu shluků, jež se pohybují se od jednoho shluku až do přibližně jedné desetiny počtu pozorování. Kritérium CCC se nemůže chovat rozumně, jestliže je průměrný počet pozorování na shluk menší než deset. Pro interpretaci CCC mohou být jako vodítko použity následující informace:

- Vrcholky v grafu s kritériem CCC větším než 2 či 3 indikují dobré shlukování.

- Vrcholky s kritériem CCC mezi 0 a 2 indikují potenciální shluky, ale měly by se interpretovat opatrně.
- Jestliže mají data hierarchickou strukturu, pak se může vytvořit několik vrcholků.
- Z velmi odlišných nehierarchických kulatých shluků obvykle vyplývá ostrý vzestup před vrcholem s následným pozvolným snižováním.
- Velmi odlišné nehierarchické elipsovité shluky často vykazují ostrý růst ke správnému počtu shluků s následným dalším pozvolným růstem a nakonec pozvolným poklesem.
- Jestliže jsou všechny hodnoty kritéria CCC negativní a snižují se pro 2 a více shluků, pak rozdělení je pravděpodobně jednomodální (s jedním módem) nebo s dlouhými konci.
- Velmi negativní hodnoty kritéria CCC (např. –30) vypovídají, že to je možná kvůli odlehlým pozorováním.
- Jestliže kritérium CCC nepřetržitě roste stejně jako roste počet shluků, pak rozdělení může být zrnité nebo data mohou být nadměrně zaokrouhlena nebo zaznamenána s malým počtem číslic (popř. desetinných míst).

Závěrečné a velmi důležité upozornění: ani CCC ani R^2 není vhodným kritériem pro shluky, které jsou vysoce zašpičatělé nebo nepravidelně tvarované [59].

Další indexy

Dále Hebák a kol. uvádí ještě další tři vybraná pravidla – indexy - pro stanovení počtu shluků

$$G_5 = \frac{\left(\frac{B}{k-1}\right)}{\left(\frac{E}{n-k}\right)}, \tag{18}$$

kde E je vnitroskupinový součet čtvercových vzdáleností vzhledem k centroidům a B je meziskupinový součet čtvercových vzdáleností,

$$G_6 = \frac{\left(S_+ - S_-\right)}{\left(S_+ + S_-\right)}, \tag{19}$$

kde S_+ označuje počet konkordantních srovnání a S_- počet diskordantních srovnání (jestliže vnitroskupinová nepodobnost je menší než meziskupinová nepodobnost, pak je srovnání konkordantní, pokud je větší, je diskordatntní) a

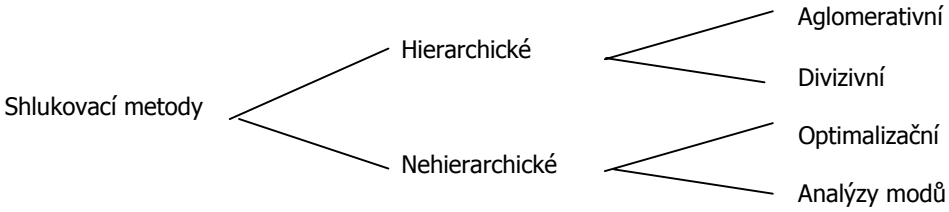
$$G_7 = \frac{\left(D(k) - D_{\min}\right)}{\left(D_{\max} - D_{\min}\right)}, \tag{20}$$

kde D(k) je součet všech vnitroskupinových nepodobností při rozdělení objektů do k shluků.

Počet shluků se na základě globálních pravidel stanovuje tak, že v případě indexů G_5 a G_6 je to maximální z vypočítaných hodnot a v případě indexu G_7 minimální (počítají se všechny varianty pro počty shluků menší než zadaná hodnota) [29].

3.4.1.4 Shlukovací metody a postupy

Existuje celá řada shlukovacích metod a postupů, protože měr podobnosti statistických jednotek a podobnosti („vzdálenosti") shluků je několik. Nejčastěji se rozlišují shlukovací metody podle cílů, k nimž směřují, do dvou základních skupin metod: hierarchické a nehierarchické metody shlukové analýzy [39].



Obrázek 6: Základní skupiny metod shlukové analýzy [39]

V literatuře se lze setkat s různými klasifikacemi metod pro shlukování, většina z nich se však shoduje v tom, že existují metody hierarchické; rozdělovací (vycházející z počátečního rozdělení buď celé množiny objektů nebo její podmnožiny do požadovaného počtu shluků např. metoda k-průměrů); metody založené na hustotě, mřížce a modelu. Pro rozsáhlé soubory dat jsou však používány jejich modifikace. Kromě toho existují algoritmy, které integrují myšlenky různých shlukovacích přístupů. Některé shlukovací metody byly vyvinuty ke speciálnímu účelu, kterým je zjišťování odlehlých pozorování [29].

3.4.1.5 Hierarchické shlukovací techniky

Hierarchické shlukovací techniky se provádí buď na základě řady postupného seskupování (aglomerativní přístup) nebo na základě řady postupného rozdělování (divizivní přístup) [32], [39].

Aglomerativní hierarchický postup

Aglomerativní hierarchický postup (též AGNES, AGlomerative NESting) je založen na postupném spojování objektů a jejich shluků do větších shluků. V prvé fázi je vypočtena základní matice vzdáleností mezi objekty. Dále se podle hodnot vícerozměrné veličiny spojují vždy dva nejpodobnější objekty (později objekt a shluk a poté dva nejpodobnější shluky) do jednoho podsouboru, tzv. shluku a opět je vypočtena matice vzdáleností pro nově vytvořené shluky. Celý postup je pak opakován tak dlouho, pokud není dosaženo zadaného počtu shluků [27], [83]. Řešitel může totiž zpravidla subjektivně rozhodnout, do kolika podsouborů (shluků) má být soubor v posledním kroku - použitém pro interpretaci výsledku - rozdělen [15].

Jako nejčastěji zmiňované aglomerativní hierarchické shlukovací procedury uvádí Hebák s Hustopeckým následujících pět algoritmů:

- **metodu nejbližšího souseda** (Nearest Neighbour, Single, *jednoduché spojení*), kde kritériem pro spojování shluků je minimum z možných

mezishlukových vzdáleností objektů. Metoda tvoří nový shluk na základě nejkratší vzdálenosti mezi shluky a neumí proto rozlišit špatně separované shluky. Při aplikaci této metody se často i značně vzdálené objekty mohou sejít ve stejném shluku, pokud větší počet dalších objektů mezi nimi vytvoří jakýsi most. Toto charakteristické řetězení objektů se považuje za nevýhodu, zvláště je-li důvod požadovat, aby shluky měly obvyklý eliptický tvar se zhutněným jádrem. Jinak má metoda velký počet příznivých vlastností, je to jedna z mála metod, která umí roztřídit a rozlišit i neeliptické shluky [42], [29].

- **metodu nejvzdálenějšího souseda** (Furthest Neighbour, Complete, úplné vazby, *úplné spojení*), která počítá vzdálenost dvou shluků jako maximum z možných mezishlukových vzdáleností. Probíhá podobně jako metoda nejbližšího souseda s jednou důležitou výjimkou: vzdálenost mezi shluky je určována vzdáleností mezi dvěma nejvzdálenějšími objekty, každý je přitom z jiného shluku [42]. Nežádoucí řetězový efekt zde odpadá, naopak je tu tendence ke tvorbě kompaktních shluků, nikoli mimořádně velkých [29].
- **metodu průměrovou** (Average, Sokalova-Sneathova), kdy se vzdálenost dvou shluků vypočte jako průměr z možných mezishlukových vzdáleností objektů. Metoda vede často k podobným výsledkům jako metoda nejvzdálenějšího souseda.
- **metodu centroidní** (Centroid, Gowerova), zde se vzdálenost shluků počítá jako euklidovská vzdálenost průměrů proměnných v jednotlivých shlucích – centroidů,
- **Wardovu metodu**, kde kritériem pro spojování shluků je přírůstek celkového vnitroshlukového součtu čtverců odchylek pozorování od shlukového průměru (21). Wardova metoda má tendenci odstraňovat malé shluky, tedy tvořit shluky zhruba shodné velikosti, což je často vítaná vlastnost [Hebák]. Wardova metoda je hierarchickým předchůdcem nehierarchických shlukovacích metod, které optimalizují určité kritérium rozkladu [32].

$$\Delta G_1 = \frac{n_h n_{h'}}{n_h + n_{h'}} \sum_{j=1}^{p} (\bar{x}_{hj} - \bar{x}_{h'j})^2 \qquad (21)$$

Dále se ještě uvádí **mediánová metoda** (Median), která je považována za vylepšení centroidní metody. Důvodem jejího zavedení byla snaha odstranit nedostatek centroidní metody [39]. Tím jsou rozdílné váhy, které centroidní metoda dává různě velkým shlukům [42].

Meloun a kol. (2005) zmiňují ještě **metodu těžiště**. U této metody jde o vzdálenost dvou těžišť shluků vyjádřených euklidovskou vzdáleností nebo čtvercem euklidovské vzdálenosti. Těžiště shluku má souřadnice odpovídající průměrným hodnotám objektů pro jednotlivé znaky. Po každém kroku shlukování se počítá nové těžiště. Poloha těžiště shluku poněkud migruje tak, jak se připojují nové objekty a vznikají větčí shluky. Mohou se objevit také zmatečné shluky. Výhodou této metody je menší ovlivnění odlehlými body, než je tomu u ostatních hierarchických metod [43].

Divizivní hierarchický postup

Divizivní hierarchický postup (též DIANA, DIvisive ANAlysis) pracuje v opačném směru než postupy aglomerativní. Všechna pozorování tvoří jeden samostatný počáteční shluk, který je rozdělen do dvou podskupin tak, že objekty v jedné podskupině jsou vzdáleny objektům ve skupině druhé. Tyto podskupiny jsou dále rozdělovány do nepodobných podskupin; tento postup pokračuje tak dlouho, dokud není tolik podskupin jako objektů nebo dokud není dosaženo požadovaného počtu shluků [32].

Výsledky obou hierarchických shlukovacích postupů lze výhodně zachytit graficky v podobě *stromu* (dendrogramu). Na vodorovnou osu se uvádí stupnice pro hladinu spojování. Vlevo začíná strom *n* větvemi a v každém kroku se spojují dvě větve v bodě, který odpovídá příslušné hladině spojení [27].

Výhodou hierarchických metod je nepotřebnost informace o optimálním počtu shluků v procesu shlukování; tento počet se určuje až dodatečně [42]. Při shlukování vznikají pouze dva základní problémy, prvním je způsob vyjádření podobnosti mezi objekty a druhým je volba vhodné shlukovací procedury [43].

Hierarchické metody poskytují rozdílné počty shluků v závislosti na úrovni abstrakce uživatele. Je mnoho způsobů, jak definovat rozdělení stromu, který je vytvořen hierarchickým shlukovacím algoritmem. Nejčastěji se používá rozříznutí stromu horizontálně. Standardní software takové řezy vytváří, jestliže uživatel definuje parametr: buď počet požadovaných tříd nebo nejvyšší úroveň uzlu, při níž má dojít k řezu. Sofistikovanější software pomáhá uživateli nalézt nejvyšší úroveň řezu výběrem hodnoty ze seznamu [12].

Pro doplnění celkového přehledu lze uvést ještě dvě grafické metody zkoumání podobnosti objektů - Sun Ray Plot a Star Symbol Plot. Tyto metody slouží k rychlému vizuálnímu posouzení podobnosti zkoumaných objektů. Každý objekt je zde znázorněn graficky. Tyto grafické metody jsou vhodné pro posouzení podobnosti u menšího počtu objektů, při větším počtu již grafické znázornění ztrácí přehlednost [83].

3.4.1.6 Modifikace hierarchických metod

K novým přístupům, jež jsou založeny na hierarchických algoritmech, patří metody frakcionizace, refrakcionizace, metoda BIRCH, dvoukroková shluková analýza a některé další postupy.

Frakcionizace a refrakcionizace

V metodách vycházejících z hierarchických algoritmů se lze setkat s pojmy *frakcionizace* a refrakcionizace. První přístup spočívá v rozdělení datového souboru do podsouborů (frakcí) a aplikování hierarchické metody na každou frakci. Shluky vzniklé ve frakcích jsou dále shlukovány do k skupin stejnou metodou shlukové analýzy (počet skupin k musí být stanoven předem). Zbylé objekty jsou přiřazeny do vytvořených k shluků na základě centroidů.

Frakcionizace je však spojena s některými problémy (zaprvé je potřeba předem specifikovat počet shluků, za druhé shluky vzniklé v určité frakci jsou převedeny na tzv. metaobjekty, které nemusí být vhodným reprezentantem skupiny). Proto byl navržen zdokonalený algoritmus nazývaný refrakcionizace, který se liší tím, že shluky vzniklé

frakcionizací vytvářejí frakce pro následující iteraci. Součástí algoritmu je odhad počtu shluků *k*.

Metoda BIRCH

Dále z hierarchického přístupu vychází třída metod **BIRCH** (*Balanced Iterative Reducing and Clustering using Hierarchies*). Algoritmus je založen na podobném principu, na kterém je založena frakcionizace. Základní myšlenkou třídy algoritmů BIRCH je, že při iterativní optimalizaci shluků není potřebné opakovaně procházet všechny věty původního souboru, jak je tomu typicky např. u algoritmu K-means, ale o jednotlivých aktuálních shlucích postačí určitá statistická informace uchovávaná ve formě vyváženého stromu (tzv. CF tree – Cluster Features tree) [87]. Objekty se uspořádají do podshluků, které jsou charakterizovány pomocí shlukovacích vlastností označovaných zkratkou CF (*Cluster Features*). Tyto podshluky se pak shlukují do *k* skupin pomocí tradiční hierarchické shlukové analýzy. Nevýhodou této metody je citlivost na pořadí objektů.

Shlukovací vlastnost je tříprvkový vektor, přičemž první hodnotou je počet objektů v daném podshluku, druhým prvkem je vektor, jehož každý prvek vyjadřuje součet hodnot příslušné proměnné (rozměr vektoru odpovídá počtu analyzovaných proměnných), a třetím prvkem opět vektor, jehož každý prvek vyjadřuje součet druhých mocnin hodnot příslušné proměnné. K vytváření a uchování shlukovacích vlastností slouží CF-strom, což je určitá forma víceúrovňové komprese dat. Tento strom se skládá z listů a nelistových uzlů, přičemž každý uzel je charakterizován určitou shlukovací vlastností. Nelistový uzel má nejvýše B potomků a list obsahuje nejvýše L vstupů. Při vytváření CF-stromu je každý objekt umístěn k nejpodobnějšímu vstupu do listu [29].

Dvoukroková shluková analýza

Princip popsaný u třídy metod BIRCH je základem procedury označované jako *dvoukroková shluková analýza* (*TwoStep Cluster analysis*). Tato metoda může být použita jak pro kvantitativní spojité, tak pro kategoriální proměnné. Dvoukroková shluková analýza je implementována v SPSS (v Clementine i v SPSS Base), kde je její součástí též možnost stanovit počet shluků (tj. možnost zadání nalezení optimálního počtu shluků podle zvoleného informačního kritéria [87]). Datovým souborem se prochází pouze jednou. Jestliže u určitého objektu není uvedena některá hodnota, je tento objekt (řádek datové matice) vynechán [29].

Jak naznačuje název, algoritmus má dva kroky. V *prvním kroku* se objekty shlukují do malých shluků (podshluků), jejichž počet je podstatně menší než počet objektů původního souboru. Objekty vstupují po sobě a hodnotí se, zda mohou být zařazeny do již vytvořeného shluku, nebo zda bude vytvořen nový shluk. Je proto vhodné, aby objekty byly náhodně uspořádány.

Algoritmus vytváří modifikovaný CF-strom, který se skládá z několika úrovní uzlů a každý uzel obsahuje určitý počet vstupů. Listy zahrnují konečné podshluky. Každý vstup je popsán charakteristikou CF, která se skládá z počtu objektů vstupu, střední hodnoty a rozptylu každé spojité proměnné a četností každé kategorie každé kategoriální proměnné.

Při vkládání objektu se postupuje následujícím způsobem. Pokud se objekt nachází v rámci prahové vzdálenosti od určitého (nejbližšího) podshluku, vstupuje do tohoto podshluku, přičemž se aktualizuje příslušná CF charakteristika. V opačném případě daný objekt vytváří vlastní podshluk. Pokud již v příslušném listu není prostor pro zařazení dalšího vstupu, je tento podshluk rozdělen na dva, a to podle nejvzdálenějších vstupů.

Ostatní vstupy se redistribuují na základě kritéria blízkostí. Pokud CF-strom roste za povolenou maximální velikost, je strom přestavěn zvětšením prahové vzdálenosti.

Všechny objekty spadající do stejné skupiny jsou reprezentovány souhrnně pomocí CF charakteristiky. Po zařazení nového objektu do vstupu se tato charakteristika přepočítává, a to na základě nového objektu a původní CF charakteristiky. Děje se tak bez znalostí individuálních objektů ve vstupu, což snižuje nároky na operační paměť. Součástí algoritmu může být zjišťování odlehlých objektů, tj. objektů, které se nehodí do žádného shluku. Je-li počet objektů ve vstupu menší než stanovený podíl (např. 25 %) velikosti největšího vstupu listového uzlu v CF-stromu, jsou tyto objekty považovány za odlehlé. CF-strom je přestavěn bez těchto objektů, a poté se zkoumá, zda by mohly být znovu zařazeny. Výsledkem prvního kroku je nová datová matice, kde v řádcích jsou charakteristiky jednotlivých podshluků (kromě odlehlých).

Ve *druhém kroku* jsou vzniklé podshluky shlukovány do stanoveného počtu shluků. Protože počet podshluků je podstatně menší než počet objektů původního souboru, mohou být již využity tradiční metody shlukování [29]. Výsledkem je to, že věty, které mají být shlukovány, jsou procházeny typicky dvakrát – což tento algoritmus činí obzvláště výhodným pro Data mining aplikace [87].

Jak v prvním, tak ve druhém kroku se používá míra vzdálenosti. *Euklidovská vzdálenost* může být použita pouze v případě, že všechny proměnné jsou kvantitativní spojité. Vzdálenost mezi dvěma shluky představuje vzdálenost mezi jejich centroidy. *Míru nepodobnosti typu věrohodnostní poměr* lze použit jak pro kvantitativní spojité, tak pro kategoriální proměnné (pro kategoriální je to jediná možnost). Vzdálenost mezi dvěma shluky je spojena s poklesem míry věrohodnostního poměru, jenž nastává při spojení dvou shluků do jednoho. Předpokladem pro použití této míry je normální rozdělení pro spojité proměnné a multinomické rozdělení pro kategoriální proměnné. Dále se předpokládá nezávislost pro každou dvojici proměnných [29].

Pokud jde o přiřazení objektů ke shlukům, pak v případě, že nejsou sledovány odlehlé shluky, je objekt přiřazen podle míry vzdálenosti k nejbližšímu shluku. V opačném případě je při použití euklidovské vzdálenosti objekt přiřazen k nejbližšímu neodlehlému shluku, jestliže euklidovská vzdálenost je menší než kritická hodnota. Pokud tato podmínka splněna není, je objekt označen jako odlehlý [29]. Podle nastavení většího nebo menšího počtu procenta odlehlých hodnot se vytváří početnější či méně početný „Outlier cluster" [87].

Při použití míry typu věrohodnostní poměr se předpokládá, že odlehlá pozorování mají rovnoměrné rozdělení. Tato míra se počítá jak na základě přiřazení objektu k odlehlému shluku, tak na základě přiřazení objektu k neodlehlému shluku. Objekt je přiřazen ke shluku, u něhož byla získána větší hodnota použité míry. Tento postup je ekvivalentní přiřazení objektu k nejbližšímu neodlehlému shluku, jestliže vypočtená vzdálenost je menší než kritická hodnota [29].

Společnost SPSS přidala přímo do funkce Two Step i vytvoření chybového grafu, který společně s grafem significance jednotlivých segmentačních proměnných pro dané shluky urychluje a zpříjemňuje interpretaci nalezených segmentů.

Oproti K-means je Two Step v SPSS více parametrizovatelný, což někdy může vést paradoxně k obtížnějšímu nalezení nejlepších parametrů. Jedním, zdaleka nikoli jediným

z postupů (velmi záleží na charakteristice vstupních dat), který může být užitečný, je tento:

1. obvyklým způsobem ověřit segmentační proměnné na průběh a závislost, realizovat transformace,
2. vypnout nebo nastavit zpracování odlehlých hodnot na minimální hodnotu,
3. s využitím BIC nebo AIC kritérií se pokusit nalézt optimální počet shluků,
4. pro interpretaci shluků využít nabídnuté grafy, zkusit zvýšit představenou hladinu významnosti např. na 99 procent,
5. vypnout parametr hledání optimálního počtu shluků – zvolit optimálně nalezený v kroku 3 nebo jemu blízký,
6. zvyšovat parametr procenta odlehlých hodnot a sledovat vliv na stabilitu jednotlivých shluků vs. velikost „Outlier" shluku.

Two Step algoritmus patří v současnosti mezi velmi nadějné algoritmy pro shlukování. Jeho implementace v SPSS je poměrně čerstvá, ale jeví se jako slibná. Two Step má velkou šanci nahradit v řadě případů dnes převládající metodu K-means [87].

Další metody

Dalšími metodami založenými na hierarchických algoritmech jsou CURE, ROCK a Chameleon. Podstata algoritmu **CURE** (*Clustering Using REpresentatives*) spočívá v tom, že každý shluk má c reprezentantů. Nejdříve se provede náhodný výběr objektů, které se rozdělí do frakcí. V každé z těchto frakcí se provede hierarchická shluková analýza. Poté se identifikují odlehlé objekty a na základě vzniklých pomocných shluků se vytvoří požadovaný počet konečných shluků. V procesu shlukování se využívá kd-strom. V poslední fázi se každý z dosud neanalyzovaných objektů přiřadí ke shluku, který obsahuje reprezentanta nejbližšího danému objektu.

Chameleon je dvoufázový algoritmus. Jeho podstatou je hierarchické shlukování, které používá dynamické modelování. V první fázi je aplikován nehierarchický algoritmus, který shlukuje objekty do velkého počtu relativně malých podshluků. Cílem druhé fáze je najít shluky opakovaným kombinováním podshluků.

Metoda **ROCK** (*RObust Clustering using linKs*) je určena pro kategoriální proměnné. Stejně jako u algoritmu CURE se nejdříve provede náhodný výběr objektů, které se shlukují do požadovaného počtu shluků, po čemž následuje přiřazení zbylých objektů. Základními používanými prostředky jsou přitom sousedé a vazby (links). Soused určitého bodu je takový bod, pro který platí, že jeho podobnost se sledovaným bodem je rovna nebo větší než stanovená prahová hodnota. Vazba mezi dvěma body je definována jako počet společných sousedů těchto bodů. Podstata metody ROCK spočívá v maximalizaci kriteriální funkce, která zohledňuje jednak maximalizaci součtů vazeb pro objekty patřící do stejného shluku, jednak minimalizaci součtů vazeb pro objekty z různých shluků.

ROCK je hierarchický shlukovací algoritmus. Dvojice shluků, pro kterou výše uvedená míra nabývá maximální hodnoty, je v daném kroku nejvhodnější dvojicí pro shlukování. V konečné fázi jsou zbylé objekty přiřazeny k vytvořeným shlukům, což se provádí následujícím způsobem. Z každého h-tého shluku je vybráno L_h objektů, podle kterých se mají zbylé objekty zařazovat. Každý zbylý objekt je přiřazen k tomu shluku, v němž má nejvíce sousedů z L_h objektů (po normalizaci) [29].

3.4.1.7 Nehierarchické (rozdělovací) shlukovací techniky

Nehierarchické metody mohou být používány na mnohem rozsáhlejší datové soubory než hierarchické techniky. Nehierarchické shlukovací postupy jsou navrženy, aby seskupovaly pozorování do skupin *k* shluků [32]. Protože kontrolovat všechny možné skupiny je výpočetně neuskutečnitelné, používá se jistá heuristika ve formě iteraktivní optimalizace. Konkrétně ji představují rozdílná schémata přemísťování, jež opakovaně znovuurčují středy mezi k shluky. Na rozdíl od tradičních hierarchických metod, v nichž shluky nejsou po jejich vytvoření znovu procházeny, optimalizační algoritmy shluky postupně vylepšují [4].

Optimalizační metody hledající takový rozklad množiny objektů určených pro klasifikaci, který je optimální podle vhodně zvoleného kritéria optimality rozkladu [39]. Optimalizační nehierarchické metody hledají optimální rozklad přeřazováním objektů ze shluku do shluku s cílem minimalizovat nebo maximalizovat nějakou charakteristiku rozkladu [43].

Pro algoritmy hledání optimálního rozkladu je typické, že začínají stanovením nebo odvozením počátečního rozkladu na *k* shluků. Tento rozklad je pak postupně zlepšován, a to buď tak, že počet shluků zůstává zachován, nebo se mění v závislosti na určitých řídících parametrech. Z tohoto hlediska se algoritmy směřující k nalezení optimálního rozkladu množiny objektů dělí do dvou skupin, z nichž první zahrnuje algoritmy zachovávající daný počet shluků a druhá algoritmy měnící počet shluků. Stanovení nebo odvození počátečního rozkladu je závažným problémem zvláště v případě první skupiny algoritmů [39].

Hledání optimálního rozkladu spočívá v hledání takového rozkladu, pro nějž nabývá zvolené kritérium extrémní hodnoty. Jedním z hlavních problémů shlukovacích technik je to, že tyto techniky vedou k nalezení lokálního nikoliv absolutního extrému. Jedinou možnou cestou k nalezením absolutního extrému je totiž zpravidla sestrojení a zhodnocení všech možných rozkladů množiny pozorování, což je reálně nezvládnutelné [39].

Počet shluků *k* může být specifikován buď předem nebo určen během shlukovací procedury jako její součást [32]. V nehierarchických shlukovacích metodách je počet shluků obvykle dán předem, i když se v průběhu výpočtu může změnit. Zůstává-li počet shluků zachován, hovoří se o nehierarchických metodách s konstantním počtem shluků, v opačném případě o nehierarchických metodách s optimalizovaným počtem shluků [43].

Mezi algoritmy hledání optimálního rozkladu patří zejména metoda k-průměrů (k-means) a metoda k-medoidů (k-medoids). Podle Berkhina, je k-tý medoid nejvhodnějším objektem daného shluku [4]. Shluk je reprezentován jeho konkrétním objektem, který je umístěn nejblíže středu [29]. Reprezentace na základě k-medoidů má dvě velké výhody. První výhodou je, že nepředstavuje žádná omezení typů proměnných. Druhou výhodou je, že volba medoidu je dána umístěním převládajícího podílu na středu uvnitř shluku a ten je tudíž méně citlivý na výskyt odlehlých pozorování. V případě metody k-průměrů je shluk reprezentován centroidem (těžištěm), který je průměrem (obvykle váženým průměrem) středů uvnitř shluku. Toto pojetí vhodně funguje jenom v případě numerických proměnných a může být negativně ovlivněno odlehlým pozorováním. Na druhou stranu, centroidy mají výhodu jasného geometrického a statistického významu [4].

Metoda k-průměrů (k-means)

Metoda k-průměrů (k-means) je jednou z nejpopulárnějších nehierarchických procedur [32]. Zjednodušeně funguje tak, že celá procedura na svém začátku „náhodně" stanoví

středy shluků. Přitom počet středů je roven počtu hledaných a požadovaných shluků. Každé pozorování je pak přiřazeno k nejbližšímu středu. Následně je střed přemístěn tak, aby jeho pozice odpovídala střední hodnotě daného shluku. Celý proces je opakován tak dlouho, dokud se změna pozice středu nestane zanedbatelně malou [54].

Jak již bylo uvedeno výše, algoritmus pracuje iterativně, je založen na přesunování objektů mezi shluky. Lze ho popsat následujícími kroky:

1. Zvolí se počáteční rozklad do *k* shluků, nejčastěji náhodně, podkladem však může být nějaká vnější informace, někdy také výsledek již provedeného shlukování, který se má zlepšit.
2. Určí se centroidy pro všechny shluky v aktuálním rozkladu.
3. Proberou se po řadě všechny objekty. Pokud má právě zkoumaný objekt nejblíže k vlastnímu centroidu, ponechá se na místě, jinak se přesune do shluku, k jehož centroidu má nejblíže. Nedošlo-li v tomto kroku k žádným přesunům, považuje se aktuální rozklad za definitivní (suboptimální) řešení úlohy. Jinak se vrací k 2. kroku [29].

Hebák s Hustopeckým uvádí, že „uvedený algoritmus se osvědčuje jako velmi efektivní", sub-optimálního řešení bývá dosaženo většinou již po malém počtu iterací. Uvedený postup patří mezi základní, někdy bývá modifikován. Např. místo počtu shluků se zadá zdrsňující parametr jako minimální přípustná vzdálenost centroidů a zjemňující parametry jako maximální přípustná vzdálenost objektu od vlastního centroidu. Shluky, které nesplňují podmínku minimální vzdálenosti centroidů, splynou a rozklad se tak zdrsní. Objekt, který je příliš daleko i od nejbližšího centroidu, se sám stane jádrem nového shluku a rozklad se zjemní [27].

Meloun a Militký uvádějí, že „při vytváření malého počtu shluků z velkého počtu objektů se metoda k-průměrů jeví nejúčinnější shlukovací metodou". Dále upozorňují, že „vyžaduje spojité proměnné a především bez odlehlých hodnot. Diskrétní data mohou být rovněž analyzována, ale mohou způsobit problémy."

Metoda k-průměrů je různými způsoby *modifikována*:

a) Proces shlukování lze zahájit s *k* vybranými objekty (např. s prvními k objekty) místo počátečního rozkladu. V první iteraci se pak přikročí hned ke 3. kroku, v němž se zvolené objekty stanou centroidy tvořených shluků; další postup je shodný (tento postup je aplikován například v systému STATISTICA).

b) Přepočet centroidů lze provést po každém přesunu objektu (nikoli tedy jen po každém cyklu); jde o velmi rozšířenou variantu, známou jako McQueenův algoritmus. Kromě počátečního rozkladu je tu průběh shlukování a výsledek závislý také na pořadí objektů, ve kterém vstupují do 3. kroku.

c) Zadáním některých parametrů řešení lze dojít k rozkladu s vhodným počtem shluků, i když tento počet není předem zadán. Jde o variantu McQueenova algoritmu, centroidy se tedy přepočítávají po každém přesunu. Místo počtu shluků musí být udán *zdrsňující* parametr jako minimální přípustná vzdálenost centroidů a *zjemňující* parametr jako maximální přípustná vzdálenost objektu od vlastního centroidu. Splnění uvedených podmínek se také kontroluje po každém přesunu. Shluky, které nesplňují podmínku minimální vzdálenosti centroidů, splynou a rozklad se tak zdrsní. Objekt, který je příliš

daleko i od nejbližšího centroidu, se sám stane jádrem nového shluku a rozklad se tak zjemní [29].

Metoda zárodečných (typických) bodů

Metoda typických bodů (Seeded) je založena na tom, že zadavatel úlohy určí na základě znalostí problému ty objekty, které jsou „typickými" představiteli nově vytvořených shluků. Kolem nich se dá očekávat vytvoření shluků [83]. Existuje několi postupů zadávání zárodků shluku a zařazování objektů do shluku. Lze využít jeden ze tří postupů:

- *Sekvenční práh*. Metoda začíná volbou jednoho zárodku shluku, do něho jsou přiřazeny všechny objekty uvnitř zvolené vzdálenosti. Když jsou všechny objekty uvnitř této vzdálenosti zahrnuty do shluku, je vybrán zárodek druhého shluku a všechny objekty uvnitř zvolené vzdálenosti jsou zahrnuty do tohoto shluku. Pak je vybrán třetí zárodek shluku a proces se opakuje. Když je jednou objekt shlukován se zárodkem, není s ním více počítáno do některého jiného shluku.
- *Paralelní práh*. Na rozdíl od předešlého postupu tento vybírá na začátku několik shlukových zárodků souběžně a zařazuje objekty uvnitř prahové vzdálenosti do nejbližšího zárodku. Jak se proces vyvíjí, prahovou vzdálenost lze nastavit tak, aby zařadila více nebo méně objektů do shluku. Některé objekty mohou zůstat nezařazeny do shluků, když se totiž nacházejí vně předspecifikované vzdálenosti od shlukového zárodku.
- *Optimalizaci*. Metoda zvaná optimalizační postup je podobná předešlým dvěma tím, že některý objekt se octne blíže jinému shluku, než ve kterém se právě nachází. Optimalizační postup ho přeřadí do jiného bližšího shluku.

Sekvenční prahový postup je vhodný pro velké datové soubory. Při jeho volbě závisí ale počáteční a konečný shluk na pořadí objektů v datové matici. Proto se provádí náhodné přeuspořádání objektů [43].

Metoda k-medoidů

V případě odlehlých pozorování není průměr vhodnou charakteristikou skupiny hodnot. Proto byly vyvinuty další metody, v nichž je shluk reprezentován jeho konkrétním objektem, který je umístěn nejblíže středu shluku. Tento objekt se nazývá medoid a takové objekty využívá *metoda k-medoidů* [29]. Medoid (optimální střed shluku) je tedy takový střední objekt, pro který platí, že průměrná vzdálenost k ostatním objektům v tomto shluku je minimální. K metodám shlukování okolo medoidů patří Späthova metoda a metoda PAM (Partition Around Medoids) [43].

Späthova metoda minimalizuje účelovou funkci (celkovou vzdálenost mezi všemi objekty ve shlucích) přemísťováním objektů z jednoho shluku do druhého. Začíná u počátečního uspořádání shluků, algoritmus pak najde lokální minimum inteligentním přesouváním objektů ze shluku do shluku. Jakmile se nepřemístí už žádný objekt, metoda končí proces. Lokální minimum však nemusí být globálním. Aby program překonal toto omezení, zopakuje se několikrát hledání vždy z jiného startovacího uspořádání a nejlepší uspořádání shluků je nakonec bráno za výsledné.

Metoda PAM (Pertition Around Medoids) minimalizuje celkovou vzdálenost mezi všemi objekty ve shlucích D takto:

1) Nalezne se reprezentativní soubor *k* objektů. První objekt má nejkratší vzdálenost ke všem ostatním objektům, čili představuje střed shluku – medoid. Pak se *k - 1* objektů hledá tak, že hodnota D je co možná nejmenší.
2) Možné alternativy polohy k objektů jsou vybírány iteračním způsobem. Algoritmus vyhledává dosud nezařazené objekty a přemisťuje je tak, aby se hodnota D snižovala. Iterace skončí, jakmile změny nezpůsobí další snížení hodnoty D [43].

Pozn. Metoda PAM je např. součástí systému S-PLUS [29].

3.4.1.8 Modifikace rozdělovacích metod

K novým přístupům, jež jsou založeny na rozdělovacích algoritmech, patří např. metody hybridní klasifikace, CLARA, CLARANS a SCA.

Jako příklad rozdělovacího přístupu lze uvést *hybridní klasifikaci*. Základem je přiřazení objektů z určité podmnožiny w do *k* shluků a potom přidělení ostatních objektů do některého ze vzniklých shluků, který je nejblíže. Podmnožina w by měla být pokud možno reprezentativní a obsahovat objekty všech tříd kompletního datového souboru. Pokud je to možné, měla by být pro zvolení objektů do podmnožiny w využita externí informace. Jestliže tato informace není k dispozici, pak se do w volí objekty, které mají velký počet sousedů. Klasifikace ostatních objektů se provádí podle některého z optimalizačních kritérií.

Na tomto principu je založena metoda CLARA (*Clustering LARge Applications*). Základem je algoritmus k-medoidů (medoid je jeden konkrétní objekt shluku). Pro velké soubory dat se používá jeho modifikace, jež se sestává ze tří kroků. V prvním se provede náhodný výběr objektů, které se rozdělí do k skupin. Ve druhém kroku se každý objekt ze souboru přiřadí k nejbližší skupině. Ve třetím kroku se zapamatuje průměrná vnitroskupinová vzdálenost. Proces se několikrát opakuje, a poté se vybere shlukování s nejmenší průměrnou vzdáleností. Vylepšením algoritmu CLARA je CLARANS (*Clustering Large Applications based upon RANdomized Search*). V prvním kroku se na základě náhodného výběru objektů vybere k medoidů. Novými medoidy se mohou stát sousedé medoidů, kteří splňuji daná kritéria. Výkonnost algoritmu může být zvýšena využitím R^*-stromů.

Jako další přístup lze uvést aplikování shlukové analýzy k-průměrů na náhodné výběry ze souboru a spojení informací získaných na základě předchozích výběrů s informacemi obdrženými ze stávajícího výběru. V jednotlivých fázích analýzy se využívá komprese dat. Algoritmus se označuje jako SCA (*Scaling Clustering Algorithms*) [29].

3.4.1.9 Pravděpodobnostní shlukování

Mezi nehierarchické metody patří i pravděpodobnostní modely, jež předpokládají, že údaje pocházejí ze směsice několika populací, u nichž chceme znát jejich rozdělení [4].

Metoda k-modů

Analýzy modů jsou skupinou metod, které jsou v souladu s použitým pravděpodobnostním přístupem [39]. Výše uvedený postup metody k-means lze samozřejmě použít pouze pro kvantitativní data. *Metoda k-modů* využívá princip metody k-průměrů pro nominální data. K tomu jsou zapotřebí vhodné míry nepodobnosti a aktualizace modů shluků je založena na četnostech [29]. Analýzy modů vycházejí z pravděpodobnostního pojetí. Při tomto přístupu se jeví reálné definovat shluky na základě existence a polohy modů frekvenční

funkce [39]. Analýzy modů představují hledání rozkladu do shluků, kde jsou shluky chápány jako místa se zvýšenou koncentrací objektů v *m*-rozměrném prostoru proměnných [42].

Kombinace metod k-průměrů a k-modů se nazývá *metoda k-prototypů* [29].

Algoritmus EM

Dále je na myšlence metody k-průměrů založen *algoritmus EM* (Expectation Maximization). Místo jednoznačného přiřazení objektu k určitému shluku jsou objektům přiřazeny váhy, které reprezentují pravděpodobnosti příslušnosti k jednotlivým shlukům [29]. EM clustering je pravděpodobnostní metoda shlukování. Pro definici shluku se použijí ty atributy, které ho nejlépe charakterizují. Tedy každý shluk může být definován jinou sadou atributů. Tato metoda je schopna pracovat i s větším počtem atributů (tj. více než 10), avšak je náročnější na přípravu dat [41].

Fuzzy shlukování

Meloun s Militkým zařazují k hierarchickému a nehierarchickému shlukování ještě další přístup tj. *Fuzzy shlukování*. To obohacuje všechny shlukovací metody tím, že umožňuje shlukování jednoho objektu do více než jednoho shluku, zatímco v běžném shlukování je každý objekt členem pouze jednoho shluku. Ve fuzzy shlukování je přítomnost objektu rozdělena do všech shluků. Pravděpodobnost, že objekt i je klasifikován do j-tého shluku je m_{ij} a musí platit, že $0 \leq m_{ij} \leq 1$ a suma těchto hodnot m_{ij} musí být rovna 1. Postup uplatňující m_{ij} se nazývá fuzifikace shlukové konfigurace [43]. Fuzzy shluková analýza je zařazena např. ve statistickém systému S-PLUS.

3.4.1.10 Nové přístupy ve shlukové analýze

Metody shlukové analýzy lze rozdělit do dvou základních skupin. Jedna skupina je založena na analýze matice vzdálenosti mezi objekty (dále *metody založené na vzdálenostech*) a druhá při analýze vychází přímo z původní zdrojové matice, kde každý řádek představuje vektor charakterizující určitý objekt (dále *metody vektorového prostoru*).

Základním problémem velkých souborů dat je, že analýza nemůže vycházet z matice vzdáleností vypočtené na základě všech objektů, neboť tento postup je velmi náročný, a to jak výpočetně, tak z hlediska uložení matice. Při použití metod vektorového prostoru jsou vytvářené shluky obvykle charakterizovány pomocí statistik vypočtených na základě jednotlivých objektů, což vede ke snížení výpočetních nákladů. K přednostem metod vektorového prostoru patří navíc to, že vytvořené shluky mají přirozenou souhrnnou reprezentaci, která vede k nalezení geometrického středu množiny objektů v p-rozměrném prostoru. Existuje však řada metod, která oba výše uvedené principy kombinuje.

Jestliže je datový soubor takového rozsahu, že nemůže být uchováván ve vnitřní paměti počítače, existují tři základní přístupy k řešení tohoto problému: přístup rozděl a panuj, postupné shlukování a paralelní implementace.

V prvním případě, **přístup rozděl a panuj**, jsou data rozdělena do p bloků. V každém z těchto bloků jsou objekty shlukovány do k shluků pomocí některého ze standardních algoritmů. Tímto způsobem se získá pk reprezentativních objektů, které jsou dále shlukovány do k shluků. Zbývající objekty jsou přiřazeny k vytvořeným shlukům. Tento algoritmus je možné rozšířit ze dvou na libovolný počet úrovní. Jde tedy o rozdělení

datového souboru do podmnožin (frakcí), na kterých je možno realizovat tradiční metody shlukové analýzy. Pokud je na každou frakci aplikována hierarchická shluková analýza, je tento postup označován jako *frakcionizace*. Shluky vzniklé na základě frakcí jsou dále shlukovány do k skupin stejnou shlukovací metodou.

Při **postupném shlukování** jsou objekty přiřazovány ke shlukům krok po kroku. Každý objekt je přiřazen buď k existujícímu shluku, nebo k novému shluku. Literatura uvádí čtyři základní typy postupného shlukování:

- shlukovací algoritmus hlavního objektu (viz níže),
- nejkratší cesta typu SSP (*Shortest Spanning Path*),
- *pavučinový* (*cobweb*) systém (postupný konceptuální shlukovací algoritmus) a
- postupný shlukovací algoritmus pro dynamické zpracování informací.

Ze starších přístupů lze uvést algoritmy rychlého rozdělování. Buď je v každém shluku určen hlavní objekt (*leader*), od kterého jsou počítány vzdálenosti jednotlivých objektů (algoritmus hlavního objektu), nebo jsou určeny prahové hodnoty pro všechny proměnné (třídicí algoritmus).

Algoritmus hlavního objektu rozčlení objekty do nepřekrývajících se shluků metodou postupného zařazování, takže jde o přistup závislý na pořadí vstupu. Je používána míra vzdálenosti D a prahová hodnota T, což je předem zvolená úroveň vzdálenosti D. Algoritmus přiřadí sledovaný objekt do prvního shluku, pro který platí, že vzdálenost hlavního objektu a daného objektu je menší než T. V případě, že zmíněná vzdálenost je pro všechny shluky rovna T nebo je větší, algoritmus zakládá nový shluk s vlastním hlavním objektem.

V *třídicím* algoritmu je pro každou (j-tou) proměnnou stanovena prahová hodnota T(j). Do daného shluku jsou objekty přiřazeny tak, aby pro všechny proměnné platilo, že variační rozpětí j-té proměnné je menší než T(j). Procedura je ekvivalentní postupu, při němž by každá proměnná byla převedena na kategoriální (kategorie by byly vytvořeny podle příslušné prahové hodnoty), a shluky by tak tvořily políčka vícerozměrné kontingenční tabulky mezi všemi proměnnými.

Metody založené na hustotě, mřížce a modelu

Metody založené na hustotě popisují shluky jako oblasti ve výběrovém prostoru, které se vyznačují značnou hustotou bodů ve srovnání s řídkými oblastmi. Základní myšlenka spočívá v tom, že každý objekt má ve výběrovém prostoru své vlastní sousedství. Pomocí neformální definice shluku lze říci, že pro objekt ve shluku platí, že jeho sousedství (dané poloměrem) musí obsahovat alespoň minimální počet dalších objektů. Metody tohoto typu mohou být použity pro zjišťování šumu a odlehlých pozorování a k odhalení shluků libovolného tvaru. Tím se daný přístup odlišuje od rozdělovacích metod založených na vzdálenostech mezi objekty, které mohou nalézt pouze shluky sférického tvaru. Jako metody založené na hustotě lze označit DBSCAN, OPTICS a DENCLUE.

V algoritmu DBSCAN (*Density-Based Spadal Clustering of Applications with Noise*) se rozlišují shluky a šum. *Shluk* je množina objektů spojených na základě hustoty a *šum* je množina objektů, které nepatří do žádného shluku. Objekt p je spojen s objektem q na základě hustoty (při daném poloměru a minimálním počtu objektů), pokud existuje objekt r takový, že jak objekt p, tak objekt q jsou z objektu r dosažitelné na základě hustoty.

Přitom objekt p (resp. q) je dosažitelný z objektu r na základě hustoty, pokud existuje posloupnost objektů p_1, p_2, ..:, p_n, kde p_1 = r a p_n = p tak, že p_{i+1} se nachází v sousedství p_i.

K přednostem algoritmu DBSCAN patří, že uživatel nemusí zadávat počet shluků. Z nedostatků lze uvést, že shluky musí mít určitý minimální počet bodů, což znemožňuje nalézt malé shluky. Rozšířením metody DBSCAN je algoritmus OPTICS (*Ordering Points To Identify the Clustering Structure*).

Algoritmus DENCLUE (*DENsity-based CLUstEring*) používá *influenční* (vlivovou) funkci, která modeluje vliv objektu na své sousedství. Hustota datového prostoru je vypočtena jako součet vlivových funkcích přes všechny objekty. Shluky (nazývané jako atraktory hustoty) jsou definovány jako lokální maxima celkové funkce hustoty [29].

Podstata **metod založených na mřížce** tkví v tom, že je datový prostor rozdělen do konečného počtu pravoúhlých buněk, které tvoří mřížkovou strukturu. Všechny shlukovací operace jsou prováděny na této struktuře. Hlavní výhodou uvedeného přístupu je nízká časová náročnost, která závisí pouze na počtu buněk v každé dimenzi.

Tyto metody reprezentuje algoritmus STING (STatistical INformation Grid), pomocí něhož je datový prostor rekurzivně rozdělen na pravoúhlé buňky. Existuje několik úrovní buněk, které odpovídají různým úrovním rozlišení, přičemž tyto buňky vytvářejí hierarchickou strukturu. Každá nelistová buňka je rozštěpena s cílem vytvořit určitý počet buněk v následující nižší úrovni. Přitom jsou pro každou buňku uchovávány potřebné statistické charakteristiky, jako je počet objektů, průměr, směrodatná odchylka, minimální a maximální hodnota a typ statistického rozdělení (sleduje se rozdělení normální, rovnoměrné, exponenciální či žádné z uvedených). STING mřížka dokáže vyhodnocovat dotazy. Každá buňka je k zadanému dotazu relevantní s určitou pravděpodobností. Irelevantní buňky jsou ignorovány, zatímco relevantní jsou sledovány, až je dosažena spodní část mřížky. Zde je k dispozici seznam relevantních buněk obsahujících informaci, která je odpovědí na dotaz.

Aby bylo uskutečněno vlastní shlukování, je potřeba provést druhou fázi algoritmu. Je testována významnost sousedů všech relevantní buněk. Shlukování pomocí ST1NG mřížky dokáže nalézt shluky libovolného tvaru, ale tyto shluky inklinují k mírně neupravené podobě vzhledem k jejich pravoúhlé podstatě. Jsou uvažována též odlehlá pozorování [29].

Metody založené na modelu předpokládají model pro každý shluk a hledají nejlepší přiřazení dat k danému modelu. Na základě standardních statistik dokážou určit počet shluků, přičemž berou v úvahu šum a odlehlá pozorování. Jde tedy o robustní shlukovací metody.

Jako metody založené na modelu lze označit *filtry částic* (*particle filters*) a algoritmus SOON. Filtrovací metoda odhaduje množství důležitosti (obvykle jde o posteriorní rozdělení s neznámými parametry) v množině N vážených částic. Filtr je tedy tvořen množinou částic a vah. Pro případný nový objekt je množina sekvenčně aktualizována.

Algoritmus SOON (*Self-Organizing Oscillator Network*) je založen na použití neuronové sítě. Organizuje množinu objektů do k stabilních a strukturovaných shluků. Metoda vychází z algoritmu SOM (Self-Organizing Map), což je samoorganizující se neuronová síť navržená Kohonenem. Pomocí algoritmu SOON je každý objekt reprezentován jako oscilátor, který je charakterizován fází a stavem [29].

Kromě výše uvedených speciálně zaměřených metod existuje přístup, který využívá všechny tři principy, tj. shlukování založené na modelu, hustotě a mřížce. Tyto **smíšené metody** reprezentuje algoritmus **DBCLASD** (*DistributionBased clustering algorithm for Clustering LArge Spatial Datasets*). Základní idea pro identifikování shluků spočívá ve shlukování založeném na hustotě ve výběrovém prostoru. Rozlišovací charakteristikou hustoty oblastí je to, aby vzdálenosti nejbližšího souseda pro objekty uvnitř oblasti byly menší než pro objekty vně oblasti. K popisu charakteristik shluků objektů může být proto použito pravděpodobnostní rozdělení vzdálenosti nejbližšího souseda. Toto rozdělení lze využit pro testování, zda by sousední objekt měl být zahrnut do shluku, či nikoli (pomocí chí-kvadrát testu se zjišťuje, zda po zahrnuti dalšího objektu do shluku je rozdělení stejné jako před přidáním objektu). K přednostem algoritmu DBCLASD patří, že dokáže nalézt shluky libovolného tvaru, určuje počet shluků a nevyžaduje žádné parametry od uživatele. Z nevýhod je možno uvést, že shlukování je závislé na pořadí objektů a že chí-kvadrát test lze použít pouze pro shluky obsahující minimálně 30 objektů. Proto na počátku musí být shluky vytvořeny bez testování. Pro nový objekt je vytvořen nový shluk tak, že je do shluku přidáno 29 nejbližších sousedních objektů [29].

Shlukování podprostorů

Metody pro shlukování podprostorů jsou určeny pro datové soubory s velkým počtem proměnných. Místo vytváření redukované matice založené na nových proměnných (získaných například lineární kombinací původních proměnných) je problém s velkým počtem dimenzí řešen zkoumáním podprostorů původního prostoru. Tento přístup je výhodný tím, že jsou zachovány původní proměnné, které mají reálný význam, zatímco lineární kombinace původních proměnných může být někdy těžko interpretovatelná.

Shlukování podprostorů vychází z metod založených na hustotě. Cílem je nalézt podmnožiny proměnných tak, aby projekce datových objektů zahrnovaly regiony s vysokou hustotou. Základem je rozdělení všech dimenzí do stejného počtu stejně dlouhých intervalů. Jsou-li určeny vhodné podprostory, úloha spočívá v nalezení shluků v odpovídajících projekcích. Shluky jsou oblasti navazujících jednotek s vysokou hustotou (v rámci určitého podprostoru). Pro jednodušší popis jsou shluky určovány jako hyperkvádry. Shlukování podprostorů umožňuje zařazovat do shluků též objekty s chybějícími údaji. Výsledky jsou přesnější než při jejich nahrazení chybějících údajů hodnotami z příslušného rozděleni.

Základní metodou uváděnou v literatuře je **CLIQUE** (*CLustering In QUEst*), která byla navržena pro kvantitativní proměnné. Tento shlukovací algoritmus využívá jak principy metod založených na hustotě, tak principy metod založených na mřížce. Shluky jsou nalezeny spojením buněk tak, že tvoří nepřekrývající se pravoúhlé oblasti s vysokou hustotou. Algoritmus zahrnuje tři následující kroky: identifikaci podprostorů obsahujících shluky, identifikaci shluků a vytvoření minimálního popisu pro shluky.

Algoritmus **ENCLUS** (*ENntropy-based CLUStering*) je založen na podobném principu jako CLIQUE, avšak používá rozdílné kritérium pro výběr podprostorů. Výpočetní náklady této metody jsou ale vysoké.

MAFIA (*Merging of Adaptive Finite Intervale (And more than a CLIQUE)*) je modifikací algoritmu CLIQUE, která funguje rychleji a nalézá shluky lepší kvality. Metoda v každé dimenzi konstruuje adaptivní mřížky. Její paralelní verze se nazývá pMAFIA.

Z dalších algoritrnů lze uvést **PROCLUS** (*PROjected CLUStering*), **ORCLUS** (*ORiented projected CLUSter generation*) a **OptiGrid** (*Optima! Grid-Clustering*). Metoda OptiGrid je založena na rekurzivním postupu. V každém kroku je datový soubor rozdělen do určitého počtu podmnožin. Podmnožiny, které obsahují alespoň jeden shluk, jsou dále analyzovány. Rozdělování je prováděno pomocí vícerozměrné mřížky [29].

Indexování objektů

Pro usnadnění některých operaci s objekty při výše uvedených typech analýz se v případě velkých souborů dat často provádí jejich indexování. Techniky používané pro účely statistické analýzy jsou založeny na principu hierarchického shlukování.

Podstatou takového indexování je vytvoření *stromu*, který se skládá z uzlů uspořádaných do různých úrovní. Na nejvyšší úrovni se nachází jediný uzel, který se nazývá *kořen*. Uzly jsou dvou typů: nelistové, které se odkazují na nižší úrovně, a listové *(listy*), které jsou na nejnižší (nulové) úrovni. Od kořene k listům vedou *větve*, přičemž všechny větve mají stejný počet úrovní. Každý uzel obsahuje informaci o skupině podobných objektů. Základní postup je takový, že každý objekt je zařazen pouze jedenkrát. Existují různé typy stromů, které se liší například klíčovou informací vztaženou k uzlu.

Indexová struktura se obvykle využívá pro urychlení vyhledávání objektů datového souboru, které mají stejné nebo podobné hodnoty proměnných jako nový objekt (v informatickém pojetí *dotaz*). Dále může být tato struktura vytvářena v procesu shlukování jako předzpracování dat před použitím některé tradiční metody shlukové analýzy.

K základním indexovým strukturám určeným pro datové soubory s velkým počtem proměnných patří R-strom a jeho varianty (R^* a R^+), X-strom, kd-strom a SS-strom. Základem těchto stromových struktur je zobrazení objektů jako bodů ve vícerozměrném prostoru [29].

3.4.1.11 Požadavky na metody shlukování

Kromě výše uvedených přístupů k modifikaci tradičních metod byly navrženy metody určené speciálně pro rozsáhlé soubory, jejichž vývoj spadá především do 90. let 20. století. Těchto metod existuje značné množství, přičemž každá má na jedné straně své přednosti, ale na druhé straně také své nedostatky.

Ideální metoda by měla splňovat určité požadavky (dosud vyvinuté metody splňuji pouze některé z nich), kterými jsou především:

- *přiměřená náročnost* (techniky pro shlukování musí být přiměřeně náročné, a to jednak pokud jde o požadavky na strojový čas, jednak pokud jde o požadavky na paměť),
- *nezávislost na pořadí vstupu* (tj. pořadí, v němž jsou objekty zařazovány do analýzy),
- *schopnost ohodnotit platnost vytvořených shluků* a
- *interpretovatelnost*.

Dále by metody měly být robustní zejména v následujících oblastech: *dimenzionalita* (rozdílnost mezi objekty by měla být zjistitelná i v případě velkého počtu proměnných), *šum* a *odlehlá pozorování* (algoritmus musí být schopen odhalit šum a odlehlá pozorováni

a eliminovat jejich negativní vlivy), statistické rozdělení, tvar shluků, velikost shluků, hustota shluků, oddělení shluků (algoritmus musí být schopen zjistit překrývající se shluky), *typy proměnných* (algoritmus by měl být určen pro různé typy proměnných, tj. kvantitativní spojité i kategoriální) [29].

3.4.2 Další vícerozměrné statistické metody

Jak již bylo uvedeno, v metodologii Data mining může být použita celá řada vícerozměrných statistických postupů. K všeobecně často používaným patří: faktorová analýza, diskriminační analýza, vícerozměrné škálování, kanonická korelační analýza, logistická, lineární a nelineární regrese.

3.4.2.1 Analýza hlavních komponent

Metoda hlavních komponent (PCA) je jedna z nejstarších a nejvíce používaných metod vícerozměrné analýzy. Cílem analýzy hlavních komponent je především zjednodušení popisu skupiny vzájemně lineárně závislých neboli korelovaných znaků čili rozklad zdrojové matice dat do matice strukturní a do matice šumové [43]. Na rozdíl od regresní analýzy neexistuje při této analýze dělení na závisle a nezávisle proměnné, všechny proměnné mají stejný status [30].

Techniku lze popsat jako metodu lineární transformace původních znaků na nové, nekorelované proměnné nazvané hlavní komponenty. Každá hlavní komponenta představuje lineární kombinaci původních znaků. Základní charakteristikou každé hlavní komponenty je její míra variability čili rozptyl. Hlavní komponenty jsou seřazeny dle důležitosti, tj. dle klesajícího rozptylu, od největšího k nejmenšímu. Většina informace o variabilitě původních dat je přitom soustředěna do první komponenty a nejméně infromace je obsaženo v poslední komponentě. Platí pravidlo, že má-li nějaký původní znak malý či dokonce žádný rozptyl, není schopen přispívat k rozlišení mezi objekty.

Standardním využitím metody hlavních komponent je snížení dimenze úlohy čili redukce počtu znaků bez velké ztráty informace, a to užitím pouze prvních několika komponent. Toto snížení dimenze úlohy se netýká počtu původních znaků. Je výhodné především pro možnost zobrazení vícerozměrných dat. Namísto vyšetřování velkého počtu původních znaků s komplexními vnitřními vazbami analyzuje uživatel pouze malý počet nekorelovaných hlavních komponent. Dále lze vybrané hlavní komponenty využít také k testu normality [43].

3.4.2.2 Faktorová analýza

Faktorová analýza (FA – Factor Analysis) je vícerozměrná technika k vyšetření vnitřních souvislostí a vztahů čili korelací a odhalení základní struktury zdrojové matice dat. Týká se analýzy struktury vnitřních vztahů mezi velkým počtem původních znaků pomocí souboru menšího počtu latentních proměnných, zvaných faktory. Nejprve jsou identifikovány faktory a pak je každému faktoru přidělen obsahový, obvykle fyzikální, význam, pomocí kterého je každý původní znak vysvětlen vybraným faktorem. Jde o dva primární cíle faktorové analýzy, a to jednak sumarizaci a jednak redukci dat. V sumarizaci dat využívá faktorová analýza faktorů tak, aby data vysvětlila a usnadnila jejich pochopení daleko menším počtem latentních proměnných, než je počet původních znaků. Redukce dat je dosaženo vyčíslením skóre pro každý faktor a následnou náhradou původních znaků novými latentními proměnnými – faktory.

Podobně jako metoda hlavních komponent patří faktorová analýza mezi metody snížení dimenze čili redukce počtu původních znaků. Ve faktorové analýze se předpokládá, že každý vystupující znak můžeme vyjádřit jako lineární kombinaci nevelkého počtu společných skrytých faktorů a jediného specifického faktoru. Na rozdíl od metody hlavních

komponent je ve faktorové analýze snaha vysvětlit závislost znaků. K nevýhodám metody patří zejména nutnost zvolit počet společných faktorů ještě před prováděním vlastní analýzy [43].

3.4.2.3 Kanonická korelační analýza

Kanonická korelační analýza (CCA – Canonical Correlations Analysis) byla navržena v souvislosti s hledáním lineární kombinace jedné skupiny znaků $x = (x_1, \ldots, x_q)$, která nejlépe koreluje s lineární kombinací druhé skupiny znaků $y = (y_1, \ldots, y_p)$. Vychází z předpokladu společného rozdělení obou skupin znaků. Podobně jako u metody analýzy komponent a faktorové analýzy se hledá lineární kombinace znaků obou skupin, tj. hypotetických kanonických proměnných, které vedou k maximálním vzájemným korelacím. Jde o krokový proces, kdy se v prvním kroku hledá lineární kombinace x a lineární kombinace y, jejichž korelace je maximální. V dalších krocích se hledají další lineární kombinace x a y, tj. kanonické proměnné takové, které mají maximální vzájemnou korelaci a přitom jsou nekorelované s kanonickými proměnnými nalezenými v předchozích krocích.

Přímé využití této metody je při snižování dimenze, kdy jsou skupiny původních znaků veliké a účelem je nalézt malý počet kanonických proměnných (lineární kombinace původních znaků), které postihují v maximální míře korelace mezi původními skupinami znaků.

Kanonická korelační analýza se často využije v situacích, ve kterých se tvoří regresní modely a v nichž existuje více než jedna závisle proměnná. Zvláště je užitečná v situacích, kdy závisle proměnné jsou vnitřně korelovány, takže nemá cenu je vyhodnocovat odděleně [43].

3.4.2.4 Diskriminační analýza

Diskriminační analýza (Discriminant Analysis) umožňuje hodnocení rozdílů mezi dvěma nebo více skupinami objektů charakterizovaných více znaky. Obyčejně se dále dělí na techniky, které interpretují rozdíly mezi předem stanovenými skupinami objektů, a techniky, kde je cílem klasifikace objektů do skupin.

Klasická klasifikační diskriminační analýza patří mezi metody zkoumání vztahu mezi skupinou p nezávislých znaků, zvaných diskriminátory, a jednou kvalitativní závisle proměnnou – výstupem. Výstupem je v nejjednodušším případě binární proměnná y nabývající hodnotu 0 pro případ, že objekt je v první třídě, respektive hodnotu 1 pro případ, že objekt je ve druhé třídě. O třídách je známé, že jsou zřetelně odlišené a každý objekt patří do jedné z nich. Účelem může být také identifikace, které znaky přispívají do procesu klasifikace. Ve vstupních datech trénovací skupiny jsou svými hodnotami diskriminátorů a výstupů všechny objekty zařazené do tříd.

Rozhodovací pravidlo, na jehož základě lze klasifikovat nový objekt se nazývá diskriminační pravidlo. U diskriminační analýzy se rozlišují jednotlivé diskriminační metody, nejčastěji se jedná o: lineární diskriminační funkci, kvadratickou diskriminační funkci, nelineární diskriminační funkci a logistickou diskriminaci. K odvození diskriminační funkce lze obecně užít dvou postupů: přímé metody a krokové metody. Mezi klíčové předpoklady odvození diskriminační funkce patří vícerozměrná normalita diskriminátorů. Data, která nevykazují vícerozměrnou normalitu, mohou způsobit problémy v odhadu diskriminační

funkce. Jiným rušivým vlivem, který může silně ovlivnit výsledky, je multikolinearita v datech.

Před užitím diskriminační analýzy se musí zvolit, které znaky budou chápány jako nezávisle proměnné, tj. diskriminátory, a které jako závisle proměnné. Závisle proměnnou je většinou nemetrická, kategorická proměnná, zatímco diskriminátory bývají především metrické proměnné. Uživatel se musí nejprve zaměřit na závisle proměnnou. Počet tříd může být roven dvěma i více, ale třídy musí být vzájemně se nepřekrývající a postačující, tzn. Každý objekt může být umístěn pouze do jediné třídy. V některých případech se závisle proměnná zařazuje do dvou tříd (dichotomie). V ostatních případech se může závisle proměnná zařazovat do několika tříd (multichotomie). Existují situace, kdy závisle proměnná není nominální proměnnou, ale může být ordinální nebo intervalovou měřenou veličinou a je třeba ji transformovat do zvoleného počtu kategorií.

Diskriminační analýza je značně citlivá na poměr velikosti výběru ve vztahu k počtu diskriminátorů. Výsledky diskriminační analýzy budou nestabilní, když bude poměr velikosti výběru a počtu diskriminátorů malý. Za minimální velikost výběru se doporučuje 5 objektů na jeden diskriminátor (optimální je 20). Navíc vedle celkové velikosti výběru je nutné brát v úvahu i velikost každé třídy, neboť při vlastním zařazování mají větší třídy nepoměrně větší šanci [43].

Diskriminační analýza je klasickou, průhlednou a statisticky bezproblémovou korektní metodou, která ovšem klade poměrně náročné požadavky na vstupní data, a proto se v komerčních aplikacích příliš nevyužívá [73].

3.4.2.5 Logistická regrese

Modelu vícenásobné logistické regrese se často využívá k odhadu pravděpodobnosti jisté události, která se přihodí danému objektu. Logistická regrese úzce souvisí s diskriminační analýzou a analýzou směsi normálních rozdělení. Je alternativní metodou klasifikace, když nejsou splněny předpoklady vícerozměrného normálního modelu. Logistická regrese nalezne lepší odhady parametrů než diskriminační analýza, když rozdělení nezávisle proměnných není vícerozměrně normální. Může se aplikovat na libovolnou kombinaci diskrétních nebo spojitých proměnných. Vyžaduje však znalost, jak závisle proměnné, tak i nezávisle proměnných u analyzovaného výběru. Výsledný model může být využit k budoucímu klasifikování, když jsou uživateli dostupné pouze vysvětlující, nezávisle proměnné [43].

Logistickou regresi lze vysvětlit jako zobecnění klasické vícerozměrné lineární regrese [73]. Logistická regrese se liší od lineární regrese v tom, že predikuje pravděpodobnost, zda se daná událost stala nebo nestala. Vypočtená pravděpodobnost je tedy rovna 0 nebo 1. Aby se vytvořila tato vazebná podmínka, užívá logistická regrese tzv. **logitovou transformaci**, která vede na sigmoidální vztah (sigmoidální funkce – křivka ve tvaru S) mezi závisle proměnnou y a vektorem nezávislých proměnných x. Při velmi nízkých hodnotách nezávisle proměnné se pravděpodobnost proměnné y blíží k nule, zatímco při vysokých hodnotách nezávisle proměnné se blíží k jedné. Rozdíl mezi logistickou a lineární regresí spočívá v tom, že logistická regrese používá kategorickou vysvětlovanou, závisle proměnnou, zatímco lineární regrese užívá pouze spojitou vysvětlovanou, závisle proměnnou. Centrální roli zde hraje logitová transformace, která vychází z tzv. **poměru šancí** či naděje. Dle typu vysvětlující proměnné se rozlišují:

1. Binární logistická regrese, která se týká binární závisle proměnné či znaku, nabývající pouze dvou možných hodnot. Vektor vysvětlujících, nezávislých proměnných či znaků může obsahovat jednu či více proměnných či znaků, a to spojitých zvaných *prediktory* anebo kategorických zvaných *faktory*.
2. Ordinální logistická regrese, která se týká ordinální závisle proměnné či znaku, nabývající tři a více možných stavů přirozeného charakteru. Vektor vysvětlujících nezávisle proměnných či znaků může obsahovat jak prediktory, tak i faktory.
3. Nominální logistická regrese, která se týká nominální závisle proměnné či znaku, o více než třech úrovních různých stavů, mezi kterými je definována pouze odlišnost. Vektor vysvětlujících nezávisle proměnných může obsahovat jak prediktory, tak i faktory [43].

Logistická regrese je výkonnou a robustní statistickou metodou pro predikci pravděpodobnosti výskytu určité události. Slovem robustní je míněno to, že model bude správně fungovat i po případných změnách, ke kterým dojde v průběhu doby. Protože logaritmus poměru šancí je lineární funkcí jednotlivých prediktorů, je hlavním problémem nalezení takové formy prediktorů, která bude co nejvíce lineární. Existuje několik způsobů, jak toho dosáhnout. Logistická regrese vnímá všechny prediktivní proměnné jako spojité. Pro nespojité proměnné tedy se používají tzv. indikátorové (příznakové) proměnné, aby je model chápal jako by byly spojité. Indikátorové proměnné jsou proměnné, které nabývají hodnoty 1 při splnění podmínky a 0 při jejím nesplnění [54].

3.4.2.6 Lineární a nelineární regrese

Prostá lineární regresní analýza je statistickou metodou, která kvantifikuje závislost mezi dvěma spojitými proměnnými: závislou proměnnou a nezávislou, prediktivní proměnnou. Predikce jedné spojité proměnné pomocí většího počtu prediktivních neboli nezávislých spojitých proměnných se nazývá vícenásobná lineární regrese.

Cílené modely vytvořené pomocí lineární regrese jsou obecně velmi robustní. V marketingu je lze použít samostatně nebo v kombinaci s jinými modely [54].

3.4.2.7 Vícerozměrné škálování

Vícerozměrné škálování (MDS = Multidimensional Scaling) je název pro skupinu exploratorních statistických metod, založených na redukci vícerozměrného prostoru objektů (pozorování) a průzkumové analýze vztahů mezi nimi [29]. MDS je technika vytvoření subjektivní mapy relativního umístění objektů v rovině dvourozměrného grafu, a to na základě vzdáleností či podobností mezi objekty, tzv. matice proximity (blízkosti) [43]. MDS pracuje s různými typy relací mezi objekty, přičemž nejčastěji jde o číselně vyjádřenou vzájemnou vzdálenost (blízkost) či nepodobnost (podobnost). Jsou však možné i jinak vyjádřené vztahy, např. korelace, asociace apod. [29]. Cílem vícerozměrného škálování je detekovat základní souřadnice, které dovolují objasnit pozorované podobnosti či vzdálenosti mezi vyšetřovanými objekty [43].

Smyslem MDS je optimálně snížit rozměr dat a zkoumat relace objektů v redukovaném prostoru. Ačkoli jsou výstupy MDS i číselné, jde hlavně o *vizuální techniku*. Objekty jsou zobrazovány v redukovaném prostoru, který se označuje *konfigurace bodů* (mapa objektů), a který bývá základním vodítkem pro interpretaci vztahů mezi objekty.

Ve své podstatě řeší MDS obdobné úlohy jako jiné vícerozměrné metody, např. faktorová analýza, korespondenční analýza, shluková analýza nebo analýza hlavních komponent. Na rozdíl od nich však nevyžaduje přímé určení matice pozorování - tu je možné určit nepřímo z matice relací mezi objekty [29]. Vícerozměrné škálování je metoda vysoce deduktivní povahy [43].

Data měřená na různých škálách je třeba z důvodu souměřitelnosti nejprve normovat - obvykle postačí přepočet na z-skóry nebo jiný typ standardizace. Data měřená na stejné škále se nazývají *profily* [29]. Vícerozměrné škálování objektů nemá omezující požadavky na metodologii, typ dat, formu vztahu mezi znaky, ale vyžaduje, aby uživatel přijal několik zásad o datech: kolísání ve volbě znaků, v důležitosti znaků a v čase [43]. V MDS se používá obvykle euklidovská vzdálenost (čtvercová, vážená). Její bezproblémové použití ale teoreticky předpokládá lineární nezávislost a vzájemnou souměřitelnost proměnných.

MDS zahrnuje celou řadu různých variant a modelů. Klasifikačních kritérií lze nalézt více a většinou závisí na pohledu, ze kterého se modely třídí. Kromě základního rozlišení na *metrické* a *nemetrické* lze modely klasifikovat podle počtu datových matic, symetrie, počtu opakování apod. [29]. Techniky subjektivního mapování objektů mohou být klasifikovány např. i dle charakteru respondentova hodnocení objektů. Prvním přístupem je *dekompoziční* (rozkladná) metoda bez užití znaků, která měří pouze celkový dojem při hodnocení objektu a pak vypočte polohu objektu ve vícerozměrném prostoru objektů, která tento dojem vystihne. Tato technika je velmi typická pro vícerozměrné škálování objektů. *Kompoziční* (skladná) metoda při užití znaků je alternativním přístupem, který využívá několik vícerozměrných technik, již diskutovaných při tvorbě samotného dojmu o objektu. Je založena na kombinaci více posuzovaných znaků o objektech [43].

3.4.2.8 Korespondenční analýza

Korespondenční analýza (CA) je grafická metoda k zobrazení skryté vnitřní závislosti, asociace v tabulce četností (kontingenční tabulce). Je to kompoziční technika, protože subjektivní mapa je založena na asociaci mezi souborem objektů v řádcích a souborem popisných znaků ve sloupcích (zadaných člověkem). Polohy bodů pak přímo vyjadřují asociaci. Její přímou aplikací je zobrazování korespondence kategorií proměnných, znaků, které jsou měřeny v nominální stupnici. Tato korespondence je základem vytváření subjektivní mapy.

Korespondenční analýza je schopna zpracovávat nemetrická data i nelineární vztahy. Sdílí s tradičními technikami vícerozměrného škálování volnost v předpokladech, a protože jde o kompoziční techniku, je zde nutná úplnost znaků. Korespondenční analýza představuje popisnou techniku, která se nehodí ke statistickému testování hypotéz. Uplatňuje se především v rámci exploratorní analýzy dat [43].

3.4.2.9 Rozhodovací stromy

Rozhodovací stromy jsou alternativním postupem k diskriminační a regresní analýze. Některé slouží pouze ke klasifikaci (klasifikační stromy), jiné umožňuji též odhadovat hodnoty kvantitativní vysvětlované proměnné (regresní stromy). Výhodou uvedených přístupů je, že vysvětlující proměnné nemusí být kvantitativní (pokud jsou spojité, jsou převedeny na kategoriální). Cílem modelování je vytvořit stromovou strukturu. Existuje řada různých algoritmů, jako příklady lze uvést C&RT (*Classification And Regression Tree*), CHAID (*Chi-Sguare Automatic Interaction Detection*), LMDT, OC1, QUEST či C5.

Kořenovým uzlem je vysvětlovaná proměnná. Pro štěpení se vybere proměnná, která má největší vliv na hodnoty vysvětlované proměnné. V případě kategoriální vysvětlované proměnné lze pro tento výběr využít například chí-kvadrát statistiku. Jiným kritériem může být nejvyšší hodnota informačního zisku [29].

Strom je vhodnou formou pro sumarizaci informací v datech, protože určuje pomocí sekvence podmínek pro prediktory několik skupin tak, že v rámci těchto skupin lze použít velmi jednoduchý prediktor závisle proměnné. Dále nevyžaduje, aby populace byla homogenní a efekty prediktorů lineární. Podobně jako u jiných exploračních technik je nutné získané výsledky pro učební soubor validizovat pomocí nového souboru dat – zkušebního souboru [30].

Regresní stromy

Pomocí **regresních stromů** se řeší problémy spojené s predikcí spojité závisle proměnné nebo klasifikací, kdy se predikuje příslušnost objektů do předem daných tříd. Může doplnit nebo v určitých případech nahradit metody regresní analýzy v případě spojité závisle proměnné nebo metody diskriminační analýzy a logistické regrese.

Při prvním kroku se rozdělí skupina objektů do dvou nebo více skupin pomocí podmínek, jež se týkají jednoho prediktoru. Vytvořené skupiny se nazývají uzly. Každá z vytvořených skupin se může opět podobným způsobem rozdělit na podskupiny podle hodnot některého z prediktorů. Nejjednodušší je tzv. binární segmentace. Hledání optimální štěpící proměnné a příslušné meze se provádí pomocí optimalizace nějakého kritéria. Pokud je podmínka splněna pro všechny koncové skupiny (listové uzly stromu), celý proces končí. Pro regresi se po ukončení procesu štěpení zjišťuje průměrná hodnota závisle proměnné v každé koncové skupině [30].

V případě regresních stromů se pro štěpení vybere proměnná, u niž byla zjištěna maximální hodnota redukce směrodatné odchylky. Princip výpočtu redukce směrodatné odchylky je obdobný výpočtu informačního zisku. V jednotlivých skupinách vzniklých na základě hodnot vysvětlující proměnné se vypočte výběrová směrodatná odchylka a ze získaných hodnot se vypočte vážený aritmetický průměr, který se odečte od výběrové směrodatné odchylky vysvětlované proměnné vypočtené pro všechny objekty.

V dalších krocích se opět vyberou vysvětlující proměnné, které mají největší vliv na hodnoty vysvětlované proměnné zjištěné u objektů zařazených do příslušného uzlu. Pro ukončení štěpení se stanoví určitá kritéria, jejichž základem může být počet objektů nebo variabilita hodnot v uzlu (například větvení bude ukončeno, jestliže výběrová směrodatná odchylka v daném uzlu bude menší než 5 % z hodnoty výběrové směrodatné odchylky vypočtené pro všechny objekty analyzovaného souboru) [29].

Klasifikační stromy

Podobně se postupuje při vytváření **klasifikačních stromů**. V tomto případě jde o vytváření skupin, jež budou stále více homogenní vzhledem k zastoupení objektů z různých tříd, do nichž se objekty klasifikují. Nejlepší konečné dělení by bylo takové, že v dané koncové skupině jsou objekty pouze z jedné klasifikované třídy [30].

Rozhodovací stromy se řadí mezi jednu z neoblíbenějších Data mining technik. Důvodů pro to je několik. Hlavní spočívá v jejich přehlednosti a snadné interpretovatelnosti, která umožňuje uživatelům rychle a lehce vyhodnocovat získané výsledky, identifikovat klíčové položky a vyhledávat zajímavé segmenty případů. Algoritmy tvorby rozhodovacích stromů

vycházejí z důkladného a mnohaletého výzkumu v oboru statistiky a umělé inteligence. I proto jsou tolik populární, neboť umožňují získávat zajímavé znalosti z podnikových databází [48]. Nevýhodou rozhodovacích stromů může být „lokální" (= vždy jen na malý subsegment se vztahující) a nespojitý (po částech konstantní) charakter těchto modelů; výhodou je naopak malá závislost na distribuci vstupních dat, umožňující analyzovat i řídké nebo značně negaussovské proměnné [73].

Regresní a klasifikační stromy mají několik potenciálních výhod ve srovnání s lineárními modely. Predikce závisle proměnné se velmi zjednodušuje, protože se pomocí jednoduchých podmínek pro nezávisle proměnné musí pouze zjistit pro objekt tzv. terminální uzel regresního stromu. Stromová struktura poskytuje jednodušší interpretaci než lineární predikční rovnice. Regresní stromy nevyžadují omezující předpoklady o rozdělení závisle proměnné.

Omezení techniky regresních a klasifikačních stromů spočívá v tom, že při analýze musí být k dispozici velké množství změřených objektů, protože při každém štěpení se změnšuje počet objektů, o které je možné se při další analýze daného uzlu opírat. Nestabilita vytvářených klasifikací je poměrně značná. Tímto nedostatkem se vyznačují všechny shlukovací techniky [30].

3.4.2.10 Analýza historie událostí

Analýza historie událostí se zabývá obecně studiem pohybu subjektů v čase mezi určitými stavy. Tyto metody se dají uplatnit všude tam, kde hlavní sledovaná proměnná je čas nějaké události (událostí) nebo přesněji doba do určité události (událostí). Použití těchto metod je mj. v demografii a pojišťovnictví, kde se nazývají aktuární nebo aktuárský počet. V medicíně se užívá zejména pro sledování délky přežívání, proto se ujal i název analýza přežívání. Důležitou roli v této metodě hraje tzv. cenzorování, tj. sledování. Existují čtyři typy nezávislého cenzorování:

- Jednoduchý typ I – všechny objekty jsou sledovány pevnou dobu;
- Progresivní typ I - všechny objekty jsou cenzorovány ve stejný časový okamžik;
- Typ II – studie trvá do uskutečnění n událostí;
- Náhodné cenzorování – časy cenzorování jsou nezávislé na době události.

Zachycení vlivu nezávislých proměnných na průběh křivek umožňují regresní modely časů události. Nejznámější z nich se nazývá Coxova regrese. Příslušný model vychází z funkce rizika h(t), která udává profil, jak se v čase mění pravděpodobnost, že dojde k události [30].

Coxovu regresi se hodí použít namísto logistické regrese např. pro modelování odchodů zákazníků a podobných procesů, neboť tato data jsou cenzorovaná – zákazníci průběžně přicházejí a odcházejí během analyzovaného období. Každý sledovaný zákazník je vystaven riziku odchodu po různou dobu, a především u zákazníků, kteří u firmy setrvávají i k okamžiku zpracování modelu, není ještě známo, kdy odejdou. Coxova regrese tento nedostatek informací bere v úvahu a umožňuje i tehdy konstruovat statisticky korektní odhady [73].

3.4.3 Nestatistické metody

Pro analýzu velkých datových souborů jsou v praxi používány i jiné metody než dosud uvedené statistické postupy. Většinou jsou zařazovány k metodám strojového učení. Patří k nim zejména klasifikátory, neuronové sítě a genetické algoritmy.

3.4.3.1 Klasifikátory

Existuje mnoho různých algoritmů pro klasifikaci případů do tříd podle zvolené proměnné, a to jak algoritmů standardních, z oblasti statistiky, tak i relativně nových, z oblasti umělé inteligence. Většina těchto algoritmů se snaží o minimalizaci počtu chyb, které udělají na trénovacích datech. Výsledek se však používá na jiných datech (předpokládá se, že jsou ze stejného rozdělení). Proto je rozhodujícím kritériem kvality počet chyb na datech testovacích [37].

Klasifikátory k-nejbližších sousedů

Pro klasifikaci a předpověď je možné použít též *klasifikátory k-nejbližších sousedů*. Jde o přístup založený na analogii, při němž se každý objekt chápe jako bod v p-rozměrném prostoru. Chceme-li pro zadané hodnoty vysvětlujících proměnných odhadnout hodnotu vysvětlované proměnné, je na základě euklidovské vzdálenosti nalezeno k nejbližších sousedů. Vlastním odhadem je pak v případě klasifikace nejčetnější kategorie vysvětlované proměnné zjištěná na základě k nalezených objektů, v případě předpovědi aritmetický průměr hodnot vysvětlované proměnné odpovídajících k objektům. Lze použít též vážený průměr, přičemž váhy se stanoví na základě vzdálenosti nového objektu od nalezených sousedů; mohou být počítány například jako převrácená hodnota čtvercové vzdálenosti.

Protože v případě rozsáhlého souboru by bylo značně časově náročné pro každý nezařazený objekt počítat vzdálenosti od všech zařazených objektů, používá se obvykle pro klasifikaci či předpověď jen výběr (tzv. trénovací množina), případně mohou být pro klasifikaci využity centroidy vypočítané pro jednotlivé skupiny vytvořené na základě kategorií vysvětlované proměnné. Pro urychlení nalezení k-nejbližších sousedů lze dále využit stromovou strukturu objektů, a to kd-stromy, v nichž jsou nelistové uzly tvořeny proměnnými a listy obsahují seznamy podobných objektů [29].

Klasifikace Support Vector Machines

Relativně nová metoda klasifikace Support Vector Machines (SVM) je zvláště vhodná v případech, kdy je k dispozici mnoho relevantních proměnných a málo případů pro učení klasifikátoru. Nachází tedy uplatnění zejména v oblastech klasifikace textů, kredit skóringu nebo odhalování podvodů.

Klasifikace SVM je nová a velice slibná metoda. Je to jeden z mnoha příkladů nové třídy metod, nahrazující tradiční schéma maximálně věrohodného odhadu (minimalizace chyby na trénovacích datech) novým schématem, založeným na minimalizaci horní meze chyby na testovacích datech. SVM je klasifikační metoda, která hledá lineární klasifikátory s maximální margin (vzdálenost). Právě díky podmínce maximální margin se značně snižuje kapacita tohoto klasifikátoru, a tedy zvyšuje jeho schopnost zobecňovat závislosti z trénovacích dat na data testovací.

Klasifikace SVM je relativně nová metoda, a proto zdaleka není dostupná ve všech softwarových „balících" pro analýzu dat. Z dostupných komerčních implementací klasifikace SVM stojí za zmínku program Statistica [37].

3.4.3.2 Neuronové sítě

Neuronové sítě jsou relativně novým nástrojem, který se ve statistice využívá pro různé typy analýz, například klasifikaci, predikci nebo shlukování. Vznik konceptu neuronových sítí patří do oblasti umělé inteligence. Prvotní myšlenkou byla touha po napodobení činnosti lidského mozku. Vznikl tak protiklad oproti symbolickému, ryze matematickému, přístupu k řešení tradičních úloh umělé inteligence. Neuronové sítě, tak jak jsou dnes chápány, nejčastěji používají velmi zjednodušený model neuronu [29]. Neuron je matematický model vycházející z představy fungování neuronu v mozku [47]. Z těchto neuronů se sestavují (propojují pomocí synapsí) různé architektury (topologie) sítí. Vytvořené sítě se pak učí danou úlohu řešit. V procesu učení se vazbám mezi neurony (synapsím) přiřazují váhy. Rozeznáváme učení s učitelem a bez učitele [29].

K základním neuronovým sítím patří *sítě dopředné*, v nichž je síť složena z vrstev neuronů. Tyto neurony jsou propojeny jen mezi vrstvami - vstupní signál se propaguje přes jednotlivé vrstvy až k vrstvě výstupní. Existují různé architektury tohoto typu sítí, jako příklady budou dále zmíněny vícevrstvý perceptron a Kohonenova mapa.

Vícevrstvý perceptron

Vícevrstvý perceptron se ve statistice používá zejména pro klasifikaci a zejména pro predikci. Je typickým představitelem sítě učené s učitelem. Zde o síť tvořenou určitým počtem (větším než dvě) vrstev neuronů [29].

Činnost vícevrstvého perceptronu lze popsat následujícím způsobem. Do neuronu přicházejí vstupy, které se v něm syntetizují do jediného výstupu. Vstupy představují vstupní proměnné. Tyto hodnoty se na vstupu do neuronu vynásobí příslušnými vahami (w_i) a takto upravené vstupují do neuronu. V rámci neuronu jsou vstupní hodnoty dále upravovány. Používají se různé převodní funkce, jejichž cílem je převést několik vstupních hodnot na jednu hodnotu výstupní. Klasicky se nejprve všechny vstupy do neuronu sečtou a na výsledný součet se aplikuje některá z funkcí, která převede vzniklý součet na výstupní hodnotu. Nejčastěji se používá funkce sigmoidy, někdy lineární, někdy skoková funkce. Do výpočtu ještě vstupuje tzv. práh, hodnota specifická pro každý neuron. Výstupem z neuronu je tedy pouze jediná hodnota vzniklá transformací vstupních hodnot.

Poskládání několika neuronů postupně vedle sebe a za sebe vznikne vícevrstevná neuronová síť. Neurony v rámci své vrstvy nejsou vzájemně propojeny, ale každý je propojen se všemi neurony následující vrstvy. Výpočet v neuronové síti probíhá následovně. Vstupní hodnoty x_i se předají neuronům ze Vstupní vrstvy, které tyto hodnoty redistribuují na Skrytou vrstvu. Zde v jednotlivých neuronech dochází k výpočtům a následně jsou výstupy předány do Výstupní vrstvy, která obsahuje jeden nebo i více neuronů. Výstupní vrstva přijme hodnoty ze Skryté vrstvy a vypočte výslednou hodnotu. Tato hodnota je výstupem celé sítě [47].

Aby neuronová síť správně předpovídala, je nutné ji ale předem „naučit". To znamená správně nastavit váhy a prahy u jednotlivých neuronů a také určit počet neuronů v jednotlivých skrytých vrstvách. Proces učení probíhá na základě historických údajů o sledované skutečnosti. Neuronová síť v procesu učení postupně prochází každý záznam

historických dat. Načte vstupní honoty a vypočítá svůj výstup. Tento výstup porovná se skutečnou hodnotou z dat a na základě rozdílu upraví váhy a prahy neuronů. Cílem učení je, aby se výstupy sítě lišily od skutečných hodnot co možná nejméně [47]. Při učení takové sítě se minimalizuje suma druhých mocnin rozdílů mezi získanými a očekávanými výstupy (známými hodnotami vysvětlované proměnné) [29].

Kohonenova mapa

Kohonenova mapa (síť) neboli SOM (*Self-Organizing Map*) se obvykle používá pro shlukování, respektive pro vizualizaci dat. Tato síť se učí bez učitele (nemá žádný cílový atribut), má tedy schopnost samoorganizace [29]. Základním principem její funkce je shluková (segmentační) analýza, tedy schopnost sítě rozdělit předkládané případy (zákazníky, smlouvy, produkty, ...) do skupin s podobnými charakteristikami sledovaných vlastností. Kohonenovy neuronové sítě dokážou toto rozdělení najít přímo v předkládaných datech, bez znalosti vnější informace [49]. Kohonenova síť je ekvivalentem algoritmu k-průměrů.

Síť je tvořena dvěma vrstvami, vstupní vrstvou a vrstvou navzájem propojených neuronů uspořádaných do čtvercové matice (Kohonenovy mřížky). Vstupní vrstva je propojena na každý neuron v mřížce. Cílem učení je přiřadit objekty k jednotlivým neuronům (shlukům) mřížky, což se obvykle provádí tak, že se upraví střed nejbližšího shluku tak, aby se zmenšila vzdálenost mezi tímto shlukem a aktuálním signálem. Učící data jsou předkládána tak dlouho, až po předložení signálu nedojde k úpravě vah. Konvergence je zajištěna průběžným zmenšováním koeficientu učení [29].

Učení sítě probíhá následujícím postupem: Kohonenově síti jsou postupně předkládány vstupní příklady. Nejčastěji (v případě segmentace zákazníků) se jedná o záznamy charakteristik a chování zákazníka (věk, plat, zisk, ...). Vstupní hodnoty musejí být číselné nebo jsou speciální technikou (za pomoci takzvaných dummy proměnných) na číselné převedeny (například pohlaví, vzdělání). Pro každý neuron z výstupní mřížky se vypočítá výše uvedená vzdálenost vah neuronu od předkládaného případu. Neuron s nejnižší hodnotou vzdálenosti je vybrán jako nejpodobnější. Váhy vybraného neuronu a jeho okolí jsou modifikovány tak, aby ještě více odrážely vstupní hodnoty. Rozsah okolí vybraného neuronu, u kterého dochází ke změně vah, se ve fázi učení postupně zmenšuje a ke konci již neobsahuje žádný další neuron (modifikován je pouze vybraný neuron). Na počátku okolí většinou zahrnuje všechny neurony v mřížce. Tímto postupem se náhodně nastavené váhy postupně upravují tak, aby odrážely skryté skupiny (segmenty) obsažené v datech. Čím jsou neurony v mřížce blíže, tím podobnější případy reprezentují. Naopak neurony na koncích diagonál mřížky reprezentují nejméně podobné případy.

Počet neuronů ve výstupní mřížce určuje sám uživatel na počátku učení, podobně může mít možnost nastavit i rozsah a rychlost redukování okolí neuronu určeného pro modifikaci vah. Jakmile je síť naučena (jsou jí předloženy všechny případy, popřípadě je splněno jiné kritérium pro ukončení fáze učení), je možné ji použít k zatřídění nových případů do příslušného segmentu. Pro vstupní případ se vypočtou výstupní hodnoty neuronů v mřížce a případ je přiřazen k segmentu, který je reprezentován neuronem s nejnižší hodnotou výstupu [49].

Kohonenova síť je oproti algoritmu K-Means a Two-Step clustering vhodná i pro segmentaci za účelem vyhledání odlehlých (extremních) případů, zejména podvodných případů. Ty se většinou vyskytují v datech ve velice malé míře a Kohonenova síť je schopna je efektivně oddělit od zbývajících případů a umístit je na svém výstupu izolovaně

od ostatních [49]. Nevýhodou neuronových sítí je jejich notoricky obtížná interpretovatelnost a omezená statistická testovatelnost [73].

3.4.3.3 Genetické algoritmy

Dalším typem biologicky inspirovaných algoritmů jsou *genetické algoritmy*, které napodobují mechanismus evoluce. Jde vlastně o kombinatorickou optimalizaci. Začíná se pracovat s určitým seskupením objektů (populací), které se postupně modifikuje (zdokonaluje) pomocí operací selekce, křížení a mutace. Jestliže je objekt určen jako vhodný, *přežívá*. Činnost algoritmu končí po splnění určité podmínky, například po dosažení maxima účelové funkce nebo vyčerpání stanoveného počtu generací. Jako příklad lze uvést algoritmus **GGA** (*Genetically Guided Algorithm*) používaný pro shlukování algoritmu k-průměrů a pro fuzzy shlukovou analýzu [29].

4 ZVOLENÉ METODY ZPRACOVÁNÍ

Tato kapitola představuje metody a postupy, které budou použity pro zpracování dat společnosti poskytující internetové připojení. Pro praktickou aplikaci vybraných statistických metod pomocí technik Data mining je nutné zajistit vhodný programový prostředek, shromáždit datový soubor, provést kvalitní přípravu dat, vybrat metodu, zvolit statistické postupy aj.

První podkapitola je zaměřena na přípravu dat tj. na možnosti, které pro práci s daty nabízí programování v základním modulu Base SAS. V procesu přípravy dat bude použita celá řada procedur a funkcí, neboť součástí řešení práce bude vypracování vlastního návrhu programu pro přípravu dat z flat file souboru na formu vhodnou pro analytickou práci.

Druhá podkapitola je věnována možnostem shlukování pomocí programování, které bude použito především k identifikaci odlehlých pozorování, sledování počtu opakování shlukování a hodnocení chování kritérií.

Specializovanému dataminingovému modulu systému SAS – Enterprise Miner je věnována třetí podkapitola. Tento modul používá pro všechny fáze práce s daty metodologii SEMMA. Každý krok této metodologie obsahuje specifické úlohy, které se sestavují do diagramu úloh.

Nejrozsáhlejší část této kapitoly se zaměřuje na metodické podklady použití uzlu Clustering, který v modulu Enterprise Miner zajišťuje kvalitní shlukovou analýzu. Shlukování pomocí uzlu Clustering bude použito pro vlastní segmentaci zákazníků.

Softwarové prostředí pro Data mining je voleno na základě disponibility České zemědělské univerzity. Jedná se o systém SAS, který patří v dané kategorii ke světově uznávaným. Jedná se o systém s celou paletou nástrojů, které uživateli umožňují vyhodnocování velkého množství údajů užitím jak jednoduchých tak náročnějších statistických i nestatistických metod a grafů.

4.1 Příprava dat pomocí programování v systému SAS

Pro vlastní statistickou analýzu v jakémkoli statistickém softwaru je nutné pracovat s daty, která jsou utříděna do struktury numerických či znakových proměnných. Kroky pro práci s daty směřují v systému SAS k tvorbě datových souborů (Data Sets), jež jsou tvořeny proměnnými (Variables) a jejich pozorováními (Observations). V této podobě lze potom provádět statistické analýzy a vhodné informační výstupy. Možnosti úpravy dat lze efektivně provádět z příkazové řádky programového editoru systému SAS [9], [10].

V systému SAS má každá tabulka (datový soubor) své jméno, datum a čas vytvoření tabulky, počet pozorování, popisku a informace o uložení. K ukládání dat se nabízejí dva typy proměnných: číselné a znakové. Každá proměnná má svůj název, typ, formát, délku záznamu, popisku a pozici. Znaková proměnná může (ve verzi SAS v8) obsahovat od 1 do 32 767 znaků. Znakem může být písmeno, číslo, speciální znak, mezery atd. Chybějící hodnoty ve znakové proměnné jsou označeny mezerou, jejich automatické zarovnání je doleva. Číselná proměnná je konvertována na pohyblivou řádovou čárku v 8 Bytech, chybějící číselná hodnota je zobrazena tečkou, automatické zarovnání je vpravo.

Tabulky (Data Sets) lze místo do složek na disku ukládat v rámci systému SAS do tzv. knihoven (Libraries). Knihovny jsou dvojího základního typu: dočasné a trvalé. Dočasný charakter má knihovna WORK (po uzavření systému SAS je její obsah vyprázdněn). Trvale existující knihovnou je knihovna SASUSER a také knihovna, kterou si uživatel sám vytvoří, pojmenuje a přistupuje do ní např. v rámci uceleného projektu. Pojmenování tabulek se proto provádí dvoustupňově, první stupeň je jméno knihovny, druhým stupněm je vlastní jméno tabulky. Knihovna nemusí obsahovat jenom tabulky s daty, ale i jejich pohledy a různé katalogy.

V systému SAS jsou zavedeny konvence pro pojmenování tabulek a proměnných. Jméno může tvořit 32 znaků, začínat může pouze písmenem (velkým i malým) nebo podtržítkem. Nelze tedy uvádět na začátku jména mezeru nebo speciální symboly.

Pro programování v systému SAS slouží editorové okno (Enhanced Editor). O průběhu zpracování programového kódu informuje okno LOG. Okno OUTPUT zobrazuje výstupy, jež jsou požadovány nejen v rámci práce s daty, ale i z kroků statistických procedur. V orientaci ve struktuře uložených informací pomáhá uživateli okno Explorer a také okno s přehledem výsledků (Results).

Pro tvorbu programu se využívají zejména příkazy a funkce. V systému SAS jsou k dispozici čtyři typy příkazů: globální příkazy, příkazy pro práci s daty, příkazy pro procedury a makro příkazy. Příkazy jsou zpravidla odděleny mezerami (zlepšuje to čitelnost programového kódu). Text programu může být proložen poznámkami a to hned dvěma způsoby. Příkazový řádek se ukončuje středníkem.

Příkazy pro práci s daty se zapisují do těla programu DATA, kde obecný zápis je v následující struktuře: DATA datový_soubor <atributy>;

Mezi základní příkazy patří například nastavení vlastností proměnné, jako je délka, popiska, formát apod. Dalším typem příkazu jsou tzv. přiřazovací příkazy, které jsou založené na logických operátorech a operátorech porovnání. Jinou kategorii tvoří příkazy pro výpis či čtení proměnných. Pro výběr proměnných slouží příkazy zajišťující vyjmutí proměnné a zařazení proměnné. Též je možné proměnnou přejmenovat. Pro výběr pozorování lze využít filtr na řádky. Údaje v příslušném řádku se dají i sčítat. Pro práci s textem slouží pole, s nimiž lze provádět opět celou řadu operací.

Pro manipulaci se znakovými hodnotami slouží např. funkce pro vyjmutí či vložení znaku, pro odsunutí znaku (doleva, doprava), pro navracení n-tého slova, pro spojení znaků ve více proměnných, pro mazání prázdných znaků na konci slova, pro převod znaků na velká/malá písmena nebo pro převod znaků na číslo či naopak.

Mezi globální příkazy patří např. tvorba titulků a patiček (jejich počet je omezen na deset a deset). Mezi příkazy pro práci s daty patří např. tvorba tabulky či zápis hodnot pozorování. Base SAS je schopen přečíst údaje téměř z jakéhokoli formátu, z jakéhokoli souboru včetně záznamů různé délky, binárních souborů nebo souborů s chybějícími hodnotami. Jak již bylo výše uvedeno, proměnná má několik atributů. Jeden z nich – formát – stojí za zvýšenou pozornost.

Formát udává informaci pro tvar výstupu a zobrazení (nikoli pro uložení). Formát znakových hodnot umožňuje přesně nadefinovat délku nebo např. ponechat všechny úvodní mezery v řetězci, zahodit všechny úvodní mezery v řetězci, vypsat vše s velkými písmeny. Formát pro číselné hodnoty se udává v počtu všech číslic a v počtu desetinných míst; ve standardním tvaru má číslo podobu 12345.67 (typicky americky psané číslo). Je

však možné zvolit i jiný formát např. německý, kdy je číslo ve tvaru 12.345,67. Správný typografický český formát 12 345,67 však použít nelze. Obdobně je to i s volbou formátu datumu, kde lze nastavit způsoby oddělování (mezera, dvojtečka, pomlčka, tečka, …). Je možné definovat i vlastní uživatelské formáty (např. telefonní čísla) [10].

Kalendářní data je nutné transformovat na formát, který podporuje tzv. „kalendářní matematiku". Tou je schopnost provádět matematické funkce s kalendářními časovými údaji. Mezi ně patří sčítání, odčítání, násobení a dělení. SAS má pro zachycení kalendářních údajů připravenu celou řadu formátů. Jakmile se převede kalendářní hodnota do takového formátu, uloží se jako celé číslo představující počet dní od 1. ledna 1960 [54].

Pro tvorbu nových proměnných lze využít i základní aritmetické výpočty založené na sčítání, odečítání, násobení, dělení a umocnění. Z logických operátorů se nabízí AND a OR, z operátorů porovnání je možné využít následující vztahy: $<$, $>$, $=$, $\neq$, $\leq$ a $\geq$ (pro jejich zápis v programovacím jazyce slouží zkratky LT, GT, EQ, NE, LE a GT).

Pro práci s daty se mohou také využít možnosti, kdy lze: zařadit jen příslušné sloupce z tabulky (příkaz KEEP), odebrat příslušné sloupce z tabulky (příkaz DROP), přejmenovat proměnnou (RENAME), vybrat jen číselnou/znakovou proměnnou, provést výběr proměnné začínající na určité písmeno, výběr proměnných z intervalu, spojení sloupců atd. Pro výběr řádků slouží následující možnosti: čtení do n-tého řádku, čtení od m-tého řádku a čtení za podmínky (WHERE).

Modul Base SAS nabízí deset kategorií funkcí, lze členit do následujících kategorií: zaokrouhlení, aritmetické funkce, matematické funkce, trigonometrické funkce, základní statistické funkce, finanční funkce, generátor náhodných čísel, funkce časové a datové, znakové funkce a speciální funkce (Truncation, Arithmetic, Mathematical, Trigonometric, Sample Statistics, Financial, Random Number, Date and Time, Character, Special).

Zaokrouhlování je možno provádět na nejmenší celé číslo, největší celé číslo, na celočíselnou část nebo dle zvolených jednotek. Mezi finanční funkce patří různé výpočty pro úročení a pro amortizaci. Z datových funkcí je zajímavé např. na základě data určit den v týdnu (WEEKDAY), měsíc v roce (MONTH) apod. nebo vypočítat, kolik uběhlo dní od zvoleného dne, měsíce, kvartálu atd. Generátor náhodných čísel používá různé typy rozdělení (Normální, Poissonovo, rozdělení χ^2, Studentovo, Gama, Beta, Exponenciální ad.). Znakové funkce umožňují ze znakového řetězce nahradit nebo vybrat znaky, odsunout prázdné znaky ze začátku, vybrat n-té slovo (dle zvoleného oddělovače), převést znaky na malá/velká písmena apod.

Při programování je též důležité znát možnosti větvení programu popř. seskupování. Modul Base SAS nabízí větvení pomocí příkazu IF-THEN/ELSE nebo DO-END, výběr skupin se provádí pomocí příkazu SELECT.

Procedurální příkazy umožňují provádět analýzu dat. Base SAS dovede tvořit analýzy formou stručných sumarizací dat např. výpočet relativních četností a kontingenčních tabulek. Pomocí programování tak je možné vypočítat různé druhy popisných statistik, včetně průměru, součtu, rozptylu, směrodatné odchylky a dalších. Lze také vypočítat korelační a asociační míry závislosti stejně jako vícerozměrné tabulky a deduktivní statistiky. Analytické schopnosti tohoto modulu lze rozšířit o další specializované části.

Pomocí procedurálních příkazů je možné zajistit také prezentaci dat a výsledků. V modulu Base SAS je možno vytvářet tzv. reporty (Reports), a to od jednoduchých výpisů tabulek

až k uživatelským reportům komplexního charakteru. Tvorba těchto reportů vyžaduje obvykle jen malé množství příkazů.

Base SAS podporuje práci s jazykem SQL (Structured Query Language), který je ANSI-standardem a jazykem široce používaným, jež umožňuje tvořit, opravit a aktualizovat databázi. Je zde zahrnuto SQL dotazovací okno (SQL Query Window) – uživatelské rozhraní, které poskytuje možnost tvořit dotazy nikoli pomocí příkazů ale kliknutím myší.

Systém SAS je rozsáhlý statistický software, jež umožňuje celou škálu statistických analýz, ale bez důkladné přípravy dat se žádný statistik neobejde. Vzhledem k různorodosti problémů týkajících se přenosu dat a práce s daty je právě přístup s programováním jedním z nejvíce efektivních [10].

4.2 Shlukování pomocí programování

Programování v systému SAS nabízí použít pro shlukovou analýzu jak hierarchické, tak nehierarchické metody. Pro analýzu shluků se využívá několik algoritmů, zařazeny jsou procedury CLUSTER a FASTCLUS (ve verzi 4.3), nově DMVQ (od verze 5.1). Hierarchické shlukování je zastoupeno v proceduře CLUSTER, nehierarchické procedurou FASTCLUS. Obě základní kategorie metod mají určité slabiny, určitá znevýhodnění. Hierarchická shluková analýza nevyžaduje předem znalost počtu shluků, ale může vyžadovat příliš vysoký výpočetní výkon. To znamená, že pro velké objemy dat by mohly výpočty vyžadovat příliš času. Nehierarchické metody jsou rychlé, ale zase vyžadují předem zadat počet shluků.

Giudici uvádí [24], že aby bylo možné vyvarovat se uvedeného znevýhodnění a pokusit se využít potenciál obou metod, lze zvolit jedno ze dvou možných řešení:

1) Vybrat z dat výběr omezené velikosti, potom provést hierarchickou shlukovou analýzu pro určení *k*, tj. optimálního počtu shluků. Když je určena hodnota *k*, vezme se těch *k* průměrů jako středy; potom se pokračuje se shlukovou nehierarchickou analýzou na celý datový soubor s použitím počtu shluků rovno *k* a přiřazováním každého objektu do jednoho z nich.

2) Alternativně se nabízí zajímavá možnost: při velkém objemu dat metodou K-Means spočítat relativně velké množství shluků (např. 50) a poté pro jejich následné hierarchické shlukování s možností zobrazení dendrogramu využít proceduru CLUSTER. Je tedy možné pracovat s celým datovým souborem, provádět nehierarchickou analýzu s rozsáhlým *k*. Potom vzít v úvahu nový datový soubor vytvořený z průměrů *k* shluků, který se obohatí dalšími dvěma mírami. Jedna míra vyjadřuje velikost shluku a druhá rozptyl uvnitř shluku. Hierarchická analýza se potom provádí na tomto datovém souboru, aby bylo vidět, zda se mohou spojit některé shluky. Je nezbytné označit četnost a variabilitu každého shluku, jinak analýza nevezme v úvahu shluky mající odlišný počet pozorování a odlišný rozptyl.

4.2.1 Procedura CLUSTER

Procedura CLUSTER hierarchicky shlukuje pozorování (objekty) použitím jedné z jedenácti metod. Procedura CLUSTER nachází hierarchické shluky objektů datového souboru. Údaje mohou mít podobu souřadnic (coordinates) nebo vzdáleností. Jestliže jsou data souřadnicemi, pak PROC CLUSTER počítá euklidovské vzdálenosti. Je možné provést shlukovou analýzu na základě jiných dat než jsou euklidovské vzdálenosti a to použitím procedury DISTANCE. Tato procedura může vytvářet vhodnou vzdálenost mezi daty, jež se potom může použít jako vstup do PROC CLUSTER. Jednou ze situací, kdy může být vhodné analyzovat jiné než euklidovské vzdálenosti mezi daty, je pokud datový soubor obsahuje kategorizované údaje. U nich se vzdálenost mezi daty počítá použitím míry asociace.

Mezi shlukovací metody procedury CLUSTER patří

1) metoda průměrné vazby (average linkage),
2) centroidní metoda,
3) metoda úplné vazby (complete linkage),
4) density linkage (zahrnující Wongův hybrid a metodu nejbližšího souseda),

5) EML – metoda založená na spojování shluků za účelem maximalizace pravděpodobnosti na každé úrovni hierarchie pro směsice kulovitých vícerozměrných normálních rozdělení se shodnými rozptyly, ale potenciálně neshodnými poměry mísení,
6) metoda flexible-beta,
7) McQuittyho analýza podobnosti,
8) mediánová metoda,
9) metoda jednoduché vazby, tj. nejbližšího souseda (single linkage),
10) two-stage density linkage a
11) Wardova metoda minimálního rozptylu.

Všechny metody jsou založeny na užití aglomerativního hierarchického shlukovacího postupu. Různé shlukovací metody se odlišují výpočtem vzdálenosti mezi shluky.

Procedura CLUSTER není praktická pro rozsáhlé datové soubory, protože u většiny metod čas práce procesoru stoupá se čtvercem či krychlí počtu objektů. Naproti tomu procedura FASTCLUS vyžaduje čas proporčně k počtu objektů a tudíž může být používána na mnohem větší objemy dat než PROC CLUSTER.

Jestliže je cílem shlukovat rozsáhlé datové soubory hierarchicky, tak se doporučuje použít nejprve PROC FASTCLUS pro předběžnou shlukovací analýzu s tím, že vytvoří velký počet shluků, a potom použít PROC CLUSTER pro hierarchické shlukování předběžných shluků [64], [68].

4.2.2 Procedura FASTCLUS

Procedura FASTCLUS poskytuje disjunktní shlukovou analýzu na základě vzdáleností vypočítaných z jedné či více kvantitativních proměnných. Objekty se dělí do shluků tak, že každý objekt náleží do jednoho a pouze jen do jednoho shluku; shluky netvoří stromovou strukturu jako v případě procedury CLUSTER. Jestliže je požadováno analyzovat výsledky v závislosti na odlišném počtu shluků, je možné spustit PROC FASTCLUS jedenkrát pro každou z analýz. Nebo je pro provedení hierarchického shlukování na velkém datovém souboru možné použít PROC FASTCLUS pro nalezení počátečních shluků a potom použít tyto počáteční shluky jako vstup do PROC CLUSTER.

Standardně používá procedura FASTCLUS euklidovské vzdálenosti, takže středy shluků jsou založeny na odhadu nejmenších čtverců. Tento typ shlukovací metody je často označován jako „k-means model", neboť středy shluků jsou průměrem objektů přiřazených každému shluku, když se spustí algoritmus pro úplné sbližování. Každá iterace redukuje kritérium nejmenších čtverců, dokud není dosaženo sbližování.

Často není nutné spouštět proceduru FASTCLUS pro sbližování, neboť je navržena k nalezení dobrých shluků (ale ne bezpodmínečně nejlepších možných) při procházení datového souboru pouze dvakrát nebo třikrát. Inicializace metody PROC FASTCLUS zaručuje, že pokud existují takové shluky, že všechny vzdálenosti mezi objekty ve stejných shlucích jsou menší než všechny vzdálenosti mezi objekty v odlišných shlucích, a pokud se proceduře FASTCLUS zadá k nalezení správný počet shluků, pak je vždy možné nalézt takové shlukování bez iterace. Dokonce u shluků, které se dobře neseparují, obvykle nachází procedura FASTCLUS počáteční středy, které jsou dostatečně dobré tak, že je

požadován malý počet iterací. Proto PROC FASTCLUS vykonává standardně pouze jednu iteraci.

Inicializační metoda používaná procedurou FASTCLUS je citlivá k odlehlým pozorováním. PROC FASTCLUS může být efektivní pro detekování odlehlých objektů, protože odlehlá pozorování se často objevují jako shluky pouze s jedním členem.

Procedura FASTCLUS je určena pro zpracování velkého objemu dat, více než 100 pozorování. S malými datovými soubory mohou výsledky vykazovat vysokou citlivost k pořadí pozorování v souboru.

Základní výsledky procedury FASTCLUS jsou sestaveny do sady tabulkových výstupů. První z nich zobrazuje počáteční středy tzn. hodnoty pozorování jednotlivých proměnných. Další se věnuje souhrnným statistikám a zobrazuje četnost objektů ve shluku, střední kvadratickou směrodatnou odchylku, euklidovskou vzdálenost středu shluku k nejvzdálenějšímu bodu, nejbližší shluk a vzdálenost mezi středy shluků (vzdálenost středů aktuálního a nejbližšího shluku). Třetí část výstupu obsahuje statistiky proměnných (směrodatnou odchylku, R^2) a hodnoty kritérií (kritérium Pseudo-F, celkové R^2, kubické shlukovací kritérium). Celkové R^2 a kubické shlukovací kritérium nejsou vhodné pro korelované proměnné.

4.2.2.1 Syntaxe a hlavní parametry procedury FASTCLUS

V proceduře FASTCLUS musí být zadán buď maximální počet shluků tj. MAXCLUSTER nebo tzv. RADIUS. Další parametry, jako CONVERGE, LEAST, MAXITER, REPLACE; jsou volitelné.

MAXCLUSTER

MAXCLUSTER udává maximální počet povolených shluků. Pokud se nezadá tato volba, předpokládá se nastavení na hodnotu 100.

RADIUS

RADIUS určuje minimální vzdálenost kritéria pro výběr nového středu. Žádné pozorování se nebere v úvahu jako nový střed, dokud minimální vzdálenost k předchozím středům nepřekročí hodnotu danou volbou RADIUS=hodnota. Implicitní hodnota je nastavená na nulu. Pokud se uvede RADIUS=RANDOM, nastavení RADIUS=hodnota se ignoruje.

CONVERGE

CONVERGE udává konvergenční kritérium. Povolená je jakákoli nezáporná hodnota. Přednastavená hodnota je 0,0001 pro všechny hodnoty *p*, jestliže je určeno LEAST=*p*; jinak je hodnota rovna 0,02. Iterace se zastaví, když je maximální relativní změna ve středu shluku menší nebo rovna konvergenčnímu kritériu a vyhovuje případným dalším podmínkám. Relativní změna ve středu shluku je vzdálenost mezi starým a novým středem vydělená faktorem vah. Jestliže není specifikována hodnota LEAST, faktor vah je minimální vzdáleností mezi počátečními středy. Když je hodnota LEAST určena, je faktor vah odhadem L_1, který se při každé iteraci přepočítá. Specifikovat CONVERGE=hodnota se doporučuje jen v případě, že MAXITER=honota je větší než 1.

LEAST

LEAST zapřičiňuje, že procedura FASTCLUS optimalizuje kritérium L_p, kde $1 < p < \infty$. Nekonečno se vyjadřuje jako LEAST=MAX. Hodnota tohoto shlukovacího kritéria se zobrazuje v iterační historii.

Pokud není uvedeno LEAST=hodnota, pak procedura FASCLUS používá kritérium nejmenších čtverců (L_2). Když je ale implicitní počet iterací roven 1 a je vynecháno nastaveni LEAST=hodnota, pak optimalizace kritéria není dokončená. Jestliže je LEAST=hodnota uvedena, pak se maximální počet iterací zvyšuje, aby měl optimalizační proces možnost konvergence. Zadání LEAST=hodnota mění také přednastavení konvergenčního kritéria z 0,02 na 0,0001.

Když je hodnota LEAST=2, pak se procedura FASTCLUS pokouší minimalizovat průměrný absolutní rozdíl mezi údaji a odpovídajícími průměry shluků.

Když je hodnota LEAST=1, pak se procedura FASTCLUS pokouší minimalizovat průměrný absolutní rozdíl mezi údaji a odpovídajícími mediány shluků.

Když je hodnota LEAST=MAX, pak se procedura FASTCLUS pokouší minimalizovat průměrný absolutní rozdíl mezi údaji a odpovídajícími středními rozpětími shluků.

Pro obecné hodnoty *p* se procedura FASTCLUS pokouší minimalizovat p-tou odmocninu průměru p-tých mocnin absolutních rozdílů mezi údaji a odpovídajícími středy shluků.

V shlukovacím kritériu je dělitelem buď počet úplných záznamů nebo součet vah, jež odpovídají všem úplným záznamům (tzn. pozorování s n úplnými záznamy přispívá n-krát danému pozorování). V průběhu iterace závisí metoda pro aktualizaci středů shluků na volbě LEAST=hodnota.

MAXITER

MAXITER udává maximální počet opakování tj. iterací pro přepočítání středů shluků. Když je nastavení MAXITER=hodnota větší než 0, pak procedura FASTCLUS uskuteční třetí ze čtyř kroků postupu. V každé iteraci je každé pozorování přiřazeno k nejbližšímu středu a středy jsou přepočítány na průměry středů. Přednastavená hodnota MAXITER=hodnota záleží na LEAST=hodnota.

Tabulka 4: Vztah mezi LEAST a MAXITER

LEAST = *p*	nespecifikováno	*p* = 1	*1* < *p* < 1,5	1,5< *p* < 2	*p* = 2	2 < *p* < ∞
MAXITER	1	20	50	20	10	20

REPLACE

REPLACE udává, jak se provádí přemisťování středu.

- FULL udává, že stávající střed bude v každém kroku programu nahrazen vypočtenou střední hodnotou (přednastavená volba).
- PART vyžaduje přemístění středu jen tehdy, když je vzdálenost mezi objektem a nejbližším středem větší než minimální vzdálenost mezi středy.
- NONE potlačuje přemisťování středu.
- RANDOM vybírá jednoduchý náhodný výběr úplných pozorování jako počáteční středy shluků.

RANDOM

RANDOM=hodnota udává kladnou celočíselnou hodnotu jako počáteční hodnotu generátoru náhodných čísel pro použití REPLACE=RANDOM. Pokud tato hodnota není nastavena, pak se pak použije aktuální denní čas.

SEED

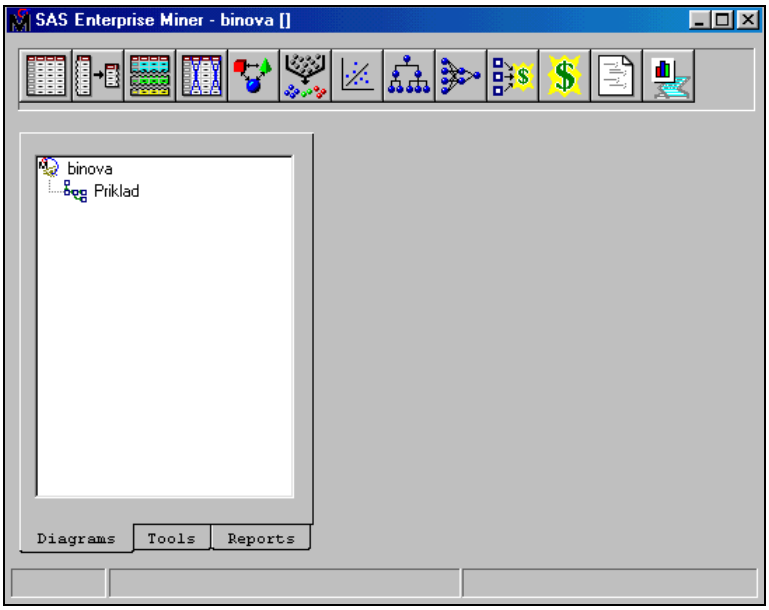
SEED=název_datového_souboru specifikuje vstupní soubor, z něhož se vybírají počáteční středy shluků. Pokud se neuvede SEED=název_datového_souboru, pak se počáteční středy vybírají z DATA= název_datového_souboru. SEED=název_datového_souboru musí obsahovat stejné proměnné jako ty, které se používají pro analýzu.

STRICT

STRICT zabraňuje tomu, aby bylo do shluku zařazeno pozorování, jestliže jeho vzdálenost k nejbližšímu středu shluku překročí hodnotu udanou STRICT=hodnota. Jestliže se k příkazu STRICT nezadá hodnota, pak se musí použít RADIUS=hodnota, která ji nahradí. Ve výstupním souboru OUT=datový_soubor je pozorováním, jež nejsou díky použití příkazu STRICT zařazena do žádného shluku, dáno záporné číslo shluku (jeho absolutní hodnota udává shluk s nejbližším středem) [64].

4.3 Shlukování v Enterprise Miner

V SAS Enterprise Miner jsou implementovány jak pokročilé statistické tak i nestatistické metody. Enterprise Miner vychází z vlastní metodiky pro dobývání znalostí z databází. Název SEMMA charakterizuje jednotlivě prováděné kroky: Sample, Explore, Modify, Model, Assess [3].



Obrázek 7: Prostředí modulu Enterprise Miner

4.3.1 Metodologie SEMMA

Prvním krokem je Sample – výběr, který umožňuje rozpoznávat vstupní datové soubory (rozpoznává vstupní data, provádí výběr z rozsáhlejších dat, rozděluje datový soubor na trénovací, validační a testovací množinu).

Druhý krok, Explore – průzkum, prozkoumává datový soubor pomocí statistických a grafických metod (kreslí grafy, získává popisné charakteristiky, identifikuje důležité proměnné, provádí asociační analýzy).

Třetí krok, Modify – úprava, obsahuje přípravu dat pro analýzy (vytváří doplňkové proměnné nebo transformuje existující proměnné pro analýzy, identifikuje odlehlá či extrémní pozorování, doplňuje chybějící hodnoty, pozměňuje postupy, jimiž jsou používány proměnné pro analýzu, provádí shlukovou analýzu, analyzuje data pomocí sítí).

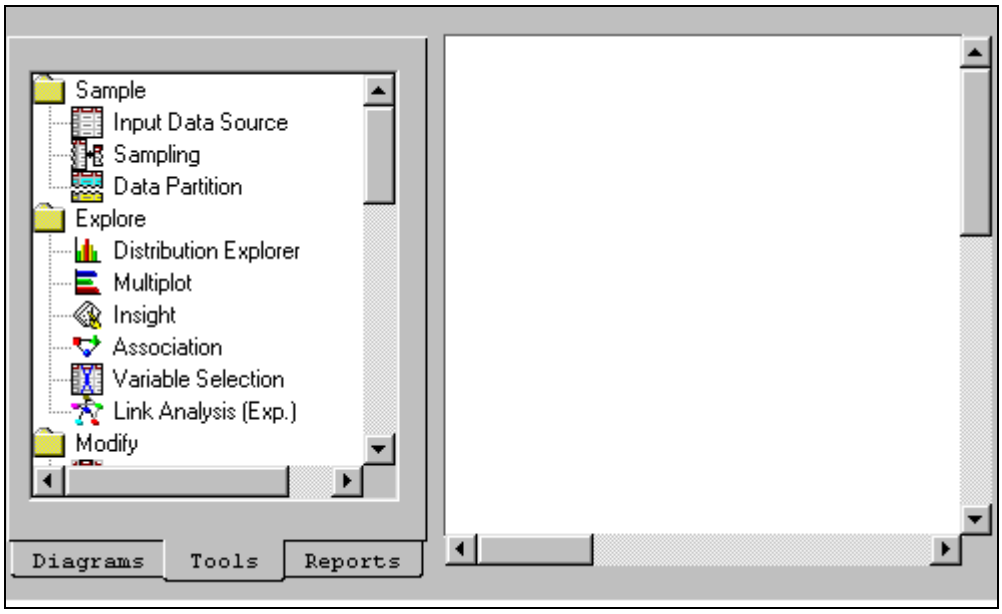
Čtvrtý krok, Model – modelování, kvalifikuje prediktivní model (modeluje cílovou proměnnou použitím regresních modelů, rozhodovacích stromů, neuronových sítí nebo uživatelem definovaných modelů).

V pátém kroku, Assess – vyhodnocení, se porovnávají vytvořené prediktivní modely (vytvořené grafy, které vyznačují procenta vysvětlení, procenta zachycených odpovědí, růstové grafy, grafy užitku)

Pro doplnění výše uvedených kroků existuje ještě sada nástrojů označovaná termínem Utilities [60].

Procesy dolování dat se definují pomocí procesních diagramů (Process Flow Diagrams) v grafickém uživatelském prostředí. Všechny metody dolování dat mají širokou škálu technik a jsou snadno modifikovatelné. Výstupy jsou ve formě přímo interpretovatelné obchodními uživateli. Integrace s technologií datových skladů usnadňuje práci s IT [53].

Práce v modulu Enterprise Miner je založena na tvorbě uzlových grafů, jejichž uzly jsou tvořeny jednotlivými úlohami v rámci výše uvedených postupů. Každý krok obsahuje sadu nástrojů, které se dají vybrat buď kliknutím na ikonku na liště nebo z nabídky uvedené v levé části pracovního okna modulu Enterprise Miner (Tools) [60].

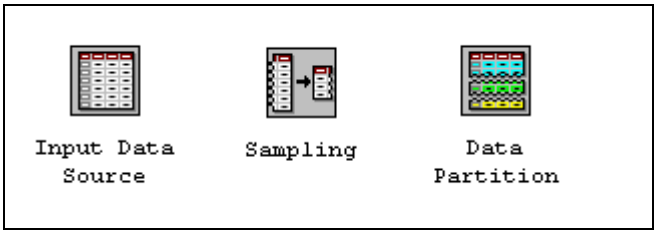


Obrázek 8: Seznam ikon uzlů

Následuje stručný popis jednotlivých činností uzlů v modulu Enterprise Miner.

4.3.2 Uzly pro výběr - SAMPLE

Pro práci se vstupními datovými soubory jsou připraveny tři ikonky: Input Data Source (vstupní datové soubory), Sampling (výběr metadat) a Data Partition (rozdělení dat).



Obrázek 9: Ikony Sample

Obsahem uzlu **Input Data Source** je načíst údaje ze zdrojů a definovat jejich vlastnosti (atributy) pro pozdější zpracování v modulu Enterprise Miner. Tento uzel může provádět různé činnosti:

1. Umožňuje přístup k datovým souborům a datovým skladům. Umožňuje přístup k datovým zdrojům ve formátu SAS, nebo ve spojení s příslušným SAS/ACCESS engine k více než padesáti druhům datových formátů, jako jsou Oracle, DB2 nebo MS SQL Server. Datové sklady se mohou definovat užitím softwaru SAS/Warehouse Administrator a pro Enterprise Miner se vytvoří použitím Enterprise Miner Warehouse Add-Ins.
2. Automaticky vytváří metadatové soubory pro každou proměnnou v datovém souboru.
3. Nastavuje počáteční hodnotu pro hladinu významnosti a role všech proměnných v modelu.
4. Zobrazuje souhrnné statistické charakteristiky pro intervalové a třídící proměnné.
5. Ve vstupním datovém souboru umožňuje definovat cílové profily pro každý cíl.

Uzel **Sampling** umožňuje vybrat náhodný výběr, stratifikovaný náhodný výběr a shluky výběrů datových souborů. Výběr se doporučuje na extrémně rozsáhlé databáze, protože může významně snížit čas trénování modelu. Jestliže je výběr dostatečně reprezentativní, mohou být vztahy nalezené ve výběru využitelné pro zobecnění na původní datový soubor, na celek. Uzel Sampling popisuje vybraná pozorování pro výstupní datový soubor a ukládá zdrojové hodnoty, které se používají pro generování náhodných čísel pro výběry tak, že je možno provádět opakovaný výběr.

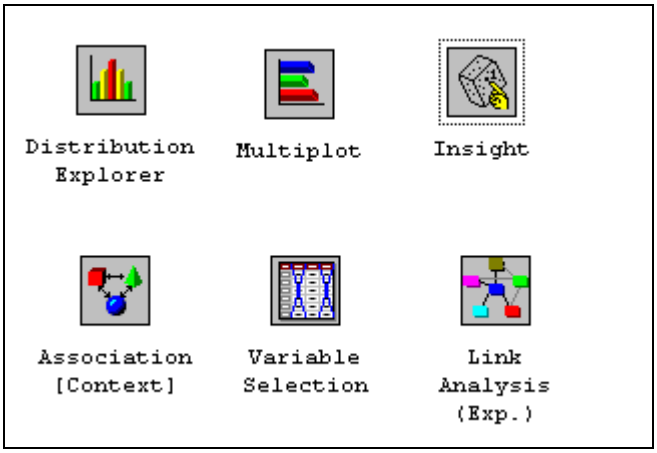
Uzel **Data Partition** umožňuje rozdělit datové soubory na soubory trénovací, testovací a validační. Trénovací datový soubor se používá pro předběžné sestavení modelu (pro odhad). Validační datový soubor se používá k monitoringu a vyladění modelu během odhadu a také se používá pro hodnocení modelu (při výběru nejvhodnějšího modelu). Testovací datový soubor je doplňkovým datovým souborem, který se může použít při vyhodnocování kvality modelu. Tento uzel používá jednoduchý náhodný výběr, stratifikovaný náhodný výběr nebo uživatelem definované nastavení pro tvorbu rozdělených datových souborů.

4.3.3 Uzly pro průzkum – EXPLORE

Uzel **Distribution Explorer** je vizualizační nástroj, který umožňuje rychle a jednoduše prozkoumat rozsáhlé objemy dat ve vícerozměrných histogramech. Pomocí tohoto uzlu se lze podívat na rozdělení až tří proměnných v čase. Pokud je proměnná binární, nominální nebo ordinální, mohou se vybírat specifické hodnoty pro vyřazení ze zobrazení. Vyloučení extrémních hodnot z intervalových proměnných lze provést nastavením rozsahu vyřazení. Uzel také generuje souhrnné statistické charakteristiky pro zobrazené proměnné.

Jiným vizualizačním nástrojem je **Multiplot**, který umožňuje graficky prozkoumat rozsáhlé objemy dat. Na rozdíl od uzlů Insight nebo Distribution Explorer vytváří Multiplot automaticky sloupcové grafy a korelační pole pro vstupní a cílové proměnné bez toho, aby uživatel procházel menu a vybíral položky. Kód vytvořený tímto uzlem může být používán pro tvorbu grafů v dávkovém prostředí, zatímco uzly Insight nebo Distribution Explorer musí být spouštěny interaktivně.

Uzel **Insight** umožňuje otevřít modul SAS/INSIGHT. Tento modul je interaktivním nástrojem pro průzkum dat a průzkumovou analýzu. S jeho pomocí lze prozkoumat data prostřednictvím grafů a analýz, které jsou propojeny přes paralelní okna. Je možno analyzovat jednorozměrné rozdělení, zkoumat vícerozměrná rozdělení a sestavit vysvětlující modely pomocí zobecnění lineárních modelů.



Obrázek 10: Ikony Explore

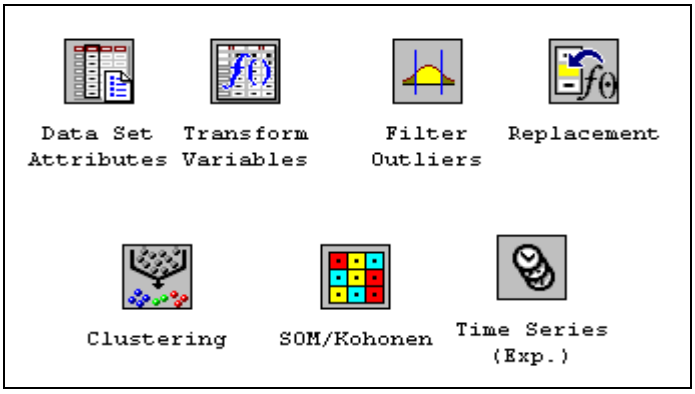
Uzel **Association** umožňuje identifikovat asociační vazby v datech. Např. jestliže zákazník kupuje bochník chleba, s jakou pravděpodobností koupí zákazník i litr mléka? Uzel také umožňuje provést sekvenční odkrývání, pokud je časová proměnná uvedená v datovém souboru.

Uzel **Variable Selection** umožňuje ohodnotit významnost vstupních proměnných v prognózování nebo klasifikování cílové proměnné. K výběru významných vstupů používá uzel buď koeficient determinace nebo kritérium χ^2. Koeficient determinace umožňuje odstranit proměnné v hierarchiích, odstranit proměnné, které mají vysoké procento chybějících hodnot, a odstranit třídící proměnné, které jsou založeny na počtu ojedinělých hodnot. Proměnné, které nejsou ve vztahu k cíli, jsou vyřazeny – je jim přiřazena role vyřazení (rejected). Ačkoli jsou vyřazené proměnné v rámci následujících uzlů uvedeny v procesních diagramech, nejsou tyto proměnné použity jako vstupy modelu do detailnějšího uzlu modelování, stejně jako do uzlu Neural Network a uzlu Tree.

Analýza spojení je zkoumání vazeb mezi efekty v komplexním systému, aby byly objeveny vzory chování, které mohou být použity k odvození užitečných závěrů. Některé aplikace zahrnují tvary analýz detekce podvodů, kriminálních síťových komplotů, vzorů telefonních provozů, struktur webových stránek a jejich využití, vizualizace databáze a analýzu sociální sítě. Uzel **Link Analysis** transformuje data z různých zdrojů do datového modelu, který může být zobrazen pomocí grafů. Datový model podporuje jednoduché statistické míry, prezentuje jednoduché interaktivní grafy pro základní analytické prozkoumání a generuje shluky výsledků ze surových dat, které mohou být používány pro redukci dat a pro segmentaci. Grafika z tohoto uzlu ukazuje vztahy mezi úrovněmi proměnných.

4.3.4 Uzly pro úpravu - MODIFY

Úprava a příprava dat je v Enterprise Mineru reprezentována sedmi uzly: Data Set Attributes (vlastnosti datových souborů), Transform Variables (transformace proměnných), Filter Outliers (filtr na odlehlá pozorování), Replacement (nahrazení), Clustering (shlukování), SOM/Kohonen a Time Series (časové řady).



Obrázek 11: Ikony Modify

Uzel **Data Set Attributes** umožňuje měnit a modifikovat vlastnosti datových souborů, např. názvy datových souborů, jejich popisy a role. Tento uzel lze použít také pro měnění metadatového souboru, který je propojen s datovým souborem a specifikuje cílové profily pro zvolený cíl. Příklad užitečné aplikace uzlu Data Set Attributes je generování datového souboru do uzlu SAS Code a změna jeho metadatového souboru tímto uzlem.

Uzel **Transform Variables** umožňuje transformovat proměnné; např. lze transformovat proměnné pomocí umocnění, pomocí přirozeného logaritmu, maximalizace korelace s cílovou proměnnou nebo normalizování proměnné. Nadto uzel podporuje uživatelsky definované formule pro transformace a poskytuje vizuální rozhraní pro seskupování proměnných intervalových hodnot. Tento uzel také automaticky ukládá intervalové proměnné do oblasti paměti použitím algoritmů, které jsou založeny na rozhodujících stromech. Transformované proměnné podobného měřítka a variability mohou zlepšit odhad modelů a následně klasifikovat a předpovídat přesnost odhadnutých modelů.

Uzel **Filter Outliers** umožňuje aplikovat filtr na trénovací datový soubor za účelem vyloučení pozorování, a to zejména takových, jako jsou odlehlá pozorování či jiná pozorování, jež není žádoucí zahrnout do analýzy pomocí metod Data mining. Uzel Filter Outliers nabízí několik možností pro filtrování na základě spojitých proměnných, především odstranění pozorování při hodnotě dané proměnné za definovaným násobkem směrodatné odchylky od průměru nebo odstraněním pozorování, pokud je hodnota za určitým percentilem. U kategoriálních proměnných lze odstranit pozorování s výjimečnou (minimálně zastoupenou) hodnotou. Uzel nefiltruje pozorování ve validačních, testovacích nebo výsledných datových souborech.

Uzel **Replacement** umožňuje doplnit hodnoty za pozorování, která mají chybějící hodnoty. Lze tedy nahradit chybějící hodnoty intervalové proměnné pomocí průměru, mediánu, středního rozpětí, prostřední minimální vzdálenosti, nahrazení založeného na

rozložení četností nebo lze k nahrazení použít tzv. M-estimator, což může být např. Tukey's biweight, Huber's Wave nebo Andrew's Wave. Chybějící hodnoty pro třídící proměnné mohou být nahrazeny nejčastěji se vyskytující hodnotou, pomocí nahrazení založeného na rozložení četností, na stromovém přičítání nebo konstantou.

Uzel **Clustering** umožňuje segmentovat data; tzn. že umožňuje rozpoznat taková pozorování, která jsou si nějakým způsobem podobná. Pozorování, která jsou si podobná, směřují do stejného shluku a pozorování, která jsou odlišná, směřují do odlišných shluků. Shlukový identifikátor pro každé pozorování může být převeden do následných uzlů v diagramu. Detailnější charakteristika tohoto uzlu je uvedena v následující podkapitole.

Uzel **SOM/Kohonen** generuje samoorganizující se mapy, Kohonenovy sítě a vektorově kvantifikační sítě. Podstatné je, že uzel provádí nekontrolované učení, ve kterém se pokouší naučit strukturu dat. Podobně jako u uzlu Clustering se mapy tvoří podle sítě a charakteristiky se mohou prohlížet graficky pomocí prohlížeče, v němž jsou uvedeny výsledky. Uzel poskytuje analýzu výsledků ve formě interaktivní mapy, která ilustruje charakteristiky shluků. Navíc poskytuje výstup, který vyjadřuje významnost každé proměnné.

Uzel **Time Series** umožňuje porozumět trendům a sezónním výkyvům v zákonitostech. Např. datový soubor může mít mnoho dodavatelů a mnoho zákazníků, stejně tak jako transakčních dat, která lze navzájem spojovat. Velikost každé množiny transakcí může být velmi rozsáhlá, což činí mnoho tradičních datamingových úloh obtížnými. S ohledem na informace v časových řadách lze objevit trendy a sezónní výkyvy ve zvycích zákazníků a dodavatelů, které nemohou být zřejmé z transakčních dat. Uzel konvertuje transakční data (časově označené údaje, které jsou sbírány po určitou dobu nespecificky často) na údaje časových řad (časově označené údaje, které jsou sumarizovány za určitou dobu se specifickou četností).

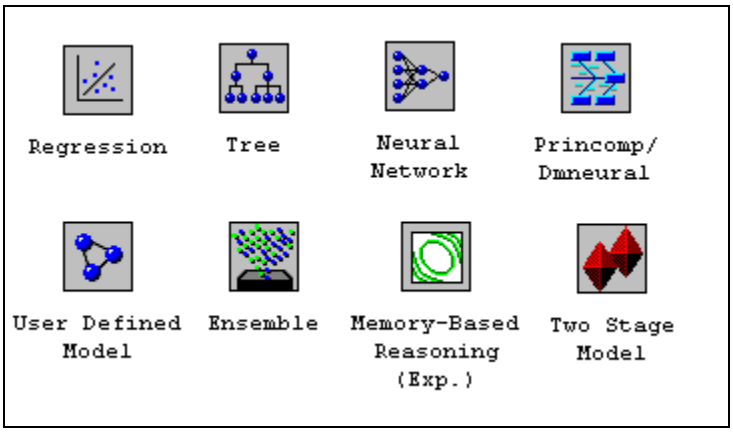
4.3.5 Uzly pro modelování - MODEL

Pro budování modelů je v modulu Enterprise Miner připraveno osm uzlů: Regression (regrese), Tree (rozhodovací strom), Neural Network (neuronová síť), Princomp/Dmneural (analýza hlavních komponent), User Defined Model (uživatelsky definovaný model), Ensemble (spojení modelů), Memory Based Reasoning (logické myšlení) a Two-Stage Model (modely dvou proměnných).

Uzel **Regression** (regrese) umožňuje odhadnout modely lineární a logistické regrese. Lze použít intervalové, ordinální nebo binární závislé proměnné. Na vstupu lze použít jak intervalové tak i diskrétní proměnné. Uzel podporuje tři metody výběru: stepwise, forward a backward. Interaktivní nástroj umožňuje vytvořit velmi přehledné a uspořádané prostředí pro modelování.

Uzel **Tree** (rozhodovací strom) umožňuje provést několika způsoby rozdělení databáze založené na nominálních, ordinálních a intervalových proměnných. Uzel podporuje automatické i interaktivní učení. Pokud je spuštěn uzel Tree v automatickém módu, automaticky do stromu rozřadí vstupní proměnné podle síly jejich rozdělení. Toto rozřazení může být použito pro výběr proměnných, které se použijí pro další modelování. Mohou být vygenerovány i umělé (dummy) proměnné, které se použijí v následném modelování. Lze též vybrat jakýkoli automatický krok s posouzením, které definuje

pravidla dělení a prořeže zřetelné uzly a podstromy. Interaktivní pěstování umožňuje prozkoumat a hodnotit širokou množinu stromů.



Obrázek 12: Ikony Model

Uzel **Neural Network** (neuronová síť) umožňuje vybudovat, vytrénovat a ověřit vícevrstvou neuronovou síť. Všeobecně je každý vstup plně zapojen do první skryté vrstvy, každá skrytá vrstva je plně zapojena do další skryté vrstvy a poslední skrytá vrstva je plně zapojena do výstupu. Uzel Neural Network podporuje mnoho obměn tohoto obecného postupu.

Uzel **Princomp/Dmneural** (analýza hlavních komponent) umožňuje odhadnout další nelineární model, který používá vybrané hlavní komponenty jako vstupy pro predikování binární či intervalové závislé proměnné. Uzel také provádí analýzu hlavních komponent a předává výsledné hlavní komponenty následujícím uzlům. Závislá proměnná (target) musí být binární nebo intervalová pro trénování neuronové sítě, ale pro analýzu hlavních komponent se žádná závislá proměnná nepožaduje.

Uzel **User Defined Model** (uživatelsky definovaný model) umožňuje generovat hodnotící statistické charakteristiky pomocí predikovaných hodnot z modelu, který se vytvoří pomocí uzlu SAS Code (např. logistický model pomocí procedury LOGISTIC v modulu SAS/STAT) nebo pomocí uzlu Variable Selection. Predikované hodnoty mohou být uloženy do datového souboru a potom pomocí uzlu Input Data Source importovány do procesního toku.

Uzel **Ensemble** (spojení modelů) umožňuje slučovat modely. Od spojených modelů se očekává, že se u nich projeví vyšší stabilita než u individuálních modelů. Jsou mnohem efektivnější než individuální modely, projevují nižší korelaci. Uzel vytváří tři různé typy spojení: Combined model, Stratified model a Bagging/Boosting modely.

Logické myšlení (Memory Based Reasoning) je proces, který identifikuje podobné případy a aplikuje informace, které jsou získány z těchto případů na nové záznamy. V Enterprise Mineru je uzel **Memory Based Reasoning** nástrojem, který používá algoritmus k-tého nejbližšího souseda pro kategorizace nebo predikci pozorování. Algoritmus k-tého nejbližšího souseda probírá datový soubor, kde každé pozorování v datovém souboru je složeno z množiny proměnných a zkoumání má jednu hodnotu pro každou proměnnou. Počítá se vzdálenost mezi pozorováním a zkoumáním. K-tá pozorování, která mají nejnižší

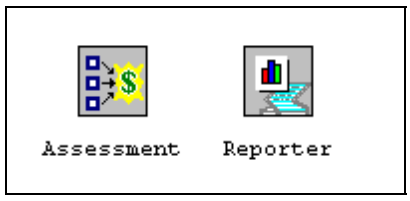
vzdálenosti ke zkoumání, jsou k-tými nejbližšími sousedy k tomu zkoumání. V Enterprise Mineru jsou k-tí nejbližší sousedé určeni euklidovskou vzdáleností mezi pozorováním a zkoumáním. Na základě cílových hodnot k-tých nejbližších sousedů volí každý z k-tých nejbližších sousedů cílovou hodnotu pro zkoumání. Zvolené hlasy jsou následujícími pravděpodobnostmi pro třídící cílovou proměnnou.

Uzel **Two-Stage Model** (modely dvou proměnných) počítá dvoufázový model pro predikci třídící cílové a intervalové cílové proměnné. Uzel automaticky rozpozná třídící proměnnou a hodnotovou proměnnou, stejně jako pravděpodobnostní, klasifikační a predikční proměnnou. Oba modely, třídící i hodnotový, jsou odhadnuty pro třídící cílovou proměnnou a intervalovou cílovou proměnnou v tomto pořadí, v první a druhé fázi. Podle definice přenosové funkce a použití volby filtru lze specifikovat, jak aplikovat predikci třídící proměnné pro třídící cílové proměnné a jestli použít všechna data nebo podmnožinu tréninkových dat v druhé fázi pro intervalovou predikci. Predikce intervalové cílové proměnné je vypočítána z hodnoty modelu a volitelně upravena následujícími pravděpodobnostmi třídící cílové proměnné prostřednictvím volby chyby vyrovnání. Probíhá také následná analýza, která zobrazí hodnotu predikce pro intervalovou cílovou proměnnou podle aktuální hodnoty a predikce třídící cílové proměnné. Výsledný kód uzlu Two-Stage Model je složený z modelů třídících proměnných a intervalových proměnných. Hodnota modelu je používána k tvorbě hodnocení zobrazení v uzlu Model Management a také v uzlu Assessment.

4.3.6 Uzly pro vyhodnocování - ASSESS

Vyhodnocování a porovnávání modelů je možné provádět pomocí uzlu Assessment (hodnocení) a tvorba výstupu o celém průběhu zpracování dat se zadává uzlem Reporter (tvorba výstupů).

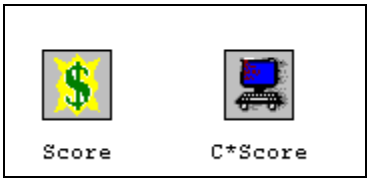
Uzel **Assessment** (hodnocení) poskytuje společný rámec pro srovnávání modelů a predikcí z jakéhokoli z uzlů pro modelování (Regression, Tree, Neural Network a User Defined Model). Porovnání je založeno na očekávaných a skutečných ziscích či ztrátách, které mohou vyplývat z použitého modelu. Uzel poskytuje několik grafů, které pomáhají popisovat užitečnost modelu tak jako grafy navýšení a grafy zisků/ztrát.



Obrázek 13: Ikony Assess

Uzel **Reporter** sestavuje z průběhu analýz výsledky do HTML výstupu, který může být prohlížen v jakémkoli webovém prohlížeči. Každý výstup obsahuje informace v hlavičce, znázornění diagramu a samostatné výstupy pro každý uzel. Výstupy jsou uvedeny do tabulky reportů tzv. Project Navigator.

Pro doplnění práce s výsledky vybraných úloh je k dispozici uzel Score a C*Score.



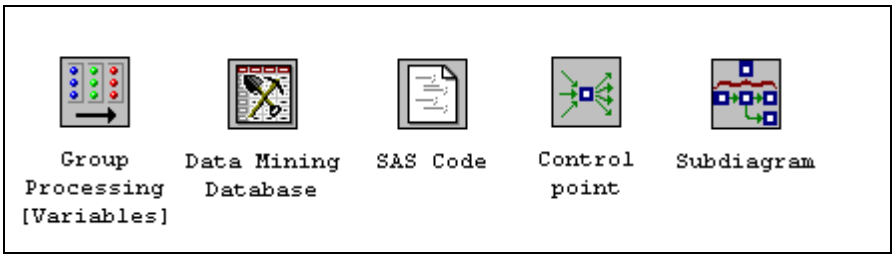
Obrázek 14: Ikony Score

Uzel **Score** umožňuje generovat a uvádět predikované hodnoty z trénovaného modelu. Předpisy pro vyhodnocení jsou vytvořeny pro hodnocení i pro predikování. Enterprise Miner generuje a uvádí předpisy pro vyhodnocení ve formě kódu postupu SAS DATA (pro práci s daty), který může být použit ve většině prostředí systému SAS i bez přítomnosti modulu Enterprise Miner.

Uzel **C*Score** překládá výsledný kód postupu SAS DATA, který se generuje nástroji modulu Enterprise Miner, do programovacího jazyka C.

4.3.7 Ostatní typy uzlů

Pro práci v modulu Enterprise Miner bylo připraveno ještě dalších pět modulů, které mohou být uživateli velmi užitečné. Jedná se o: Group Processing, Data Mining Database, SAS Code, Control Point a Subdiagram.



Obrázek 15: Pomocné uzly

Uzel **Group Processing** umožňuje provést zpracování třídících proměnných v rámci skupin (BY group). Tento uzel lze také použít pro analýzu několikanásobných cílů a opakovaného zpracování stejného datového zdroje podle nastavení módu skupinového zpracování do indexu.

Uzel **Data Mining Database** umožňuje vytvořit dataminingovou databázi (DMDB) pro dávkové zpracování. Pro nedávkové zpracování jsou DMDB databáze automaticky vytvořeny podle potřeby.

Uzel **SAS Code** umožňuje začlenit nový nebo existující kód systému SAS do diagramu úloh. Schopnost napsat kód systému SAS umožňuje zahrnout další procedury systému SAS do analýz Data mining. K vytvoření uživatelského výsledkového kódu je možné též použít postup SAS DATA, podmíněně ke zpracování dat a ke vertikálnímu či horizontálnímu spojení existujících datových souborů.

Uzel **Control Point** umožňuje založit kontrolní bod ke snížení počtu spojení, která jsou vytvořena v diagramu úloh.

Uzel **Subdiagram** umožňuje seskupit část diagramu úloh do subdiagramu. Pro komplexní diagramy úloh lze požadovat vytvoření subdiagramů pro zlepšení náčrtu a kontroly toku úloh.

4.3.8 Nové uzly v SAS Enterprise Miner 5.1

SAS Enterprise Miner 5.1 v systému SAS 9.1.3 obsahuje následující nové uzly:

Segment Profile Node – uzel slouží pro hodnocení datových souborů pomocí štěpících algoritmů s cílem vytvořit segmentované údaje. Segmentované údaje jsou údaje, jež jsou seskupeny podle segmentové proměnné, založené na společných vlastnostech nebo hodnotách mezi specifikovanými vstupními proměnnými. Kromě toho počítá Segment Profile souhrné statistiky pro stanovené proměnné.

Credit Scoring Nodes – jedná se o sadu uzlů, které se objevují na kartě Credit Scoring tab. SAS Enterprise Miner zahrnuje následující čtyři uzly pro kredit skóring:

- Interactive Grouping,
- Scorecard,
- Reject Inference,
- Credit Exchange.

4.4 Uzel Clustering

Uzel **Clustering** (shlukování) patří v datamingové SEMMA metodologii do kategorie „Modify". Uzel Clustering provádí shlukování pozorování, které může být použito k segmentování databází. Shlukování zařazuje objekty do skupin nebo shluků navržených na základě struktury dat. Objekty v každém shluku mají v určitém smyslu tendenci být si navzájem podobné a objekty v odlišných shlucích mají tendenci být si nepodobné. Jestliže se mohou objevovat zřejmé shluky nebo seskupení již před vlastní analýzou, potom shlukovací analýza může být provedena jednoduchým setříděním údajů.

Uzel shlukování implementuje směs obou dříve uvedených pojetí v třífázovém procesu. Jeho stručné shrnutí je následující:

První fázi tvoří následující postup.

a) Nejprve se spustí procedura nehierarchického shlukování na celý soubor, vybere se vysoká hodnota *k*. Středy jsou nastaveny jako prvních *k* dostupných pozorování.

b) Potom se spustí iterační procedura, v každém kroku procedury se vytvářejí dočasné shluky, každé pozorování je přiřazováno do shluku se středem jemu nejbližším. Pokaždé, když je objekt přiřazen do shluku, je střed nahrazen průměrem shluku nazývaným centroid (těžiště).

c) Proces se opakuje, dokud se nedosáhne konvergence, tzn. dokud se nedosáhne stavu, kdy již nedochází k podstatným změnám ve středech shluků.

d) Na konci procedury je dostupných celkem *k* shluků s odpovídajícími centroidy shluků. To je vstupem do druhé fáze.

V druhé fázi se na výběrový soubor spouští metoda hierarchického shlukování s cílem nalézt optimální počet shluků. Protože počet shluků nemůže být větší než *k*, procedura je aglomerativní, začíná na *k* a pracuje sestupně. Předchozí průměry shluků se použijí jako středy a nehierarchická procedura se spouští, aby rozdělila objekty do shluků. Zvláštním aspektem této fáze je, že optimální počet shluků se vybírá s ohledem na testové kritérium, které je postaveno na indexu R^2 a je známé jako Cubic Clustering Criterion – CCC (kubické shlukovací kritérium).

Když se vybere optimální počet shluků, pokračuje algoritmus nehieararchickým shlukováním, aby přiřadil objekty do *k* vybraných skupin, jejichž počáteční středy jsou centroidy získané v předchozím kroku. Tímto způsobem dosáhneme definitivního uspořádání objektů. Tato procedura je podobná té, jež byla použita v první fázi a spočívá v opakování následujících dvou kroků, dokud nedosáhne konvergence:

a) prohlížej data a každé pozorování přiřaď nejbližšímu středu (na základě euklidovské vzdálenosti),

b) nahraď každý střed průměrem pozorování, jež jsou k danému shluku přiřazena.

Uzel Clustering je představován sedmi kartami s následujícími názvy:

Tabulka 5: Základní struktura uzlu Clustering

Název okna anglicky	Význam okna česky
Data Tab	Data
Variables Tab	Proměnné
Clusters Tab	Shluky
Seeds Tab	Středy
Missing Values Tab	Chybějící hodnoty
Output Tab	Výstup
Notes Tab	Poznámky

Pokud je uzel otevřen, zobrazuje se implicitně karta Variables (Proměnné).

4.4.1 Okno Data (Data)

Okno Data se v uzlu Clustering používá k výběru datových souborů na trénovací, validační, testovací a skórovací datové soubory. Po spuštění procesu se jednotlivá pozorování v trénovacím datovém souboru shlukují podle defaultního nastavení. Po specifikaci uživatelem je možno výsledky shlukování použít na validační, testovací nebo skórovací datové soubory.

Obvykle se diagram či schéma začíná sestavovat pomocí uzlu Input Data Source, ve kterém se vybere datový soubor a proměnným se určí role modelu. Vybraný datový soubor je zpracován a analyzován v dalších uzlech. Když je předpříprava hotová, datový soubor se použije jako vstup do uzlu Clustering. Nutno podotknout, že pro provedení shlukovací analýzy není vyžadována žádná cílová proměnná (target variable). Jakákoli proměnná, jíž byla přiřazena role cílové proměnné, není do shlukové analýzy zařazena.

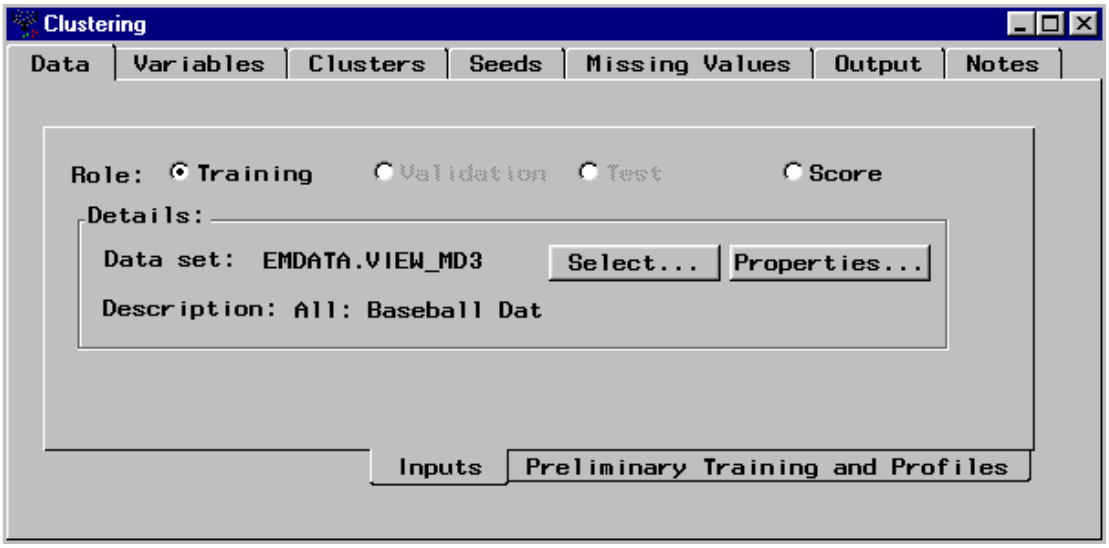
Okno Data obsahuje dvě karty: Input Subtab (vstupy) a Preliminary Training and Profiles Subtab (předběžné trénování a profily).

4.4.1.1 Okno Data: karta Inputs

Karta vstupů (Inputs Subtab) uvádí předcházející datové zdroje v procesním schématu, jímž se přiděluje jedna z následujících rolí:

- trénování - používá se k provedení počáteční shlukové analýzy,
- validace - používá se k dolaďování modelu a k jeho zhodnocení,
- testování - používá se k získání definitivního odhadu chyby modelu,
- skórování - používá se k předpovědi cílových hodnot pro nový datový soubor.

Datové soubory, které jsou zabezpečeny přímo z uzlu Input Data Source, se považují za trénovací datové soubory.



Obrázek 16: Shlukování – okno Data – karta Inputs

Pro natrénování modelu (tj. provedení počáteční shlukové analýzy) se vybírá role trénovací. V tomto případě jsou přepínače „Validation" a „Test" potemnělé, protože v procesním toku neexistují žádné předchozí validační ani testovací datové soubory. V případě, že je do procesního schématu zařazeno několik předcházejících datových souborů (např. dva trénovací soubory), vybírá uzel Replacement automaticky jeden z datových zdrojů.

K náhledu jména a popisky další role datového zdroje slouží přepínače „Validation" a „Test" a „Score". V případě výběru role datového zdroje, pro niž neexistuje žádný platný soubor, pak budou detailní pole prázdná.

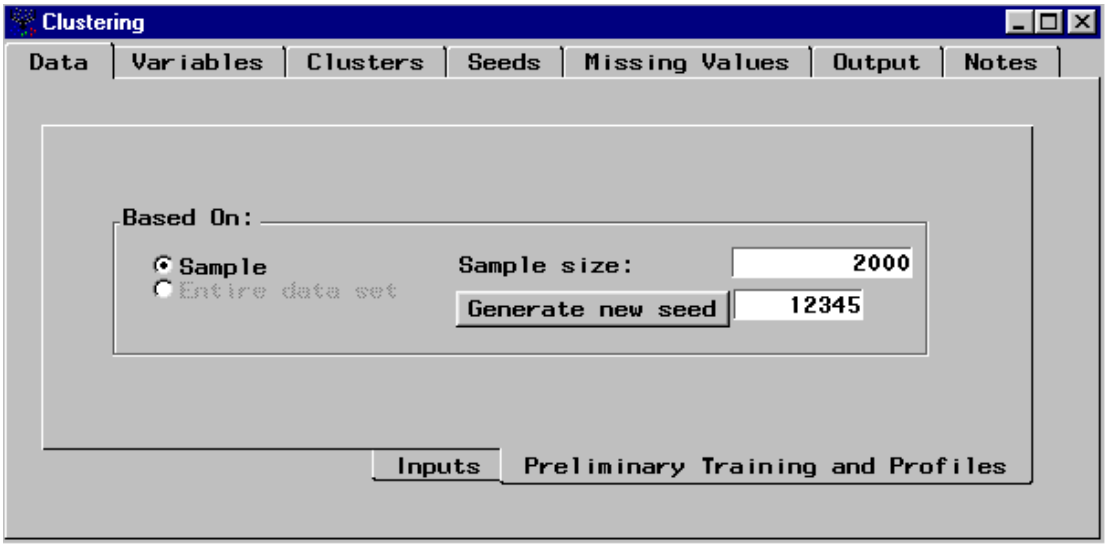
Pro přiřazení rolí datovému zdroji se kliknutím na přepínače *Role* vybere požadovaná role a kliknutím na tlačítko *Select* otevře průvodce importem, v němž se vybere požadovaný datový soubor.

K prohlédnutí detailů a obecných vlastností datových souborů a k nahlédnutí na datový soubor (datovou tabulku) slouží tlačítko *Properties*. Po jeho zmáčknutí se otevře stránka s detaily o datovém souboru a v tabulce Information se zobrazí obecné a souhrnné údaje. Aby bylo možné prohlížet tabulku náhledu na data, vybere se ze stránky Data set details tabulka Table View.

4.4.1.2 Okno Data: karta Preliminary Training and Profiles

Na kartě pro předběžné trénování a profily (Preliminary Training and Profiles Subtab) lze specifikovat velikost výběru, který se vytváří z tréninkového souboru pro předběžné trénování. Přepínač Entire data set (úplný datový soubor) je v této podtabulce potemnělý a nedostupný.

Implicitně je nastaven náhodný výběr tvořený 2000 pozorováními z trénovacích dat s použitím hodnoty středu 12345. Pro změnu velikosti výběru je nutno do pole zadat novou hodnotu.

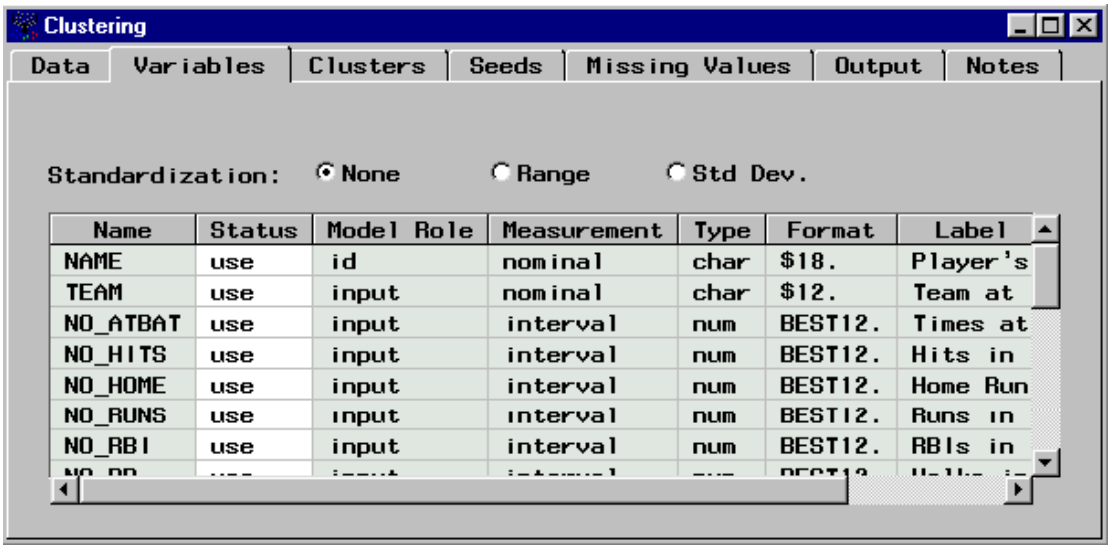


Obrázek 17: Shlukování – okno Data – karta Preliminary Training and Profiles

V téže tabulce je také možné specifikovat hodnotu náhodného středu. Pro jeho změnu se buď zadá do pole nová hodnota nebo se vybere tlačítko Generate new seed, které automaticky vygeneruje nový střed. Použitím stejné hodnoty středu v následných krocích replikuje náhodný výběr vybraných pozorování. Pro změnu náhodně vybraných pozorování je zapotřebí měnit hodnotu středu.

4.4.2 Okno Variables (Proměnné)

Okno Variables se používá k prohlédnutí proměnných, jejich statutu, jejich rolí v modelu a dalších atributů a vlastností. Zobrazeny jsou všechny vstupní proměnné stejně jako jejich četnost a ID, což už bylo možné specifikovat v uzlu Input Data Source.



Obrázek 18: Shlukování – okno Variables

Tlačítka **Standardization** umožňují nastavit jednu z vnitřních standardizačních metod.

- **None** (žádná) specifikuje, že proměnné nejsou před shlukováním standardizovány.
- **Range** (rozpětí) specifikuje, že hodnoty proměnné jsou vyděleny rozpětím. Průměr se neodečítá.
- **Std Dev.** (směrodatná odchylka) specifikuje, že hodnoty proměnné jsou vyděleny směrodatnou odchylkou. Průměr se neodečítá.

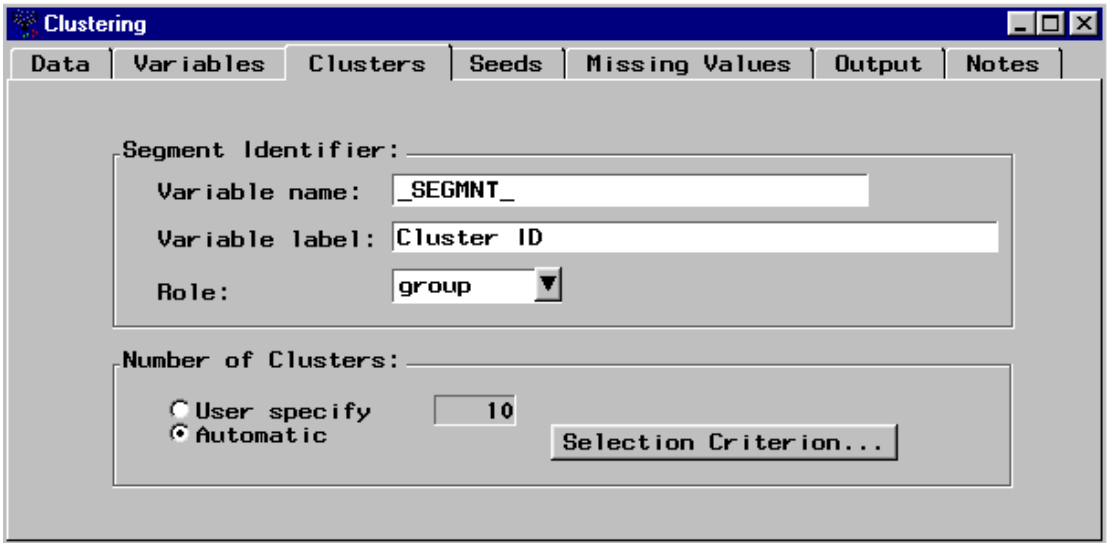
Ve sloupci Status je pro každou vstupní proměnnou implicitně nastaveno **use**. Zamítnuté proměnné jsou označené jako **don't use**. Status **don't use** vylučuje proměnné ze shlukové analýzy. Následujícím způsobem lze vybrané proměnné vyloučit:

1. vyberou se řádky těch proměnných, které se mají vyloučit,
2. klikne se na pravé tlačítko myši ve sloupci Status a z překryvného menu se vybere Set Status,
3. z druhého překryvného menu se vybere **don't use**.

Proměnné, které mají status **don't use**, jsou vyloučeny ze shlukové analýzy, ale dále v datových souborech zůstávají.

4.4.3 Okno Clusters (Shluky)

Okno Clusters se používá pro specifikaci možností pro segmentový identifikátor a pro specifikaci počtu shluků. Pojem „segment" se vztahuje ke shluku pozorování; pro praktické účely se tudíž shoduje pojem „segment" s pojmem „shluk".



Obrázek 19: Shlukování – okno Clusters

4.4.3.1 Segmentový identifikátor

Segmentový identifikátor se skládá ze tří položek: název proměnné, popiska proměnné, role.

Název proměnné (Variable name) určuje název segmentového identifikátoru. Implicitně je nastaven název _SEGMNT_. Pro specifikaci odlišného pojmenování je nutné vepsat nový název do políčka.

Popiska proměnné (Variable label) uvádí označení shluku, implicitní nastavení popisky je CLUSTER ID. Pro specifikaci odlišného pojmenování je nutné vepsat novou popisku do políčka.

Role je rolí modelu, která je přidělena proměnné, jež se používá pro utváření shluků. Implicitně se segmentovému identifikátoru přiřazuje role skupiny. Pro volbu odlišné role modelu se z rozevíracího seznamu vybírá vhodná role modelu. Mezi modelové role patří skupina, ID, vstup a cíl. Skupinová role modelu je také užitečná při zpracovávání na základě třídění (BY group). Segmentový identifikátor si ponechává vybranou roli proměnné tak, jak uvádějí další nástroje v procesním vývojovém diagramu.

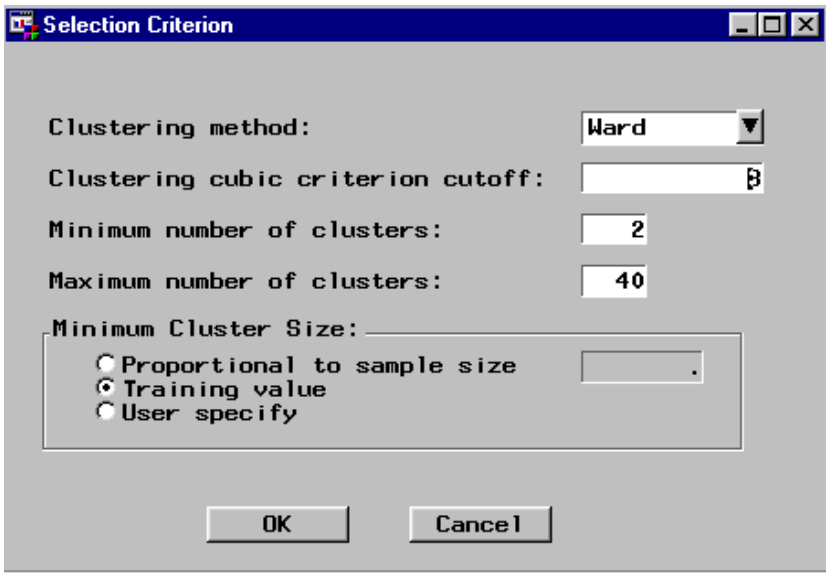
4.4.3.2 Počet shluků

Uzel Clustering implicitně používá pro určení počtu shluků automatické kritérium. Přepínač „User Specify" však umožňuje zadat počet shluků podle přání uživatele. Standardně je v tomto poli nastaveno 10 shluků, ale je možno volit jakoukoli kladnou celočíselnou hodnotu větší nebo rovnu 2.

Je možné požadovat provedení shlukové analýzy pomocí rozdílných hodnot maximálního počtu shluků. Předběžná shluková analýza může identifikovat odlehlá pozorování.

Automatický výběr počtu shluků

V případě výběru tlačítka Selection Criterion se otevře dialogové okno, ve kterém lze měnit způsoby automatického určení počtu shluků.



Obrázek 20: Shlukování –okno Clusters – Selection Criterion

V dialogovém okně „Selection Criterion" se mohou měnit následující položky:

- Metoda shlukování. Poklikáním na šipku lze vybírat metodu:
  - *Průměrovou* – vzdálenost mezi dvěma shluky je průměrnou vzdáleností mezi dvojicemi pozorování, jedno v každém shluku.
  - *Centroidní* - vzdálenost mezi dvěma shluky je definována jako (kvadratická) euklidovská vzdálenost mezi jejich těžišti (centroidy) nebo průměry.
  - *Wardovu* (přednastavená metoda) - vzdálenost mezi dvěma shluky je dána součty čtverců analýzy rozptylu mezi dvěma shluky, které jsou spočítány ze všech proměnných.
- Zastavení shlukovacího kubického kritéria (Clustering Cubic Criterion) je automaticky nastaveno na hodnotu 3. Problematice je věnováno několik poznámek na jiném místě této práce.
- Minimální počet shluků (implicitně je nastaveno 2).
- Maximální počet shluků (implicitně je nastaveno 40).
- Minimální velikost shluku. Minimální velikost shluku se používá v přípravné trénovací fázi k výběru optimálního počtu shluků. Minimální velikost shluku lze volit jako:
  - Proporcionální k výběrovému souboru (Proportional to sample size): Enterprise Miner nalézá poměr mezi minimální velikostí shluku pro trénovací data a mezi počtem pozorování v trénovacích datech, potom aplikuje tento poměr na výběrový soubor k určení výběrové minimální velikosti shluku. Pokud by měl například trénovací soubor 10 tis. pozorování a minimální velikost shluku by byla 500, znamenalo by použití proporcionální velikosti k výběrovému souboru, že pokud je výběrový soubor o velikosti 2000 pozorování, je minimální velikost shluku (2000*[500/10000]) nebo-li 100.
  - Trénovací hodnotu (Training value): specifikuje stejnou minimální velikost shluku, která byla použita pro trénování dat. Použitím příkladu výše uvedených hodnot by minimální velikost shluku byla zadána 500.
  - Stanovenou uživatelem (User specify): jakékoli číslo, které si uživatel zadá.

Potvrzením OK se uloží změny v nastavení tohoto dialogového okna.

Automatický výběr počtu shluků pracuje následovně:

1. Procedura PROC FASTCLUS (nově DMVQ) probíhá na přípravném výběru, aby byly vytvořeny počáteční shluky. K určení počtu počátečních shluků se použije hodnota „Maximální počet shluků".
2. Spustí se procedura PROC CLUSTER, která na vstupu používá průměrné hodnoty počátečních shluků. Vybírá se nejnižší počet shluků, pro který platí, že
   - musí být větší nebo roven minimálnímu počtu shluků, jež je uveden v okně Selection Criterion,
   - CCC překročí mez zastavení shlukovacího kubického kritéria (Clustering Cubic Criterion = CCC).

4.4.4 Okno Seeds (Středy)

V okně Seeds jsou tři karty: General, Initial a Final.

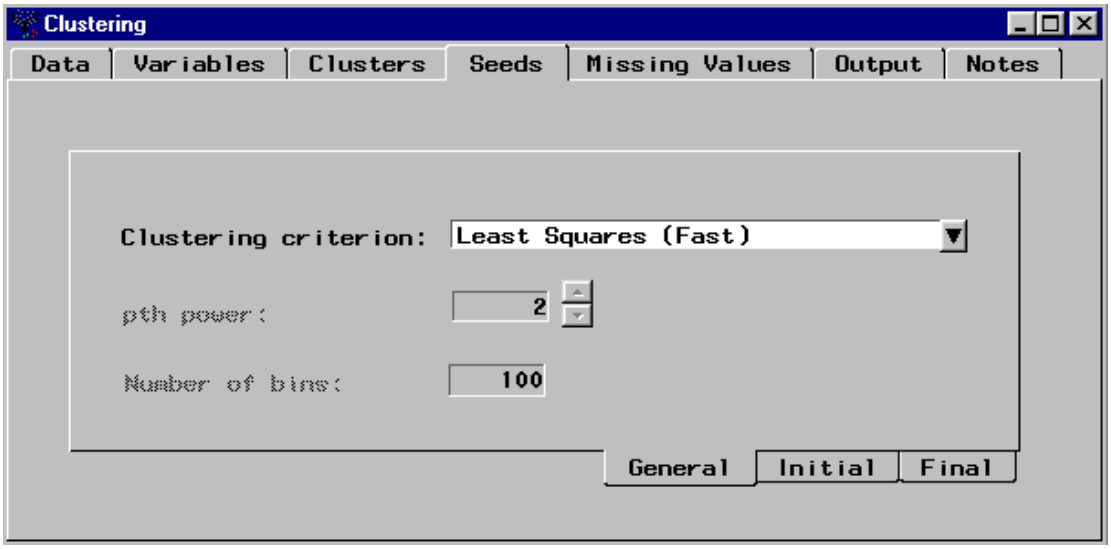
4.4.4.1 Okno Seeds: karta General

Karta General se používá pro definování shlukovacího kritéria. Implicitní nastavení shlukovacího kritéria je Least Squares (Fast) – metoda nejmenších čtverců, kdy se shluky konstruují tak, že součet čtverců vzdáleností pozorování od průměru shluku je minimalizován. Provádí se jenom jedna iterace.

Ke stanovení odlišného shlukovacího kritéria slouží nabídka v rozevíracím seznamu, z níž lze vybrat vhodné kritérium. Dostupná shlukovací kritéria jsou následující:

- *Least Squares (Fast)* – (Metoda nejmenších čtverců – rychlá) minimalizuje součet čtverců vzdáleností datových bodů od průměru shluku. Provádí se jenom jedna iterace. Implicitně je možné provést maximálně 1 iteraci.
- *Mean Absolute Deviation (Median)* (Kritérium průměrné absolutní odchylky) vyžaduje specifikovat počet zásobníků, má implicitně nastaveno 100 zásobníků. Pro specifikaci jiné hodnoty se musí vybrat pole Number of bins a vepsat do něj vhodnou hodnotu.
- *Modified Ekblom-Newton* (Modifikované Ekblomovo-Newtonovo kritérium) vyžaduje specifikovat p-tou mocninu. Její implicitní hodnota je 1,5; uživatel může zvolit hodnotu v rozpětí od 1 do 2. Pro specifikaci odlišné hodnoty slouží šipky nahoru a dolů; každé kliknutí na šipku změní hodnotu o 0,001. Implicitně lze provést maximálně 20 opakování.
- *Least Squares* (Metoda nejmenších čtverců) minimalizuje součet čtverců vzdáleností datových bodů od průměrů shluků. Oproti *Least Squares (Fast)* se obvykle provádí více než jedna iterace.
- *Newton* (Newtonovo kritérium) vyžaduje nastavení p-té mocniny. Její implicitní hodnota je 2,001. Pro specifikaci jiné hodnoty slouží šipky nahoru a dolů; každé kliknutí na šipku změní hodnotu o 1,0. Zvolit lze jakékoli číslo větší než 2. Implicitně lze provést maximálně 20 opakování.
- *Midrange* (Střední rozpětí) minimalizuje vzdálenost středního rozpětí datových bodů od průměru shluků.

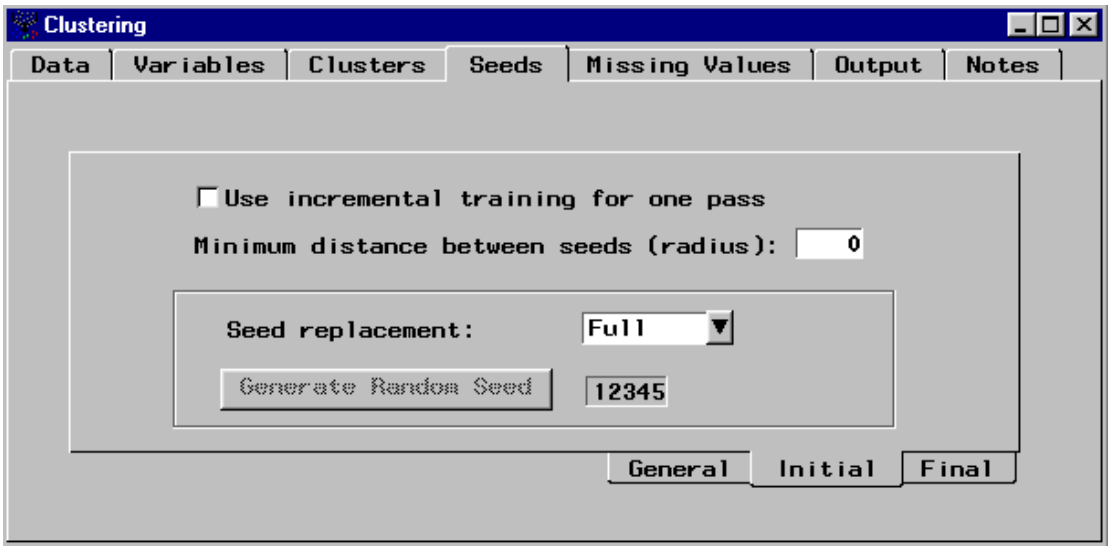
Maximální počet iterací pro shlukovací kritéria je možné upřesnit na kartě Final v okně Seeds.



Obrázek 21: Shlukování – okno Seeds – karta General

4.4.4.2 Okno Seeds: karta Initial

Podtabulka Initial (Počáteční) na kartě Seeds slouží ke specifikaci způsobu inicializace středů shluků.



Obrázek 22: Shlukování – okno Seeds – karta Initial

Pokud je zaškrtnuté pole vedle „Use incremental training for one pass" (Použij dodatečné trénování pro jedinou realizaci), potom se středy mohou pohybovat jako počáteční středy algoritmizovaných výběrů. Počáteční středy musí být úplnými případy (tj trénovací případy, které nemají žádné chybějící hodnoty) a vyžadují, aby byly odděleny na základě euklidovské vzdálenosti, tj. nejméně o hodnotu stanovenou pro minimální vzdálenost mezi středy shluků (Minimum distance between cluster seeds). Implicitně je nastaveno, že se

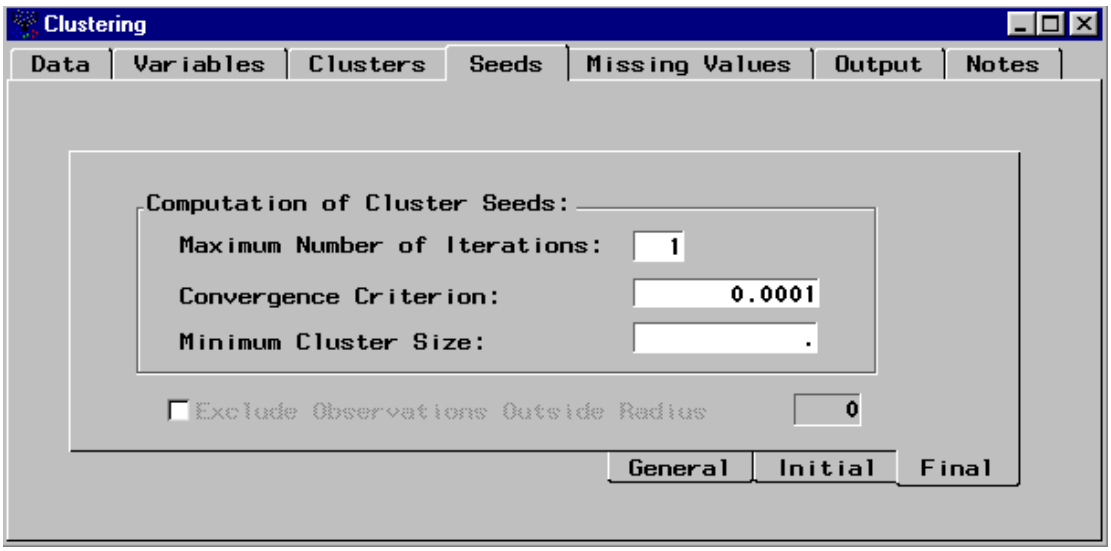
středy volí tak vzdálené, jak jen to je možné; tj. nahrazení středu je nastaveno na **Full** (maximální).

Volbu nahrazení středu lze nicméně změnit na Partial (částečně), None (žádný) a Random (náhodně).

- Jestliže se vybere **Partial** (částečně), potom se přemisťují jen ty středy, které nesplňují požadavek minimální vzdálenosti (od průměru shluku).
- Jestliže se vybere **None** (žádný), jsou počátečními středy pro *n* shluků první n-tá úplná pozorování v datovém souboru. Nutno poznamenat, že výběrem None se dosahuje nejrychlejšího výpočetního času; aby se však dosáhlo kvalitních shluků, doporučuje se uvést do pole vhodnou hodnotu pro minimální vzdálenost mezi středy shluků (tj. rádius shluku).
- Jestliže se pro nahrazení středu vybere **Random** (náhodně), jsou středy shluků náhodně vybranými úplnými případy. Implicitně je pseudonáhodné číslo středu nastaveno na 12345 (neplést se středy shluků). Pro změnu středu se do pole zapisuje nová hodnota nebo se zmáčkne tlačítko Generate Random Seed, které automaticky generuje nový střed.

4.4.4.3 Okno Seeds: karta Final

Karta Final (Definitivní) v okně Seeds slouží ke kontrole ukončovacích kritérií pro generování středů shluků.

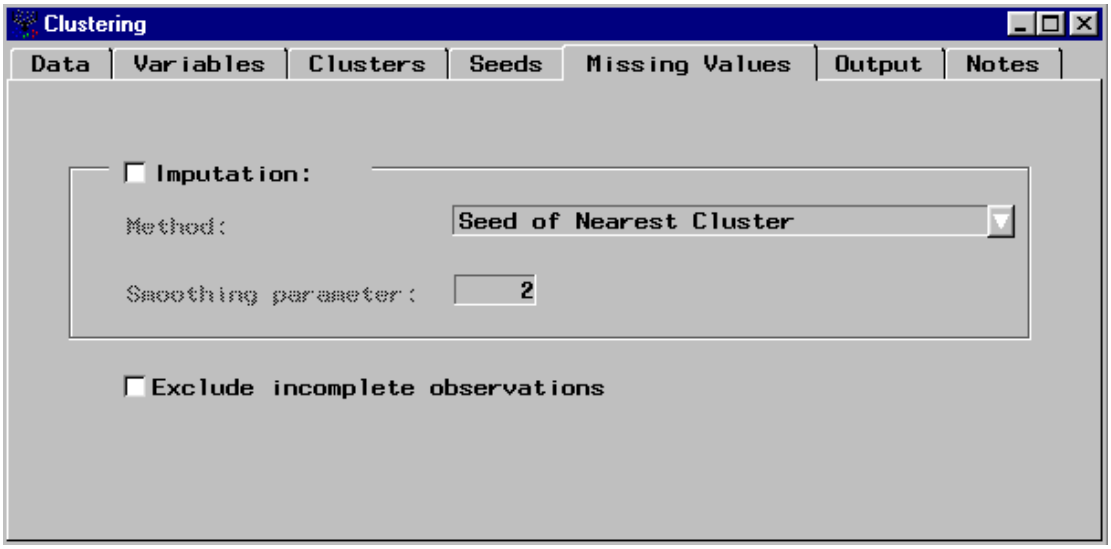


Obrázek 23: Shlukování – okno Seeds – karta Final

Na této kartě lze specifikovat maximální počet shlukovacích iterací, kritérium konvergence a minimální velikost shluku. Implicitní nastavení těchto polí závisí na shlukovacím kritériu, které bylo stanoveno na kartě General v okně Seeds.

4.4.5 Okno Missing Values (Chybějící hodnoty)

Okno Missing Values (Chybějící hodnoty) slouží ke specifikaci jak zacházet s pozorováními, která obsahují nějaké chybějící hodnoty. Ta pozorování, jež mají chybějící hodnoty, nemohou být použita jako středy shluků. Pozorování, jež pro každou proměnnou obsahují chybějící pozorování, jsou z analýzy vyloučena.



Obrázek 24: Shlukování – okno Missing Values

V okně Missing Values jsou dvě volby, které umožňují vypořádat se s daty obsahujícími chybějící hodnoty: případy s chybějícími hodnotami vyloučit během algoritmu shlukování nebo nahradit chybějící hodnoty ve výstupním datovém souboru.

Aby se mohly z algoritmu shlukování vyloučit případy, jež obsahují chybějící hodnoty, musí se zaškrtnout pole „Exclude incomplete observation" (vyloučit neúplná pozorování). Tato možnost nezabraňuje nahrazení chybějících hodnot ve výstupním datovém souboru.

Nahrazení hodnot se zajistí zaškrtnutím pole Imputation a specifikací přisuzování hodnot. K dispozici jsou následující metody přisuzování:

- *Střed nejbližšího shluku* (Seed of Nearest Cluster).
- *Průměr nejbližšího shluku* (Mean of Nearest Cluster) – může se použít, jestliže je nastavena metoda nejmenších čtverců.
- *Podmíněný průměr* (Conditional Mean) – může se použít, pokud je nastavena metoda nejmenších čtverců. Jestliže je vybrán podmíněný průměr, potom se musí specifikovat také vyrovnávací parametr (smoothing parametr). Vyrovnávací hodnota je multiplikována vnitroshlukovou směrodatnou odchylkou. Implicitně se nastavuje na hodnotu 2, ale lze uznat jakékoli kladné číslo. Podmíněný průměr chybějící proměnné se počítá pomocí integrálu odhadu smíšené hustoty (jež je založena na proměnných bez chybějících hodnot) s vnitroshlukovou směrodatnou odchylkou zvětšenou faktorem, který je specifikován vyrovnávacím

parametrem. Čím větší je vyrovnávací parametr, tím je hustota odhadu vyrovnanější.

4.4.6 Okno Output (Výstup)

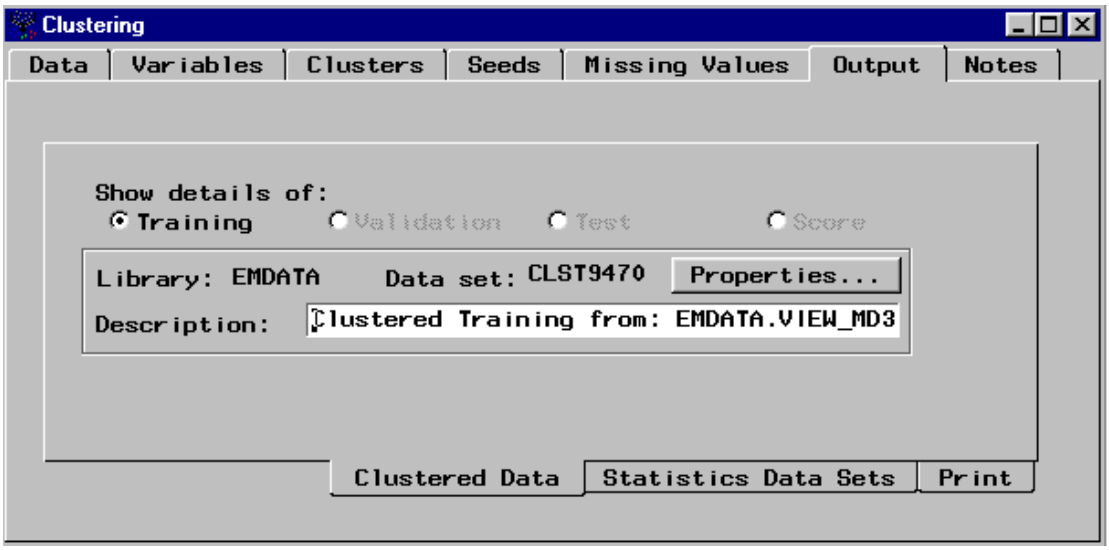
Okno Output (Výstup) se skládá z tří karet: Clustered Data (shlukovaná data), Statistics Data Sets (statistiky datových souborů), Print (tisk).

4.4.6.1 Okno Output – karta Clustered Data

Karta **Clustered Data** (shlukovaná data) v okně Output zaznamenává knihovny dat a výstupní datové soubory pro trénování, validaci, testování a skórování. Tyto datové soubory jsou uloženy v knihovně projektu.

Soubory shlukovaných dat zahrnují původní data, segmentovou proměnnou a proměnnou vzdáleností (vzdálenost každého pozorování od středu shluku). Proměnná vzdáleností má přiřazenou roli „rejected". Jestliže se nahrazují chybějící hodnoty, potom je do datového souboru přidána další proměnná (_IMPUTE_).

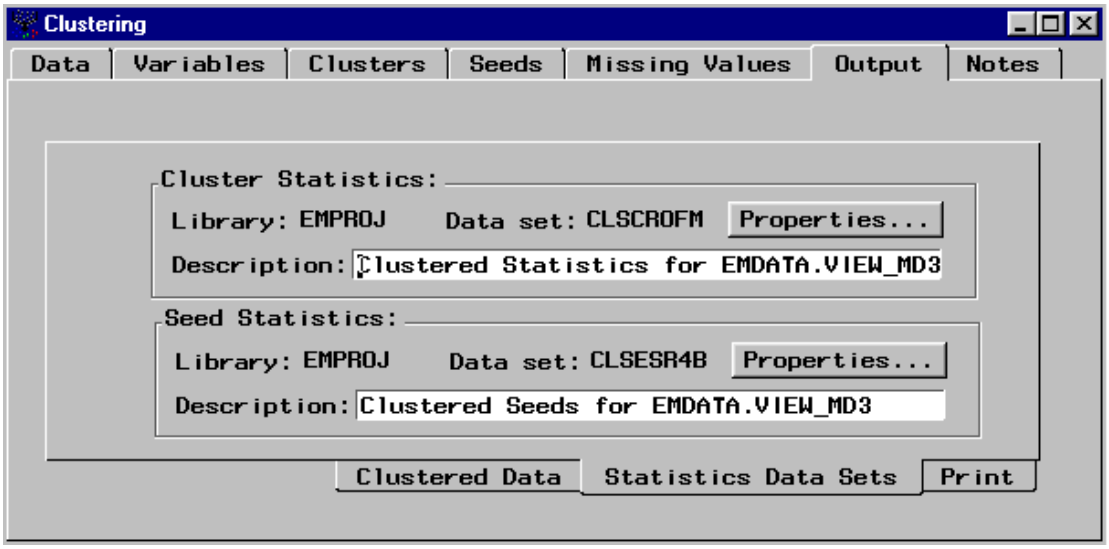
Jedinečné názvy datového souboru se přiřazují automaticky. K prohlédnutí základních informací o shlukovaném datovém souboru slouží tlačítko „Properties".



Obrázek 25: Shlukování – okno Output – karta Clustered Data

4.4.6.2 Okno Output – Karta Statistics Data Sets

Karta **Statistics Data Sets** (statistiky datových souborů) z okna Output zaznamenává datové soubory, které obsahují charakteristiky shluků a středů. Datový soubor s charakteristikami shluků zahrnuje statistiky o každém shluku. Datový soubor s charakteristikami středů zahrnuje informace o středech shluků, které mohou být použity pro shlukování jiných datových souborů pomocí procedury DMVQ (dříve FASTCLUS).



Obrázek 26: Shlukování – okno Output – karta Statistics Data Sets

Jedinečné názvy datového souboru se opět přiřazují automaticky. K prohlédnutí základních informací o charakteristikách datových souborů a k prohlédnutí datových tabulek slouží tlačítko „Properties".

4.4.6.3 Okno Output– karta Print

Karta **Print** (tisk) z okna Output slouží ke specifikaci výstupu. Implicitní nastavení výstupu je navoleno *Cluster Statistics* (charakteristiky shluků). Volitelně je možné uvést *Distance between cluster mean* (vzdálenost mezi průměry shluků) nebo *Cluster Listing* (výpis shluků). Lze také potlačit všechny výstupy a to pomocí *Suppress Output*. (potlačení výstupu) [62].

4.4.7 Prohlížeč výsledků uzlu Clustering

Po spuštění uzlu **Clustering** je možné výsledky prohlížet pomocí prohlížeče výsledků (Results Browser). Prohlížeč výsledků shlukování je okno, které obsahuje karty uvedené v následující tabulce.

Tabulka 6: Přehled karet v okně Results Browser (prohlížeč výsledků) pro uzel Clustering

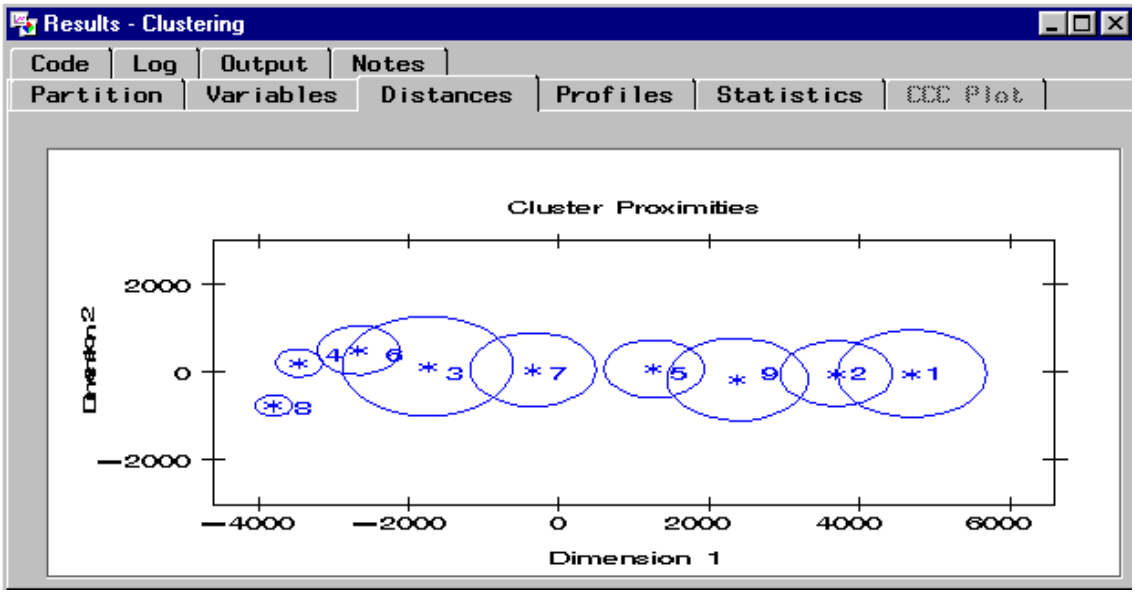
Partition Tab	Segment
Variables Tab	Proměnné
Distances Tab	Vzdálenosti
Profiles Tab	Profily
Statistics Tab	Statistické charakteristiky
CCC Plot Tab	Graf CCC
Output Tab	Výstupy
Log Tab	Záznam
Code Tab	Kód programu
Notes Tab	Poznámky

Karta **Partition** (Segment) poskytuje grafické znázornění klíčových charakteristik a vlastností shluků. Na levé straně karty je třírozměrný koláčový graf s následujícími nastaveními: šířky, výšky a barvy dílku.

Na pravé straně karty je typicky zobrazen síťový graf vstupních průměrů pro všechna trénovací data napříč všemi shlukovými segmenty. Síťový graf vstupních průměrů může porovnávat vstupní průměry za celý datový soubor s průměry vybraného shluku (např. na obrázku 14 se porovnávají vstupní průměry všech shluků se shlukem č. 12).

Karta **Variables** (Proměnné) uvádí všechny ze vstupních proměnných (které byly používány ve shlukovací analýze) a míry, typy a popisky pro každou vstupní proměnnou. Pro každou proměnnou se počítá tzv. *hodnota významnosti* (Importance), která se pohybuje v intervalu od 0 do 1 a představuje míru zásluh dané proměnné, když se vytvářely shluky.

Karta **Distances** (Vzdálenosti) poskytuje grafické znázornění velikosti každého shluku a vztahy mezi shluky.



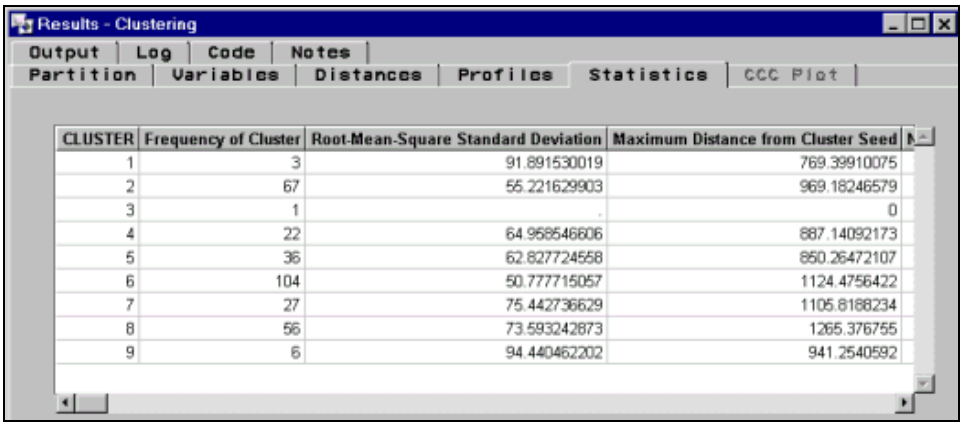
Obrázek 27: Výsledky shlukování graficky na kartě Distances

Osy jsou určeny na základě analýzy vícerozměrného škálování za použití matice vzdáleností mezi průměry shluků na vstupu. Hvězdičky jsou středy shluků a kroužek představuje rádius shluku. Shluk, který obsahuje pouze jedno pozorování, se zobrazuje jako hvězdička. Rádius každého shluku závisí na nejvíce vzdáleném pozorování v daném shluku. Pozorování nesmějí být rovnoměrně rozdělena v rámci shluků. Z tohoto důvodu se může vyskytnout situace, že se shluky překrývají, ale v podstatě je každé pozorování přiřazeno pouze do jednoho shluku. Vzdálenosti mezi shluky jsou založeny na kritériu, které se specifikuje při sestavování shluků.

Karta **Profiles** (Profily) poskytuje grafické znázornění kategoriálních a spojitých proměnných pro každý shluk. Implicitně se jako první zobrazuje graf kategoriálních proměnných.

Karta **Statistics** (Statistické charakteristiky) implicitně zobrazuje informace o každém shluku v tabulkové formě. Pro výpočet následujících statistických charakteristik je použit celý tréninkový soubor dat:

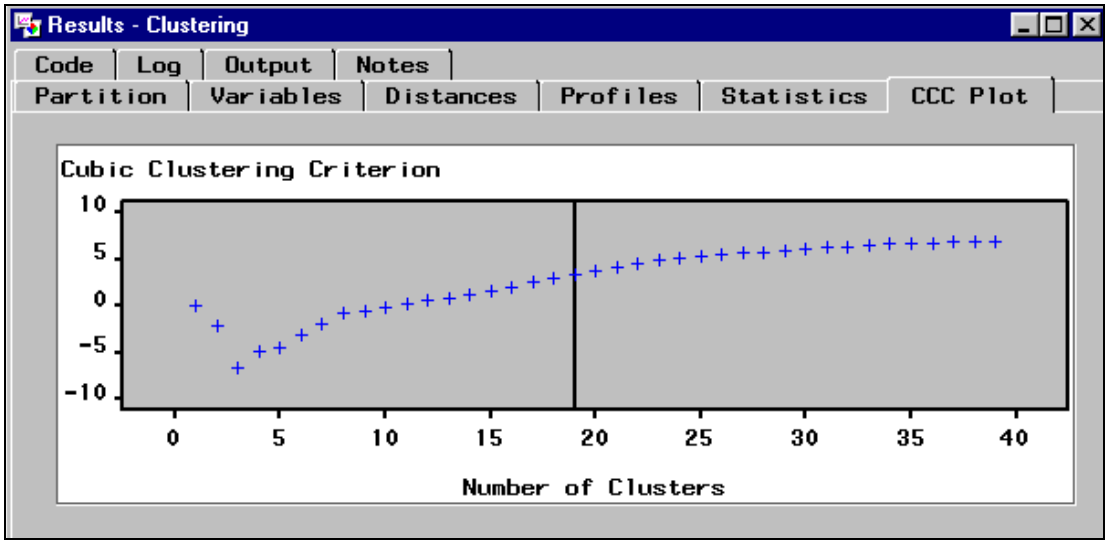
- *Frequency of the Cluster* (Četnost shluku) – počet pozorování (případů) v jednotlivém shluku.
- *Root-Mean-Square Standard Deviation* (Střední kvadratická chyba směrodatné odchylky) – střední kvadratická chyba směrodatných odchylek shluku ze všech proměnných, shoduje se se střední kvadratickou vzdáleností mezi pozorováními v daném shluku.
- *Maximum Distance from Cluster Seed* (Maximální vzdálenost od středu shluku) – maximální vzdálenost od středu shluku k jakémukoli případu (pozorování) v daném shluku.
- *Nearest Cluster* (Nejbližší shluk) – číslo shluku s průměrem, který je nejblíže k průměru aktuálního shluku.
- *Distance to Nearest Cluster* (Vzdálenost k nejbližšímu) – vzdálenost mezi centroidy (průměry) aktuálního shluku a nejbližšího jiného shluku.
- *Mean* (Průměr) – průměr pro jednotlivé vstupní proměnné.



CLUSTER	Frequency of Cluster	Root-Mean-Square Standard Deviation	Maximum Distance from Cluster Seed
1	3	91.891530019	769.39910075
2	67	55.221629903	969.18246579
3	1	.	0
4	22	64.958546606	887.14092173
5	36	62.827724558	850.26472107
6	104	50.777715057	1124.4756422
7	27	75.442736629	1105.8188234
8	56	73.593242873	1265.376755
9	6	94.440462202	941.2540592

Obrázek 28: Výsledky shlukování v číslech na kartě Statistics

Karta **CCC Plot** (Graf CCC) představuje graf kubického shlukovacího kritéria (Cubic Clustering Criterion) v závislosti na počtu shluků. Optimální počet shluků vybere uzel Clustering automaticky.



Obrázek 29: Výsledky shlukování na kartě CCC Plot

Karta **Output** (Výstupy) zobrazuje výstupy, které byly zaznamenány v průběhu výpočtu procedur FASTCLUS a CLUSTER ze základního modulu SAS/STAT. Jestliže je vybrán pevný počet shluků, potom je vidět výstup z průběhu procedury FASTCLUS na základě tréninkových dat a použitého nastavení počtu shluků. Jestliže se použije automatického výběru počtu shluků, pak lze shlédnout tři části výstupu:

1. výsledky spuštění procedury FASTCLUS na základě počátečního výběru,
2. výsledky spuštění procedury CLUSTER na základě shlukových průměrů získaných z procedury FASTCLUS,
3. výsledky spuštění procedury FASTCLUS na základě trénovacích dat, když se použije (automaticky) vybraný počet shluků.

Karta **Log** (Záznam) zobrazuje poznámky a záznamy, které se generují během procesu shlukování.

Karta **Code** (Kódy) umožňuje zhlédnout kód, který se automaticky vygeneroval pro shlukování dat.

Karta **Notes** (Poznámky) se využívá pro zápis a uložení poznámek o projektu [62].

5 VÝSLEDKY DISERTAČNÍ PRÁCE

Mezi stěžejní výsledky disertační práce patří prozkoumání aplikace technik Data mining ve vybraném softwarovém programu a evaluace různých přístupů. Výzkumná práce je realizována na základě sběru původních e-mailových hlášení monitorovacího systému lokálního poskytovatele připojení k internetu (více než 11 miliónů záznamů).

První podkapitola se zaměřuje na jedno z těžišť práce, a to na vlastní návrh přípravy dat do vhodné struktury. Příprava dat je v procesu vytváření modelu jedním z nejdůležitějších kroků a z hlediska času představuje nejnáročnější část. Pomocí programování v systému SAS bude upraven flat file (textový soubor *.txt) na celou sadu proměnných vhodných pro statistické zpracování a modelování.

Druhá podkapitola představuje aplikaci shlukovací metody včetně různých aspektů jejího použití pomocí programování v systému SAS. Práce porovnává a hodnotí možnosti použití různých vstupních dat a parametrů procedury. Tou je vzhledem k velkému objemu dat procedura FASTCLUS. Cílem této části je popsat různé aspekty řešení a vyvodit závěry pro další praktické využití i vědecko-výzkumnou činnost.

Třetí podkapitola se věnuje praktickému řešení ukázkové úlohy v SAS Enterprise Miner. Zaměří se především na prozkoumání funkčnosti uzlu Clustering (shlukování), který zabezpečuje použití statistické vícerozměrné metody „shlukování" za účelem segmentace. Zpracování této části analýz umožní porovnat výhody a nevýhody použití programování a na míru připraveného modulu pro Data Mining.

Pro disertační práci je používán SAS/Enterprise Miner ve verzi 4.3.

5.1 Příprava dat

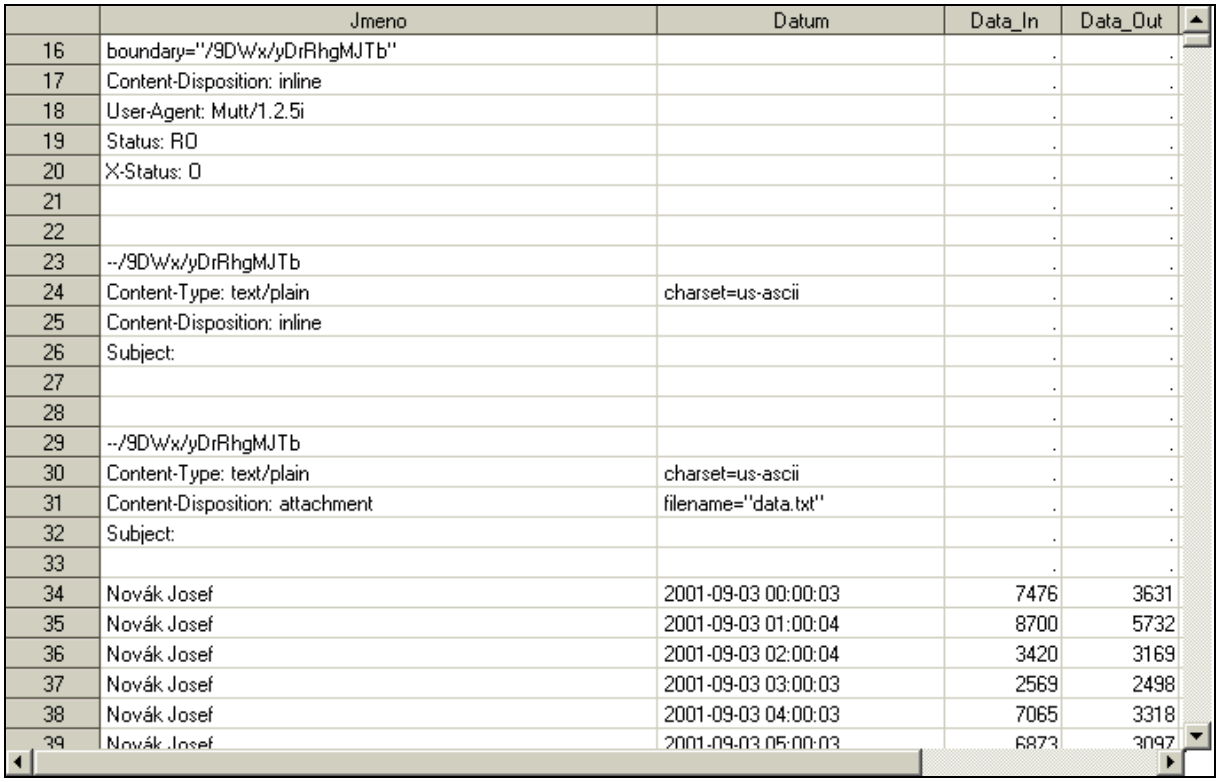
Příprava dat patří k časově nejnáročnějším úkonům spojeným s vlastní realizací technik Data mining. Pro účely disertační práce byly po dobu necelých čtyř let (září 2001 – květen 2005) sbírány údaje transakčního charakteru, historické záznamy představují aktivitu zákazníků telekomunikační firmy. Jedná se tedy o interní datový zdroj, který představuje jistou záruku kvality.

Údaje byly sbírány denně s jednou měsíční pauzou (chybí údaje od 20. dubna 2004 do 19. května 2004), kterou zapříčinila porucha monitorovacího systému. Ten každodenně sestavoval výstupní zprávy ve formě e-mailu. Jejich zápis je uložen ve dvou souborech ve formátu TXT, první soubor obsahuje e-maily do 19. dubna 2004, druhý od 20. května 2004. Tyto dva „ploché" soubory je tedy nutné importovat do systému SAS. Soubor DataNetA zahrnuje údaje za září 2001 – duben 2004 a obsahuje 7 437 883 řádků, soubor DataNetB (květen 2004 – květen 2005) představuje 3 633 640 záznamů.

V obou souborech je využito řetězení denních záznamů s každodenní aktualizací. To znamená, že transakční údaje (cca 8 400 položek za den) se přidávají každý den do souboru a za necelé čtyři roky sledování obsahují přes 10 miliónů položek. Každý e-mail však obsahuje údaje o odeslání, které představují cca 40 řádků. Ty jsou tedy vkládány mezi jednotlivé jednodenní zápisy. Struktura standardní věty záznamu je následující:

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXX; YYYY-MM-DD HH:MM:SS; NNNNNNNNNN; NNNNNNNNNN¶
```

Pro úplnost, znak X představuje alfanumerický znak včetně mezery, ostatní zástupné symboly jsou číslicové hodnoty (rok, měsíc, den, hodina, minuta, sekunda, číslo).



	Jmeno	Datum	Data_In	Data_Out
16	boundary="/9DWx/yDrRhgMJTb"		.	.
17	Content-Disposition: inline		.	.
18	User-Agent: Mutt/1.2.5i		.	.
19	Status: RO		.	.
20	X-Status: O		.	.
21			.	.
22			.	.
23	--/9DWx/yDrRhgMJTb		.	.
24	Content-Type: text/plain	charset=us-ascii	.	.
25	Content-Disposition: inline		.	.
26	Subject:		.	.
27			.	.
28			.	.
29	--/9DWx/yDrRhgMJTb		.	.
30	Content-Type: text/plain	charset=us-ascii	.	.
31	Content-Disposition: attachment	filename="data.txt"	.	.
32	Subject:		.	.
33			.	.
34	Novák Josef	2001-09-03 00:00:03	7476	3631
35	Novák Josef	2001-09-03 01:00:04	8700	5732
36	Novák Josef	2001-09-03 02:00:04	3420	3169
37	Novák Josef	2001-09-03 03:00:03	2569	2498
38	Novák Josef	2001-09-03 04:00:03	7065	3318
39	Novák Josef	2001-09-03 05:00:03	6873	3097

Obrázek 30: Vzhled importované e-mailové hlavičky a transakčních dat

V prvním kroku se uvedený soubor v textovém formátu importuje do systému SAS pomocí následujícího zápisu.

```
data SASUSER.DataNetA ;
  infile 'C:\Dokument\DataNet' delimiter = ';' MISSOVER DSD lrecl=32767 ;
      informat Jmeno $50. ;
      informat Datum $20. ;
      informat Data_In 10. ;
      informat Data_Out 10. ;
      format Jmeno $50. ;
      format Datum $20. ;
      format Data_In 10. ;
      format Data_Out 10. ;
  input
       Jmeno $
       Datum $
       Data_In
       Data_Out ;
run;
```

Provedený import dat je v datovém souboru ve formě čtyř proměnných. Obsah proměnné je dán oddělovačem, kterým je v tomto případě středník. Každá proměnná má svůj název a formát. Proměnné Jmeno a Datum jsou znakové, Data_In a Data_Out jsou numerické proměnné. Pro první proměnnou je vymezena délka 50 znaků, pro datum 20 znaků a pro poslední dvě po deseti cifrách.

V dalším kroku přípravy dat je potřeba spojit oba importované soubory s denními e-mailovými záznamy. K tomu lze využít proceduru APPEND nebo SET. Výhodou procedury APPEND je, že nezpracovává pozorování v původním datovém souboru. Pouze přidává pozorování v druhém datovém souboru přímo na konec toho původního. Příkaz SET lze využít i pro tvorbu nového resp. záložního datového souboru.

```
DATA SASUSER.DataNetC; /* Ze souboru DataNetA se vytvoří DataNetC */
   SET SASUSER.DataNetA;
RUN;

proc append base = SASUSER.DataNetC
            data = SASUSER.DataNetB force; /* Spojení DataNetC a DataNetB */
run;
```

Po spojení obou souborů byl získán soubor DataNetC s celkem 11 071 523 záznamy. Některé řádky nově vytvořeného datového souboru obsahují i nadbytečné hodnoty, konkrétně údaje z hlavičky e-mailu, které je nutné odstranit. Údaje v hlavičce byly odděleny maximálně jedním středníkem, čili zápis těchto informací proběhl pouze do prvních dvou proměnných. Pro tvorbu nového souboru SASUSER.DataNetD lze proto využít příkazu WHERE (alternativně i příkaz IF), jenž vybere ze souboru SASUSER.DataNetC jen ty řádky, které neobsahují v proměnné Data_In a Data_Out žádné hodnoty.

```
DATA SASUSER.DataNetD; /*Redukce záznamů, kde jsou Data_In a Out bez zaznamu*/
   SET SASUSER.DataNetC;
   WHERE (Data_In ^=.) AND (Data_Out^=.);
RUN;
```

Redukce nadbytečných informací v hlavičce e-mailu snížila počet řádků na 11 017 908 záznamů.

	Jmeno	Datum	Data_In	Data_Out
1	Novák Josef	2001-09-03 00:00:03	7476	3631
2	Novák Josef	2001-09-03 01:00:04	8700	5732
3	Novák Josef	2001-09-03 02:00:04	3420	3169
4	Novák Josef	2001-09-03 03:00:03	2569	2498
5	Novák Josef	2001-09-03 04:00:03	7065	3318
6	Novák Josef	2001-09-03 05:00:03	6873	3097
7	Novák Josef	2001-09-03 06:00:03	2908	2712
8	Novák Josef	2001-09-03 07:00:03	464637	36410
9	Novák Josef	2001-09-03 08:00:04	11329	4044
10	Novák Josef	2001-09-03 09:00:03	15481	11564
11	Novák Josef	2001-09-03 10:00:03	3272	2965
12	Novák Josef	2001-09-03 11:00:04	52871	7533

Obrázek 31: Struktura dat po odmazání e-mailové hlavičky

V proměnné Datum jsou zapsány údaje o datu a čase. Proto je vhodné tuto jednu znakovou proměnnou rozdělit na dvě. K tomu lze využít funkce SCAN (pro znakové proměnné), která vrátí n-té slovo znakového řetězce, jenž je tvořen slovy a oddělovači. Konkrétně je v programovém kódu zadáno vytvořit novou proměnnou Cas, která bude obsahovat druhé slovo, a oddělovačem slov bude mezera. Nová proměnná tedy obsahuje údaje o čase ve formátu HH:MM:SS.

```
DATA SASUSER.DataNetE;
   set SASUSER.DataNetD;
   Cas = SCAN(Datum, 2, ' '); /*SCAN vrátí druhé slovo znakového řetězce */
RUN;
```

	Jmeno	Datum	Data_In	Data_Out	Cas
1	Novák Josef	2001-09-03 00:00:03	7476	3631	00:00:03
2	Novák Josef	2001-09-03 01:00:04	8700	5732	01:00:04
3	Novák Josef	2001-09-03 02:00:04	3420	3169	02:00:04
4	Novák Josef	2001-09-03 03:00:03	2569	2498	03:00:03
5	Novák Josef	2001-09-03 04:00:03	7065	3318	04:00:03
6	Novák Josef	2001-09-03 05:00:03	6873	3097	05:00:03
7	Novák Josef	2001-09-03 06:00:03	2908	2712	06:00:03
8	Novák Josef	2001-09-03 07:00:03	464637	36410	07:00:03
9	Novák Josef	2001-09-03 08:00:04	11329	4044	08:00:04
10	Novák Josef	2001-09-03 09:00:03	15481	11564	09:00:03

Obrázek 32: Vytvoření nové proměnné „Cas"

Obdobně lze využít tento postup pro tvorbu proměnné osahující údaje jen o datu. Pro názornost je provedena obměna funkce SCAN za funkci SUBSTR, která vybere od určité pozice ze řetězce stanovený počet znaků. Konkrétně se vybírá slovo o délce deseti znaků (yyyy-mm-dd) počínaje první hodnotou v řetězci.

```
DATA SASUSER.DataNetF;
  set SASUSER.DataNetE;
  Datum2 = substr(Datum, 0, 10);
RUN;
```

	Jmeno	Datum	Data_In	Data_Out	Cas	Datum2
1	Novák Josef	2001-09-03 00:00:03	7476	3631	00:00:03	2001-09-03
2	Novák Josef	2001-09-03 01:00:04	8700	5732	01:00:04	2001-09-03
3	Novák Josef	2001-09-03 02:00:04	3420	3169	02:00:04	2001-09-03
4	Novák Josef	2001-09-03 03:00:03	2569	2498	03:00:03	2001-09-03
5	Novák Josef	2001-09-03 04:00:03	7065	3318	04:00:03	2001-09-03
6	Novák Josef	2001-09-03 05:00:03	6873	3097	05:00:03	2001-09-03
7	Novák Josef	2001-09-03 06:00:03	2908	2712	06:00:03	2001-09-03
8	Novák Josef	2001-09-03 07:00:03	464637	36410	07:00:03	2001-09-03
9	Novák Josef	2001-09-03 08:00:04	11329	4044	08:00:04	2001-09-03
10	Novák Josef	2001-09-03 09:00:03	15481	11564	09:00:03	2001-09-03

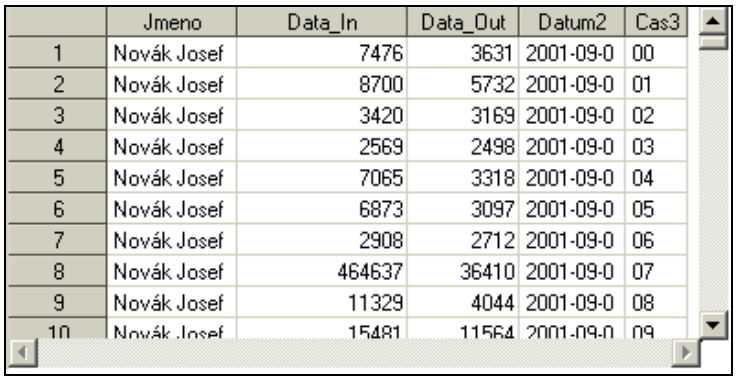
Obrázek 33: Vytvoření nové proměnné „Datum2"

Proměnná Cas může být však uváděna pouze v hodinách a minutách nebo jen v hodinách. Sekundy totiž neposkytují užitečnou informaci, jedná se o zpoždění monitorovacího systému. Proto byly vytvořeny nové datové soubory, které tyto úpravy za pomoci funkce znakových proměnných SUBSTR obsahují.

```
data SASUSER.DataNetG;
  set SASUSER.DataNetF (drop=_OBSTAT_ Datum Cas);   /* DROP odstraní proměnné */
  Cas3 = substr (Cas, 1, 2);   /* SUBSTR vytvoří proměnnou Cas3 ve formátu HH */
run;
```

Příkaz DROP odstraní nadbytečné proměnné. Opačným postupem může být výběr proměnných, které budou zařazeny do nově vytvářeného datového souboru. K tomu lze

využít příkaz KEEP (ve spojení s přehledem vybraných proměnných, které jsou od sebe odděleny mezerou, a vše je zadáno do kulatých závorek). Tato volba umožní zmenšit velikost datového souboru a urychlit tak práci s uvedenými daty.

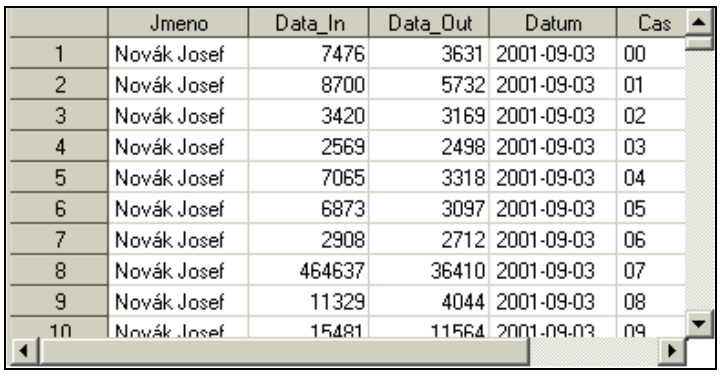


	Jmeno	Data_In	Data_Out	Datum2	Cas3
1	Novák Josef	7476	3631	2001-09-0	00
2	Novák Josef	8700	5732	2001-09-0	01
3	Novák Josef	3420	3169	2001-09-0	02
4	Novák Josef	2569	2498	2001-09-0	03
5	Novák Josef	7065	3318	2001-09-0	04
6	Novák Josef	6873	3097	2001-09-0	05
7	Novák Josef	2908	2712	2001-09-0	06
8	Novák Josef	464637	36410	2001-09-0	07
9	Novák Josef	11329	4044	2001-09-0	08
10	Novák Josef	15481	11564	2001-09-0	09

Obrázek 34: Vytvoření nové proměnné „Cas3" a redukce proměnných

Pro přejmenování názvů proměnných slouží příkaz RENAME. Nový datový soubor tedy stále obsahuje stejné datové proměnné, ale pod jinými názvy (Jmeno, Data_In, Data_Out, Datum, Cas).

```
data SASUSER.DataNetH;
   set SASUSER.DataNetG (rename=(Datum2=Datum) rename=(Cas3=Cas));
run;
```



	Jmeno	Data_In	Data_Out	Datum	Cas
1	Novák Josef	7476	3631	2001-09-03	00
2	Novák Josef	8700	5732	2001-09-03	01
3	Novák Josef	3420	3169	2001-09-03	02
4	Novák Josef	2569	2498	2001-09-03	03
5	Novák Josef	7065	3318	2001-09-03	04
6	Novák Josef	6873	3097	2001-09-03	05
7	Novák Josef	2908	2712	2001-09-03	06
8	Novák Josef	464637	36410	2001-09-03	07
9	Novák Josef	11329	4044	2001-09-03	08
10	Novák Josef	15481	11564	2001-09-03	09

Obrázek 35: Přejmenování proměnných

Vzhledem k tomu, že proměnná Datum je proměnná znaková, která obsahuje pro oddělení údajů pomlčky, je nutné z ní vhodným způsobem získat proměnnou číselnou. Jednou z možností je rozdělit znakové proměnné podle oddělovače (pomlčky) na tři nové proměnné (Rok, Mesic, Den) a z nich sestavit nové datové proměnné (Date1). Pro sloučení tří proměnných slouží dva vykřičníky !!. Pro očištění možných nadbytečných mezer slouží funkce TRIM.

```
data SASUSER.DataNetI ; /* Vytvoří ze znakového data tři proměnné */
   set SASUSER.DataNetH ;
   Rok= scan(Datum, 1,  '-');
   Mesic= scan(Datum, 2,  '-');
   Den= scan(Datum, 3,  '-');
   Date1= trim(Den)!! trim(Mesic)!! trim(Rok);   /* Spojí tři znakové proměnné */
   DROP Rok Mesic Den Datum;   /* DROP odstraní nadbytečné proměnné */
RUN;
```

	Jmeno	Data_In	Data_Out	Datum	Cas	Rok	Mesic	Den	Date1
1	Novák Josef	7476	3631	2001-09-03	00	2001	09	03	03092001
2	Novák Josef	8700	5732	2001-09-03	01	2001	09	03	03092001
3	Novák Josef	3420	3169	2001-09-03	02	2001	09	03	03092001
4	Novák Josef	2569	2498	2001-09-03	03	2001	09	03	03092001
5	Novák Josef	7065	3318	2001-09-03	04	2001	09	03	03092001
6	Novák Josef	6873	3097	2001-09-03	05	2001	09	03	03092001
7	Novák Josef	2908	2712	2001-09-03	06	2001	09	03	03092001
8	Novák Josef	464637	36410	2001-09-03	07	2001	09	03	03092001
9	Novák Josef	11329	4044	2001-09-03	08	2001	09	03	03092001
10	Novák Josef	15481	11564	2001-09-03	09	2001	09	03	03092001

Obrázek 36: Nová proměnná uvádějící datum – Date1

Ani proměnná Date1 není plně akceptovatelnou časovou proměnnou pro další zpracování. SAS má pro zachycení kalendářních údajů připravenu celou řadu formátů, proto je nutné transformovat kalendářní data na formát, který podporuje tzv. „kalendářní matematiku". Převést hodnoty proměnné Date1 do formátu SAS lze pomocí funkce MDY, která přijímá hodnoty ze závorek a přiřazuje je ke měsíci, dni a roku. Pro výběr hodnot se používá již použitý příkaz SUBSTR. Hodnoty měsíců začínají na třetí pozici a obsahují dva znaky, hodnoty dnů se vyskytují na první a druhé pozici, hodnoty roku okupují místa od páté pozice a obsahují čtyři znaky.

```
data sasuser.DataNetR;
   set sasuser.DataNetP;
   Datum2=mdy(substr(Date1,3,2), substr(Date1, 1,2), substr(Date1, 5,4));
   format Datum2 ddmmyy6.;
   Den = weekday(Datum2);
run;
```

Funkcí WEEKDAY lze z formátu systému SAS získat den v týdnu. Ten má podobu čísla, kdy jedna znamená neděli a sedm představuje sobotu.

V této formě již lze datový soubor využít pro průzkumovou analýzu i pokročilejší statistické metody.

5.1.1 První hodnocení kvality dat

Pro první hodnocení kvality dat poslouží základní statistické charakteristiky pro sumarizované záznamy. Tato sumarizace je prováděna podle jména zákazníků. Pro hodnocení bylo vybráno sedm základních charakteristik: minimum, maximum, součet, průměr, směrodatná odchylka, šikmost a špičatost.

```
proc means data=SASUSER.DATANETJ    maxdec=2    vardef=DF
MIN MAX SUM MEAN STD SKEWNESS KURTOSIS;
var Data_In Data_Out ;
class Jmeno ;
output out=Sasuser.NetJmeno
   MIN=MIN1-MIN2     /* index 1 – údaje na vstupu, index 2 – údaje na výstupu */
   MAX=MAX1-MAX2
   SUM=SUM1-SUM2
   MEAN=MEAN1-MEAN2
   STD=STD1-STD2
   SKEWNESS=SKEW1-SKEW2
   KURTOSIS=KURT1-KURT2 ;
run;
```

Při 11 017 846 záznamech jsou celkové vlastnosti záznamů následující:

Tabulka 7: Základní charakteristika neočištěných dat

Proměnná	maximum	součet	průměr	směrodatná odchylka	šikmost	špičatost
Data_In	105 900 094 754	64 727 781 305 395	5 874 812,7	129 865 239,3	273,7	160 371,0
Data_Out	76 857 244 447	35 777 369 278 985	3 247 219,9	89 957 933,7	343,3	220 151,6

Ze základních informací je zřejmé, že je evidováno na 955 potenciálních zákazníků. U některých z nich (23 zákazníků) nedošlo po dobu jejich sledování k žádnému přenosu dat. Ti z velké většiny představují zákazníky, jejichž připojení pod daným názvem bylo pouze krátkodobé. Jejich další zahrnutí do analýz není žádoucí. Proto byly ze sumarizačních záznamů zkušebně vyřazeni.

Nejdříve byl vygenerován soubor se záznamy 23 zákazníků, který posloužil k vizuální kontrole správnosti výsledku.

```
/*Vytvoří soubor se zákazníky, již nemají žádná odeslaná a zároveň přijatá data*/
data Sasuser.NetJmeno2;
   set Sasuser.NetJmeno;
   where (max1 LT 1 AND max2 LT 1)
   AND (min1 LT 1 AND min2 LT 1)
   AND (sum1 EQ 0 AND sum2 EQ 0);
   /* LT znamená less than (méně než), EQ znamená equal to (rovno)*/
run;
```

Seznam neaktivních zákazníků byl použit při psaní kódu pro jejich vyřazení. Nejprve byly neaktivní zákazníci zkušebně vyřazeni ze sumarizačního souboru (zůstalo 932 zákazníků), poté z úplného (zůstalo 10 989 310 pozorování).

```
/* Odstraní ty zakazníky, kteří nemají žádný přenos dat */
data Sasuser.DataNetk;
   set Sasuser.DataNetj;
   if Jmeno='Archiv__01'
   OR Jmeno='Archiv__02'
   …
   OR Jmeno='Archiv__23' then delete ;
run;
```

Z důvodu zachování anonymity firemních údajů je jedním z důležitých momentů i převedení jmen zákazníků na identifikační čísla. Tato činnost byla částečně prováděna „ručně", za účasti zástupce firmy. Část z potenciálních zákazníků totiž představují servisní přenosy.

Jedná se o záznamy činností routerů a serverů pro jednotlivé lokality sítě, dále o činnosti server hostingů, kabelových televizí a v neposlední řadě i vnitrofiremní servery. Všechny tyto záznamy by neadekvátně zasahovaly do struktury vedených informací pro nadměrný objem dat na vstupu či na výstupu. Dalším doprovodným jevem jejich existence je faktická nesouměřitelnost, cílem tedy je odstranit provozní jednotky z evidence v analyzovaném souboru. V případě, že by k tomu nedošlo, by byly údaje velmi nekonzistentní, což by narušovalo kvalitu výstupů statistických analýz.

Procedura byla nejprve spuštěna na sumarizačním souboru a poté na úplném. V prvním případě bylo smazáno 43 záznamů - potenciálních zákazníků, zůstalo 889 záznamů o klientech. V druhém souboru bylo odstraněno 492 506 záznamů (z 10 989 310 pozorování), čili zůstalo 10 496 804 záznamů o přenosu dat.

```
data sasuser.NetJmeno4;   /*přiřazení ID v sumarizačním souboru */
   set sasuser.NetJmeno3 ;
   ID=0; /*Vytvori novou promennou ID, ktera obsahuje misto jmena cislo*/
   if  Jmeno='Novák Josef' then  ID= 1;
   if  Jmeno='Novotný Karel' then  ID= 2;
   …
   if  Jmeno='Terminal_server' then  ID= . ;
   …
   if  Jmeno='Zatloukal Petr' then  ID= 889;
run;

data sasuser.DataNetL;   /* přiřazení ID v úplném souboru */
   set sasuser.DataNetK ;
   ID=0;   /*Vytvoří novou proměnnou ID, která obsahuje místo jména číslo*/
   if  Jmeno='Novák Josef' then  ID= 1;
   if  Jmeno='Novotný Karel' then  ID= 2;
   …
   if  Jmeno='Terminal_server' then delete; /* vymazání záznamu */
   …
   if  Jmeno='Zatloukal Petr' then ID= 889;
   drop Jmeno;
run;
```

Druhým důvodem pro „ruční" přidělování ID byla duplicita evidence zákazníků, resp. jejich připojení. Tato duplicita v záznamech byla zapříčiněna zejména díky administrátorským a administrativním změnám. Ty představují především odchylky způsobené zkušebním provozem připojení, stěhováním firmy, změnou připojení, přejmenováním firmy, atd. Tato redundance v datech tedy vedla k dalším úpravám datového souboru. V některých případech bylo nutné zohlednit existenci více poboček jedné firmy, jejichž provoz internetové sítě je nezávislý (jiná lokalita, jiné připojení, jiná konkurence poskytovatelů internetu), a není vhodné hodnotit firmu jako celek.

Pro sumarizaci redundantních údajů byl vytvořen z aktuálního úplného datového souboru pomocný soubor Redundant, který obsahoval 1 125 204 záznamů.

```
data sasuser.Redundant;
   set sasuser.DataNetL;
   where (ID= 39 or ID= 55 or ID= 58 or ID= 85 or ID= 912 or ID= 89 or ID= 105
   or ID= 118 or ID= 121 or ID= 165 or ID= 193 or ID= 213 or ID= 236 or ID= 244
   or ID= 253 or ID= 254 or ID= 256 or ID= 274 or ID= 277 or ID= 309 or ID= 352
   or ID= 395 or ID= 434 or ID= 450 or ID= 464 or ID= 468 or ID= 487 or ID= 488
   or ID= 491 or ID= 492 or ID= 539 or ID= 547 or ID= 548 or ID= 551 or ID= 618
   or ID= 623 or ID= 626 or ID= 634 or ID= 636 or ID= 660 or ID= 665 or ID= 679
   or ID= 718 or ID= 730 or ID= 740 or ID= 743 or ID= 745 or ID= 751);
run;
```

Pro další využití se všechny redundantní záznamy musejí sumarizovat podle ID, času a data. Tomu však předchází nutnost setřídění údajů (podle ID, data a času).

```
proc sort data=sasuser.redundant out=sasuser.redundant2;
   by ID Date Cas;
run;

/* Výpočet součtu Data_In a Data_Out za stejné ID Cas a Date*/
PROC SUMMARY  DATA=sasuser.redundant2  ;
 CLASS ID Date Cas ;
 VAR Data_In Data_Out;
 OUTPUT out=sasuser.redsum
 SUM=Data_In Data_Out;
run;

proc contents data=sasuser.redsum;   /* Vypíše základní info o souboru redsum*/
run;
```

Na základě výše uvedeného kódu procedury byly vytvořeny součty všech kombinací obměn tří proměnných (tzn. ID, Date a Cas), celkem tedy osm možných kombinací (nic; Cas; Date; Cas a Date; ID; ID a Cas; ID a Date; ID, Cas a Date). Z nich pro další účely poslouží jen posledně uvedená kombinace - současný výskyt obměn proměnné ID, času a data. Tato kombinace představuje 845 785 záznamů, sníží tedy celkový počet záznamů o 279 419 a přispěje k posílení principu jednoznačně vedených zákazníků.

```
DATA sasuser.redsum2;
   set sasuser.redsum;
   where _TYPE_=7; /* vybere současný výskyt obměn proměnné ID, času a data*/
run;

DATA sasuser.redsum3;
   set sasuser.redsum2 (drop = _TYPE_ _FREQ_); /* odstraní nadbytečné proměnné*/
run;
```

Dále je potřebné odstranit z úplného souboru ty záznamy, jejichž ID je redundantní, k tomu se použije příkaz DELETE a vznikne soubor s 9 371 600 záznamy. K tomuto souboru se následně přidá, pomocí procedury APPEND, soubor sumarizovaných záznamů – vznikne datový soubor s 10 217 385 záznamy.

```
data sasuser.DataNetM;
   set sasuser.DataNetL;
   if (ID= 39 or ID= 55 or ID= 58 or ID= 85 or ID= 912 or ID= 89 or ID= 105
   or ID= 118 or ID= 121 or ID= 165 or ID= 193 or ID= 213 or ID= 236 or ID= 244
   or ID= 253 or ID= 254 or ID= 256 or ID= 274 or ID= 277 or ID= 309 or ID= 352
   or ID= 395 or ID= 434 or ID= 450 or ID= 464 or ID= 468 or ID= 487 or ID= 488
   or ID= 491 or ID= 492 or ID= 539 or ID= 547 or ID= 548 or ID= 551 or ID= 618
   or ID= 623 or ID= 626 or ID= 634 or ID= 636 or ID= 660 or ID= 665 or ID= 679
   or ID= 718 or ID= 730 or ID= 740 or ID= 743 or ID= 745 or ID= 751)then delete;
run;

proc append base = SASUSER.DataNetN
            data = SASUSER.Redsum3 force;
run;
```

V této formě již lze datový soubor využít pro hodnocení statistických charakteristik a popř. pro užití pokročilejších statistických metod.

5.1.2 Druhé hodnocení kvality dat

Pro další práci je již k dispozici soubor, který obsahuje tři třídící proměnné: Identifikace – 795 obměn (795 zákazníků), Čas – 24 obměn (24 hodin) a Datum – 1200 obměn (1200 dní).

5.1.2.1 Základní charakteristiky pro numerické proměnné

Z výpisu procedury UNIVARIATE pocházejí následující základní charakteristiky pro numerické proměnné.

```
proc univariate data=sasuser.datanetp plot;
   var Data_In Data_Out;
   run;
```

The UNIVARIATE Procedure
Variable: Data_In

Basic Statistical Measures

Location				Variability	
N	10217385	Mean	2689997	Std Deviation	38371374
Sum	2.74847E13	Median	1260	Variance	1.47236E15
Skewness	230.320778	Mode	0	Coeff Variation	1426.44693
Kurtosis	116305.048				

Tests for Location: Mu0=0

Test	-Statistic-		-----p Value------	
Student's t	t	224.0858	Pr > |t|	<.0001
Sign	M	3193397	Pr >= |M|	<.0001
Signed Rank	S	1.02E13	Pr >= |S|	<.0001

Quantiles

Quantile	Estimate
100% Max	29492815454
75% Q3	34464
50% Median	1260
25% Q1	0
0% Min	0

Extreme Observations

----Lowest----		------Highest-------	
Value	Obs	Value	Obs
0	1.02E7	21117525931	5.09E6
0	1.02E7	22152854280	5.52E6
0	1.02E7	22428768591	5.09E6
0	1.02E7	23160858456	5.94E6
0	1.02E7	29492815454	5.07E6

```
                    Histogram                          #     Boxplot
 2.9E10+*                                              1           *
       .
       .
       .*                                             3           *
       .*                                             2           *
       .*                                             1           *
       .*                                             2           *
 1.5E10+*                                             3           *
       .*                                             3           *
       .*                                             3           *
       .*                                             6           *
       .*                                            11           *
       .*                                             9           *
       .*                                            68           *
    1E9+************************************************1.02E7        +--0--+
       ----+----+----+----+----+----+----+----+----+---
          * may represent up to 212860 counts
```

Údaje o odesílání či přijímání dat z internetové sítě činí 10 217 385 záznamů. V průměru se na vstupu odebere 2,7 miliónu Bytů a na výstupu odešle 1,2 miliónu Bytů. Hodnota mediánu je však mnohem nižší, 1260 B se odebere na vstupu a 0 B se odešle na výstupu. Z toho je zřejmá velká asymetrie (230 B a 310 B) a špičatost (114 kB a 167 KB), jež je zapříčiněna příliš vysokým výskytem nulových hodnot (především nečinností v průběhu noci). Variační koeficient tento fakt taktéž potvrzuje, jeho hodnoty se vyšplhaly až na úroveň 1426 % a 2515 %.

Částečnou sumarizaci informací pro posouzení polohy, asymetrie a identifikaci odllehlých pozorování umožňuje i krabicový graf. Ten napovídá, že 112 záznamů o přijatých datech a 66 záznamů o odeslaných datech představuje abnormálně vysoké hodnoty. Testy polohy (jednovýběrový t-test i oba neparametrické testy) zamítají nulovou hypotézu jak o nulovém průměru přijímaných, tak odesílaných dat.

The UNIVARIATE Procedure
Variable: Data_Out

Basic Statistical Measures

Location		Variability			
N	10217385	Mean	1201814	Std Deviation	30228251
Sum	1.22794E13	Median	0	Variance	9.13747E14
Skewness	310.43493	Mode	0	Coeff Variation	2515.2192
Kurtosis	170641.483				

Tests for Location: Mu0=0

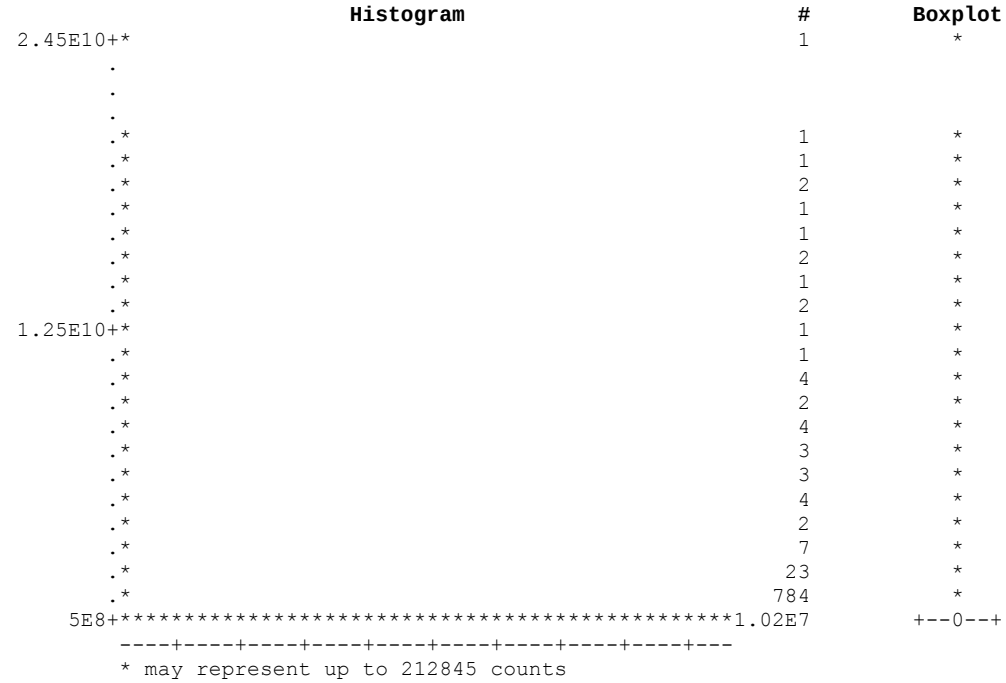
Test	-Statistic-		-----p Value------	
Student's t	t	127.0849	Pr > |t|	<.0001
Sign	M	2462343	Pr >= |M|	<.0001
Signed Rank	S	6.063E12	Pr >= |S|	<.0001

Quantiles

Quantile	Estimate
100% Max	24168811360
75% Q3	21649
50% Median	0
25% Q1	0
0% Min	0

Extreme Observations

----Lowest----		-------Highest-------	
Value	Obs	Value	Obs
0	1.02E7	18109802470	5.09E6
0	1.02E7	18907828701	5.52E6
0	1.02E7	19675109877	5.94E6
0	1.02E7	20706950702	5.09E6
0	1.02E7	24168811360	5.07E6



Součet přijatých dat představuje 25 TB dat a odeslaných 11 TB. Zákazníci jsou tedy více než dvojnásobně aktivnější při stahování dat z internetu. Maximálně bylo za jednu hodinu přijato 27,5 GB a odesláno 22,5 GB.

Některé ze záznamů zaznamenávají i neaktivitu zákazníka v určitý den a hodinu. Tyto záznamy lze odstranit a získat pouze informace o aktivních transakcích.

```
DATA sasuser.DataNetO;
   set sasuser.DataNetN;
   if (Data_In=0 AND Data_Out=0) then delete;
   /* Vymazání záznamů, kde není žádný pohyb dat v obou směrech */
run;
```

Vyřazení nulových hodnot představí chování zákazníků v trochu jiném zorném úhlu. Pokud je zákazník aktivní, pak v průměru odebere za jednu hodinu 4,3 miliónu B a odešle 1,9 miliónu B. Z dalších základních charakteristik je zajímavý medián, ten se u přijatých dat zvýšil na 6249 B a u odeslaných na 4864 B. Variabilita se u aktivních transakcí podle předpokladu snížila (1130 % a 1994 %), podobně i šikmost (183 B a 247 B) a špičatost (72 kB a 105 kB).

5.1.2.2 Základní charakteristiky setříděné dle vybrané kategorické proměnné

Pro detailnější sumarizační přehledy dle vybrané kategorické proměnné je nutné soubor nejprve setřídit dle dané třídící proměnné. V prvním případě je soubor setříděn podle času (24 obměn), v druhém případě je setříden podle dne v týdnu.

Charakteristiky pro kategorie času

Procedura pro setřídění záznamů podle hodnot kategorické proměnné je následující.

```
proc sort data=sasuser.datanetp ;
   by Cas; /* Třídění souboru podle hodnot kategorické proměnné*/
run;
```

Přehledné tabulkové výstupy lze získat díky proceduře TABULATE. Požadované statistické charakteristiky musejí být do programu zadány.

```
proc tabulate data=sasuser.datanetp ;
   class Cas;
   var Data_In;
   table Cas, Data_In *(N SUM MEAN VAR MIN MAX);
run;
```

Tabulka 8: Základní statistiky pro Data_In podle času

Cas	N	Sum	Mean	Var	Min	Max
0	424 959	1,06E+12	2504316,83	1,34E+15	0	11 455 912 543
1	424 441	8,4E+11	1985013,26	1,26E+15	0	12 320 801 611
2	421 451	5,1E+11	1200961,14	8,28E+14	0	13 000 094 452
3	425 398	7,0E+11	1634168,96	1,19E+15	0	13 672 280 540
4	426 839	6,4E+11	1507830,01	1,19E+15	0	14 413 819 620
5	426 302	5,7E+11	1343943,93	1,15E+15	0	14 955 410 318
6	425 680	5,7E+11	1331841,23	1,17E+15	0	15 761 133 663
7	426 550	6,9E+11	1617474,50	1,29E+15	0	16 724 673 730
8	424 981	10,0E+11	2346713,46	1,45E+15	0	17 866 423 699
9	427 036	1,28E+12	3007504,94	1,83E+15	0	19 110 333 176
10	427 013	1,40E+12	3268607,13	1,84E+15	0	20 087 178 311
11	424 117	1,63E+12	3831578,34	5,50E+15	0	29 492 815 454
12	426 177	1,62E+12	3806248,82	2,32E+15	0	22 152 854 280
13	426 351	1,42E+14	3806331,98	2,46E+15	0	23 160 858 456
14	427 398	1,58E+12	3705005,48	8,38E+14	0	3 836 403 174
15	426 853	1,54E+12	3601508,98	7,63E+14	0	2 697 049 367
16	424 329	1,41E+12	3331846,93	7,65E+14	0	1 869 788 322
17	425 392	1,33E+12	3135174,43	8,54E+14	0	1 951 763 641
18	424 723	1,29E+12	3027485,95	1,14E+15	0	6 500 321 620
19	426 186	1,24E+12	2914675,46	1,05E+15	0	4 958 833 758
20	426 838	1,24E+12	2906058,08	1,15E+15	0	6 260 460 855
21	426 828	1,26E+12	2949266,06	1,23E+15	0	7 734 995 380
22	425 935	1,26E+12	2961311,91	1,34E+15	0	9 175 372 401
23	425 608	1,20E+12	2817104,11	1,38E+15	0	10 374 466 022

Za každou hodinu existuje v průměru 425 724 záznamů. V době od deseti do jedenácti hodin (kategorie 11) bylo dosaženo jak nejvyššího průměru (3,7 MB), tak maximálního množství přijatých dat (27,5 GB) u jednoho zákazníka. Z hlediska průměru i dosaženého maxima se na druhém místě umístila doba od dvanácti do jedné hodiny (kategorie 13) a na třetím od jedenácti do dvanácti (kategorie 12). V době kolem poledne je tedy aktivita ve stahování dat z internetové sítě nejvyšší.

Minimální průměrná hodnota stažených dat je od jedné do druhé hodiny ranní (následuje pátá až šestá, potom čtvrtá až pátá). Nejnižší maximální zaznamenaná hodnota se objevuje v intervalu od tří do čtyř hodin odpoledne.

Podle ukazatele variability, variačního koeficientu (VAR), lze usoudit na největší proměnlivost stažených dat v době od deseti do jedenácti hodin dopoledne (11). Nejnižší variabilita v množství přijatých dat se projevuje od dvou do tří hodin odpoledne (15).

```
proc tabulate data=sasuser.DataNetR ;
   class Den;
   var Data_In;
   table Cas, Data_In *(N SUM MEAN VAR MIN MAX);
run;
```

Tabulka 9: Základní statistiky pro Data_Out podle času

Cas	N	Sum	Mean	Var	Min	Max
0	424959	5,14732E+11	1211251,22	6,85E+14	0	9 490 071 439
1	424441	4,51590E+11	1063963,96	7,20E+14	0	10 583 271 693
2	421451	2,68121E+11	636184,73	5,28E+14	0	11 439 781 241
3	425398	4,37232E+11	1027818,34	8,46E+14	0	12 336 870 277
4	426839	4,25959E+11	997938,12	8,88E+14	0	13 007 323 409
5	426302	4,10235E+11	962310,67	9,27E+14	0	13 708 565 531
6	425680	4,08459E+11	959544,01	9,82E+14	0	14 462 630 558
7	426550	4,21757E+11	988763,68	1,04E+15	0	15 231 268 491
8	424981	4,72810E+11	1112544,54	1,10E+15	0	15 951 399 389
9	427036	5,24129E+11	1227365,47	1,18E+15	0	16 628 640 529
10	427013	5,43863E+11	1273646,23	1,24E+15	0	17 289 659 567
11	424117	6,50229E+11	1533135,39	3,86E+15	0	24 168 811 360
12	426177	6,13221E+11	1438889,19	1,51E+15	0	18 907 828 701
13	426351	6,12856E+11	1437443,81	1,59E+15	0	19 675 109 877
14	427398	5,94839E+11	1391768,88	5,56E+14	0	8 727 429 677
15	426853	5,71475E+11	1338809,11	3,43E+14	0	1 761 589 263
16	424329	5,40368E+11	1273463,67	3,34E+14	0	1 751 948 801
17	425392	5,29838E+11	1245529,24	3,65E+14	0	1 950 317 712
18	424723	5,37302E+11	1265064,19	4,59E+14	0	3 508 952 852
19	426186	5,38825E+11	1264295,07	4,83E+14	0	5 095 007 549
20	426838	5,47626E+11	1282983,45	5,06E+14	0	5 946 186 445
21	426828	5,59733E+11	1311377,36	5,53E+14	0	6 738 345 191
22	425935	5,62666E+11	1321013,57	6,10E+14	0	7 509 502 783
23	425608	5,41530E+11	1272368,04	6,42E+14	0	8 705 086 233

Největší objem odeslaných dat do internetové sítě byl za sledované období přenesen v poledních hodinách, tj. od deseti do třinácti hodin (kategorie 11, 12 a 13). Nejnižší objemy dat byly odeslány v noci od jedné do druhé hodiny. Nejvyšší maximální přenos u jednoho zákazníka lze vysledovat od deseti do jedenácti hodin dopoledne, ve stejnou denní dobu je vykázán i nejvyšší průměrný odeslaný objem.

Na základě variačního koeficientu lze usoudit na největší proměnlivost odeslaných dat v době od šesti do třinácti hodin. Nejnižší variabilita v množství odeslaných dat se projevuje od dvou do pěti hodin odpoledne (kategorie 15, 16 a 17).

Charakteristiky pro kategorie dne v týdnu

Pro tvorbu základních charakteristik za jednotlivé kategorie dne v týdnu musí dojít k setřídění údajů (číslo 1 označuje neděli, číslo sedm sobotu).

```
proc sort data=sasuser.DataNetR ;
   by Den;
run;

proc tabulate data=sasuser.DataNetR ;
   class Den;
   var Data_In Data_Out;
   table Den, Data_In *(N SUM MEAN VAR MIN MAX);
   table Den, Data_Out *(N SUM MEAN VAR MIN MAX);
run;
```

Tabulka 10: Základní statistiky pro Data_In podle dne v týdnu

Den	Den	N	Sum	Mean	Var	Min	Max
1	Neděle	1 435 416	3,01E+12	2 096 785	8,34E+14	0	4 970 958 755
2	Pondělí	1 439 287	4,00E+12	2 781 571	7,88E+14	0	6 714 007 825
3	Úterý	1 459 628	4,17E+12	2 856 109	9,82E+14	0	10 374 466 022
4	Středa	1 480 915	4,91E+12	3 315 582	4,42E+15	0	23 160 858 456
5	Čtvrtek	1 506 616	4,27E+12	2 833 795	7,12E+14	0	2 955 398 638
6	Pátek	1 461 191	3,98E+12	2 723 626	7,54E+14	0	6 500 321 620
7	Sobota	1 434 332	3,14E+12	2 191 515	1,78E+15	0	29 492 815 454

Nejvyšší počet záznamů přijatých dat se vyskytl ve čtvrtek, nejnižší v sobotu. Největší objem dat směrem k zákazníkovi byl zaznamenán ve středu, nejnižší v neděli. Středa patří k průměrně nejaktivnějším dnům, co se stahování dat týká. Nejnižší průměrný počet stažených dat lze vypozorovat v dnech pracovního klidu. Nejvyšší variabilita v objemu přijatých dat je ve středu. Největší přenesený objem dat směrem k zákazníkovi je 29 492 815 454 B a vyskytl se v sobotu.

Tabulka 11: Základní statistiky pro Data_Out podle dne v týdnu

Den	Den	N	Sum	Mean	Var	Min	Max
1	Neděle	1 435 416	1,56E+12	1 089 198	4,57E+14	0	3 902 937 938
2	Pondělí	1 439 287	1,72E+12	1 194 360	4,17E+14	0	3 342 440 043
3	Úterý	1 459 628	1,77E+12	1 215 479	5,88E+14	0	8 727 429 677
4	Středa	1 480 915	2,21E+12	1 495 084	3,19E+15	0	19 675 109 877
5	Čtvrtek	1 506 616	1,79E+12	1 186 342	3,29E+14	0	2 037 192 124
6	Pátek	1 461 191	1,66E+12	1 137 052	3,04E+14	0	3 324 911 884
7	Sobota	1 434 332	1,56E+12	1 087 521	1,09E+15	0	24 168 811 360

Počet záznamů s údaji o odeslaných datech je totožný v obou tabulkách pro obě proměnné. Největší objem odeslaných dat patří středě, nejnižší pak sobotě a neděli. Průměrné množství dat dosahuje nejvyšší hodnoty ve středu, nejnižší opět o víkendu. Největší rozdíly v odeslaných datech lze zaznamenat ve středu, nejnižší pak v pátek. Nejvyšší objem přenesených dat do internetové sítě je od jednoho zákazníka zaznamenán v sobotu.

5.2 Shlukování pomocí programování v systému SAS

Nehierarchická shluková analýza v rámci systému SAS je zastoupena shlukovací metodou, jež je nazývána K-means model. Její použití lze aplikovat v rámci programování, k čemuž se využívá procedura FASTCLUS. Počet shluků *k* je v tomto případě specifikován předem.

V následující části textu bude uvedeno řešení, které se doporučuje pro dlouhodobá (longitudinální) data. Tzn. shrnout údaje o jednom zákazníkovi do jednoho záznamu. Ten může mít podobu součtu nebo i jiných charakteristik. Nejprve je zvolen průměr, později součet. První program je sestaven pro segmentování zákazníků na základě průměrného objemu přijatých a odeslaných dat ze všech záznamů, druhý na základě nenulových záznamů. Pokaždé je kód programu pozměněn v části stanovení počtu shluků pro porovnání výsledků shlukování pro různý počet skupin. Shlukování je možné iterovat a to maximálně 100 krát. Všechny uvedené způsoby segmentace jsou ilustrativní a poukazují na změny, které mohou nastat se změnou počtu skupin. Výsledky jsou pro praxi použitelné k identifikaci odlehlých pozorování, v tomto případě nadprůměrných až výjimečných zákazníků.

5.2.1 Shrnutí údajů o jednom zákazníkovi

Ve fázi přípravy dat byly z celé databáze údajů spočítány základní charakteristiky o 795 zákaznících pomocí procedury MEANS. Průměrný údaj za každého zákazníka pro obě zvolené proměnné nese označení Mean1 pro průměrně přijaté množství dat a Mean2 pro průměrně odeslané množství dat. Celkový objem přenesených dat je označet SUM1 pro přijatá data a SUM2 pro odeslaná data.

Kód pro výpočet základních charakteristik o objemu přijatých a odeslaných dat ze všech záznamů setříděných podle ID zákazníka je následující:

```
proc means data=SASUSER.DATANETN noprint maxdec=2 vardef=DF
   N MIN MAX SUM MEAN STD;
   var Data_In Data_Out ;
   class ID ;
   output out=Sasuser.NetJmeno5
   N=N1-N2   MIN=MIN1-MIN2   MAX=MAX1-MAX2
   SUM=SUM1-SUM2   MEAN=MEAN1-MEAN2   STD=STD1-STD2
   /* index 1 - údaje na vstupu, index 2 - údaje na výstupu */
run;
```

Po odmazání záznamů s nulovou hodnotou přijatých a zároveň odeslaných dat lze získat odlišné charakteristiky polohy. Kód pro výpočet základních charakteristik sumarizovaných nenulových záznamů podle ID je podobný kódu předcházejícímu:

```
proc means data=SASUSER.DataNetO noprint maxdec=2 vardef=DF
   N MIN MAX SUM MEAN STD;
   var Data_In Data_Out ;
   class ID ;
   output out=Sasuser.NetJmeno5
   N=N1-N2   MIN=MIN1-MIN2   MAX=MAX1-MAX2
   SUM=SUM1-SUM2   MEAN=MEAN1-MEAN2   STD=STD1-STD2 ;
   /* index 1 - údaje na vstupu, index 2 - údaje na výstupu */
run;
```

5.2.2 Shluková analýza průměrných údajů za zákazníky

Před vlastním shlukováním jsou pomocí procedury UNIVARIATE zjištěny základní charakteristiky proměnných MEAN1 a MEAN2 v obou souborech (soubor s úplnými a soubor s nenulovými záznamy). Při hodnocení extrémních hodnot je nalezeno pozorování, které pro obě dvě proměnné v obou dvou souborech vykazuje několikanásobně převyšující hodnoty druhého pozorování v pořadí extrémních hodnot. Jedná se o zákazníka s ID 491, který patří k providerům dalších služeb a dlouhodobě přijímá a odesílá velké množství dat. Tento zákazník by s jistotou vytvořil samostatný shluk. Toto odlehlé pozorování je tedy z obou souborů odstraněno.

```
proc univariate data=meanclus.datamean_A;
   var MEAN1 MEAN2;
   title 'Zakladni charakteristika souboru bez nulových dat';
run;
data meanclus.datamean_A_FO; /*filter outliers*/
   set meanclus.datamean_A;
   if (mean1 GT 800000000) then delete;
run;
proc univariate data=meanclus.datamean_A_FO;
   var MEAN1 MEAN2;
   title 'Zakladni charakteristika souboru bez nulových dat po odstranění odl. poz.';
run;
```

Protože algoritmus pro učení shluků vyžaduje, aby měřítka proměnných byla podobná, je nutné zvážit, zda ještě před spuštěním analýzy shluků není nutné provést standardizaci proměnných (pomocí procedury STANDARD). Jelikož obě proměnné udávají množství ve stejných jednotkách, není nutné tuto úpravu dat provádět. V opačném případě by proměnná s největším měřítkem celému procesu dominovala.

```
proc standard data=meanclus.datamean_A_FO /*s nenulovými záznamy*/
              out=meanclus.stand_A mean=0 std=1;
   var Mean1 Mean2;
   /* Mean1 je průměrný počet přijatých dat a Mean2 je průměrný počet odeslaných dat*/
run;

proc standard data=meanclus.datamean_B_FO  /*s úplnými záznamy*/
              out=meanclus.stand_B mean=0 std=1;
    var Mean1 Mean2;
   /* Mean1 je průměrný počet přijatých dat a Mean2 je průměrný počet odeslaných dat*/;
run;
```

Pro shlukování průměrů úplných záznamů byla použita procedura FASTCLUS. Program pro určení shluků je poměrně jednoduchý. Shlukování lze provádět jednorázově nebo s přepočítáním. Shlukování s iterací lze nastavit příkazem MAXITER. Počet shluků se nastavuje pomocí MAXCLUS.

Následující kód programu je příkladem pro shlukování do dvou a do tří shluků. Shlukování bylo prováděno opakovaně pro počet shluků v rozmezí od dvou do patnácti. V příloze jsou uvedeny výsledky shlukování, resp. jejich část, pro oba soubory (soubor průměrů úplných záznamů a soubor s průměry nenulových záznamů).

```
proc fastclus data= meanclus.stand_B maxclusters=2 maxiter=100
              out=meanclus.B02_s; /* s = se standardizací*/
   var Mean1 Mean2;
   title '2 shluky - Shlukování průměrů z úplných dat a se standardizací';
run;
proc plot;
   plot Mean1*Mean2 = cluster;
run;
```

```
proc fastclus data= meanclus.stand_B maxclusters=3 maxiter=100
              out=meanclus.B03_s; /* s = se standardizací*/
   var Mean1 Mean2;
   title '3 shluky - Shlukování průměrů z úplných dat a se standardizací';
run;
proc print data= meanclus.B03_s ; /* ve výstupu zobrazí ID a číslo shluku*/
   var ID cluster;
   where (cluster=2); run;
proc plot;
   plot Mean1*Mean2 = cluster;
run;
```

5.2.2.1 Porovnání počtu iterací v závislosti na počtu shluků

Z níže uvedené tabulky vyplývá, že počet iterací je kolísavý, nezáleží na zvyšování počtu shluků, ale na výběru středů.

Tabulka 12: Porovnání počtu iterací při shlukování průměrů úplných záznamů

Počet shluků	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Počet iterací	4	2	3	3	3	3	8	5	5	8	6	5	9	9

Tabulka 13: Porovnání počtu iterací při shlukování průměrů nenulových záznamů

Počet shluků	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Počet iterací	4	8	4	5	5	5	4	11	6	6	12	10	6	6

5.2.2.2 Porovnání hodnot konvergenčního kritéria v závislosti na počtu shluků

Na základě níže uvedených hodnot konvergenčního kritéria lze konstatovat, že s rostoucím počtem shluků jeho hodnota klesá. V případě shlukování průměrů úplných záznamů byla výsledná hodnota pro daný shluk pokaždé nižší než při shlukování průměrů nenulových záznamů.

Tabulka 14: Hodnoty konvergenčního kritéria při shlukování průměrů úplných záznamů

Počet shluků	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Krit.	0,507	0,388	0,241	0,229	0,217	0,206	0,172	0,154	0,149	0,112	0,110	0,102	0,090	0,085

Tabulka 15: Hodnoty konvergenčního kritéria při shlukování průměrů nenulových záznamů

Počet shluků	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Krit.	0,575	0,425	0,285	0,254	0,244	0,235	0,228	0,192	0,183	0,180	0,147	0,137	0,125	0,120

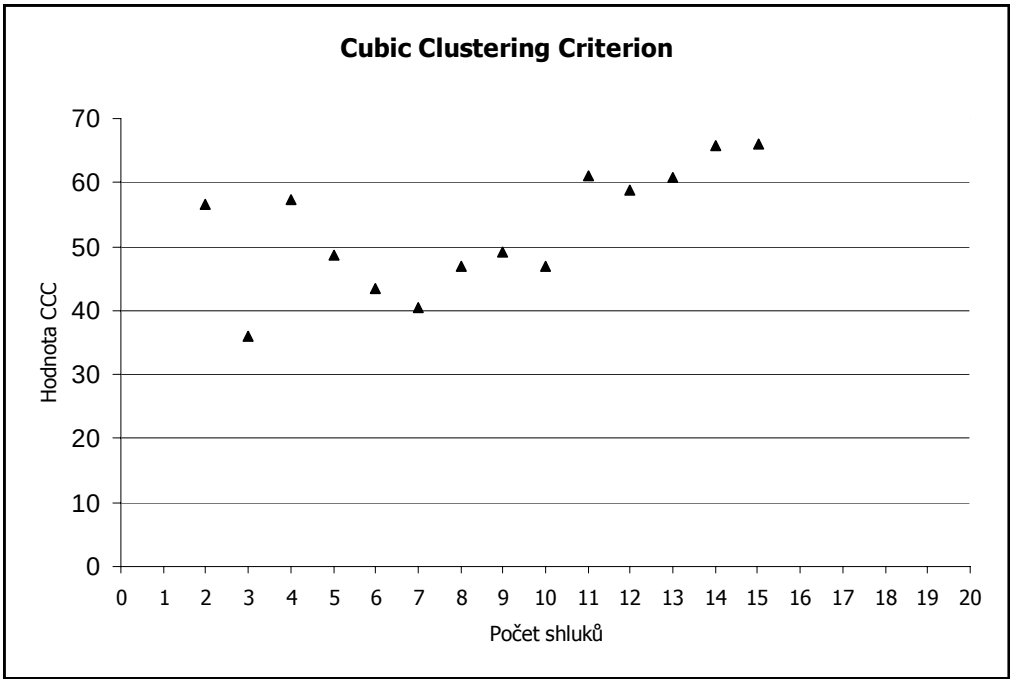
5.2.2.3 Porovnání hodnot kritérií pro stanovení počtu shluků

Procedura FASTCLUS automaticky počítá pro stanovení optimálního počtu shluků následující tři kritéria.

Tabulka 16: Hodnoty kritérií při shlukování průměrů z úplných dat a se standardizací

Počet shluků	Pseudo F Statistic	Approximate Expected Over-All R-Squared	Cubic Clustering Criterion
2	2276,84	0,37659	56,621
3	2221,74	0,66824	35,958
4	4247,52	0,75190	57,348
5	3562,77	0,80208	48,718
6	3173,17	0,83553	43,536
7	2956,03	0,85942	40,429
8	3684,21	0,87733	46,868
9	4039,40	0,89127	49,226
10	3803,35	0,90241	46,889
11	6105,12	0,91153	61,143
12	5743,38	0,91912	58,838
13	6220,80	0,92555	60,921
14	7370,17	0,93106	65,745
15	7538,76	0,93583	66,119

Graf 1: Vztah kritéria CCC a počtu shluků u úplného souboru



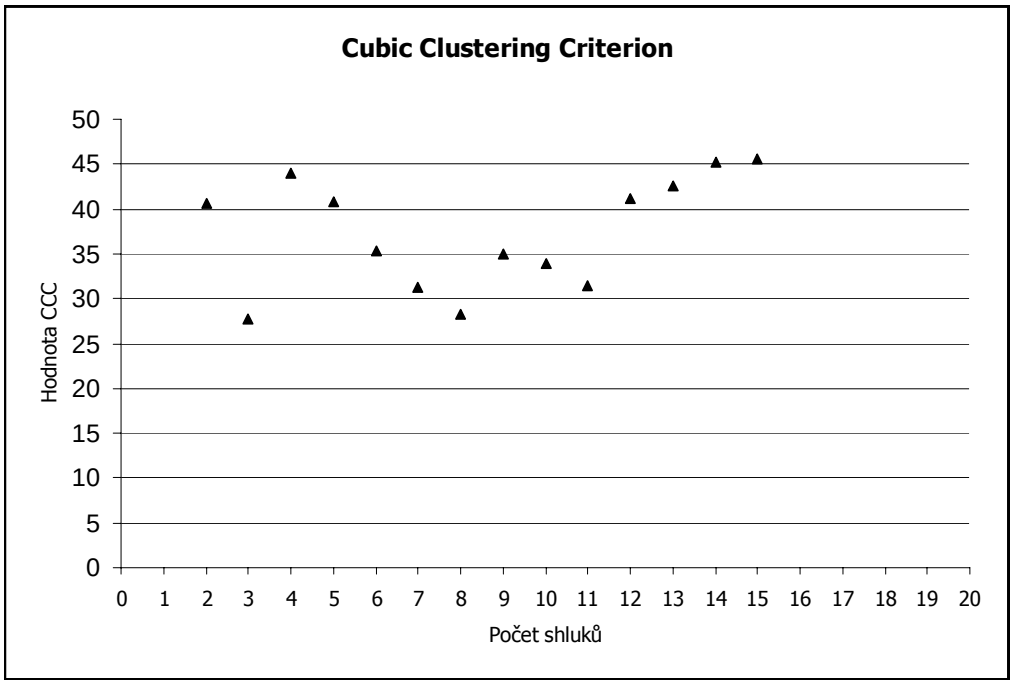
Na základě zobrazení hodnot ve výše uvedeném grafu lze podle lokálních vrcholů vybrat jako vhodný počet shluků dva, čtyři nebo jedenáct shluků. Výpočet CCC pro větší počet shluků však může být mírně zavádějící, neboť shluky neobsahují dostatečný počet pozorování.

Tabulka 17: Hodnoty kritérií při shlukování průměrů z nenulových dat a se standardizací

Počet shluků	Pseudo F Statistic	Approximate Expected Over-All R-Squared	Cubic Clustering Criterion
2	1598,58	0,37659	40,620
3	1790,47	0,66824	27,753
4	2961,43	0,75190	44,075
5	2835,05	0,80208	40,846
6	2478,02	0,83553	35,378
7	2227,26	0,85942	31,357
8	2037,80	0,87733	28,181
9	2564,27	0,89127	35,007
10	2498,63	0,90241	33,909
11	2329,35	0,91153	31,532
12	3221,12	0,91912	41,159
13	3398,93	0,92555	42,554
14	3750,00	0,93106	45,304
15	3799,51	0,93583	45,497

Při sledování hodnot kritéria CCC pro stanovení počtu shluků v případě shlukování průměrů nenulových dat lze doporučit dva, čtyři popř. devět shluků.

Graf 2: Vztah kritéria CCC a počtu shluků u souboru nenulových dat



5.2.2.4 Shlukování průměrů úplných záznamů

Na základě vývoje hodnot kubického shlukovacího kritéria jsou prezentovány výsledky pro dva, čtyři a jedenáct shluků.

Dva shluky – dvě skupiny zákazníků

Je zřejmé, že již počáteční středy shluků vykazují znatelný rozdíl v hodnotě průměrného množství odebraných a odeslaných dat. Tudíž i výsledné rozdělení pozorování je nerovnoměrné.

Ve shluku číslo jedna je zařazeno 784 zákazníků, ve shluku číslo dva jich je pouze 7. Je tedy zřejmé, že při segmentování zákazníků do dvou skupin se oddělí do dvou shluků ti zákazníci, kteří mají extrémně odlišné využití připojení k internetu.

2 shluky - Shlukování průměrů z úplných dat a se standardizací
The FASTCLUS Procedure
Replace=FULL Radius=0 Maxclusters=2 Maxiter=100 Converge=0.02

Initial Seeds

Cluster	MEAN1	MEAN2
1	-0.23568241	-0.16836328
2	14.62951133	17.09013157

Minimum Distance Between Initial Seeds = 22.77783

Iteration History

Iteration	Criterion	Relative Change in Cluster Seeds 1	2
1	0.6934	0.00987	0.2162
2	0.5768	0.00169	0.1716
3	0.5105	0.000435	0.0392
4	0.5070	0	0

Convergence criterion is satisfied. Criterion Based on Final Seeds = 0.5070

Cluster Summary

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	784	0.3665	4.6977	2	13.0047
2	7	4.0442	9.6706	1	13.0047

Pseudo F Statistic = 2276.84
Approximate Expected Over-All R-Squared = 0.37659
Cubic Clustering Criterion = 56.621
WARNING: The two values above are invalid for correlated variables.

Cluster Means

Cluster	MEAN1	MEAN2
1	-0.080366524	-0.082377364
2	9.001050650	9.226264743

Cluster Standard Deviations

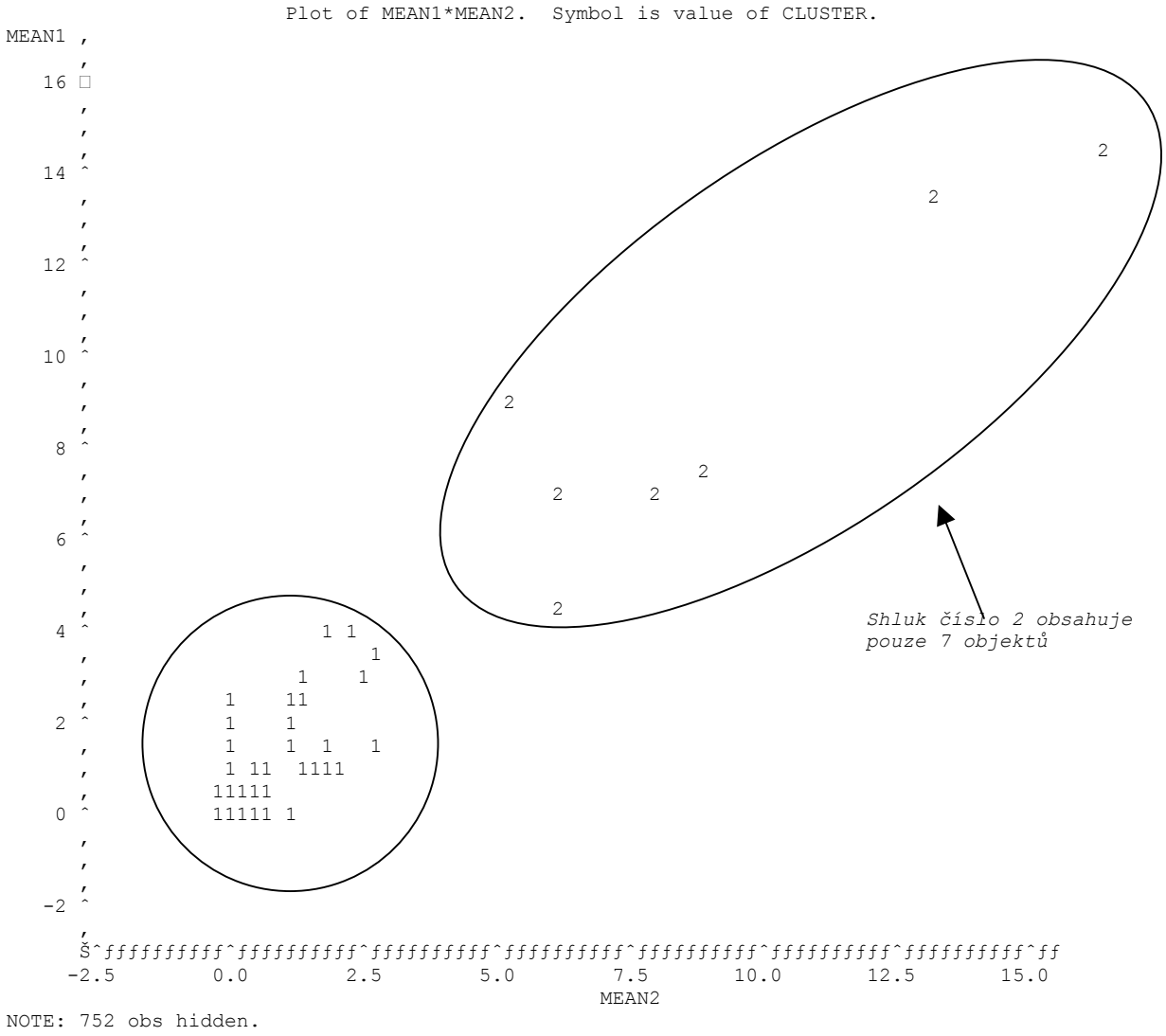
Cluster	MEAN1	MEAN2
1	0.419937432	0.303811231
2	3.645176662	4.407193644

Obrázek 37: Shluková analýza pro průměrný objem přenesených dat – 2 skupiny zákazníků

Charakteristiky ohledně polohy a variability je velmi obtížné interpretovat, neboť po standardizaci na nulový průměr a jednotkovou směrodatnou odchylku nezachovávají proměnné svůj původní rozměr.

Z následujícího obrázku je patrné, že jednotlivé skupiny (shluky) jsou poměrně jednoznačně odděleny. Pro toto grafické zobrazení výsledku je nutné ještě naprogramovat výstup pomocí procedury PLOT.

Graf 3: Dva shluky - Shlukování průměrů z úplných dat a se standardizací



NOTE: 752 obs hidden.

Čtyři shluky – čtyři skupiny zákazníků

Ve shluku číslo dva jsou zařazeny pouze 2 zákazníci, ve shluku číslo tři pouze 5 zákazníků, ve shluku čtyři 19 zákazníků. Většina zákazníků obsazuje shluk číslo jedna. I při segmentování do čtyř shluků je zřejmé, že tři shluky reprezentují zákazníky s nadstandardním využitím internetové sítě.

Cluster Summary

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	765	0.1489	1.4990	4	2.8071
2	2	2.0124	2.0124	3	10.8645
3	5	1.5628	2.6547	4	7.2168
4	19	0.9189	1.9979	1	2.8071

Cluster Means

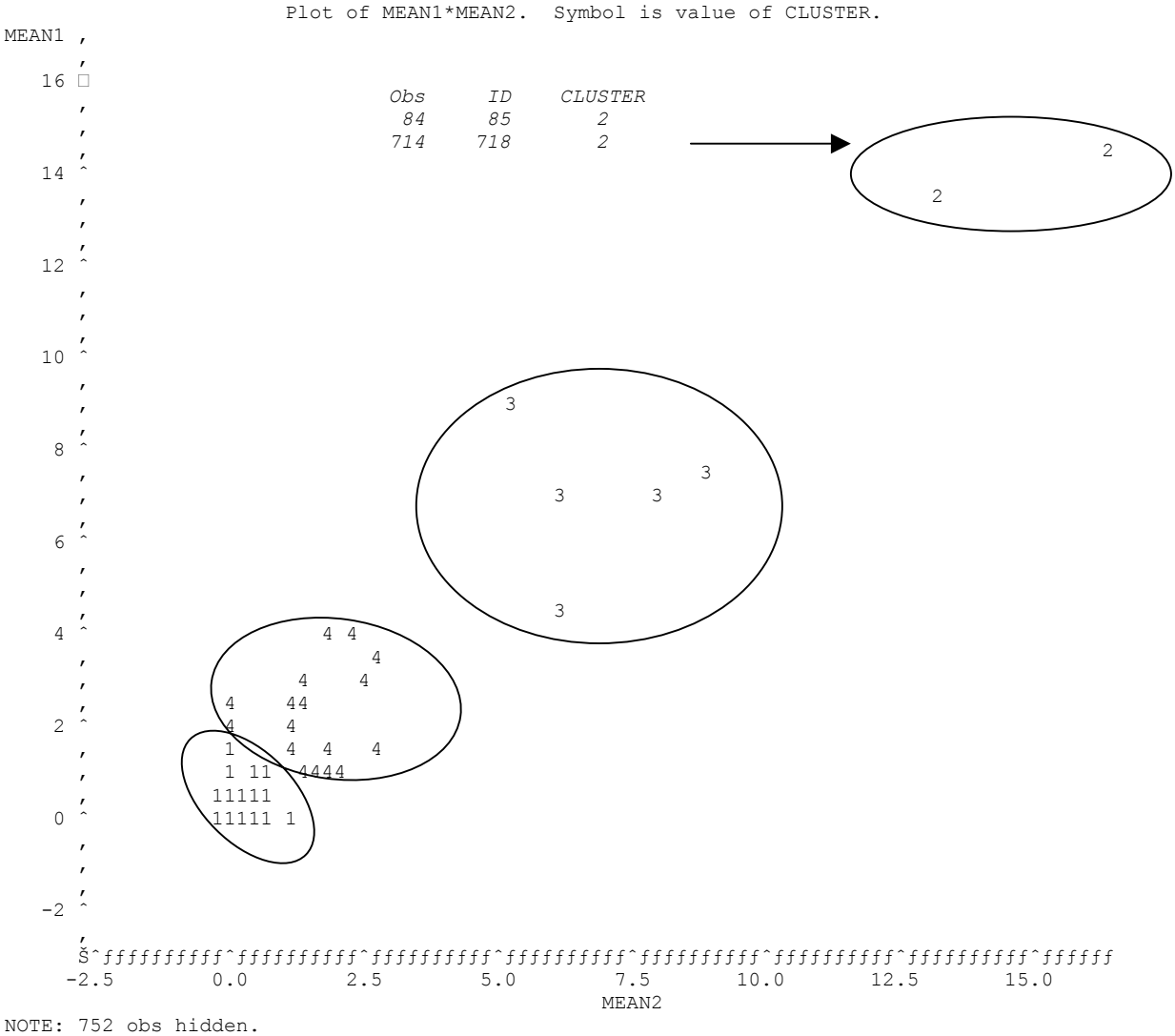
Cluster	MEAN1	MEAN2
1	-0.13562376	-0.12205689
2	13.96648938	15.19014334
3	7.01487516	6.84071330
4	2.14446419	1.51524580

Cluster Standard Deviations

Cluster	MEAN1	MEAN2
1	0.175431202	0.116419352
2	0.937654632	2.686989121
3	1.565970322	1.559667447
4	1.005090424	0.823845338

Obrázek 38: Shluková analýza pro průměrný objem přenesených dat – 4 skupiny zákazníků

Graf 4: Čtyři shluky - Shlukování průměrů z úplných dat a se standardizací



Jedenáct shluků – jedenáct skupin zákazníků

Na základě shlukování do jedenácti skupin je zřejmé, že byla identifikována celá řada případných odlehlých pozorování, neboť celkem pět skupin je tvořeno pouze jedním objektem. Méně než 10 pozorování je obsaženo v devíti z jedenácti shluků.

Cluster Summary

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	718	0.0583	0.3710	8	0.6809
2	1	.	0	7	4.0247
3	2	0.5150	0.5150	11	2.3884
4	1	.	0	11	2.5648
5	4	0.3809	0.7329	8	1.7060
6	1	.	0	11	2.0425
7	1	.	0	2	4.0247
8	46	0.2728	0.9824	1	0.6809
9	9	0.3939	0.9688	8	1.7308
10	7	0.6105	1.0227	9	1.9496
11	1	.	0	6	2.0425

Obrázek 39: Shluková analýza pro průměrný objem přenesených dat – 11 skupin zákazníků

5.2.2.5 Shlukování průměrů nenulových záznamů

Shlukování průměrů nenulových záznamů probíhá obdobně jako v případě souboru průměrů úplných záznamů. Počet záznamů se v tomto případě snížil na 6 433 348. Kód programu se opět liší pouze v proměnlivém počtu požadovaných shluků (od dvou do patnácti).

Přehled „Cluster Summary" je uveden v příloze. V následujím odstavci je představen pouze výsledek pro devět skupin, grafické zobrazení shlukování do dvou a do čtyř skupin je velmi podobné tomu, co již bylo prezentováno v případě souboru průměrů úplných záznamů.

```
proc fastclus data= meanclus.stand_A maxclusters=9 maxiter=100 out=meanclus.A09_s;
   var Mean1 Mean2; /* s = se standardizací*/
   title '9 shluků - Shlukování průměrů z aktivních dat a se standardizací'; run;
proc print data= meanclus.A09_s ;
   var ID cluster;
   where (cluster=2 or cluster=3 or cluster=5 or cluster=7 or cluster=9); run;
proc plot;
   plot Mean1*Mean2 = cluster;
run;
```

Děvět shluků – devět skupin zákazníků

Pro vytvoření devíti shluků proběhlo celkem jedenáct iterací (konvergenční kritérium bylo rovno hodnotě 0,1915). Vypočtené kubické shlukovací kritérium CCC dosáhlo hodnoty 35.

Co do počtu zákazníků je nejobsazenějším shlukem shluk číslo jedna, který obsahuje 745 zákazníků. Tři shluky jsou tvořeny pouze jedním objektem (zákazníci s ID 85, 718 a 257). Další dva shluky představují pouze dva objekty (ve shluku číslo tři jsou zákazníci s ID 264 a 490, ve shluku číslo dva jsou zákazníci s ID 492 a 86).

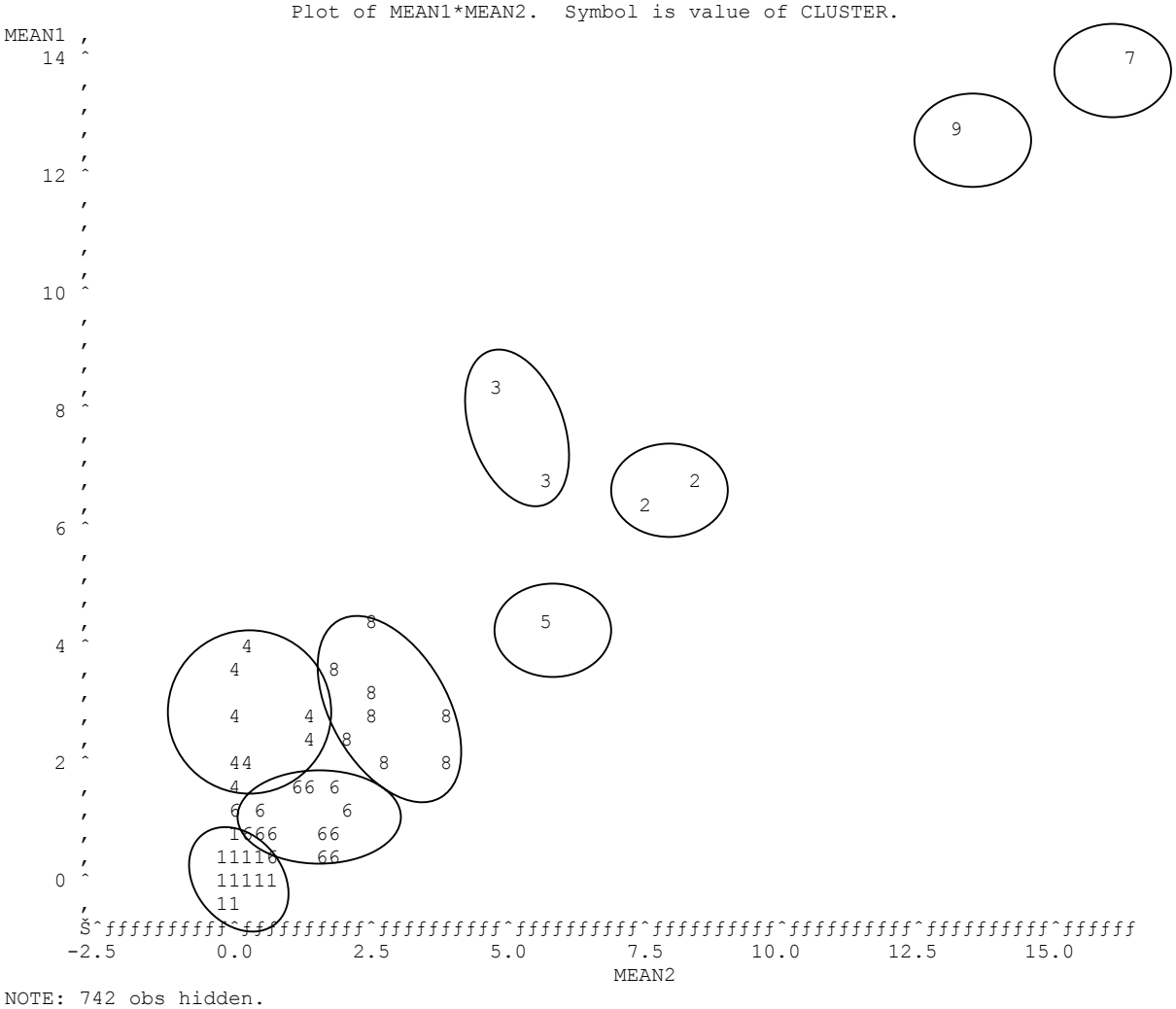
Cluster Summary

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	745	0.1267	0.9421	6	1.6161
2	2	0.4858	0.4858	3	2.8640
3	2	0.9588	0.9588	2	2.8640
4	11	0.6927	1.3744	6	1.8315
5	1	.	0	2	3.1675
6	20	0.5698	1.1194	1	1.6161
7	1	.	0	9	3.6284
8	8	0.8410	1.6657	4	2.3726
9	1	.	0	7	3.6284

Pseudo F Statistic = 2564.27
Approximate Expected Over-All R-Squared = 0.89127
Cubic Clustering Criterion = 35.007

Obrázek 40: Shluková analýza pro průměrný objem přenesených dat – 9 skupin zákazníků

Graf 5: Devět shluků - Shlukování průměrů z nenulových záznamů a se standardizací



5.2.3 Shluková analýza sumarizovaných údajů za zákazníky

Při shlukování součtů údajů za jednotlivé zákazníky (celkem 795) bylo zjištěno, že je několik zákazníků, kteří odebírají a odesílají extrémní množství dat. Proto byla tato úloha koncipována pro eliminaci výskytu odlehlých pozorování, jež shluková analýza pomáhá identifikovat.

Nalezení odlehlých pozorování bylo realizováno na základě opakovaného spouštění procedury FASTCLUS s postupně se měnícím počtem shluků v rozsahu od dvou do patnácti. Následující tabulka udává, při jakém počtu shluků bylo ve shluku nalezeno méně než deset pozorování.

Tabulka 18: Identifikace odlehlých pozorování na základě různého počtu shluků

Číslo shluku		Počet shluků													
		2	3	4	5	6	7	8	9	10	11	12	13	14	15
ID zákazníka	85	1	3	3	1	**6**	6	6	5	4	**4**	6	7	1	**13**
	86				3	**4**	1	8	4	10	**11**	12	13	5	**10**
	111									3	**1**	4	5	10	**11**
	125												12	12	**3**
	175					**3**		1	8	9	**10**	11	3	3	**5**
	176												12	12	**3**
	240												12	12	**3**
	249					**3**		1	3	1	**5**	1	6	13	**14**
	264									3	**7**	8		11	**7**
	362									3	**1**	4	5	10	**11**
	364												11		
	415					**3**		1	8	9	**10**	9	4	7	**9**
	488													12	**3**
	491		1	4	5	**5**	5	7	1	7	**9**	3	8	8	**6**
	492									3	**7**	8	11	11	**7**
	529												12	12	**3**
	616					**3**		1	8	9	**10**	11	3	3	**5**
	674												12	12	**15**
	692												12	12	**15**
	703												12		**3**
	718		1	4	5	**5**	3	3	7	8	**8**	5	10	9	**8**
	745									3	**1**	4	5	10	**11**
	751										**7**	8	11	11	**7**
	905												12	12	**3**

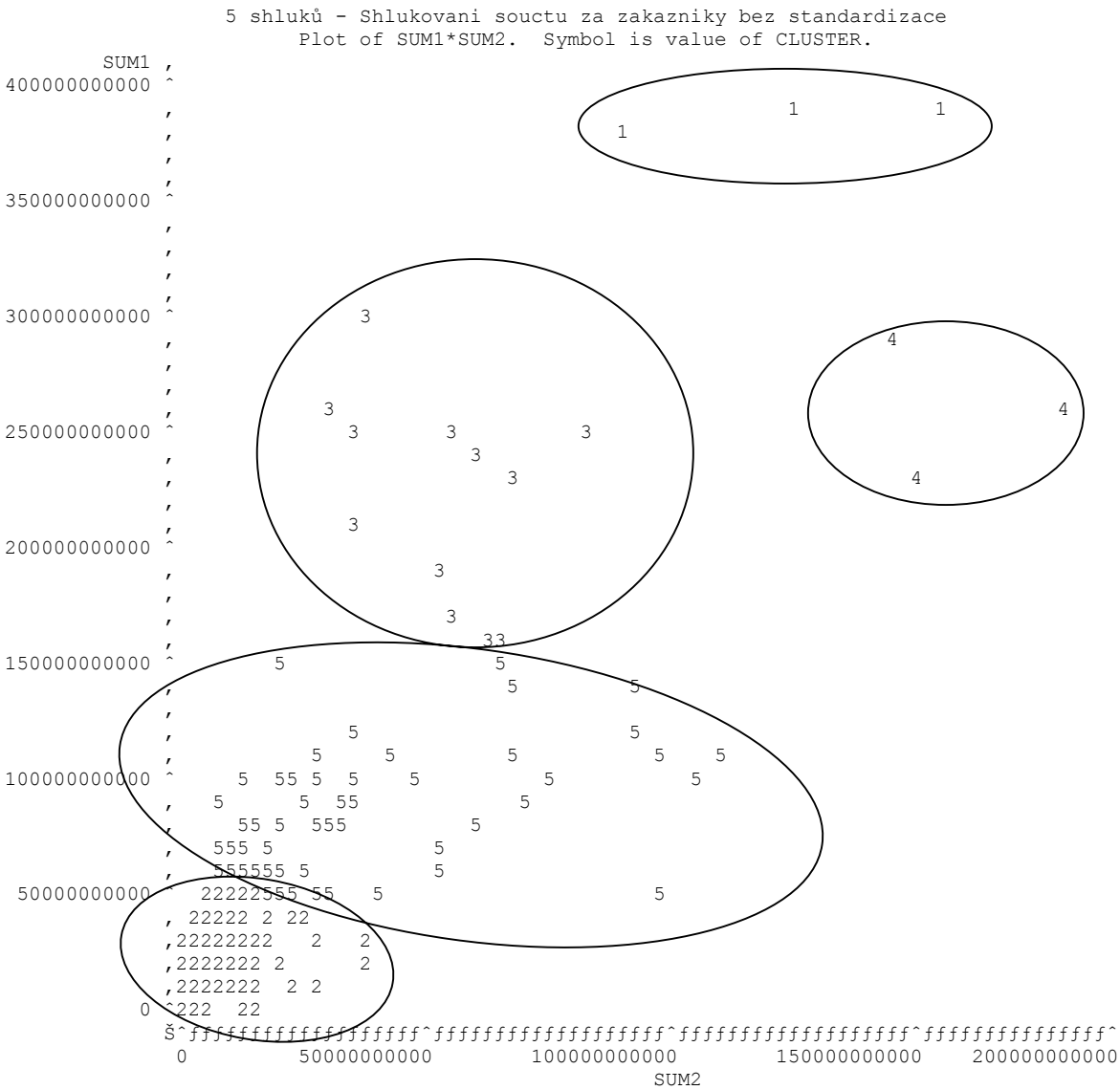
V první vlně byly nejprve odstraněny sumarizace za extrémně odlišné zákazníky, kteří byli identifikováni při shlukování do šesti skupin. Tuto skupinu „lukrativních zákazníků" tvoří osm klientů, kteří fungují víceméně jako provideři.

```
data sumclus.DataSumC; /* vyřadí lukrativní zákazníky */
   set sasuser.DataSumB;
   if (ID=85 or ID=86 or ID=175 or ID=249 or ID=415 or ID=491 or
       ID=616 or ID=718) then delete;
run;
```

Po výše uvedeném vyloučení bylo opět provedeno shlukování. Postupně byly vytvořeny skupiny od dvou do pěti. Již při shlukování do pěti shluků došlo k odtržení zákazníků, kteří

byli v tabulce „Identifikace odlehlých pozorování" nalezeni při shlukování do jedenácti shluků. Tito klienti jsou nyní prezentováni jako shluk číslo jedna a číslo čtyři.

Graf 6: Pět shluků - Shlukování součtů po vyloučení 8 zákazníků – 1. vlna



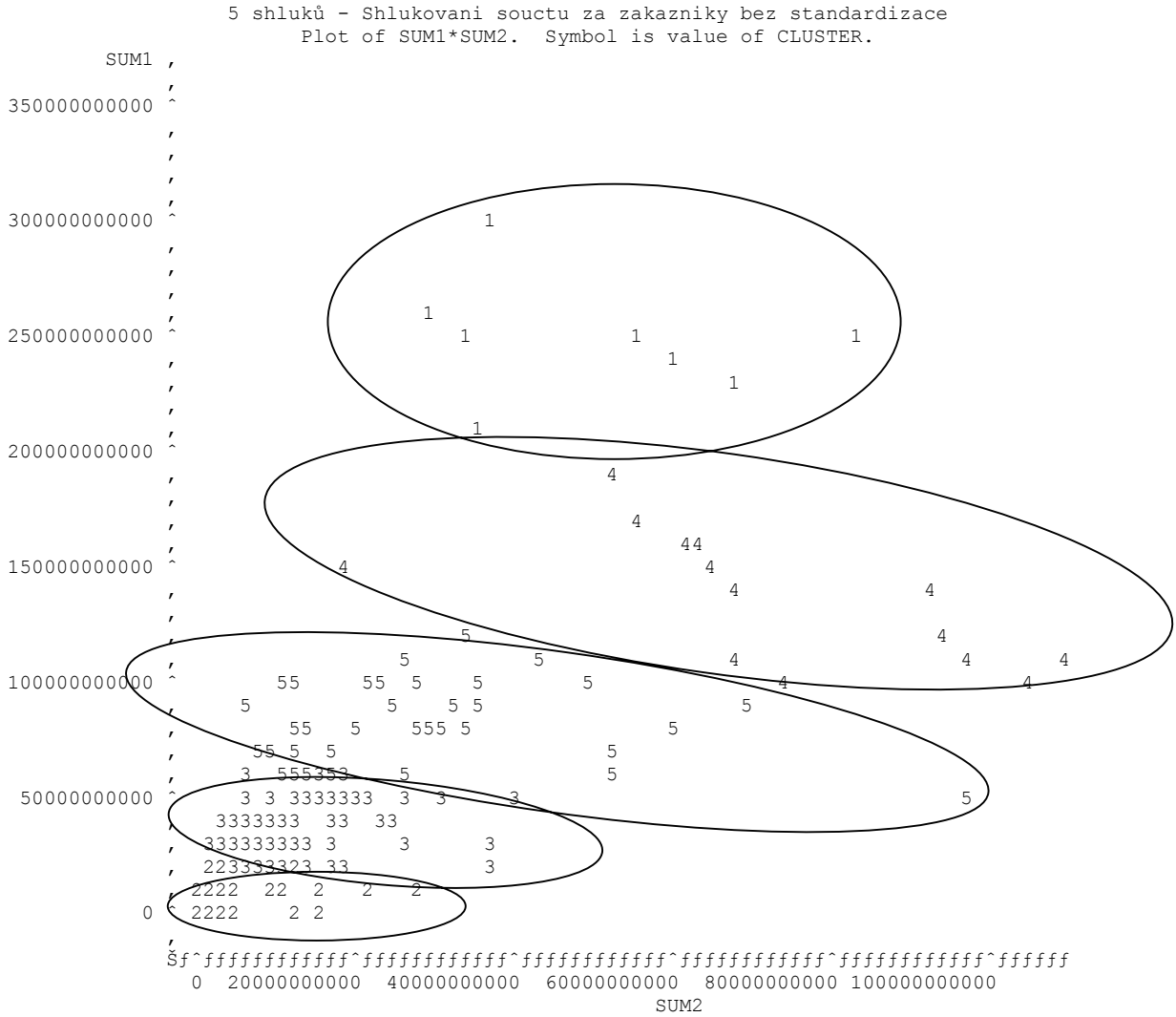
NOTE: 672 obs hidden.

Výše uvedených šest zákazníků ve shluku č. 1 a 4 bylo tedy na základě hodnot odebraných a odeslaných dat a chování při shlukování označeno jako odlehlá pozorování a v dalším kroku vypuštěno (2. vlna). Vznikla tedy skupina „lukrativních zákazníků" o 14 členech.

```
data sumclus.DataSumD; /* vyřadí lukrativní zákazníky */
   set sasuser.DataSumB;
   if (ID=85 or ID=86 or ID=175 or ID=249 or ID=415 or ID=491 or ID=616 or
       ID=718 or ID=111 or ID=264 or ID=362 or ID=492 or ID=745 or ID=751) then delete;
run;
```

Po výše uvedeném, již druhém, vyloučení odlehlých objektů (zůstalo 781 sumarizovaných záznamů) bylo opět provedeno shlukování. Postupně byly vytvořeny skupiny od dvou do pěti, přičemž už u pěti shluků začalo docházet k vymezení zákazníků, kteří ještě nebyli vyloučeni a v tabulce byli v původním shlukování do patnácti shluků (nyní jako shluk číslo jedna).

Graf 7: Pět shluků - Shlukování součtů po vyloučení 14 zákazníků – 2. vlna



NOTE: 659 obs hidden.

Následovalo tedy již třetí odfiltrování odlehlých pozorování (3. vlna). Vytvořila se tak skupina „lukrativních" zákazníků o 23 klientech.

```
data sumclus.DataSumE; /* vyřadí lukrativní zákazníky */
   set sasuser.DataSumB;
   if (ID=85 or ID=86 or ID=175 or ID=249 or ID=415 or ID=491 or
   ID=616 or ID=718 or ID=111 or ID=264 or ID=362 or ID=492 or
   ID=745 or ID=751 or ID=125 or ID=176 or ID=240 or ID=488 or
   ID=529 or ID=674 or ID=692 or ID=703 or ID=905 ) then delete;
run;
```

Při dalším shlukování byly výsledky následující:

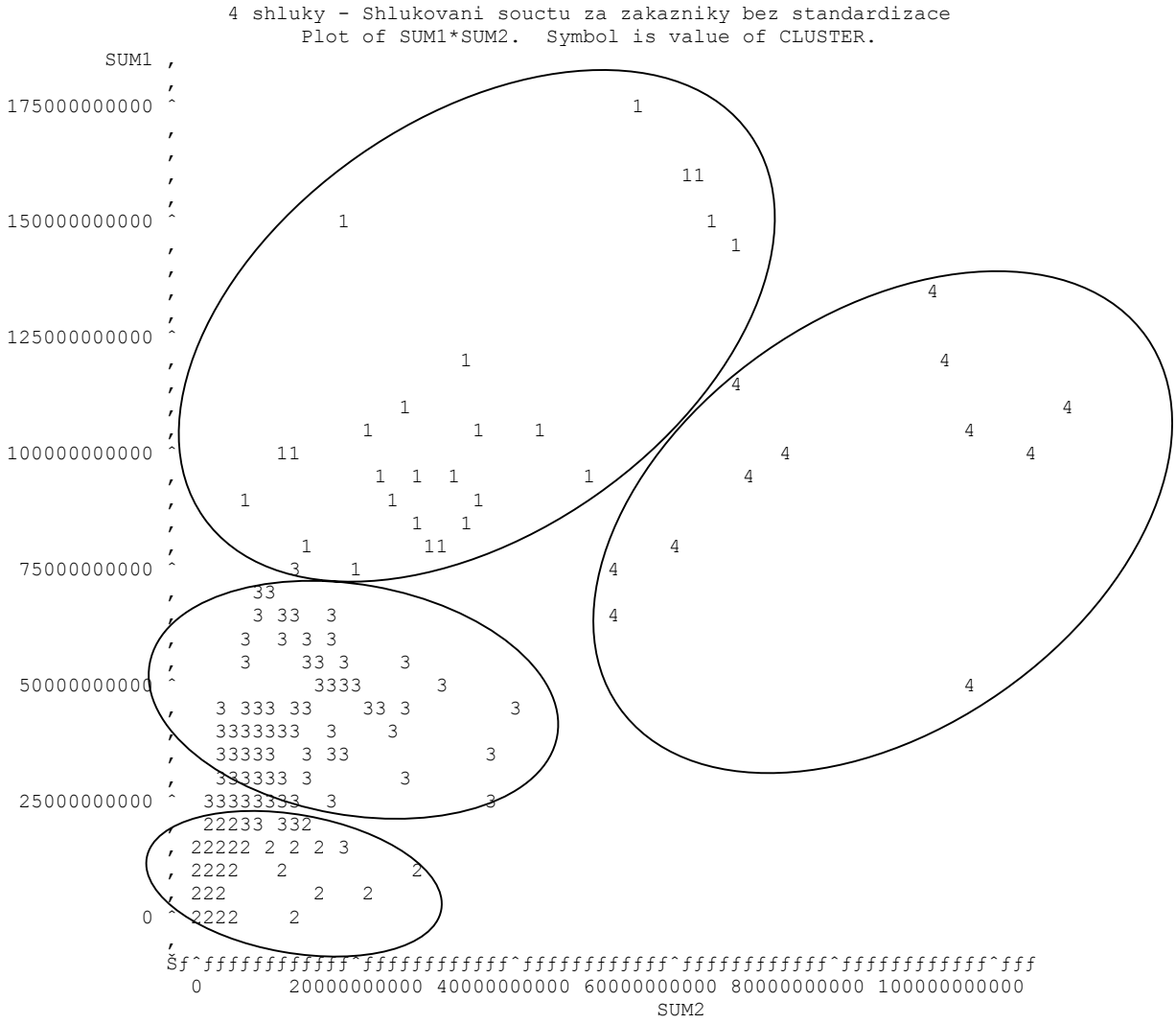
Cluster Summary

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	61	2.965E10	8.68E10	2	8.574E10
2	711	8.51E9	4.421E10	1	8.574E10
1	36	2.83E10	7.277E10	3	7.9E10
2	605	4.0907E9	2.61E10	3	3.409E10
3	131	1.118E10	4.38E10	2	3.409E10
1	**26**	**2.371E10**	**6.885E10**	**4**	**4.919E10**
2	**600**	**3.9344E9**	**2.618E10**	**3**	**3.296E10**
3	**134**	**1.067E10**	**3.868E10**	**2**	**3.296E10**
4	**12**	**2.29E10**	**4.867E10**	**1**	**4.919E10**
1	**6**	1.488E10	3.672E10	5	6.342E10
2	581	3.4371E9	2.643E10	3	2.802E10
3	138	8.507E9	3.518E10	2	2.802E10
4	37	1.524E10	3.922E10	3	5.147E10
5	10	2.083E10	5.215E10	1	6.342E10
1	38	1.564E10	3.899E10	3	5.171E10
2	581	3.4371E9	2.643E10	3	2.802E10
3	138	8.507E9	3.518E10	2	2.802E10
4	**1**	.	0	5	6.076E10
5	8	1.516E10	2.743E10	6	5.686E10
6	6	1.488E10	3.672E10	5	5.686E10
1	8	1.516E10	2.743E10	4	5.686E10
2	559	2.9552E9	2.663E10	3	2.273E10
3	134	6.0996E9	2.998E10	2	2.273E10
4	6	1.488E10	3.672E10	1	5.686E10
5	41	9.3104E9	2.956E10	3	2.868E10
6	23	1.342E10	3.444E10	5	4.12E10
7	1	.	0	1	6.076E10
1	559	2.9552E9	2.663E10	8	2.273E10
2	7	1.393E10	2.606E10	5	3.532E10
3	6	1.488E10	3.672E10	7	6.06E10
4	1	.	0	2	5.35E10
5	19	1.059E10	2.557E10	2	3.532E10
6	41	9.3104E9	2.956E10	8	2.868E10
7	5	1.19E10	2.345E10	2	4.676E10
8	134	6.0996E9	2.998E10	1	2.273E10
1	19	1.059E10	2.557E10	6	3.505E10
2	3	6.0404E9	8.2151E9	7	2.719E10
3	6	1.488E10	3.672E10	7	4.341E10
4	559	2.9552E9	2.663E10	9	2.273E10
5	41	9.3104E9	2.956E10	9	2.868E10
6	6	1.255E10	2.207E10	1	3.505E10
7	3	1.352E10	2.024E10	2	2.719E10
8	1	.	0	6	5.147E10
9	134	6.0996E9	2.998E10	4	2.273E10
1	1	.	0	8	5.436E10
2	552	2.7359E9	2.12E10	4	2.189E10
3	44	9.2007E9	2.894E10	4	2.783E10
4	137	6.1871E9	2.995E10	2	2.189E10
5	16	1.113E10	2.508E10	9	2.428E10
6	3	1.178E10	1.708E10	8	2.634E10
7	1	.	0	10	4.407E10
8	5	1.304E10	2.178E10	6	2.634E10
9	8	1.072E10	1.903E10	5	2.428E10
10	5	8.6226E9	1.78E10	7	4.407E10

Po postupném, trojím, odstranění odlehlých objektů zůstalo v souboru 772 zákazníků. Na tomto datovém souboru byla opakovaně spouštěna procedura FASTCLUS s postupně se měnícím počtem shluků v rozsahu od 2 do 10. Již při shlukování do šesti skupin došlo k oddělení jednoho zákazníka do samostatného shluku. Při pohledu na grafické znázornění lze usoudit, že tento zákazník vykazuje sice odlišné chování, ale podle hodnot odebraných a odeslaných dat jej lze ponechat mezi „nelukrativními" zákazníky. Tento klient nebyl identifikován v tabulce před začátkem filtrování odlehlých pozorování.

Na základě posouzení získaných výsledků lze pro marketingové účely doporučit výsledné čtyři skupiny zákazníků (maximálně pět skupin, kde ale pátá skupina obsahuje pouze šest zákazníků). Grafické zobrazení uvedeného řešení je následující:

Graf 8: Čtyři shluky - Shlukování součtů po vyloučení 23 zákazníků – 3. vlna



NOTE: 634 obs hidden.

Hodnocení shlukování dle kritérií

Ve všech případech shlukování lze pozorovat potvrzení teorie, že jestliže počet shluků roste, stoupá R^2 a zvyšuje se homogenita v rámci shluku. Hodnota kritéria Pseudo-F statistiky je se zvyšujícím se počtem shluků proměnlivá.

Tabulka 19: Hodnotící kritéria při shlukování sumarizovaných záznamů po odstranění 8 zákazníků

Počet shluků	Pseudo F Statistic	Approximate Expected Over-All R-Squared	Cubic Clustering Criterion
2	1537,95	0,64300	1,846
3	1935,42	0,76738	12,427
4	1722,78	0,82604	9,840
5	1435,21	0,86124	4,901

Tabulka 20: Hodnotící kritéria při shlukování sumarizovaných záznamů po odstranění 14 zákazníků

Počet shluků	Pseudo F Statistic	Approximate Expected Over-All R-Squared	Cubic Clustering Criterion
2	1515,14	0,65208	0,801
3	1646,07	0,77585	6,026
4	2145,63	0,83238	15,381
5	2297,00	0,86630	17,918

Potvrzuje se, že kubické shlukovací kritérium má tendenci být konzervativním, nejnižších hodnot dosahuje pro 2 shluky.

Tabulka 21: Hodnotící kritéria při shlukování sumarizovaných záznamů po odstranění 23 zákazníků

Počet shluků	Pseudo F Statistic	Approximate Expected Over-All R-Squared	Cubic Clustering Criterion
2	1526,32	0,60021	6,363
3	1811,07	0,73428	16,755
4	1541,40	0,80130	12,050
5	1784,56	0,84151	16,749
6	1515,02	0,86830	11,850
7	1856,41	0,88744	17,946
8	1759,28	0,90179	16,319
9	1584,05	0,91295	13,219
10	1472,71	0,92188	11,091

Hodnocení shlukování na základě počtu iterací

Z následující tabulky je zřejmé, že s rostoucím počtem shluků dochází k proměnlivému počtu iterací.

Tabulka 22: Počet iterací při shlukování sumarizovaných záznamů po odstranění odlehlých pozorování

Počet iterací		Počet shluků								
		2	3	4	5	6	7	8	9	10
Typ souboru	po odstranění 8 zákazníků	5	8	7	5					
	po odstranění 14 zákazníků	6	5	14	18					
	po odstranění 23 zákazníků	5	13	11	9	11	14	15	8	5

5.2.4 Shluková analýza neagregovaných záznamů

Cílem shlukování neagregovaných záznamů je identifiakce vybočujících pozorování. Pro získání porovnatelných výsledků bylo zvoleno shlukování pro stejný počet shluků, tj. 20, ale s odlišným způsobem zpracování. Jedna realizace představuje shlukování se standardizací, druhá bez standardizace, třetí s vyloučením nulových záznamů a se standardizací a čtvrtá s vyloučením nulových záznamů bez standardizace. Výpočet předložených výsledků vyžaduje vzhledem ke zpracování vysokého počtu záznamů (přes 10 milionů) vysoký výpočetní výkon a je velmi časově náročný.

5.2.4.1 Shlukování úplného souboru dat se standardizací

```
/* Shluková analýza úplného souboru dat se standardizací*/
Proc standard data=sasuser.DataNetN
              out=sasuser.standN mean=0 std=1;
    var Data_in Data_out; /* Mean1 je průměrný počet přijatých dat a Mean2 je průměrný
počet odeslaných dat*/
run;

proc fastclus data= sasuser.standN maxclusters=20 maxiter=100
              out=sasuser.outclusN;
    var Data_in Data_out;
run;
```

Shlukování do 20 skupin proběhlo na základě 23 iterací. Bylo vytvořeno 6 shluků s jedním záznamem a 8 shluků obsahujících do deseti záznamů. Celkem 14 shluků z dvaceti přispívá k identifikaci odlehlých hodnot. Většina záznamů je shlukována do skupiny č. 10.

Cluster Summary

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	**1**	.	0	2	157.0
2	7	18.6460	36.3939	7	69.8559
3	4	29.1059	49.7429	16	108.7
4	6333	5.2723	32.9434	12	12.5779
5	**1**	.	0	8	92.6439
6	4	17.9144	31.6716	7	42.3619
7	**1**	.	0	6	42.3619
8	867	8.8286	33.0475	9	23.9472
9	2419	6.9463	54.7611	4	17.2016
10	1.015E7	0.1215	4.2776	12	3.9298
11	38	18.9404	62.9204	8	52.5540
12	56461	1.9448	12.5008	10	3.9298
13	3	26.2624	37.3422	15	60.3161
14	**1**	.	0	11	103.6
15	**1**	.	0	13	60.3161
16	3	23.9961	34.1089	18	86.9010
17	7	20.9047	36.9855	2	84.9017
18	5	25.4991	41.3815	6	69.8651
19	**1**	.	0	15	216.8
20	3	25.8016	37.4144	13	110.8

Pseudo F Statistic = 6773420
Approximate Expected Over-All R-Squared = 0.95000
Cubic Clustering Criterion = -1310.45
WARNING: The two values above are invalid for correlated variables.

Na základě vysoké záporné hodnoty (-1310) kubického shlukovacího kritéria CCC je zřejmé, že v souboru jsou obsažena odlehlá pozorování.

5.2.4.2 Shlukování úplného souboru dat bez standardizace

```
/* Shluková analýza úplného souboru dat bez standardizace */
proc fastclus data= sasuser.DataNetN maxclusters=20 maxiter=100
              out=sasuser.outclusN_bezSTAND;
    var Data_in Data_out;   run;
```

V tomto shlukování bylo vytvořeno pět shluků obsahujících pouze jeden záznam a dalších osm shluků obsahuje do deseti záznamů. Všech třináct shluků tedy identifikuje odlehlá pozorování, jež by v další etapě případného shlukování záznamů měla být odstraněna a posuzována odděleně.

Cluster Summary

Cluster	Freq	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	1.015E7	4287639	1.8947E8	3	1.3465E8
2	37	6.6153E8	1.8668E9	11	1.8011E9
3	59232	60099590	4.4843E8	1	1.3465E8
4	**1**	.	0	15	2.7963E9
5	4	9.8298E8	1.6924E9	20	4.3601E9
6	2	6.3258E8	6.3258E8	16	2.0432E9
7	6457	1.613E8	1.0611E9	3	3.6667E8
8	**1**	.	0	13	4.9534E9
9	1387	2.4408E8	1.6492E9	10	5.1705E8
10	2448	1.8498E8	9.8659E8	7	4.6873E8
11	552	2.7761E8	1.6831E9	9	6.6715E8
12	7	6.3464E8	1.2845E9	14	2.8995E9
13	**1**	.	0	11	2.8167E9
14	7	7.2048E8	1.4005E9	15	1.4826E9
15	**1**	.	0	14	1.4826E9
16	2	6.536E8	6.536E8	6	2.0432E9
17	5	1.1756E9	2.0376E9	19	3.6025E9
18	**1**	.	0	6	7.7901E9
19	7	9.3185E8	1.6363E9	12	3.0711E9
20	3	9.1832E8	1.3361E9	16	3.223E9

Pseudo F Statistic = 7508746
Approximate Expected Over-All R-Squared = 0.95139
Cubic Clustering Criterion = -1078.88
WARNING: The two values above are invalid for correlated variables.

Cluster Means

Cluster	Data_In	Data_Out
1	1207818	402556
2	3248057518	1914896437
3	126376657	50032353
4	6714007825	536786249
5	15463759333	14102446997
6	22794813524	20191030290
7	403688703	289916502
8	1729569603	8727429677
9	1154063574	821903664
10	860137051	396524296
11	1530534197	1372683854
12	7123764284	6704110266
13	404566632	3954524395
14	5429976238	4350831025
15	6500321620	3324911884
16	21635190106	18508815586
17	12168856606	10961504326
18	29492815454	24168811360
19	9126395767	9032393314
20	19021311729	16623233162

Cluster Standard Deviations

Cluster	Data_In	Data_Out
1	5136546	3222358
2	634932556	687104425
3	63739477	56224554
4	.	.
5	1006500328	958889713
6	517665708	729621644
7	137428597	182071645
8	.	.
9	275342825	208176919
10	183543667	186403076
11	289949565	264704869
12	542002821	715388487
13	.	.
14	684480439	754765221
15	.	.
16	732087696	564289760
17	1287293271	1052208486
18	.	.
19	802069694	1045650963
20	1113050482	669146476

Ačkoli první shluk obsahuje většinu záznamů, z hlediska průměrných odebraných i odeslaných hodnot představuje z 20 shluků záznamy s nejnižší aktivitou.

5.2.4.3 Shlukování s vyloučením nulových záznamů a se standardizací

Po vyřazení nulových hodnot klesne počet záznamů o 37 % na 6 433 348.

```
proc standard data=sasuser.DataNetO
              out=sasuser.standO mean=0 std=1;
    var Data_in Data_out; /* Mean1 je průměrný počet přijatých dat a Mean2 je průměrný
počet odeslaných dat*/
run;

/* se standardizací - 6 433 348 pozorování = soubor bez nulových pozorování */
proc fastclus data= sasuser.standO maxclusters=20 maxiter=100
              out=outclusO;
    var Data_in Data_out;
run;
```

Shlukování do 20 skupin bylo realizováno pomocí 38 iterací. Pět shluků je tvořeno pouze jedním záznamem, dalších osm shluků obsahuje do deseti pozorování. Většina záznamů, tj. 99 %, je shlukována do segmentu č. 19.

Cluster Summary

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	30	13.9970	37.0901	6	44.3227
2	**1**	.	0	6	69.7908
3	6340	3.6825	19.9784	18	8.2096
4	5	20.2498	32.8518	10	56.6320
5	6	12.8710	22.9284	10	56.3249
6	491	6.6607	27.1771	15	16.8042
7	3	20.8595	29.6599	8	47.8840
8	**1**	.	0	7	47.8840
9	3	19.0549	27.0852	4	69.0340
10	5	16.2770	28.6379	5	56.3249
11	3	20.4959	29.7211	7	88.0293
12	**1**	.	0	1	80.7200
13	**1**	.	0	8	172.2
14	7	16.8091	31.8450	5	62.7735
15	1360	5.7224	49.6325	16	13.1204
16	2774	4.5626	26.2200	3	10.6254
17	**1**	.	0	5	124.1
18	60945	1.4013	8.2149	19	2.8689
19	6361367	0.1147	3.1465	18	2.8689
20	4	23.1156	39.5064	9	86.3054

Pseudo F Statistic = 4747337
Approximate Expected Over-All R-Squared = 0.95000
Cubic Clustering Criterion = -771.335
WARNING: The two values above are invalid for correlated variables.

Hodnota kubického shlukovacího kritéria je opět záporná (-771), čili potvrzuje skutečnost o výskytu odlehlých pozorování.

5.2.4.4 S vyloučením nulových záznamů a bez standardizace

```
/* bez standardizace - 6 433 348 pozorování = soubor bez nulových pozorování */
proc fastclus data= sasuser.DataNetO maxclusters=20 maxiter=100
              out=outclusO_bezSTAND;
    var Data_in Data_out;
run;
```

V průběhu shlukování do dvaceti skupin bylo dosaženo 22 iterací, než bylo splněno konvergenční kritérium. Opět bylo separováno 5 záznamů jako samostatné shluky a 8 shluků o méně jak deseti záznamech.

Cluster Summary

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	6364252	5389118	1.8923E8	3	1.3536E8
2	37	6.6153E8	1.8668E9	11	1.8004E9
3	58253	60545084	4.4838E8	1	1.3536E8
4	**1**	.	0	15	2.7963E9
5	4	9.8298E8	1.6924E9	20	4.3601E9
6	2	6.3258E8	6.3258E8	16	2.0432E9
7	6403	1.6163E8	1.0587E9	3	3.6869E8
8	**1**	.	0	13	4.9534E9
9	1390	2.4362E8	1.652E9	10	5.1598E8
10	2418	1.8483E8	9.834E8	7	4.6935E8
11	553	2.7758E8	1.6823E9	9	6.6764E8
12	7	6.3464E8	1.2845E9	14	2.8995E9
13	**1**	.	0	11	2.8186E9
14	7	7.2048E8	1.4005E9	15	1.4826E9
15	**1**	.	0	14	1.4826E9
16	2	6.536E8	6.536E8	6	2.0432E9
17	5	1.1756E9	2.0376E9	19	3.6025E9
18	**1**	.	0	6	7.7901E9
19	7	9.3185E8	1.6363E9	12	3.0711E9
20	3	9.1832E8	1.3361E9	16	3.223E9

Pseudo F Statistic = 4742349
Approximate Expected Over-All R-Squared = 0.95138
Cubic Clustering Criterion = -847.813
WARNING: The two values above are invalid for correlated variables.

Cluster	Cluster Means Data_In	Data_Out	Cluster	Cluster Standard Deviations Data_In	Data_Out
1	1936672	646326	1	6433236	4086400
2	3248057518	1914896437	2	634932556	687104425
3	127761355	50538342	3	64242221	56606990
4	6714007825	536786249	4	.	.
5	15463759333	14102446997	5	1006500328	958889713
6	22794813524	20191030290	6	517665708	729621644
7	405469323	293053367	7	138045692	182194805
8	1729569603	8727429677	8	.	.
9	1150611783	822974051	9	275356702	207069640
10	863829625	394026540	10	184214347	185440826
11	1531769127	1371124114	11	288881778	265803805
12	7123764284	6704110266	12	542002821	715388487
13	404566632	3954524395	13	.	.
14	5429976238	4350831025	14	684480439	754765221
15	6500321620	3324911884	15	.	.
16	21635190106	18508815586	16	732087696	564289760
17	12168856606	10961504326	17	1287293271	1052208486
18	29492815454	24168811360	18	.	.
19	9126395767	9032393314	19	802069694	1045650963
20	19021311729	16623233162	20	1113050482	669146476

Shluk číslo jedna tvoří opět 99 % všech záznamů zařazených do shlukové analýzy. Průměrná hodnota přijatých dat v tomto shluku je 1 936 672 B a odeslaných 646 326 B. Tento shluk vykazuje i nejnižší směrodatnou odchylku.

Je zřejmé, že pro identifikaci odlehlých hodnot je v tomto případě vliv standardizace minimální.

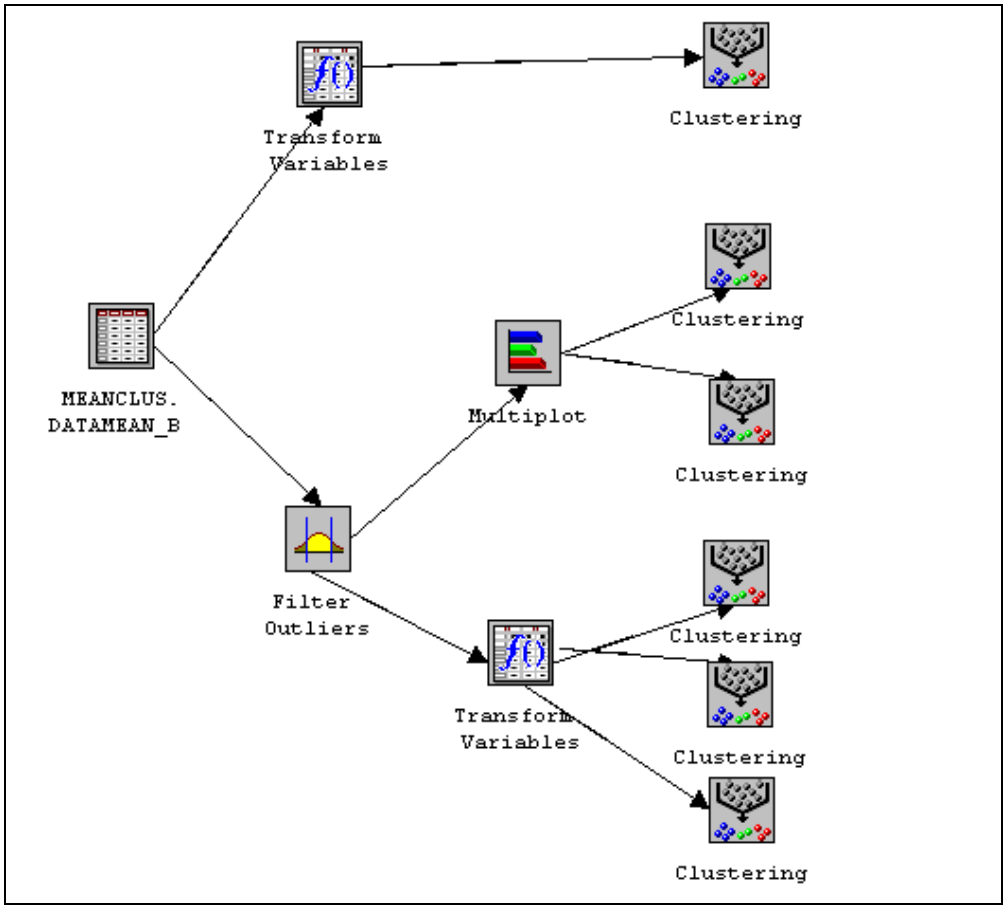
5.3 Shlukování v SAS Enterprise Miner

V prostředí SAS Enterprise Miner jsou realizovány dvě úlohy. Cílem první z nich je získat pomocí shlukování takové rozdělení zákazníků, jež by bylo vhodné pro stanovení obecného obchodního přístupu. Druhá úloha je zaměřena na vytvoření shluků agregovaných záznamů na základě použití kategorizovaných proměnných DEN a CAS.

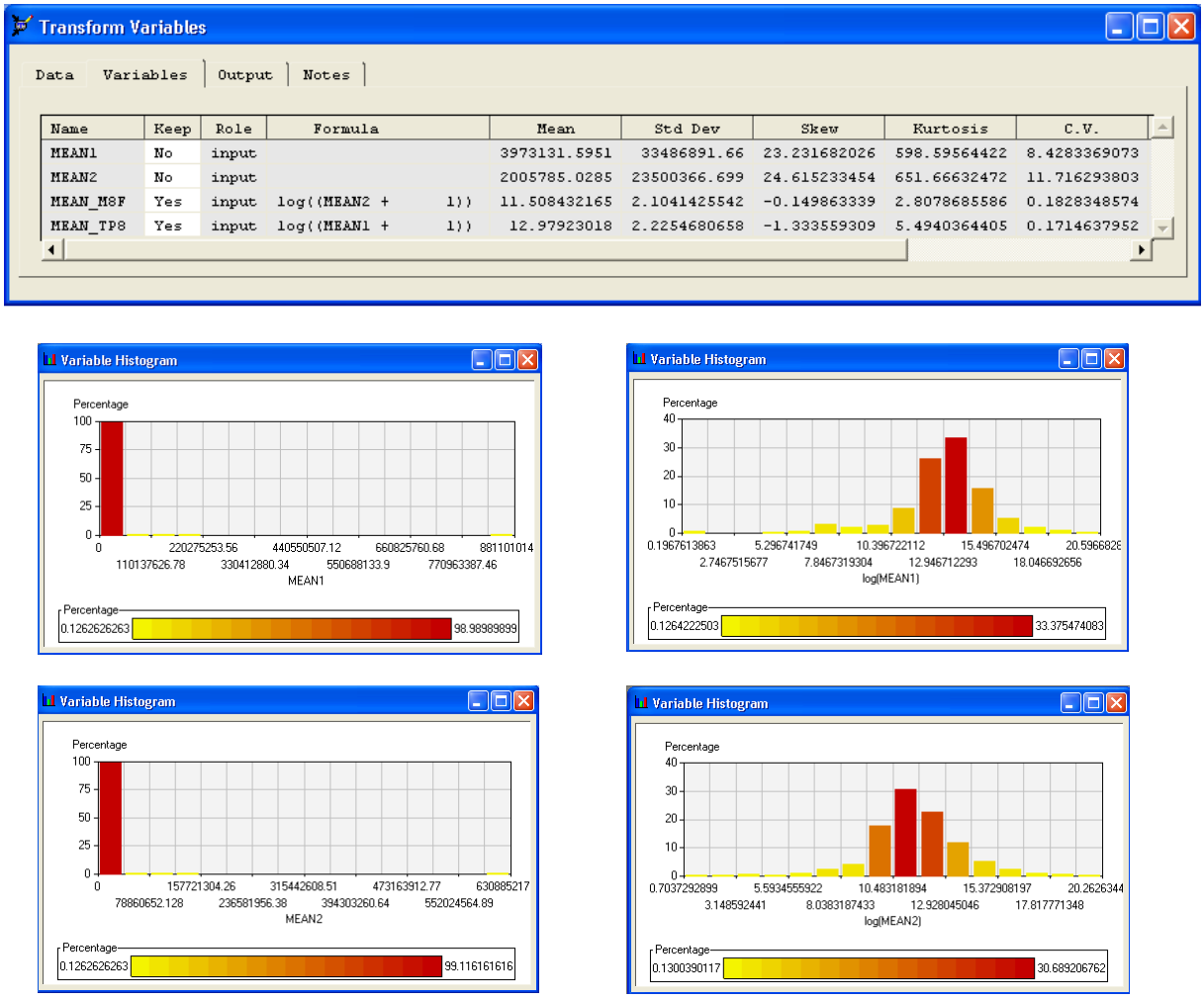
5.3.1 Segmentace zákazníků dle průměrného objemu přenesených dat

Pro segmentaci zákazníků dle průměrného množství přenesených dat je připraven soubor o 792 zákaznících. Pro úspěšné nalezení segmentů je nutné zvolit správný postup a v jeho rámci vyzkoušet takové parametry, které povedou k dobrému řešení.

Na následujícím obrázku je uvedeno několik použitých řešení, která mohou připadat v úvahu. Prvním z nich je logaritmická transformace proměnných a následné shlukování. Tento postup rozčlenil zákazníky do devíti skupin.



Obrázek 41: Diagram pro shlukování dat

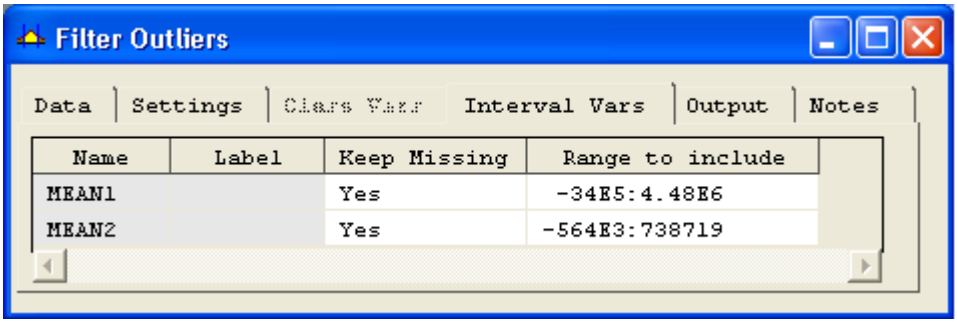


Obrázek 42: Logaritmická transformace v uzlu Transform Variables

Z dalších možností, jak dosáhnout kvalitního shlukování, je provést před transformací proměnných odstranění extrémních hodnot.

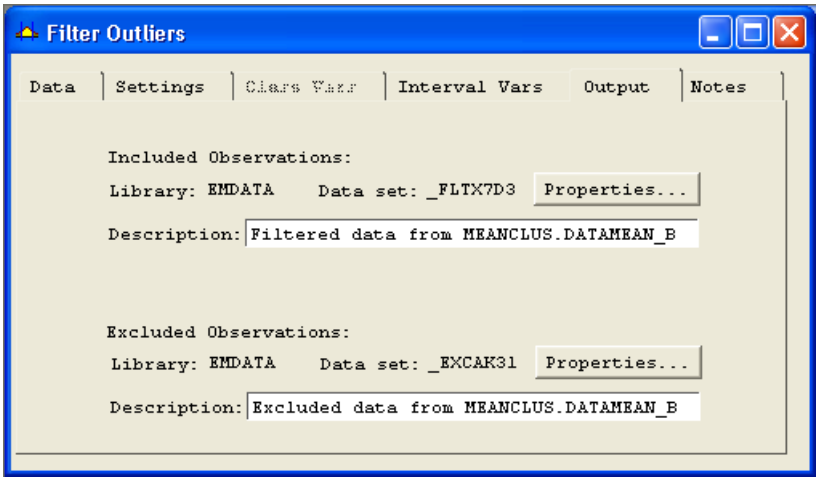
5.3.1.1 Analýza extrémních hodnot a odstranění odlehlých pozorování

V Enterprise Mineru nabízí uzel Filter Outliers několik možností pro filtrování na základě spojitých proměnných, především odstranění pozorování při hodnotě dané proměnné za definovaným násobkem směrodatné odchylky od průměru nebo odstraněním pozorování, pokud je hodnota za určitým percentilem. U kategoriálních proměnných lze odstranit pozorování s výjimečnou (minimálně zastoupenou) hodnotou.



Obrázek 43: Filtrování 9% MAD v uzlu Filtr Outliers

Výstupem z tohoto uzlu jsou dva soubory; jeden obsahuje filtrovaná data a druhý pozorování s extrémními hodnotami. Tato tak zvaná odlehlá pozorování mohou vyžadovat samostatnou segmentaci, případně mohou být považována za samostatný segment „neobvyklých případů". Pro analýzu odlehlých pozorování lze využít uzly Multiplot, Distibution Explorer nebo Insight.



Obrázek 44: Dva soubory: filtrovaná data × pozorování s extrémními hodnotami

Poté, co byla odstraněna odlehlá pozorování, zůstalo v souboru 685 zákazníků, u nichž bylo zkušebně provedeno shlukování. Při maximálním počtu shluků dvacet a maximálním počtu iterací 10, bylo vytvořeno 20 shluků. Na základě grafu kubického shlukovacího kritéria lze usoudit, že počet iterací nebyl dostatečný. Proto byla v novém uzlu Clustering nastavena možnost opakování výpočtu stokrát a standardizace směrodatnou odchylkou.

5.3.1.2 Standardizace

Jak již bylo předestřeno v předchozím odstavci, třetím krokem přípravy dat je standardizace proměnných. Ta je odůvodněna tím, že proměnné, které mají větší rozptyl, mají pak při využití shlukové analýzy v segmentačním modelu relativně větší význam než proměnné s menším rozptylem.

Uzel Clustering umožňuje automatickou standardizaci na proměnné se směrodatnou odchylkou rovnou jedné a mediánem nebo průměrem na jejich původní úrovni.

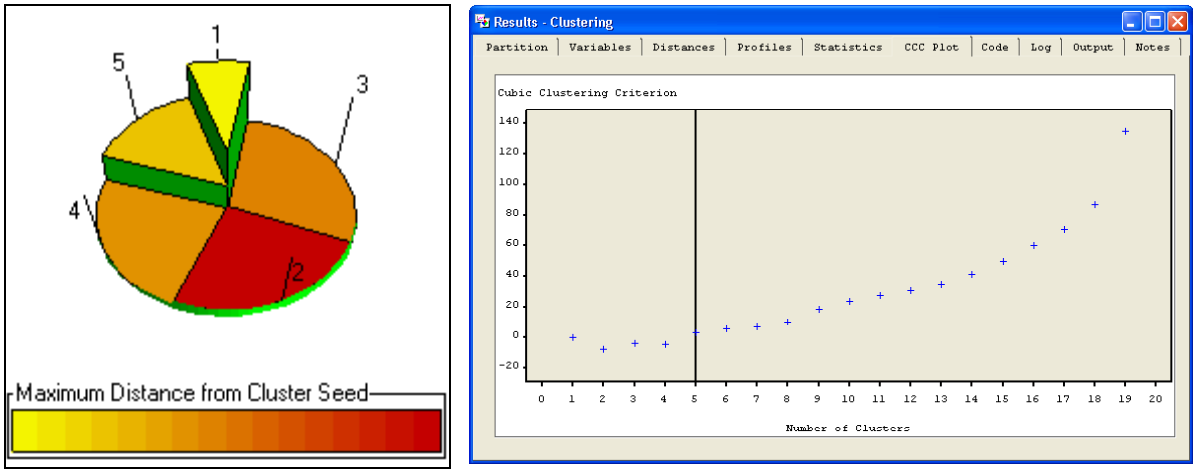
Transformaci proměnných lze realizovat také v uzlu Transform Variables. Ten může posloužit i v případě, že je potřeba dát proměnným do shlukové analýzy větší váhu.

5.3.1.3 Metody segmentace

V dosud realizované ukázkové úloze je pro výběr počtu shluků nastaveno CCC = 3 a maximum počtu shluků k = 20. Pro výpočet fáze hierarchického shlukování se používá Wardova metoda, shlukovací kritérium je nastaveno na Least Squares (metoda nejmenších čtverců) a konvergenční kritérium s prahem 0,02. Minimální počet shluků je dán automaticky volbou 2.

5.3.1.4 Profilace a interpretace segmentů

Nově vytvořené shluky představují již jen pět skupin. Jejich charakteristiky a grafické výstupy jsou uvedeny v příloze. První segment zahrnuje největší skupinu zákazníků, tj. 387 klientů, kteří přenášejí v průměru malý objem dat v obou směrech. Druhý segment obsahuje ty zákazníky (celkem 57), kteří stahují velké objemy dat, ale odesílají nepoměrně nízké množství dat. Třetí segment tvoří zákazníci, kteří odebírají i odesílají velké objemy dat, celkem jich je 24. Čtvrtý shluk představují zákazníci, kteří v průměru přijímají malé objemy dat, ale odesílají nepoměrně více (32 zákazníků). Poslední shluk 185 zákazníků je jistou kombinací shluku prvního a druhého, neboť se jedná zákazníky, kteří více odebírají a odesílají více než ti ve shluku č. 1.



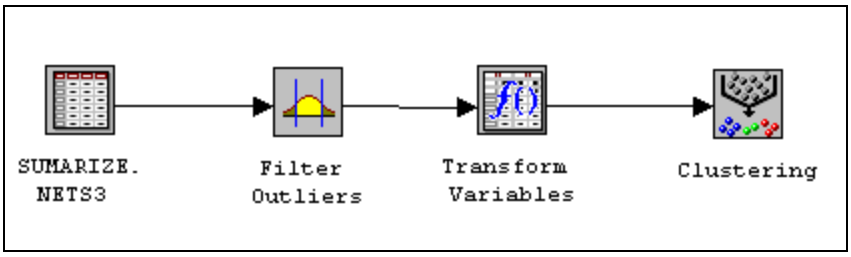
Obrázek 45: Segmentace zákazníků v Enterprise Miner - pět skupin

5.3.1.5 Zhodnocení výsledků

Z uvedených obrázků je zřejmé, že obě segmentační proměnné mají ve všech segmentech různé rozložení, proto se obě dobře uplatnily jako rozlišovací proměnné segmentace. Vytvořené shluky dávají obchodnímu oddělení dobrou příležitost v souvislosti s mírou objemu přenesených dat připravit pro zákazníky speciální služby, tarify a různá zvýhodnění.

5.3.2 Shluková analýza agregovaných údajů dle kategorizovaných proměnných

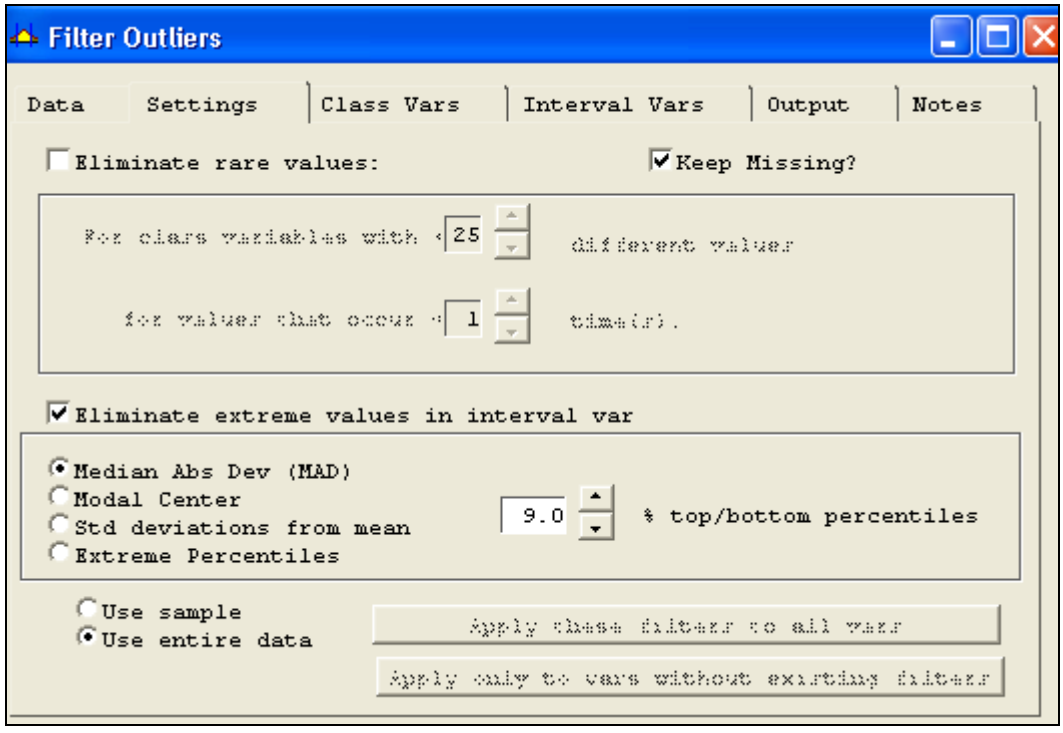
Pro shlukování údajů o průměrném objemu přenesených dat na základě hodiny, dne v týdnu a ID byly údaje agregovány pomocí programování. V modulu Enterprise Miner byl vytvořen procesní diagram, který zohledňuje výběr souboru, existenci odlehlých pozorování, potřebu transformace dat a shlukování.



Obrázek 46: Diagram pro shlukování dat

5.3.2.1 Analýza extrémních hodnot a odstranění odlehlých pozorování

Po výběru analyzovaného souboru přichází na řadu příprava dat. Ta zpravidla začíná analýzou a odstraněním pozorování s extrémními hodnotami. V uzlu Filter Outliers byla pro tyto účely vybrána možnost filtrování hodnot spojitých proměnných při hodnotě dané proměnné za definovaným násobkem směrodatné odchylky od průměru tj. 9 %.



Obrázek 47: Nastavení uzlu Filter Outliers

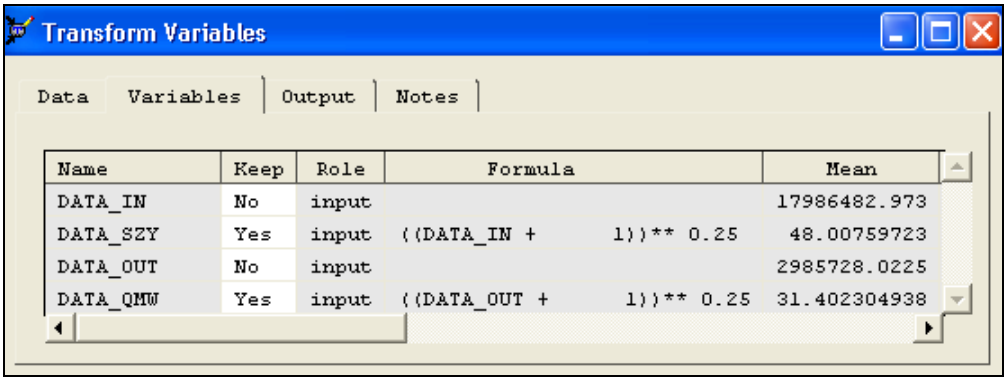
Výstupem z tohoto uzlu jsou dva soubory; jeden obsahuje filtrovaná data a druhý pozorování s extrémními hodnotami. Tato odlehlá pozorování mohou být předmětem samostatné segmentace, nebo mohou být považována za samostatný segment „neobvyklých případů".

5.3.2.2 Příprava kategoriálních proměnných

Druhým krokem je vytvoření nula-jedničkových tak zvaných „dummy" proměnných pro každou kategorii tak, jak to vyžaduje většina algoritmů shlukové analýzy (metod segmentace). Tyto „dummy" proměnné si však uzly pro modelování a shlukovou analýzu pro kategoriální proměnně vytvářejí samy.

5.3.2.3 Standardizace

Třetím krokem přípravy dat je standardizace proměnných, kterou lze provést v uzlu Transform Variables. Ten nabízí celou škálu možností, včetně dosažení normálního rozdělení. V uzlu Clustering je možné také standardizovat, ale nabídka je omezená.

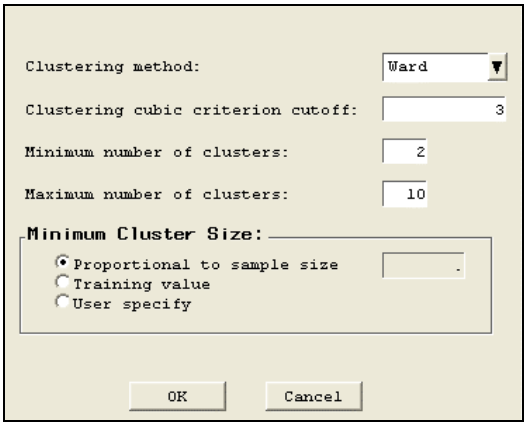


Obrázek 48: Nastavení uzlu Transform Variables

5.3.2.4 Metody segmentace

V první ukázkové úloze je shlukování nastaveno pro trénovací soubor o 2000 pozorováních. Do shlukování je zařazeno celkem pět proměnných, tři nominální (ID v roli „id", DEN a CAS v roli „input") a dvě intervalové (standardizované proměnná Data_In a Data_Out).

Hierarchická fáze se provádí na výběru 2000 objektů z dostupných dat, tento výběr se provádí náhodně s nahrazením. Vzdálenosti se počítají na základě euklidovské vzdálenosti a pro výpočet se používá Wardova metoda. Pro výběr počtu shluků se nastaví kubické shlukovací kritérium CCC = 3 a maximum počtu shluků k = 10. Minimální velikost shluku je úměrná velikosti výběru.

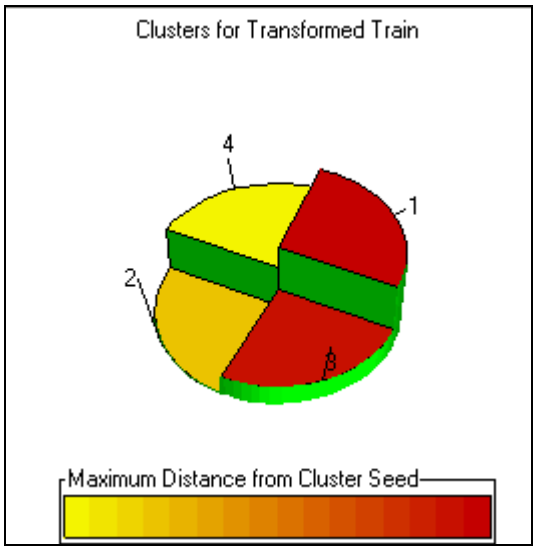


Obrázek 49: Nastavení výpočtu počtu shluků

5.3.2.5 Profilace a interpretace segmentů

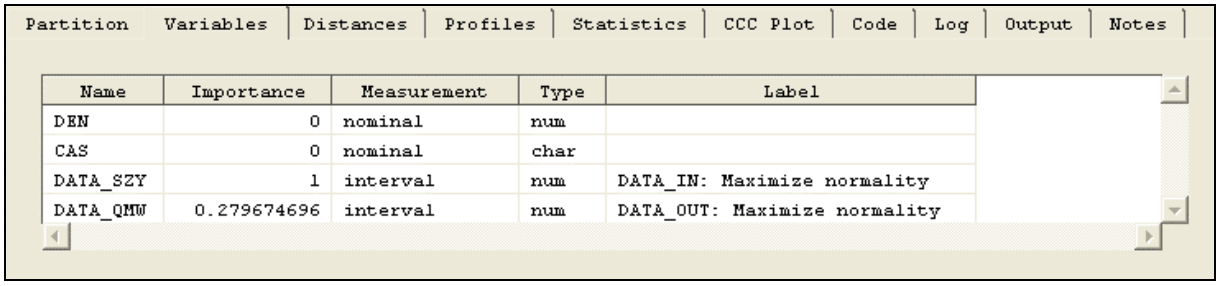
Posledním krokem procesu segmentace je určení profilů a charakteristika odlišností jednotlivých segmentů.

V prezentovaném příkladu byly vytvořeny čtyři skupiny přibližně podobného rozsahu. Následující graf shrnuje tři statistiky každého ze čtyř shluků do jednoho zobrazení. Výška dílku odpovídá počtu pozorování v každém shluku. Nejvíce pozorování je prvním shluku, nejméně pak ve třetím. Šířka dílku vyjadřuje směrodatnou odchylku, konkrétně střední kvadratickou směrodatnou odchylku mezi objekty v daném shluku. Barva vyjadřuje radius, tj. vzdálenost nejvzdálenějšího pozorování od středu shluku.



Obrázek 50: Okno Partitions

Následující tabulka udává relativní významnost proměnných pro určení shluků.



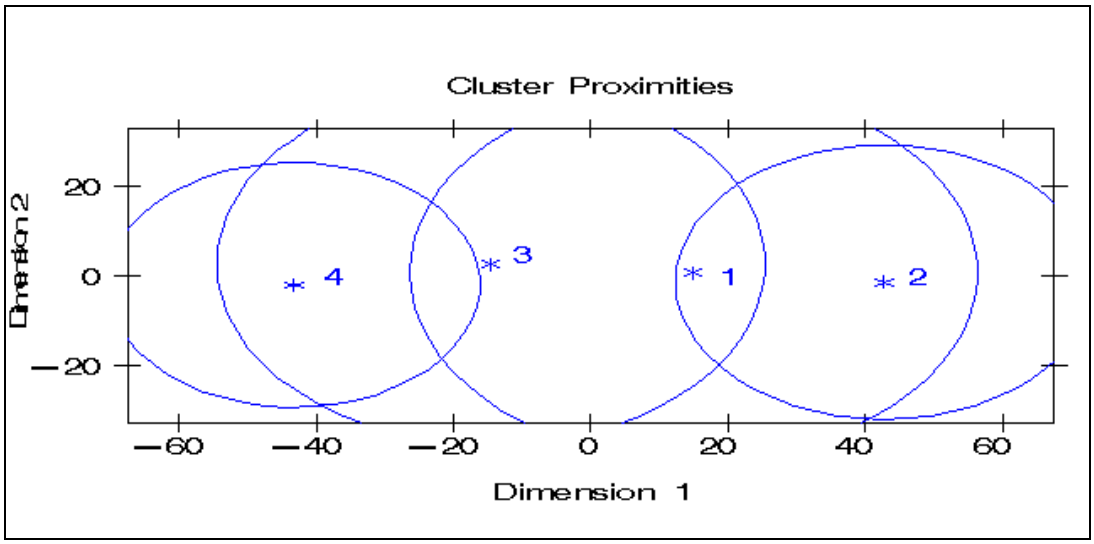
Partition | Variables | Distances | Profiles | Statistics | CCC Plot | Code | Log | Output | Notes

Name	Importance	Measurement	Type	Label
DEN	0	nominal	num	
CAS	0	nominal	char	
DATA_SZY	1	interval	num	DATA_IN: Maximize normality
DATA_QMW	0.279674696	interval	num	DATA_OUT: Maximize normality

Obrázek 51: Okno Variables

Míra důležitosti se vyjadřuje v intervalu od nuly do jedné. Vyšší hodnoty značí vyšší vlivnost, tzn. důležitější proměnnou. Pro rozdělení záznamů do shluků byly nejdůležitější údaje o průměrně odebraném objemu dat (100 %). Proměnná průměrného objemu odeslaných dat byla pro shlukování použita pouze z 28 %.

Okno vzdáleností poskytuje grafické zobrazení velikosti každého shluku a vztahy mezi shluky. Osy jsou určeny pomocí analýzy multidimenzionálního škálování a matice vzdáleností průměry shluků. Hvězdičky jsou středy shluku a kruhy odpovídají rádiím shluků. Radius každého shluku závisí na nejvzdálenějším pozorování v daném shluku. V případě překrytí shluků se jedná pouze o nedokonalost zobrazovací metody, ve skutečnosti je každé pozorování přiřazeno pouze jednomu shluku.



Obrázek 52: Okno Distances

Uzel Clustering automaticky počítá tabulku Statistics se základními charakteristikami shluků. Je z nich např. čitelné, že záznamy jsou rozděleny do segmentů relativně rovnoměrně.

Partition | Variables | Distances | Profiles | Statistics | CCC Plot | Code | Log | Output | Notes

CLUSTER	_FREQ_	Root-Mean-Square Standard Deviation	Maximum Distance from Cluster Seed	Nearest Cluster	Distance to Nearest Cluster
1	1752891	1.9493646191	41.381093582	2	27.8513018
2	1501655	1.8700486094	30.412811879	1	27.8513018
3	1570705	1.8761990615	39.969671282	4	29.090438744
4	1670847	1.7768152171	27.110181423	3	29.090438744

CLUSTER	DATA_IN: Maximize normality	DATA_OUT: Maximize normality	DEN:1	DEN:2	DEN:3	DEN:4
1	61.102330083	40.052696357	0.1442194637	0.1308079053	0.1356273722	0.1637278074
2	86.03630585	52.461904759	0.1338503185	0.138032371	0.1431840203	0.1567317393
3	34.807394252	26.593413004	0.1712734091	0.1301994964	0.1311201021	0.1280259501
4	12.501045768	7.9207437736	0.178796742	0.1426893067	0.1368832694	0.0889740353

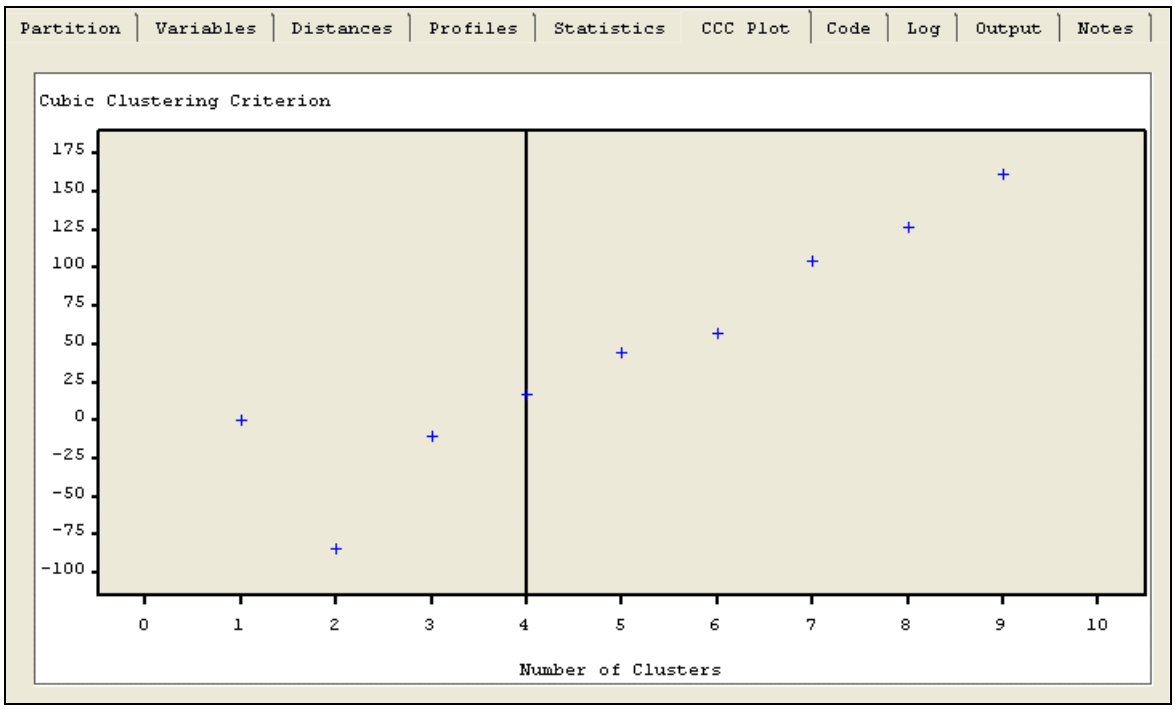
CLUSTER	DEN:5	DEN:6	DEN:7	CAS:00	CAS:01	CAS:02	CAS:03	CAS:04
1	0.1392254282	0.1386988695	0.1476931538	0.0423631589	0.0390554803	0.0377832963	0.0324007597	0.0313333801
2	0.1420832348	0.1441662699	0.1419520462	0.0356266919	0.0285465037	0.0207710826	0.0205233559	0.0170505209
3	0.1345949749	0.1354761079	0.1693099595	0.0548721752	0.0599679762	0.0590072611	0.060083211	0.0572723713
4	0.1420722544	0.1402929173	0.1702914749	0.0489931155	0.0684856244	0.0912980063	0.0947286017	0.1053645247

Obrázek 53: Okno Statistics

Okno Profiles umožňuje prozkoumat tři proměnné najednou, dvě kategoriální a jednu spojitou. Ukázka grafu pro čtyři shluky a analýzy z hlediska kategoriálních proměnných DEN a CAS a spojité proměnné Data_In a Data_Out je uvedena v příloze.

Z grafu je patrné, že chování zákazníků se v daném čase v jednotlivých kalendářních dnech velmi liší ve všech shlucích. Pro druhý shluk jsou typické vysoké hodnoty odebraných dat, pro čtvrtý naopak velmi nízké. Druhému shluku opět dominují vysoké objemy dat na výstupu do internetové sítě a stejně tak čtvrtému shluku dominuje nejnižší množství dat. Největší objem dat se přenese v sobotu od jedenácti do dvanácti.

Graf 9: Kubické shlukovací kritérium CCC



Díky zobrazení výsledných hodnot kritéria CCC je zřejmé, že odfiltrování odlehlých hodnot pomocí uzlu Filter Outliers bylo dostatečné. Pro počet shluků od čtyř výše nabývá kubické shlukovací kritérium kladných hodnot, které stále rostou.

5.3.2.6 Konfirmace výsledků segmentace

Z hlediska obchodní interpretace a optimalizace segmentace jsou čtyři segmenty dobrým výsledkem. Nesmí se však zapomenout na vyřazená odlehlá pozorování, která by měla být brána v potaz jako pátý shluk.

Pro marketingové účely se v dalších krocích nabízí profilace výsledků segmentace tak, že se popíší všechny zajímavé odchylky segmentačních proměnných od průměrného rozložení v celé množině případů.

5.3.2.7 Zhodnocení výsledků

Výsledky segmentace je potřeba zhodnotit především z hlediska splnění cílů definovaných na začátku procesu. Z hlediska dostatečné velikosti segmentů může být shlukování hodnoceno jako přijatelné a vyvážené. Pro uplatnění obchodních cílů firmy a dosažitelnost nástroji marketing mixu jsou výsledky relevantní. Ze shlukování agregovaných záznamů je zřejmá diferencovanost v jednotlivé dny v týdnu a denní doby.

6 DISKUSE

Samotná rozmanitost metod, možnosti jejich použití i množství různých přístupů, které mají posloužit k nalezení zajímavých vztahů anebo ke konstrukci použitelných modelů, s sebou nese mnoho otázek, na něž je třeba nalézat odpovědi. Mezi ně patří různá specifika velkých datových souborů, potřeby přípravy dat a požadavky na statistické postupy včetně obtížnosti volby metod a jejich parametrů.

6.1 Hodnocení přípravy dat

Práce s daty je jak časově, tak i programátorsky náročná. Příprava dat vyžaduje znalost programátorských postupů, vhodných funkcí a příkazů a jejich syntaxe. Zahrnuje řešení zajištění dat (kombinování dat z několika zdrojů), slučování souborů, vhodného uspořádání dat, tvorbu odvozených proměnných, agregace a celé řady dalších témat.

Všechny tyto základní body byly vzaty v úvahu pro navrhování vlastního řešení programu úpravy dat ve formě flat file na soubor, který je vhodný pro další analýzy. Jednotlivé fáze vývoje programu jsou uvedeny v kapitole výsledků disertační práce. Program obsahuje jednotlivé jednoduše oddělitelné části podprogramu, které mohou být opakovaně využity pro podobná řešení přípravy dat.

V rámci tvorby kódu byly vymezeny oblasti, které jsou stěžejní pro přípravu souboru z e-mailových hlášení. Jedná se o následující body zpracování transakčních záznamů:

1) importovat textové soubory do systému,
2) spojit importované soubory,
3) vymazat řádky pocházející z hlavičky e-mailu,
4) zpracovat textový záznam data a času jedné proměnné do dvou proměnných,
5) převést datum na kalendářní matematiku,
6) vytvořit kategorickou proměnnou den v týdnu (neděle – pondělí),
7) snížit dimenzi souboru (odstranit pomocné a nadbytečné proměnné),
8) vypočítat základní charakteristiky pro hodnocení kvality dat,
9) odstranit duplicity v záznamech,
10) odstranit záznamy technického charakteru,
11) sloučit duplicitně vedené zákazníky,
12) zajistit agregace a sumarizace podle kategorických proměnných.

Všechny výše uvedené body lze podle charakteru vstupního souboru zaměnit, nelze však ani jeden z bodů opomenout.

Práce potvrzuje, že získat velký datový soubor není příliš obtížné, náročné je však s tak velkým souborem pracovat a zabezpečit kvalitu dat. Při zpracování výsledků práce bylo ověřeno, že redukce ve smyslu snížení počtu záznamů je v určitých fázích zpracování nezbytným úkonem. Zejména u transakčních dat lze doporučit provedení sumarizace nebo zprůměrování záznamů, ačkoli se tím informační charakter dat snižuje.

6.2 Hodnocení shlukování

Některé ze segmentačních algoritmů umožňují navrhnout optimální počet segmentů v určitém rozsahu, jiné vyžadují pevně zadat počet segmentů předem. Automatické nalezení počtu segmentů může být vhodné pro první přiblížení. V praxi se častěji upřednostňuje ruční nastavení počtu segmentů vzhledem k uplatnění obchodních cílů firmy - „Dobrá segmentace je obchodně užitečná". Je zřejmé, že s rostoucím počtem segmentů neukáže další rozčleňování již žádné obchodně zajímavé odlišnosti. Počet zákazníků v některém shluku může být již natolik malý, že využití takovéhoto segmentu rovněž postrádá marketingově-obchodní smysl.

Volba segmentů je tedy ve shlukové analýze velice obtížně řešitelným úkolem. V práci se lze setkat s aplikací a hodnocením použití dvou přístupů. Jedním je vypracovat celou sérii analýz, a to opakovaně, s různými počty shluků a na základě nejlepších výsledků vybrat optimální počet shluků. Druhým použitým postupem je využít, v rámci datamingového modulu, kritérium pro určení počtu shluků. Oba případy jsou ve své podstatě velmi jednoduché, ale neřeší fakt, jak postupovat, pokud se výsledné řešení odlišuje.

6.2.1 Hodnocení shlukování pomocí programování

Druhou část výsledků práce představovala aplikace shlukovacích metod pomocí programování, jež ověřila různé aspekty použití procedury FASTCLUS. Předmětem práce bylo:

1) segmentování zákazníků na základě průměrného objemu přijatých a odeslaných dat

- ze všech záznamů,
- z nenulových záznamů;

2) segmentování zákazníků na základě sumarizovaných údajů;

3) shluková analýza neagregovaných záznamů.

Poznatky

Na základě realizace shlukování pomocí programování byly získány a potvrzeny následující poznatky:

- efektivní odstranění extrémně odlišujících se pozorování probíhá iterativně,
- počet iterací je kolísavý, nezáleží na zvyšování počtu shluků, ale na výběru středů,
- s rostoucím počtem shluků klesá dosažená hodnota konvergenčního kritéria,
- potvrzení teorie, že jak počet shluků roste, tak stoupá R^2 a zvyšuje se homogenita v rámci shluku,
- standardizace dat pro identifikaci odlehlých pozorování je při stejném měřítku pořízených záznamů nevýznamná.

6.2.2 Hodnocení shlukování v SAS Enterprise Miner

Třetí část výsledků práce byla věnována modulu Enterprise Miner a ověřila aspekty použití uzlu Clustering (shlukování) s cílem:

1) segmentování zákazníků na základě průměrného objemu přenesených dat
   - se zapojením uzlu Transform Variable – logaritmická transformace,
   - se zapojením uzlu Filter Outliers a standardizací v uzlu Clustering,
   - při dalších různých kombinacích;
2) shluková analýza agregovaných údajů dle kategorizovaných proměnných
   - kategorizace pomocí proměnných DEN, CAS, ID - programování,
   - se zapojením uzlu Filter Outliers a uzlu Transform Variable.

Poznatky

Realizace shlukování pomocí modulu Enterprise Miner potvrdila, že pro nalezení rozumného množství segmentů je nutné při použití tohoto nástroje zejména:

- dbát na vhodné sestavení procesního diagramu,
- měnit zadávané parametry a opakovat realizaci analýzy tak dlouho, dokud není nalezeno uspokojivé řešení,
- zabezpečit dostatečný výkon počítače.

6.3 Klíčové momenty realizace úlohy Data mining

Multidisciplinarita je základní charakteristikou technik Data mining, proto používat tyto postupy znamená vybavit se odbornými pracovníky, již mají adekvátní znalosti. K nim patří znalosti z oboru (bankovnictví, telekomunikace, atd.), znalosti práce s daty (programátor, správce datového skladu atd.), dovednost použít metody (statistik), um analyzovat potřeby (analytik) a rozhodovat (manažer).

Na základě průběhu výzkumné práce lze potvrdit, že realizace úlohy Data mining se bez odborníků různých profesí neobejde. Zejména při hodnocení kvality dat a rozhodování, které záznamy ještě mohou a které již nemohou být zařazeny do souboru vhodného pro analýzu dat, je asistence odborného pracovníka z terénu nezbytně nutná.

Na základě realizované výzkumné úlohy bylo detekováno několik klíčových momentů, které jsou předmětem úvah při zpracování transakčních dat. Mezi tyto klíčové body patří:

- jak naložit s odlehlými hodnotami – ponechat × vyřadit,
- co s nulovými záznamy, mají pro analýzu význam či nikoli,
- použít či nepoužít transformaci proměnných, pokud ano, tak jakou,
- zapojit či vynechat kategorické proměnné, jakou jim dát v modelu váhu,
- jaký postup shlukování použít,
- kolik požadovat segmentů,
- jak stanovit minimální počet pozorování v jednom shluku,

- jaké SW prostředí zvolit,
- zda dát přednost programování či modulu navrženému přímo pro Data mining,
- po různém zvolení kritérií a získání různých výsledků vybrat ten „pravý",
- jak správně interpretovat výstupy.

Způsoby interpretace

Problematika interpretace výsledků a tvorba závěrů spočívá nejen ve statistické interpretaci výsledků, ale důležitou roli hraje i věcná interpretace, která obvykle vyžaduje znalosti odborníka v daném oboru.

Pro marketingové využití vybraného segmentu v rámci kampaně je nutné výsledky segmentace ještě prověřit (konfirmovat). Běžné je využití „ručně" stanovených dodatečných konfirmačních pravidel nebo metod rozhodovacího stromu. Odborníci uvádějí [86], že po uplatnění konfirmačních pravidel může dojít ke „ztrátě" například 20 % zákazníků. Realizaci takového profilování je vhodné provést interaktivně s pracovníky odpovědnými za marketing nebo obchod.

7 ZÁVĚR

Předložená práce se zabývala nalezením a ověřením postupů zpracování dat technikou Data mining. Poskytuje základní přehled charakteristik techniky Data mining a vybraných metod, především shlukové analýzy, jejichž uplatnění se v praxi využívá pro segmentační úlohy. Aplikace shlukové analýzy na velké objemy dat vyžaduje při své realizaci řešení některých specifických problémů spojených s přípravou dat, volbou postupu a parametrů shlukování, odstraněním odlehlých pozorování, určením počtu shluků. Možnosti těchto aplikací předznamenávají data, která jsou za tímto účelem shromážděna. Konkrétně je nutné vypracovat vlastní řešení návrhu přípravy dat, jež pocházejí z transakčního zpracování. Dále je nutné zohlednit možnosti programového vybavení a představit výsledky shlukování realizovaného pomocí programování a pomocí modulu pro Data mining. Přestože shluková analýza patří v odborné literatuře k velmi často uváděným metodám vhodným pro segmentační úlohy, jejím aplikacím na transakční data velkého objemu není dán dosud dostatečný prostor. Tato práce představuje v tomto ohledu původní příspěvek, pro jehož řešení jsou získána originální, dosud nikde nezpracovaná a nepublikovaná data.

K metodám zpracování je nutno uvést, že aspekty použitého programového vybavení hrají velkou roli a determinují úroveň a možnosti zpracování. Vybraný prostředek – systém SAS – je v tomto ohledu adekvátní, je schopen zajistit jak vlastní přípravu dat, tak i prostředí pro statistické zpracování. K němu bylo využito obou nabízených variant, jednak programování, jednak modul pro techniky Data mining.

Pro naplnění cílů, zformulovaných v kapitole 2, byla celá studie realizována na souboru dat pocházejících z monitorovacího systému lokálního poskytovatele internetu. K výsledkům práce patří návrh vlastního postupu řešení přípravy dat z transakčních údajů o zákaznících. Jedná se o originální řešení, kdy předmětem činnosti navrženého kódu programu bylo zpracovat soubor s jedenácti milióny záznamů.

Návrh řešení zahrnuje zajištění dat (kombinování dat z několika zdrojů), slučování souborů, vhodného uspořádání dat, tvorbu odvozených proměnných, agregace, deduplikaci záznamů a celé řady dalších témat. Jednotlivé úpravy dat z formy flat file na soubor, který je vhodný pro další analýzy, jsou opakovatelně použitelné i pro jiné soubory. Jednotlivé fáze vývoje programu jsou uvedeny v kapitole výsledků disertační práce.

Příprava transakčních dat zahrnuje i volbu vhodné agregátní úrovně. Ukazuje se, že možností se nabízí několik. Jednou z nich je volba výpočtu průměrů, a to jak z úplných záznamů, tak pouze z nenulových hodnot. Další z možností je výpočet součtů, zjištění maximálně dosažených hodnot apod. Úroveň agregace hraje také důležitou roli, je nutné zvážit, zda agregovat pouze za zákazníky celkem či i dle kategorických proměnných, např. DEN (v týdnu) či CAS (hodina během dne). Na základě charakteru prezentovaných dat lze doporučit řešení, které se osvědčilo: používat agregáty založené na průměru úplného souboru (včetně nulových záznamů) a pro „rychlé" shlukování volit agregaci za zákazníka.

Jednou z vlastností shlukové analýzy je, že umí identifikovat extrémní odchylky vícerozměrných veličin. Z uvedených příkladů je zřejmé, že při řešení segmentační úlohy je nutné tuto vlastnost brát v úvahu a při shlukování přistupovat k opakovaným řešením stanovení úrovně, kdy je ještě pozorování odlehlé a kdy již ne. Práce přinesla zajímavý a důležitý poznatek – vícenásobnost resp. opakovaná řešení jsou typickým rysem při přípravě dat. Další významným přínosem práce je zhodnocení vlivu standardizace při

shlukování. Standardizace dat pro shlukovou analýzu je v literatuře často doporučovaná, avšak u proměnných z transakčních dat stejných jednotek a řádů nenachází pro identifikaci odlehlých pozorování patřičné uplatnění. Na druhou stranu, transformace dat za účelem nalezení segmentů bez identifikace odlehlých pozorování je téměř vždy nezbytná.

Dalším cílem práce bylo zhodnotit možnosti používání shlukovacích metod pomocí programování a pomocí speciálního modulu pro Data mining. Hlavní výhodou programování je jeho variabilita a jednoznačnost při zadávání parametrů. Práce se zabývala výsledky shlukování při změnách různých vybraných parametrů a naznačila řešení problémů, které jsou v odborné literatuře zmiňovány často neurčitě respektive rozdílně. Klíčovým výsledkem práce je zjištění, že počet iterací je značně kolísavý - v realizovaných úlohách nebyla pro optimální rozdělení pozorování do shluků dostačující pouze jedna iterace. Nevýhodou programování je, že je velmi náročné na znalosti uživatele (nutnost studia pro zajištění výběru správné procedury, konstrukce příkazů a volby parametrů). K výhodám modulu Enterprise Miner patří především jeho příjemné uživatelské rozhraní vhodné pro začátečníka. K dalším kladům patří využití celé řady nástrojů v podobě uzlů. Při shlukování se osvědčilo použít pro přípravu dat především uzel Filter Outliers a uzel Transform Variables, který na rozdíl od uzlu Clustering (nabízí pouze dvě transformace) nabízí celou řadu transformací proměnných. Z dalších nevýhod použití modulu Enterprise Miner lze na základě praktické zkušenosti jmenovat zejména: větší náročnost na výkon počítače; vytvoření výsledného řešení, ačkoli během výpočtu došlo k chybám či nenaplnění předpokladu.

Do metodologie Data mining se řadí celá škála metod a postupů. Za posledních více než deset let jim bylo věnováno mnoho diskusí, odborných textů, workshopů, seminářů a konferencí. Je to téma, které nelze ve světě zpracování dat a statistiky opominout. Tomu odpovídá i vývoj softwaru, zvláště u význačných společností produkujících statistický software. I v oblasti databázových produktů se lze setkat se zařazováním nových nebo znovu objevených, vylepšených výpočetních postupů. V současné době se pojem Data mining z větší části snoubí s užitím řešení tzv. „Business inteligence". Zejména komerční sféra se svojí potřebou analyzovat velké objemy dat představuje tradiční základnu uživatelů dataminingových produktů. V poslední době roste zájem a potřeba využívat tyto moderní přístupy i v menších firmách, neboť i ty „skladují" ve svých firemních databázích velké objemy dat. Tato práce může být návodná pro uplatnění těchto technik v praxi a přispět k jejich rozvoji.

Práce shrnuje některé výsledky studijní, pedagogické, výzkumné a praktické činnosti podpořené grantem interní grantové agentury PEF ČZU v oblasti shlukových metod v rámci dataminingových technik.

8 POUŽITÁ LITERATURA A ZDROJE

[1] ARNOŠT, Daniel. Co je to CRM? *Doly na data o vztazích s klienty*, 2001, str. I a III. Příloha měsíčníku Bankovnictví č. 2/2001 a týdeníku Ekonom 8/2001.

[2] BERKA, Petr. Data mining. *Statistika*, 2003, roč. 83, č. 1, str. 2 – 25. 83. ročník řady odborných ekonomicko-statistických časopisů statistické služby ČR. ISSN 0322-788x

[3] BERKA, Petr. *Dobývání znalostí z databází.* 1. vyd. Praha: Akademia, 2003. 366 str. ISBN 80-200-1062-9.

[4] BERKHIN, Pavel. *Survey of Clustering Data Mining Techniques* [online]. Accrue Software, 2002. [citováno 26. červen 2006]. Dostupné z http://www.ee.ucr.edu/~barth/EE242/clustering_survey.pdf

[5] BERRY, Michael J. A. – LINOFF, Gordon. *Data Mining Techniques : For Marketing, Sales, and Customer Support.* 1st edition. USA: John Wiley & Sons, 1997. 444 p. ISBN 0-471-17980-9.

[6] BÍNOVÁ, Dagmar. A criterion for determining the number of clusters for cluster analysis in the Enterprise Miner of the SAS system. In *Sborník příspěvků z doktorandského semináře 2004*. 1. vyd. Praha: ČZU PEF, 2004.

[7] BÍNOVÁ, Dagmar. Charakteristika metodologie Data mining. In *Sborník příspěvků z doktorandského semináře 2002*. Vydání 1. Praha: Credit, 2002. ISBN 80-213-0880-X.

[8] BÍNOVÁ, Dagmar. Principy technik Data Mining v systému SAS. In *Sborník z mezinárodního odborného semináře „Statistické programové systémy SAS a Statistica ve vědě, výzkumu a výuce", v Praze 26. listopadu 2003*. Vydání 1. Praha: ČZU PEF, 2004.

[9] BÍNOVÁ, Dagmar. Příprava dat pomocí programování v systému SAS. In *Sborník z mezinárodního odborného semináře „Statistické programové systémy SAS a Statistica ve vědě, výzkumu a výuce", v Praze 26. listopadu 2003*. Vydání 1. Praha: ČZU PEF, 2004.

[10] BÍNOVÁ, Dagmar. Úvod do práce s daty pomocí programování v systému SAS. In *Kvantitatívne metódy v ekonómii, metodologické aspekty výskumu v období vstupu do EU.* 1. vyd. Nitra (Slovenská republika): Slovenska Poľnohospodárska Univerzita, 2003. ISBN 80-8069-299-8

[11] BÍNOVÁ, Dagmar. Využití shlukové analýzy jako jedné z dataminingových metod v telekomunikacích. In *Sborník příspěvků z doktorandského semináře 2003*. 1. vyd. Praha: ČZU PEF, 2003. ISBN 80-213-1016-2

[12] BOUDAILLIER, Eric – HÉBRAIL, Georges. Interactive Interpretation of Hierarchical Clustering. In *Principles of Data Mining and Knowledge Discovery : First European Symposium, PKDD'97 in Trondheim, Norway, June 1997.* 1st edition. Berlin: Springer, 1997. 288-298 p. ISBN 3-540-63223-9

[13] BRABENEC, Vladimír – KÁBA, Bohumil – MACHÁČEK, Otakar. Modern data processing and statistical data analysis. In *Agric. Econ*. Praha: ČAZV, 2001, vol. 47, no. 10, p. 433 – 439. CS ISSN 0139570X

[14] BRABENEC, Vladimír – ŠAŘECOVÁ, Pavla. *Statistické metody v marketingu a obchodu : Vybrané přednášky a příklady.* 1. vyd. Praha: Credit, 2001. 134 str. Skriptum. ISBN 80-213-0747-1

[15] BRABENEC, Vladimír. Metody vícerozměrné statistické analýzy jako nástroj poznání vlastností dat. In *Zpracování dat a matematické modelování v zemědělství : Sborník příspěvků ze semináře kateder statistiky a operační a systémové analýzy*. 1. vyd. Praha: Credit, 15. prosince 2000. Str. 7-16. ISBN 80-213-0706-4.

[16] ČÁBELA, Miroslav. Řízení vztahů a hodnoty zákazníka (CRM jako nástroj pro CVM). In *Moderní řízení.* Měsíčník Hospodářských novin. 2002, ročník 37, číslo 5, str. 18 – 20. ISSN 0026-8720.

[17] ČERMÁKOVÁ, Anna. Příspěvek ke stanovení počtu shluků při shlukování pomocí nehierarchických metod. In *Sborník vědeckých prací z mimořádného setkání kateder statistiky a operačního výzkumu 2002 : Vydáno ku příležitosti oslav 50. výročí vzniku PEF ČZU v Praze*. 1. vyd. Praha: Česká zemědělská univerzita, 2002. Str. 38-42. ISBN 80-213-0921-0.

[18] DASU, Tamraparni – JOHNSON, Theodore. *Exploratory Data Mining and Data Cleaning.* First edition. USA: John Wiley & Sons, 2003. 131 p. ISBN 0-471-26851-8

[19] DIGIMINE. *Call for papers : Data Mining and Knowledge Discovery* [online]. [citováno 17. leden 2002]. Dostupné z: http://www.digimine.com

[20] *Dolování dat* [online]. [citováno 19. červen 2001]. Dostupné z: http://kit.vse.cz/cs/knihovna/tekit01.htm#d

[21] DOSTÁL, Michal. Segmentace v softwaru SAS Enterprise Miner. *Data Mining Magazine*, 2003, roč. 1, č. 3, str. 5 - 7. Adastra Corporation.

[22] DUFEK, Jaroslav. Shluková analýza demografického vývoje v krajích České republiky za rok 1998. In *Sborník příspěvků z Mezinárodní vědecké konference kateder statistiky a systémové a operační analýzy „Nové směry vědecko-výzkumné a pedagogické činnosti v oblasti kvantitativních metod" konané v Poděbradech 20.-22. září 1999*. 1. vyd. Praha: CREDIT, 1999. Str. 100-106. ISBN 80-213-0561-4.

[23] *Faq around Data Mining* [online]. [citováno 14. prosinec 2001]. Dostupné z: http://www.web-datamining.net

[24] GIUDICI, Paolo. *Applied Data Mining : statistical methods for business and industry*. 4th edition. USA: John Wiley & Sons, 2003. 356 p. ISBN 0-470-84679-8.

[25] HALKIDI, Maria – VAZIRGIANNIS, Michalis. A Data Set Oriented Approach for Clustering Algorithm Selection. In *Principles of Data Mining and Knowledge Discovery : 5th European Symposium, PKDD 2001 in Freiburg, Germany, September 2001.* 1st edition. Berlin: Springer, 2001. 165-179 p. ISBN 3-540-42534-9.

[26] HANYŠ, Marek. Některá specifika analýzy sekundárních dat. *Statistika*, 2003, roč. 83, č. 1, str. 46 – 52. 83. ročník řady odborných ekonomicko-statistických časopisů statistické služby ČR. ISSN 0322-788x

[27] HEBÁK, Petr – HUSTOPECKÝ, Jiří. *Vícerozměrné statistické metody s aplikacemi.* Vydání první. Praha: SNTL – Alfa, 1987. 452. str. DT 519.237:33(075.8). 04-323-87.

[28] HEBÁK, Petr a kol. *Vícerozměrné statistické metody [1].* Vydání první. Praha: Informatorium, 2004. 239 str. ISBN 80-7333-025-3.

[29] HEBÁK, Petr a kol. *Vícerozměrné statistické metody [3].* Vydání první. Praha: Informatorium, 2005. 255 str. ISBN 80-7333-039-3.

[30] HENDL, Jan. *Přehled statistických metod zpracování dat : Analýza a metaanalýza dat.* Vydání 1. Praha: Portál, 2004. 584 str. ISBN 80-7178-820-1

[31] JOBSON, J.D. *Applied Multivariate Data Analysis*. USA: Springer-Verlag, 1992. 714 p, 1 diskette. Volume II: Categorical and Multivariate Methods. ISBN 0-387-97804-6.

[32] JOHNSON, Richard A. – WICHERN, Dean W. *Applied Multivariate Statistical Analysis*. 4th edition. USA: Prentice-Hall, 1998. 808 p. ISBN 0-13-834194-X.

[33] KÁBA, Bohumil. Metodologické otázky hodnocení kvality dat. In *Kvantitatívne metódy v ekonómii : metodologické a praktické aspekty výskumu v období vstupu do EÚ.* 1. vyd. Nitra (Slovenská republika): Slovenska Poľnohospodárska Univerzita, 2003. ISBN 80-8069-299-8.

[34] KÁBA, Bohumil. Statistické aspekty analýzy rozsáhlých datových souborů. In *Metody statistické analýzy dat : Sborník příspěvků k první etapě výzkumného záměru „Zpracování dat a matematické modelování v zemědělství".* 1. vyd. Praha: Credit, 1999. Str. 7-17. ISBN 80-213-0568-1.

[35] KD Nuggets. [online]. Dostupné z: http://www.kdnuggets.com

[36] KOČKA, Tomáš. Organizace projektu segmentace. *Data Mining Magazine*, 2004, roč. 2, č. 2, str. 6 - 8. Adastra Corporation.

[37] KOČKA, Tomáš. Support Vector Machines. *Data Mining Magazine*, 2004, roč. 2, č. 1, str. 9 - 12. Adastra Corporation.

[38] KUPKA, Karel. Poznámky. *Statistika*, 2003, roč. 83, č. 1, str. 26 – 30. 83. ročník řady odborných ekonomicko-statistických časopisů statistické služby ČR. ISSN 0322-788x

[39] LUKASOVÁ, Alena – ŠARMANOVÁ, Jana. *Metody shlukové analýzy.* 1. vydání. Praha: SNTL, 1985. 210 str. DT 519.681 + 519.25

[40] MÁRA, Lubor. *Doly na data* [článek na CD ROM]. CHIP 1998, č. 11.

[41] MÁŠA, Petr. *Data Mining : Metody dataminingu*. Softwarové noviny, 9/2004. [online]. Dostupné z: http://www.adastracorp.sk/

[42] MELOUN, Milan – MILITKÝ, Jiří. *Kompendium statistického zpracování dat : Metody a řešené úlohy včetně CD.* Vydání první. Praha: ACADEMIA, 2002. 764 s., 1 CD-ROM. ISBN 80-200-1008-4

[43] MELOUN, Milan – MILITKÝ, Jiří – HILL, Martin. *Počítačová analýza vícerozměrných dat v příkladech.* Vydání první. Praha: ACADEMIA, 2005. 448 s., 1 CD-ROM. ISBN 80-200-1335-0

[44] MOHELSKÁ, Libuše. Pan marketing – názory Philipa Kotlera na aktuální trendy v marketingu. In *E-biz.* Květen 2002, vychází měsíčně, str. 33 - 35. ISSN 1213-063X

[45] NOVOTNÝ, Ota – POUR, Jan – SLÁNSKÝ, David. *Business intelligence : jak využít bohatství ve vašich datech.* 1. vyd. Praha: Grada Publishing, 2005. 254 s. ISBN 80-247-1094-3

[46] PERNER, Petra. *Data Mining on Multimedia Data.* First edition. Berlin (Germany): Springer, 2002. 131 p. ISBN 3-540-00317-7

[47] PIRKL, David. Neuronové sítě určené pro predikční úlohy. *Data Mining Magazine*, 2003, roč. 1, č. 2, str. 4 - 7. Adastra Corporation.

[48] PIRKL, David. Rozhodovací stromy a výhody jejich implementace v produktu Angoss KnowledgeSTUDIO. *Data Mining Magazine*, 2003, roč. 1, č. 1, str. 4 - 6. Adastra Corporation.

[49] PIRKL, David. Segmentace s využitím Kohonenových neuronových sítí. *Data Mining Magazine*, 2003, roč. 1, č. 3, str. 8 - 10. Adastra Corporation.

[50] PITTNER, Kamil. Záchranný kruh data miningu. In *Business World*. 7-8/2006. 23. srpen 2006 [citováno 8. září 2006]. Dostupné na http://www.businessworld.cz/bw.nsf/temata

[51] *Poklady v kupě dat* [článek na CD ROM]. CHIP 1995, č. 12.

[52] PRYKE, Andy. *Introduction to Data Mining* [online]. [citováno 14. prosince 2001] Dostupné z: http://www.andypryke.com/university/dm_docs/dm_intro.html nebo http://www.the-data-mine.com/bin/view/Misc/IntroductionToDataMining

[53] PULPÁN, Jaroslav. *SAS Enterprise MinerTM - řešení dolování dat v reálném podnikovém prostředí* [online]. 1. červen 1999 [citováno 7. březen 2004]. Dostupné na http://www.inforum.cz/inforum99/pulpan/

[54] RUD, Olivia Parr. *Data mining: praktický průvodce dolováním dat pro efektivní prodej, cílený marketing a podporu zákazníků (CRM).* Z angličtiny přeložil Ivo Magera a Milan Daněk. 1. vyd. Praha: Computer Press, 2001. 329 s, 1 CD-ROM. Rychle a jistě, databáze. ISBN 80-7226-577-6.

[55] ŘEZANKOVÁ, Hana. Postupy používané při analýze dat. In *Banka dat a modelů ekonomiky ČR* [online]. [citováno 19. červen 2001, 6. březen 2004]. Dostupné z http://badame.vse.cz/clanky/analyza-dat.php

[56] ŘEZANKOVÁ, Hana. Použití statistických metod. *Statistika*, 2003, roč. 83, č. 1, str. 2. 83. ročník řady odborných ekonomicko-statistických časopisů statistické služby ČR. ISSN 0322-788x

[57] SAARENVIRTA, Gary. Data mining jako proces. *Data Mining Magazine*, 2003, roč. 1, č. 1, str. 2. Adastra Corporation.

[58] SARLE, Warren S. *The Number of Clusters* [online]. [citováno: 2. leden 2004]. URL http://www.pitt.edu/~wpilib/clusfaq.html

[59] SARLE, Warren S. SAS *Technical Report A-108: Cubic Clustering Criterion.* 2nd printing. USA: SAS Institute, 1998 (1983). ISBN 1-58025-185-4.

[60] SAS Institute. *Applying Data Mining Techniques Using Enterprise Miner.* 305 p. USA: SAS Institute, 2002. Course Notes.

[61] SAS Institute. *Business Intelligence* [on-line]. [citováno 2. květen 2006]. Dostupné z http://www.sas.com/offices/europe/czech/technologies/bi/index.html

[62] SAS Institute. Enterprise Miner Software: Clustering Node. In *Sas System Help*. Cary: SAS Institute Inc., 2003.

[63] SAS Institute. *New Features in Version 8e of the SAS System.* 54 p. USA: SAS Institute, 2001. Course Notes.

[64] SAS Institute. PROC CLUSTER, PROC FASTCLUS. In *SAS System Help*. Cary: SAS Institute Inc., 2003.

[65] SAS Institute. *SAS Enterprise MinerTM* [on-line]. [citováno 7. březen 2004]. Dostupné z http://www.sas.com/technologies/analytics/datamining/miner/

[66] SAS Institute. *SAS System Programming Approach.* 275 p. USA: SAS Institute. Course Notes.

[67] SAS Institute. *SAS/Base SAS* [on-line]. [citováno 3. září 2003]. Dostupné z http://www.sas.com/technologies/bi/appdev/base/

[68] SAS Institute. *SAS/STAT: User's Guide*. 4th edition. USA: SAS Institute, 1994. 943 p. Volume 1: ACECLUS-FREQ. ISBN 1-55544-376-1

[69] SAS Institute. *Technical Support* [on-line]. [citováno 2. leden 2003]. Dostupné z http://www.sas.com/service/techsup/intro.html

[70] SAS Institute. *Turn Raw Data into Business Gold with Data Mining* [on-line]. [citováno 14. prosinec 2001]. Dostupné z http://www.sas.com/technologies/data_mining/index.html

[71] SAS Institute. *What's New in SAS Enterprise Miner 5.1* [on-line]. [citováno 8. červenec 2006]. http://support.sas.com/software/91x/emgui51whatsnew900.htm

[72] SKALSKÁ, Hana. Statistika, analýza dat a znalostní management. *Statistika*, 2003, roč. 83, č. 1, str. 39 – 45. 83. ročník řady odborných ekonomicko-statistických časopisů statistické služby ČR. ISSN 0322-788x

[73] SPOUSTA, Jan. Data mining, predikční úlohy a logistická regrese. *Data Mining Magazine*, 2003, roč. 1, č. 2, str. 7 - 12. Adastra Corporation.

[74] SPSS. *Clementine - Data mining, Predictive models* [online]. [citováno 24. březen 2004]. http://www.spss.com/clementine/

[75] SPSS. *Služby a data mining* [online]. [citováno 7. březen 2004]. Dostupné na http://www.spss.cz/sl_datamining.html#crisp

[76] STANKOVIČOVÁ, Iveta – FICOVÁ, Jana. Systém SAS na univerzitách. In *Sborník z mezinárodního odborného semináře „Statistické programové systémy SAS a Statistica ve vědě, výzkumu a výuce", v Praze 26. listopadu 2003*. Vydání 1. Praha: ČZU PEF, 2004.

[77] STATISTICKÝ SYSEL. *Zpráva o datamining* [online]. [citováno 14. prosinec 2001]. Dostupné z: http://ssysel.hyperlink.cz/data/datamin/zprava.htm

[78] STATSOFT. *Electronic Statistics Textbook*. Tulsa, OK: StatSoft, 1999. WEB: http://www.statsoft.com/textbook/stathome.html

[79] STATSOFT. *Unique Features of STATISTICA Data Miner* [online]. [citováno 30. duben 2006]. Dostupné z: http://www.statsoft.cz/page/index2.php?dm_popis

[80] STATSOFT. *Vytěžování dat a systém Statistica* [online]. [citováno 19. červen 2001]. Dostupné z: http://www.statsoft.cz/dmining/dmining.html

[81] SVATOŠOVÁ, Libuše – HŘÍBAL, Jan – VOLMA, Marián. *Systém SAS : Příručka pro uživatele.* 90 str. Praha: CREDIT, 2000. Skriptum PEF ČZU. ISBN 80-213-0597-5

[82] SVATOŠOVÁ, Libuše. Metody vícerozměrné statistické analýzy. In *Metody statistické analýzy dat : Sborník příspěvků k první etapě výzkumného záměru „Zpracování dat a matematické modelování v zemědělství"*. 1. vyd. Praha: Credit, 1999. Str. 21-27. ISBN 80-213-0568-1.

[83] SVATOŠOVÁ, Libuše. Využití vícerozměrných statistických metod při zpracování velkých datových souborů. In *Zpracování dat a matematické modelování v zemědělství : Sborník příspěvků ze semináře kateder statistiky a operační a systémové analýzy.* 1. vyd. Praha: Credit, 15. prosince 2000. Str. 42-49. ISBN 80-213-0706-4.

[84] ŠÁLY, Martin. Analýza a predikce ztráty zákazníků. *Data Mining Magazine*, 2003, roč. 1, č. 2, str. 2 - 4. Adastra Corporation.

[85] ŠÁLY, Martin. Příprava dat pro data mining. *Data Mining Magazine*, 2004, roč. 2, č. 2, str. 2 - 6. Adastra Corporation.

[86] ŠÁLY, Martin. Segmentace a data mining. *Data Mining Magazine*, 2003, roč. 1, č. 3, str. 2 - 4. Adastra Corporation.

[87] ŠÁLY, Martin. Two Step Clustering – první zkušenosti s implementací v produktu SPSS 11.5. *Data Mining Magazine*, 2003, roč. 1, č. 1, str. 7 - 8. Adastra Corporation.

[88] TALAVERA, Luis – BÉJAR, Javier. Efficient Construction of Comprehensible Hierarchical Clusterings. In *Principles of Data Mining and Knowledge Discovery : Second European Symposium, PKDD'98 in Nantes, France, September 1998.* 1st edition. Berlin: Springer, 1998. 93-101 p. ISBN 3-540-65068-7.

[89] *Vydolujte peníze z informací* [článek na CD ROM]. CHIPWeek 1998, č. 45.

[90] WESSLING, Harry. *Aktivní vztah k zákazníkům pomocí CRM : Strategie, praktické příklady a scénáře.* Vydání první. Praha: GRADA, 2003. 196 s. Manažer. ISBN 80-247-0569-9

[91] ZAVORAL, Petr. Data jako zdroj konkurenční výhody [online]. In *Moderní řízení*. 17. července 2006 [citováno 8. září 2006]. Dostupné na http://ihned.cz/2-18860750-000000_d-f8

9 SEZNAM OBRÁZKŮ

Obrázek 1: Schéma procesu Data mining [57] 9
Obrázek 2: Metodika SEMMA [3] 12
Obrázek 3: Vizuální programování v produktu Clementine od SPSS 13
Obrázek 4: Metodologie CRISP-DM 14
Obrázek 5: Data Miner - sekvence kroků 15
Obrázek 6: Základní skupiny metod shlukové analýzy [39] 45
Obrázek 7: Prostředí modulu Enterprise Miner 81
Obrázek 8: Seznam ikon uzlů 82
Obrázek 9: Ikony Sample 82
Obrázek 10: Ikony Explore 84
Obrázek 11: Ikony Modify 85
Obrázek 12: Ikony Model 87
Obrázek 13: Ikony Assess 88
Obrázek 14: Ikony Score 89
Obrázek 15: Pomocné uzly 89
Obrázek 16: Shlukování – okno Data – karta Inputs 93
Obrázek 17: Shlukování – okno Data – karta Preliminary Training and Profiles 94
Obrázek 18: Shlukování – okno Variables 94
Obrázek 19: Shlukování – okno Clusters 95
Obrázek 20: Shlukování –okno Clusters – Selection Criterion 96
Obrázek 21: Shlukování – okno Seeds – karta General 99
Obrázek 22: Shlukování – okno Seeds – karta Initial 99
Obrázek 23: Shlukování – okno Seeds – karta Final 100
Obrázek 24: Shlukování – okno Missing Values 101
Obrázek 25: Shlukování – okno Output – karta Clustered Data 102
Obrázek 26: Shlukování – okno Output – karta Statistics Data Sets 103
Obrázek 27: Výsledky shlukování graficky na kartě Distances 104
Obrázek 28: Výsledky shlukování v číslech na kartě Statistics 105
Obrázek 29: Výsledky shlukování na kartě CCC Plot 106
Obrázek 30: Vzhled importované e-mailové hlavičky a transakčních dat 108
Obrázek 31: Struktura dat po odmazání e-mailové hlavičky 109
Obrázek 32: Vytvoření nové proměnné „Cas" 110
Obrázek 33: Vytvoření nové proměnné „Datum2" 110
Obrázek 34: Vytvoření nové proměnné „Cas3" a redukce proměnných 111

Obrázek 35: Přejmenování proměnných 111
Obrázek 36: Nová proměnná uvádějící datum – Date1 112
Obrázek 37: Shluková analýza pro průměrný objem přenesených dat – 2 skupiny zákazníků..... 127
Obrázek 38: Shluková analýza pro průměrný objem přenesených dat – 4 skupiny zákazníků..... 129
Obrázek 39: Shluková analýza pro průměrný objem přenesených dat – 11 skupin zákazníků 130
Obrázek 40: Shluková analýza pro průměrný objem přenesených dat – 9 skupin zákazníků 131
Obrázek 41: Diagram pro shlukování dat..... 142
Obrázek 42: Logaritmická transformace v uzlu Transform Variables..... 143
Obrázek 43: Filtrování 9% MAD v uzlu Filtr Outliers..... 144
Obrázek 44: Dva soubory: filtrovaná data × pozorování s extrémními hodnotami 144
Obrázek 45: Segmentace zákazníků v Enterprise Miner - pět skupin 145
Obrázek 46: Diagram pro shlukování dat..... 146
Obrázek 47: Nastavení uzlu Filter Outliers 146
Obrázek 48: Nastavení uzlu Transform Variables..... 147
Obrázek 49: Nastavení výpočtu počtu shluků..... 148
Obrázek 50: Okno Partitions..... 148
Obrázek 51: Okno Variables..... 149
Obrázek 52: Okno Distances 149
Obrázek 53: Okno Statistics 150
Obrázek 54: Výsledek shlukování - okno Variables 173
Obrázek 55: Výsledek shlukování - okno Distances..... 173
Obrázek 56: Výsledek shlukování - okno Profiles - proměnná Data_In..... 174
Obrázek 57: Výsledek shlukování - okno Profiles - proměnná Data_Out 174
Obrázek 58: Výsledek shlukování - okno Statistics..... 175
Obrázek 59: Výsledek shlukování - okno Code 175

10 SEZNAM TABULEK

Tabulka 1: Některé rozdíly mezi statistikou a procesy DM - KDD [72]....8
Tabulka 2: Systémy pro Data mining [upraveno podle 3]....11
Tabulka 3: Přehled úloh a metod při technikách Data mining [55]....16
Tabulka 4: Vztah mezi LEAST a MAXITER....79
Tabulka 5: Základní struktura uzlu Clustering....92
Tabulka 6: Přehled karet v okně Results Browser (prohlížeč výsledků) pro uzel Clustering....103
Tabulka 7: Základní charakteristika neočištěných dat....113
Tabulka 8: Základní statistiky pro Data_In podle času....119
Tabulka 9: Základní statistiky pro Data_Out podle času....120
Tabulka 10: Základní statistiky pro Data_In podle dne v týdnu....121
Tabulka 11: Základní statistiky pro Data_Out podle dne v týdnu....121
Tabulka 12: Porovnání počtu iterací při shlukování průměrů úplných záznamů....124
Tabulka 13: Porovnání počtu iterací při shlukování průměrů nenulových záznamů....124
Tabulka 14: Hodnoty konvergenčního kritéria při shlukování průměrů úplných záznamů....124
Tabulka 15: Hodnoty konvergenčního kritéria při shlukování průměrů nenulových záznamů....124
Tabulka 16: Hodnoty kritérií při shlukování průměrů z úplných dat a se standardizací....125
Tabulka 17: Hodnoty kritérií při shlukování průměrů z nenulových dat a se standardizací....126
Tabulka 18: Identifikace odlehlých pozorování na základě různého počtu shluků....132
Tabulka 19: Hodnotící kritéria při shlukování sumarizovaných záznamů po odstranění 8 zákazníků....137
Tabulka 20: Hodnotící kritéria při shlukování sumarizovaných záznamů po odstranění 14 zákazníků....137
Tabulka 21: Hodnotící kritéria při shlukování sumarizovaných záznamů po odstranění 23 zákazníků....137
Tabulka 22: Počet iterací při shlukování sumarizovaných záznamů po odstranění odlehlých pozorování....137

11 SEZNAM GRAFŮ

Graf 1: Vztah kritéria CCC a počtu shluků u úplného souboru 125
Graf 2: Vztah kritéria CCC a počtu shluků u souboru nenulových dat 126
Graf 3: Dva shluky - Shlukování průměrů z úplných dat a se standardizací 128
Graf 4: Čtyři shluky - Shlukování průměrů z úplných dat a se standardizací 129
Graf 5: Devět shluků - Shlukování průměrů z nenulových záznamů a se standardizací 131
Graf 6: Pět shluků - Shlukování součtů po vyloučení 8 zákazníků – 1. vlna 133
Graf 7: Pět shluků - Shlukování součtů po vyloučení 14 zákazníků – 2. vlna 134
Graf 8: Čtyři shluky - Shlukování součtů po vyloučení 23 zákazníků – 3. vlna 136
Graf 9: Kubické shlukovací kritérium CCC 150
Graf 10: Analýza shluků dle proměnné DEN, CAS a Data_In v okně Profiles 176
Graf 11: Analýza shluků dle proměnné DEN, CAS a Data_Out v okně Profiles 177

12 PŘÍLOHY

12.1 Výsledek shlukování průměrů pomocí programování

12.1.1 Shlukování průměrů úplných záznamů

Cluster Summary

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	784	0.3665	4.6977	2	13.0047
2	7	4.0442	9.6706	1	13.0047
1	784	0.3665	4.6977	3	9.9132
2	2	2.0124	2.0124	3	10.8645
3	5	1.5628	2.6547	1	9.9132
1	765	0.1489	1.4990	4	2.8071
2	2	2.0124	2.0124	3	10.8645
3	5	1.5628	2.6547	4	7.2168
4	19	0.9189	1.9979	1	2.8071
1	765	0.1489	1.4990	2	2.8071
2	19	0.9189	1.9979	1	2.8071
3	2	2.0124	2.0124	5	10.7726
4	2	1.0213	1.0213	5	2.7867
5	3	1.4260	2.2515	4	2.7867
1	765	0.1489	1.4990	4	2.8071
2	1	.	0	6	4.0247
3	3	1.4260	2.2515	5	2.7867
4	19	0.9189	1.9979	1	2.8071
5	2	1.0213	1.0213	3	2.7867
6	1	.	0	2	4.0247
1	765	0.1489	1.4990	5	2.8071
2	2	0.5150	0.5150	7	3.0532
3	1	.	0	6	4.0247
4	1	.	0	2	3.3773
5	19	0.9189	1.9979	1	2.8071
6	1	.	0	3	4.0247
7	2	1.0213	1.0213	2	3.0532
1	755	0.1108	0.9796	7	1.6926
2	9	0.8217	1.5768	7	2.0161
3	1	.	0	4	4.0247
4	1	.	0	3	4.0247
5	2	0.5150	0.5150	8	3.0532
6	1	.	0	5	3.3773
7	20	0.6201	1.6674	1	1.6926
8	2	1.0213	1.0213	5	3.0532
1	755	0.1108	0.9796	2	1.7284
2	18	0.5657	1.5639	6	1.5020
3	1	.	0	8	3.3773
4	1	.	0	5	4.0247
5	1	.	0	4	4.0247
6	5	0.5069	0.9738	2	1.5020
7	6	0.5703	0.9672	6	2.1198
8	2	0.5150	0.5150	9	3.0532
9	2	1.0213	1.0213	8	3.0532

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	755	0.1108	0.9796	8	1.7284
2	1	.	0	3	2.0425
3	1	.	0	2	2.0425
4	1	.	0	10	4.0247
5	1	.	0	3	2.5648
6	5	0.5069	0.9738	8	1.5020
7	6	0.5703	0.9672	6	2.1198
8	18	0.5657	1.5639	6	1.5020
9	2	0.5150	0.5150	3	2.3884
10	1	.	0	4	4.0247
1	718	0.0583	0.3710	8	0.6809
2	1	.	0	7	4.0247
3	2	0.5150	0.5150	11	2.3884
4	1	.	0	11	2.5648
5	4	0.3809	0.7329	8	1.7060
6	1	.	0	11	2.0425
7	1	.	0	2	4.0247
8	46	0.2728	0.9824	1	0.6809
9	9	0.3939	0.9688	8	1.7308
10	7	0.6105	1.0227	9	1.9496
11	1	.	0	6	2.0425
1	721	0.0605	0.3774	5	0.7174
2	1	.	0	11	2.0425
3	1	.	0	11	2.5648
4	2	0.3047	0.3047	10	1.0253
5	44	0.2903	0.9881	1	0.7174
6	7	0.5865	0.9949	8	1.6389
7	1	.	0	12	4.0247
8	8	0.3611	0.9149	6	1.6389
9	2	0.5150	0.5150	11	2.3884
10	2	0.1615	0.1615	4	1.0253
11	1	.	0	2	2.0425
12	1	.	0	7	4.0247
1	721	0.0605	0.3774	4	0.7174
2	1	.	0	11	2.0425
3	2	0.3047	0.3047	5	1.0253
4	44	0.2903	0.9881	1	0.7174
5	2	0.1615	0.1615	3	1.0253
6	3	0.1963	0.2800	13	1.3113
7	2	0.5150	0.5150	11	2.3884
8	1	.	0	10	4.0247
9	1	.	0	11	2.5648
10	1	.	0	8	4.0247
11	1	.	0	2	2.0425
12	8	0.3611	0.9149	13	1.4281
13	4	0.4134	0.7108	6	1.3113
1	702	0.0496	0.2920	5	0.4819
2	1	.	0	7	2.5648
3	2	0.3047	0.3047	14	0.9887
4	3	0.1963	0.2800	12	1.2876
5	52	0.2026	1.0578	1	0.4819
6	1	.	0	7	2.0425
7	1	.	0	6	2.0425
8	1	.	0	10	4.0247
9	2	0.5150	0.5150	7	2.3884
10	1	.	0	8	4.0247
11	11	0.2550	0.5636	5	0.6898
12	4	0.3665	0.5837	13	1.1337
13	7	0.3322	0.8424	12	1.1337
14	3	0.4390	0.6923	3	0.9887

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	702	0.0496	0.2920	5	0.4819
2	11	0.2550	0.5636	5	0.6898
3	6	0.2916	0.5820	12	0.7767
4	3	0.1963	0.2800	15	1.3829
5	52	0.2026	1.0578	1	0.4819
6	2	0.5150	0.5150	9	2.3884
7	1	.	0	9	2.0425
8	1	.	0	14	4.0247
9	1	.	0	7	2.0425
10	2	0.3047	0.3047	11	1.0253
11	2	0.1615	0.1615	10	1.0253
12	3	0.3262	0.4889	3	0.7767
13	1	.	0	9	2.5648
14	1	.	0	8	4.0247
15	3	0.3143	0.4817	10	1.3082

12.1.2 Shlukování průměrů nenulových záznamů

Cluster Summary

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	784	0.4566	5.2961	2	12.3480
2	7	4.0370	9.5294	1	12.3480
1	771	0.2807	3.0334	3	5.2692
2	2	1.8142	1.8142	3	15.0341
3	18	2.1490	5.6934	1	5.2692
1	755	0.1583	1.6719	3	2.8184
2	2	1.8142	1.8142	4	10.9636
3	29	1.0932	2.5540	1	2.8184
4	5	1.4694	2.5042	3	6.5630
1	754	0.1524	1.2672	4	2.7846
2	5	1.4694	2.5042	5	6.3160
3	2	1.8142	1.8142	2	10.9636
4	14	0.7738	1.6766	5	1.8733
5	16	0.9659	2.7967	4	1.8733
1	754	0.1524	1.2672	6	2.7846
2	16	0.9659	2.7967	6	1.8733
3	2	1.8142	1.8142	4	10.8892
4	3	1.3380	2.1117	5	2.6257
5	2	0.9588	0.9588	4	2.6257
6	14	0.7738	1.6766	2	1.8733
1	754	0.1524	1.2672	3	2.7846
2	3	1.3380	2.1117	5	2.6257
3	14	0.7738	1.6766	7	1.8733
4	1	.	0	6	3.6284
5	2	0.9588	0.9588	2	2.6257
6	1	.	0	4	3.6284
7	16	0.9659	2.7967	3	1.8733
1	753	0.1490	1.2389	5	2.2488
2	1	.	0	4	3.6284
3	1	.	0	7	3.1675
4	1	.	0	2	3.6284
5	23	0.9122	2.2943	8	2.0770
6	2	0.9588	0.9588	7	2.8640
7	2	0.4858	0.4858	6	2.8640
8	8	0.8410	1.6657	5	2.0770

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	745	0.1267	0.9421	6	1.6161
2	2	0.4858	0.4858	3	2.8640
3	2	0.9588	0.9588	2	2.8640
4	11	0.6927	1.3744	6	1.8315
5	1	.	0	2	3.1675
6	20	0.5698	1.1194	1	1.6161
7	1	.	0	9	3.6284
8	8	0.8410	1.6657	4	2.3726
9	1	.	0	7	3.6284

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	745	0.1267	0.9421	3	1.6161
2	1	.	0	7	3.6284
3	20	0.5698	1.1194	1	1.6161
4	2	0.9588	0.9588	10	2.8640
5	5	0.6711	1.0743	9	1.6307
6	11	0.6927	1.3744	3	1.8315
7	1	.	0	2	3.6284
8	1	.	0	10	3.1675
9	3	0.5727	0.7875	5	1.6307
10	2	0.4858	0.4858	4	2.8640

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	745	0.1267	0.9421	6	1.6161
2	11	0.6927	1.3744	6	1.8315
3	1	.	0	9	3.6284
4	1	.	0	10	1.9176
5	3	0.5727	0.7875	7	1.6307
6	20	0.5698	1.1194	1	1.6161
7	5	0.6711	1.0743	5	1.6307
8	1	.	0	4	2.4440
9	1	.	0	3	3.6284
10	1	.	0	4	1.9176
11	2	0.4858	0.4858	4	2.2145

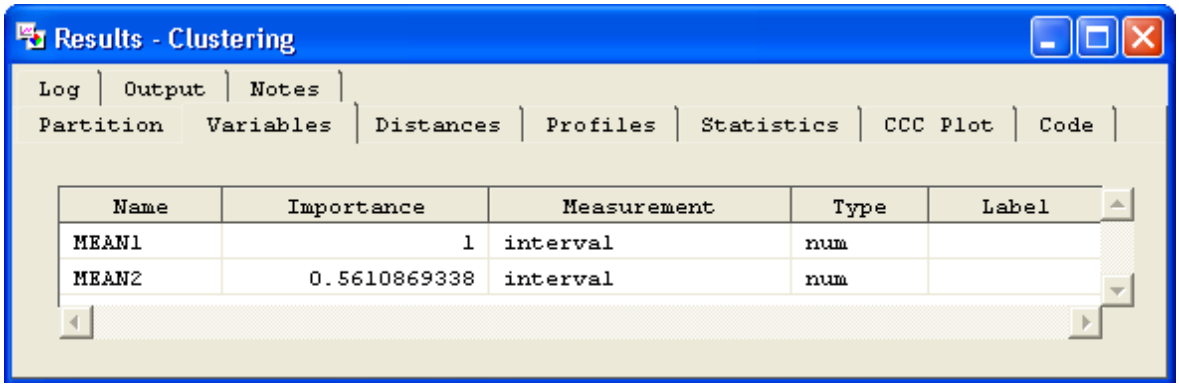
Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	696	0.0763	0.3501	9	0.6526
2	1	.	0	12	3.6284
3	2	0.4858	0.4858	11	2.2145
4	4	0.6191	1.0469	7	1.8995
5	1	.	0	11	2.4440
6	9	0.6511	1.2093	8	2.0936
7	2	0.5266	0.5266	4	1.8995
8	13	0.5451	1.1715	9	1.8785
9	60	0.2707	1.1339	1	0.6526
10	1	.	0	11	1.9176
11	1	.	0	10	1.9176
12	1	.	0	2	3.6284

Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	694	0.0753	0.3517	11	0.6433
2	1	.	0	10	1.9176
3	1	.	0	8	3.6284
4	3	0.5727	0.7875	6	1.6322
5	7	0.6044	1.2554	6	1.7376
6	8	0.5740	1.0819	13	1.3787
7	2	0.5266	0.5266	4	2.1132
8	1	.	0	3	3.6284
9	1	.	0	10	2.4440
10	1	.	0	2	1.9176
11	62	0.2695	1.1444	1	0.6433
12	2	0.4858	0.4858	10	2.2145
13	8	0.3294	0.5799	6	1.3787

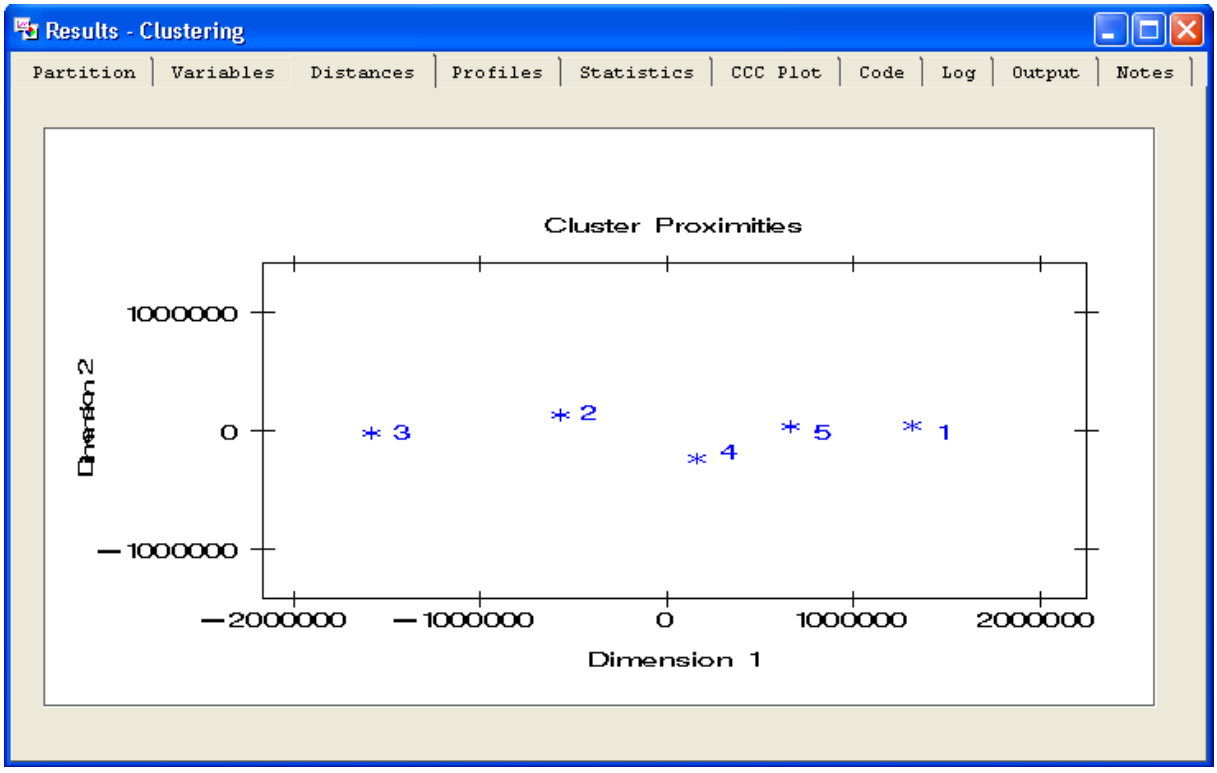
Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	7	0.5538	1.0048	10	1.5345
2	3	0.4267	0.6518	8	1.6221
3	1	.	0	5	2.4440
4	1	.	0	7	3.6284
5	1	.	0	6	1.9176
6	1	.	0	5	1.9176
7	1	.	0	4	3.6284
8	6	0.4045	0.8050	14	1.6220
9	10	0.4020	0.8260	1	1.5655
10	2	0.5769	0.5769	1	1.5345
11	683	0.0702	0.3220	14	0.5694
12	2	0.4858	0.4858	5	2.2145
13	2	0.5266	0.5266	1	1.9308
14	71	0.2303	0.7807	11	0.5694
1	683	0.0702	0.3220	11	0.5694
2	1	.	0	10	1.9176
3	1	.	0	15	1.9625
4	10	0.4020	0.8260	13	1.6503
5	1	.	0	2	2.4440
6	6	0.4045	0.8050	11	1.6220
7	1	.	0	9	3.6284
8	3	0.4267	0.6518	15	1.4917
9	1	.	0	7	3.6284
10	1	.	0	2	1.9176
11	71	0.2303	0.7807	1	0.5694
12	2	0.5266	0.5266	13	1.4345
13	4	0.4291	0.6761	15	1.1089
14	2	0.4858	0.4858	2	2.2145
15	4	0.4421	0.8247	13	1.1089

12.2 Výsledek shlukování v SAS Enterprise Miner

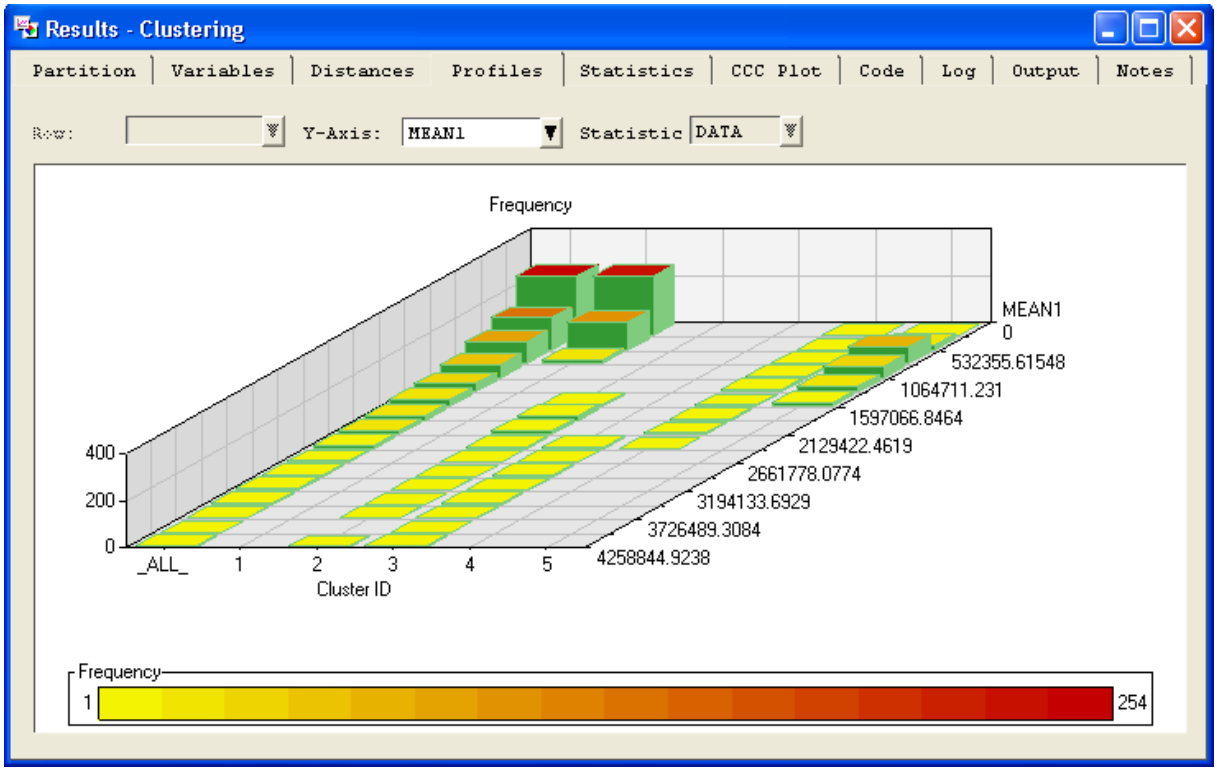
12.2.1 Po filtrování a standardizaci směrodatnou odchylkou



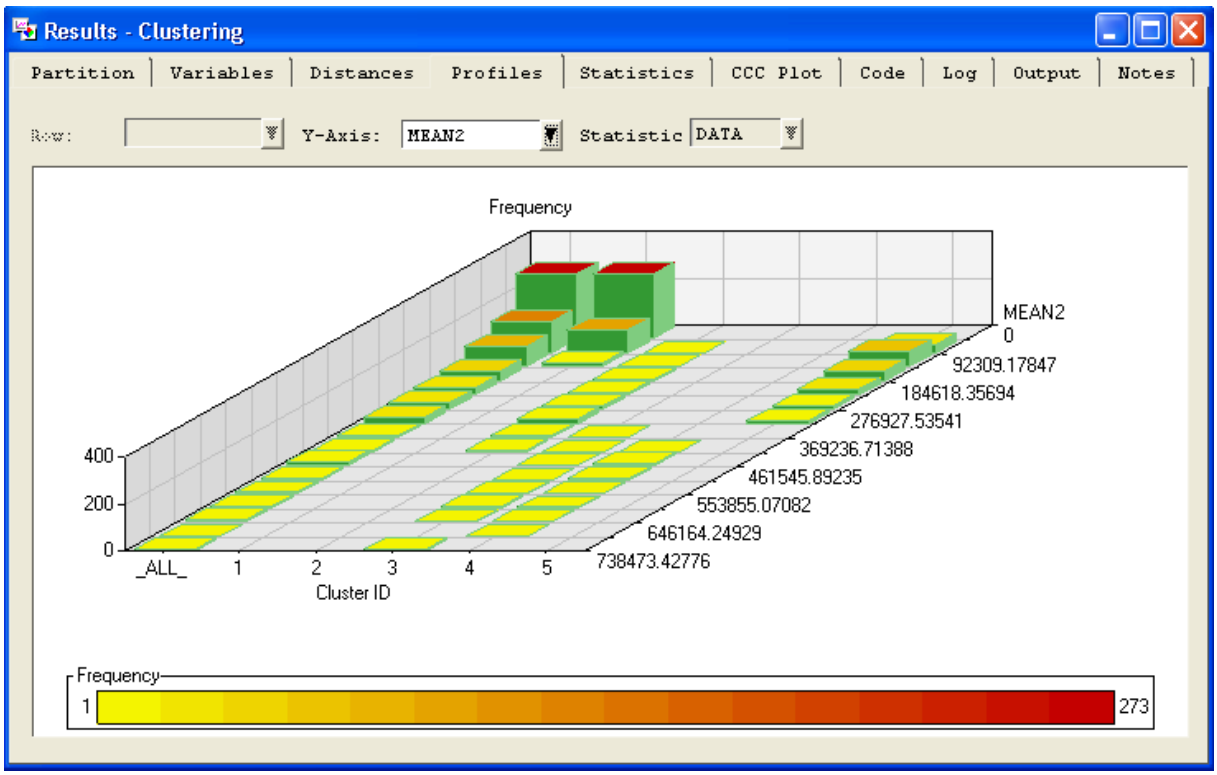
Obrázek 54: Výsledek shlukování - okno Variables



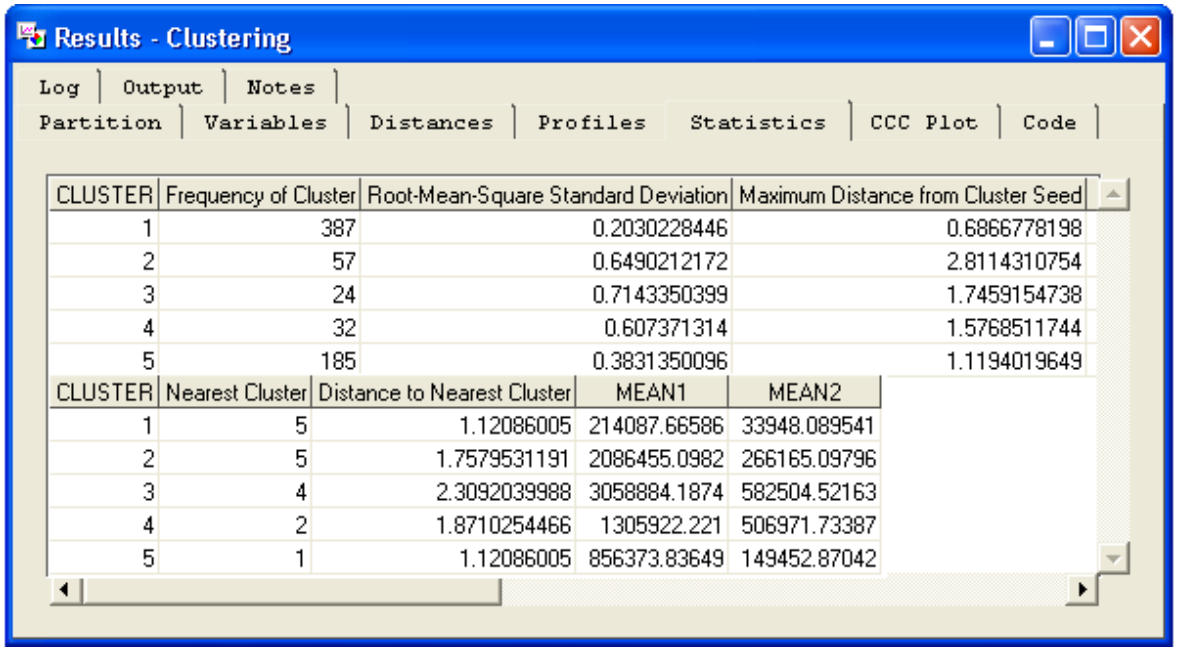
Obrázek 55: Výsledek shlukování - okno Distances



Obrázek 56: Výsledek shlukování - okno Profiles - proměnná Data_In



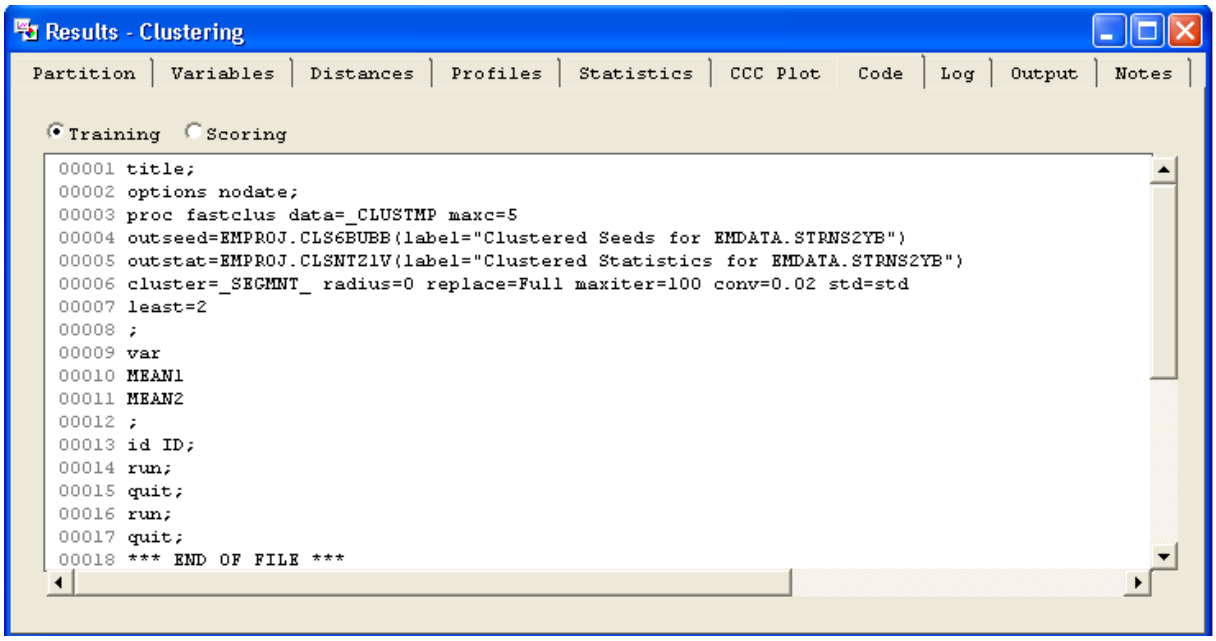
Obrázek 57: Výsledek shlukování - okno Profiles - proměnná Data_Out



CLUSTER	Frequency of Cluster	Root-Mean-Square Standard Deviation	Maximum Distance from Cluster Seed
1	387	0.2030228446	0.6866778198
2	57	0.6490212172	2.8114310754
3	24	0.7143350399	1.7459154738
4	32	0.607371314	1.5768511744
5	185	0.3831350096	1.1194019649

CLUSTER	Nearest Cluster	Distance to Nearest Cluster	MEAN1	MEAN2
1	5	1.12086005	214087.66586	33948.089541
2	5	1.7579531191	2086455.0982	266165.09796
3	4	2.3092039988	3058884.1874	582504.52163
4	2	1.8710254466	1305922.221	506971.73387
5	1	1.12086005	856373.83649	149452.87042

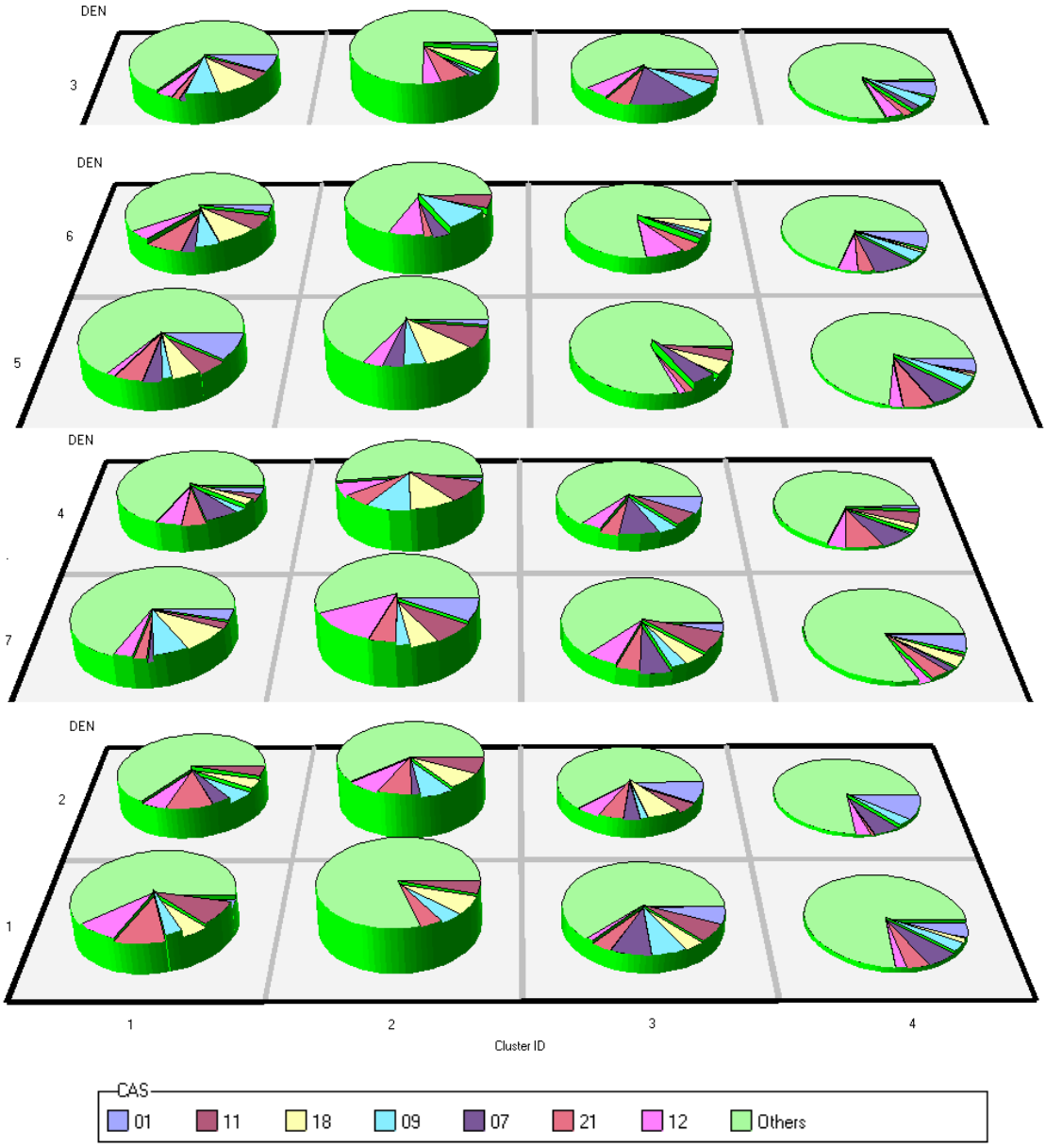
Obrázek 58: Výsledek shlukování - okno Statistics



Obrázek 59: Výsledek shlukování - okno Code

12.2.2 Při použití kategorizovaných proměnných

Graf 10: Analýza shluků dle proměnné DEN, CAS a Data_In v okně Profiles



Graf 11: Analýza shluků dle proměnné DEN, CAS a Data_Out v okně Profiles