VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV MATEMATIKY

INSTITUTE OF MATHEMATICS

METODY SHLUKOVÉ ANALÝZY V MATEMATICKÝCH PROGRAMECH

CLUSTER ANALYSIS IN MATHEMATICAL SOFTWARE

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE — **JOSEF STARÝ**

AUTHOR

VEDOUCÍ PRÁCE — **doc. RNDr. LIBOR ŽÁK, Ph.D.**

SUPERVISOR

BRNO 2021

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ FAKULTA STROJNÍHO INŽENÝRSTVÍ

Zadaní bakalářské práce

Ústav:	Ústav matematiky
Student:	**Josef Starý**
Studijní program:	Aplikované vědy v inženýrství
Studijní obor:	Matematické inženýrství
Vedoucí práce:	**doc. RNDr. Libor Žák, Ph.D.**
Akademický rok:	2020/21

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

Metody shlukové analýzy v matematických programech

Stručná charakteristika problematiky úkolu:

Shluková analýza je jednou z mnoha metod, které se snaží získat z dat informace. Tyto metody jsou součástí matematických programů, ale v různém tvaru a kvalitě. Cílem je popsat tyto metody v dostupných programech (Mathematica, Matlab, Maple, Minitab, Statistica,...) a porovnat jejich funkčnost.

Cíle bakalářské práce:

Popis základních shlukovacích metod.
Popis metod shlukové analýzy v matematických programech se zaměřením na rozdíly.
Použití shlukové analýzy na řešení reálného problému.

Seznam doporučené literatury:

ANDERBERG, M. R. Cluster Analysis for Applications. Academic Press, New York, 1973.

LUKASOVÁ, A., ŠARMANOVÁ, J. Metody shlukové analýzy, SNTL, Praha, 1985.

TRIOLA, M. F. Minitab Software Manual. Pearson Education, 1997. ISBN-13: 9780201859249.

Maple User Manual. Copyright © Maplesoft, a division of Waterloo Maple Inc. 2014. Dostupné z:

https://www.maplesoft.com/documentation_center/maple18/usermanual.pdf

BERANOVÁ, P., BLAŽKOVÁ, L., ULDRICH, M. Manuál k ovládání programu STATISTICA. StatSoft CR s.r.o. 2012. Dostupné z:

http://www.statsoft.cz/file1/PDF/manualy/Manual_k_ovladani_programu_STATISTICA.pdf

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2020/21

V Brně, dne

L. S.

prof. RNDr. Josef Šlapal, CSc. ředitel ústavu	doc. Ing. Jaroslav Katolický, Ph.D. děkan fakulty

Abstrakt
Tato bakalářská práce se zaměřuje na metody shlukové analýzy v matematických programech. Jejím cílem je popsat základní shlukovací metody, popsat způsob jejich implementace v matematických programech, metody pak použít ke shlukování připravených dat a porovnat funkčnost zvolených programů.

Summary
This bachelor thesis is focused on methods of cluster analysis in mathematical software. The goal is to describe basic methods of cluster analysis, to describe their implementation in mathematical software, to use the methods for clustering of prepared data and to compare the functionality of chosen software.

Klíčová slova
shluková analýza, shlukování, hierarchické shlukování, nehierarchické shlukování, k-means, matematické programy, dolování dat

Keywords
cluster analysis, clustering, hierarchical clustering, nonhierarchical clustering, k-means, mathematical software, data mining

STARÝ, J.*Metody shlukové analýzy v matematických programech.* Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2021. 42 s. Vedoucí doc. RNDr. Libor Žák, Ph.D.

Prohlašuji, že jsem bakalářskou práci na téma *Metody shlukové analýzy v matematických programech* vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím informačních zdrojů uvedených v seznamu použité literatury na konci práce.

Josef Starý

Chtěl bych poděkovat RNDr. Liboru Žákovi, Ph.D. za poskytnutí odborných rad při tvorbě této bakalářské práce a mamince za vytvoření příjemného pracovního prostředí v těžké koronavirové době.

Josef Starý

Obsah

Úvod

Termín *shluková analýza* použil jako první Robert Tryon v roce 1939. Jedná se o metody a algoritmy sloužící k nalezení souvislostí v datech, o nichž často nic nevíme. Cílem těchto metod je sdružovat data s podobnými vlastnostmi do množin, které jsou nazývány shluky.

V této bakalářské práci budou nejprve popsány základní pojmy spjaté se shlukovou analýzou, jako jsou shluk, nepodobnost objektů nebo koeficient nepodobnosti shluků. Představeny budou základní shlukovací metody, které se svojí podstatou dělí na hierarchické a nehierarchické.

Další náplní práce bude metody shlukové analýzy vyzkoušet ve zvolených matematických programech. Metody jsou v programech v různé formě, porovnáme tedy jejich funkčnost, dosažené výsledky a uživatelskou přívětivost. Vybrané výsledky také zobrazíme v grafických výstupech. Používat budeme software MATLAB, programovací jazyk R a dva softwary zaměřené na statistiku a analýzu dat - Minitab a STATISTICU. Pro potřeby práce byly vytvořeny fiktivní datové soubory, se kterými budeme v programech pracovat.

1. Shluková analýza

Shluková analýza je jednou z metod k získání informací z dat. Jejím cílem je v dané množině objektů najít podmnožiny (takzvané shluky) takovým způsobem, aby objekty ve shluku byly v jistém smyslu navzájem podobné a aby se dostatečně odlišovaly od objektů mimo tento shluk. (V kapitole jsou informace čerpány z [8], [1], [4].)

1.1. Základní pojmy

Nechť X je množina n objektů popsaných p-ticí znaků. Systém podmnožin

$$\Omega = \{C_1, C_2, \ldots, C_m\}$$

množiny X nazveme rozklad množiny X, jestliže platí

1. $C_i \neq \emptyset$ pro $1 \leq i \leq m$,
2. $C_i \cap C_j = \emptyset$ pro $i \neq j$,
3. $C_1 \cup C_2 \cup \ldots \cup C_m = X$.

Vytvoříme matici $\boldsymbol{X}$ o n řádcích a p sloupcích, v níž i-tý řádek patří i-tému objektu a j-tý sloupec odpovídá j-tému znaku. Tato matice se nazývá datová matice.

$$\boldsymbol{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ x_{n1} & \cdots & \cdots & x_{np} \end{pmatrix}$$

Znakům zpravidla přiřazujeme číselné hodnoty. Objekty pro shlukovou analýzu jsou pak tedy vektory p čísel. Znaky objektů mohou být:

1. Kvantitativní znaky
   - Množinu stavů znaku tvoří interval reálné osy (například pro znak výška) nebo konečná či spočetná množina čísel (např. pro znak počet žeber).
2. Kvalitativní znaky
   - Příkladem jsou binární znaky, např. znak „mít zelené oči“ (pravdivý/nepravdivý), které nejčastěji kódujeme tak, že stav „pravdivý“ bývá kódován jedničkou, stav „nepravdivý“ nulou.
   - Množinou stavů znaku je konečná množina popisných termínů. Tento typ znaků většinou převedeme na soustavu binárních znaků. Pokud například množinou stavů znaku „barva plodu“ je trojice (červená, zelená, žlutá), lze zavést namísto toho tři znaky „červený plod“, „zelený plod“, „žlutý plod“, z nichž každý nabývá hodnoty „pravdivý“ nebo „nepravdivý“. V některých případech lze množinu popisných termínů uspořádat. Jestliže znak kupříkladu nabývá stavů (světlá, střední, tmavá), můžeme mimo binární způsob kódovat také uspořádanou množinou přirozených čísel. V uvedeném případě by čísla 1, 2, 3 udávala stupeň tmavosti.

Velmi často získáváme hodnoty jednotlivých znaků v různých jednotkách. To může způsobovat, že některé znaky se chovají jako dominantní, a tedy ovlivňují průběh shlukové analýzy větší měrou. Abychom tomu zabránili a dali všem znakům stejnou váhu, je vhodné provést standardizaci dat. Nechť $\boldsymbol{Y}$ je datová matice s rozměry $n \times p$. Pro všechny znaky (sloupce) y_j vypočítáme střední hodnoty $\overline{y_j}$ a směrodatné odchylky s_j pomocí vzorců:

$$\overline{y_j} = \frac{1}{n} \sum_{i=1}^{n} y_{ij},$$
$$s_j = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_{ij} - \overline{y_j})^2}.$$

Dále vypočítáme standardizované hodnoty znaků:

$$x_{ij} = \frac{y_{ij} - \overline{y_j}}{s_j}.$$

Dostaneme tak transformovanou datovou matici $\boldsymbol{X}$, standardizované hodnoty v této matici mají nyní střední hodnotu rovnu 0 a rozptyl roven 1. Jednotlivé znaky se navíc stávají bezrozměrnými veličinami.

1.1.1. Podobnost objektů

Jedním z důležitých problémů shlukové analýzy je pojetí vzájemné *podobnosti objektů* a kvantitativní vyjádření této podobnosti. Cílem je stanovit vhodný předpis $\pi : X \times X \mapsto \mathbb{R}$, jenž každé dvojici objektů (O_i, O_j) přiřadí reálné číslo $\pi\ (O_i, O_j)$, které budeme považovat za míru podobnosti objektů, tak, aby byly splněny následující podmínky:

$$\pi(O_i, O_j) \geq 0,$$
$$\pi(O_i, O_j) = \pi(O_j, O_i).$$

Tento předpis vyjadřující podobnost objektů dává tím větší hodnotu $\pi(O_i, O_j)$, čím větší je vzájemná podobnost objektů O_i, O_j. Má tedy smysl požadovat, aby hodnota $\pi(O_i, O_j)$ byla maximální, pokud $O_i = O_j$.

Ve shlukovacích metodách se však vychází spíše z duálního pojmu. *Nepodobnost objektů* $d : X \times X \mapsto \mathbb{R}$ splňuje podmínky:

$$d(O_i, O_j) = 0 \iff O_i = O_j,$$
$$d(O_i, O_j) \geq 0,$$
$$d(O_i, O_j) = d(O_j, O_i).$$

Jedním ze způsobů vyjádření podobnostních vztahů mezi objekty jsou metriky. Pokud například pohlížíme na objekty jako na body v p-rozměrném euklidovském prostoru $\mathbb{E}_p$, pak určíme euklidovskou vzdálenost pro dva body $A = (a_1, a_2, \ldots, a_p)$, $B = (b_1, b_2, \ldots, b_p)$ jako

$$\rho(A, B) = \sqrt{\sum_{i=1}^{p} (a_i - b_i)^2}.$$

Obecně je metrikou zobrazení $\rho : \mathbb{E}_p \times \mathbb{E}_p \mapsto \mathbb{R}$ splňující čtyři podmínky ($A, B, C \in \mathbb{E}_p$):

$$\begin{aligned}
&\rho(A,B) = 0 \iff A = B,\\
&\rho(A,B) \geq 0,\\
&\rho(A,B) = \rho(B,A),\\
&\rho(A,C) \leq \rho(A,B) + \rho(B,C).
\end{aligned}$$

1.1.2. Shluk

Cílem shlukové analýzy je rozdělení objektů z množiny objektů do shluků. Bonner popsal shluk intuitivní definicí jako množinu, ve které jsou si prvky podobné, ale nejsou příliš podobné prvkům mimo tento shluk.

[8]Van Rijsbergen zpřesnil tento náznak definice takto:

Nechť je dána množina objektů $O = \{O_1, O_2, \ldots, O_n\}$, d je nepodobnost objektů a předpokládejme, že chceme najít k shluků ($1 < k < n$). Pak hledáme zobrazení $f : O \mapsto \Omega$, jež každý objekt z množiny O přiřadí do shluku $C_i \in \Omega$, $\Omega = \{C_1, C_2, \ldots, C_k\}$, $\cup_{i=1}^{k} C_i = O$. Shlukem nazveme takovou podmnožinu C množiny objektů O, pro kterou platí

$$\max_{O_i, O_j \in C} d(O_i, O_j) < \min_{\substack{O_k \in C \\ O_l \notin C}} d(O_k, O_l).$$

Shluky jakožto výsledky aplikace shlukovacího algoritmu na danou množinu objektů O tvoří systém, jenž je buď hierarchický, nebo nehierarchický.

1.1.3. Koeficient nepodobnosti shluků

Důležitou roli ve shlukování má výpočet nepodobnosti shluků. *Koeficient nepodobnosti shluků* definujeme jako zobrazení $D : \Omega \times \Omega \mapsto \mathbb{R}$, které přiřazuje každé dvojici shluků C_i, C_j z rozkladu Ω číslo $D(C_i, C_j)$ splňující následující podmínky:

$$\begin{aligned}
&D(C_i, C_i) = 0,\\
&D(C_i, C_j) \geq 0,\\
&D(C_i, C_j) = D(C_j, C_i).
\end{aligned}$$

Čtvercovou matici $\mathbf{D}$ o rozměrech $k \times k$, kde $D_{i,j} = D(C_i, C_j), 1 \leq i, j \leq k, k = |\Omega|$, nazveme *maticí nepodobností shluků*.

Zmíníme nejznámější metody zavádějící tento koeficient. Nechť d je koeficient nepodobnosti objektů, A a B jsou shluky rozkladu Ω a $D(A, B)$ je koeficient nepodobnosti shluků.

Metoda nejbližšího souseda

Pro shluky A, B definujeme jejich koeficient nepodobnosti následovně:

$$D(A,B) = \min_{\substack{O_i \in A \\ O_j \in B}} d(O_i, O_j).$$

Nepodobnost dvou shluků A, B je tedy vyjádřena pomocí nepodobnosti dvou nejméně nepodobných objektů ze shluků A a B. Metodu se také nazývá *single linkage* a je tak označována v matematických programech.

Metoda nejvzdálenějšího souseda

Metodu nejvzdálenějšího souseda definujeme podobně jako předcházející metodu nejbližšího souseda. Tentokrát koeficient nepodobnosti shluků A, B uvažujeme jako nepodobnost dvou nejvíce nepodobných objektů z těchto shluků:

$$D(A,B) = \max_{\substack{O_i \in A \\ O_j \in B}} d(O_i, O_j).$$

Metoda se nazývá také *complete linkage*.

Centroidní metoda

Centroidní metoda vyjadřuje koeficient nepodobnosti shluků A, B jako nepodobnost centroidů (těžišť) těchto shluků.

Nechť

$$A = \{O_k, k = 1, \ldots, |A|\}, O_k = (o_{k,1}, o_{k,2}, \ldots, o_{k,p}),$$
$$B = \{O_h, h = 1, \ldots, |B|\}, O_h = (o_{h,1}, o_{h,2}, \ldots, o_{h,p})$$

jsou shluky. Jejich těžiště jsou

$$\overline{A} = (\overline{a}_1, \overline{a}_2, \ldots, \overline{a}_p),$$
$$\overline{B} = (\overline{b}_1, \overline{b}_2, \ldots, \overline{b}_p),$$

kde

$$\overline{a_j} = \frac{1}{|A|} \sum_{k=1}^{|A|} o_{k,j},\ j = 1, \ldots, p,$$
$$\overline{b_j} = \frac{1}{|B|} \sum_{h=1}^{|B|} o_{h,j},\ j = 1, \ldots, p.$$

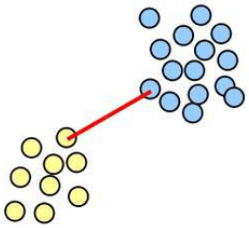
Pak definujeme nepodobnost shluků jako

$$D(A,B) = d(\overline{A}, \overline{B}).$$

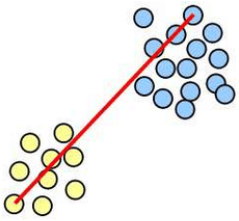
Metoda průměrné nepodobnosti objektů

V této metodě určujeme koeficient nepodobnosti shluků jako průměrnou nepodobnost všech dvojic objektů z daných shluků:

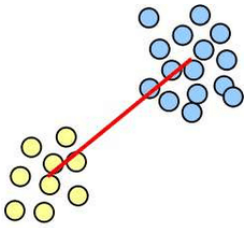
$$D(A,B) = \frac{1}{|A||B|} \sum_{O_i \in A} \sum_{O_j \in B} d(O_i, O_j).$$



(a) Metoda nejbližšího souseda



(b) Metoda nejvzdálenějšího souseda



(c) Centroidní metoda

Obrázek 1.1: Koeficient nepodobnosti shluků [6]

Lance-Williamsova pružná strategie

Zmínili jsme některé metody k určení koeficientu nepodobnosti shluků. Lance a Williams došli k závěru, že z hlediska praktického je výhodné použít rekurzivní schéma, díky kterému jsme schopni provést efektivnější výpočet. Matice nepodobností skluků rozkladu Ω_i lze vypočítat pomocí matice z předcházejícího rozkladu Ω_{i-1} tímto způsobem:

1. $D(\{O_i\}, \{O_j\} = d(O_i, O_j)$
2. Nechť $R = S \cup T$ je shluk rozkladu Ω_i získaný sjednocením shluků $S, T \in \Omega_{i-1}$. Pak pro všechny shluky C z rozkladu Ω_{i-1} přecházející beze změny do rozkladu Ω_i platí:

$$D(R, C) = \alpha_S D(S, C) + \alpha_T D(T, C) + \beta D(S, T) + \gamma |D(S, C) - D(T, C)|,$$

koeficienty $\alpha_S, \alpha_T, \beta, \gamma$ pro výše uvedené metody najdeme v tabulce níže.

Tabulka 1.1: Koeficienty Lance-Williamsova vzorce

Shlukovací metoda	α_S	α_T	β	γ
Metoda nejbližšího souseda	$1/2$	$1/2$	0	$-1/2$
Metoda nejvzdálenějšího souseda	$1/2$	$1/2$	0	$1/2$
Centroidní metoda	$\frac{\lvert S\rvert}{\lvert S\rvert+\lvert T\rvert}$	$\frac{\lvert T\rvert}{\lvert S\rvert+\lvert T\rvert}$	$\frac{-\lvert S\rvert\lvert T\rvert}{(\lvert S\rvert+\lvert T\rvert)^2}$	0
Metoda průměrné nepodobnosti	$\frac{\lvert S\rvert}{\lvert S\rvert+\lvert T\rvert}$	$\frac{\lvert T\rvert}{\lvert S\rvert+\lvert T\rvert}$	0	0

1.2. Hierarchické shlukování

Hierarchické shlukování má charakter posloupnosti rozkladů množiny objektů, ve které je každý rozklad zjemněním rozkladu následujícího. Na jedné straně tedy máme triviální rozklad, v němž každý objekt tvoří jednoprvkový shluk, a na druhé straně dostáváme triviální rozklad s jedním shlukem obsahujícím celou množinu objektů. Metody hierarchického shlukování dělíme podle směru postupu při shlukování na:

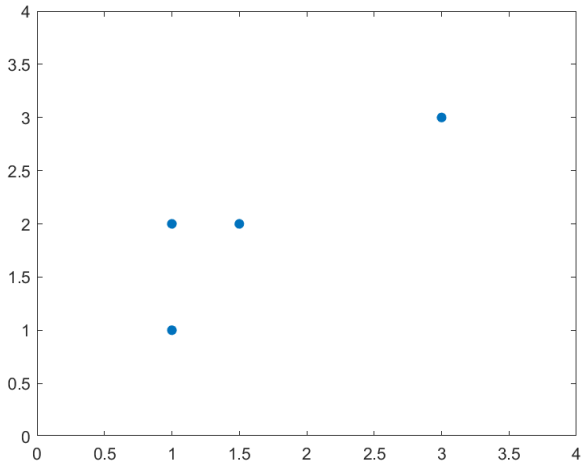
- Aglomerativní metody
- Divizní metody

1.2.1. Aglomerativní metody

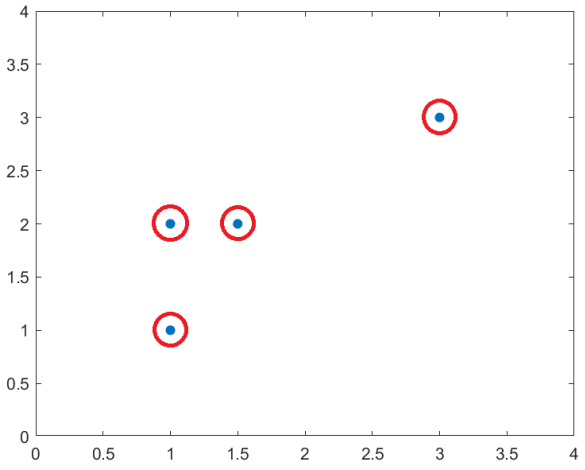
Aglomerativní procedura shlukování začíná nultým rozkladem Ω_0, ve kterém každý objekt O_i z množiny objektů $O = \{O_1, O_2, \ldots, O_n\}$ vytváří jednoprvkový shluk. V každém kroku procesu shlukování vybereme dva shluky s nejmenším koeficientem nepodobnosti (který zavedeme například jedním ze způsobů z kapitoly 1.1.3), spojíme je v jeden a vypočítáme koeficienty nepodobnosti pro nově vytvořené shluky. Tento krok opakujeme. V každém dalším kroku se snižuje počet shluků o jeden, proto s-tý rozklad Ω_s obsahuje $n - s$ shluků. Tento rozklad je složen z nového shluku vzniklého z dvou vzájemně si nejpodobnějších shluků z předchozího rozkladu Ω_{s-1} a z ostatních nezměněných shluků rozkladu Ω_{s-1}. V matici koeficientů nepodobností shluků tedy můžeme hodnoty odpovídající nezměněným shlukům převzít z předchozího kroku. Procedura končí, když dostaneme rozklad Ω_{n-1}, v němž všechny objekty z množiny objektů vytváří jediný shluk.

Příklad

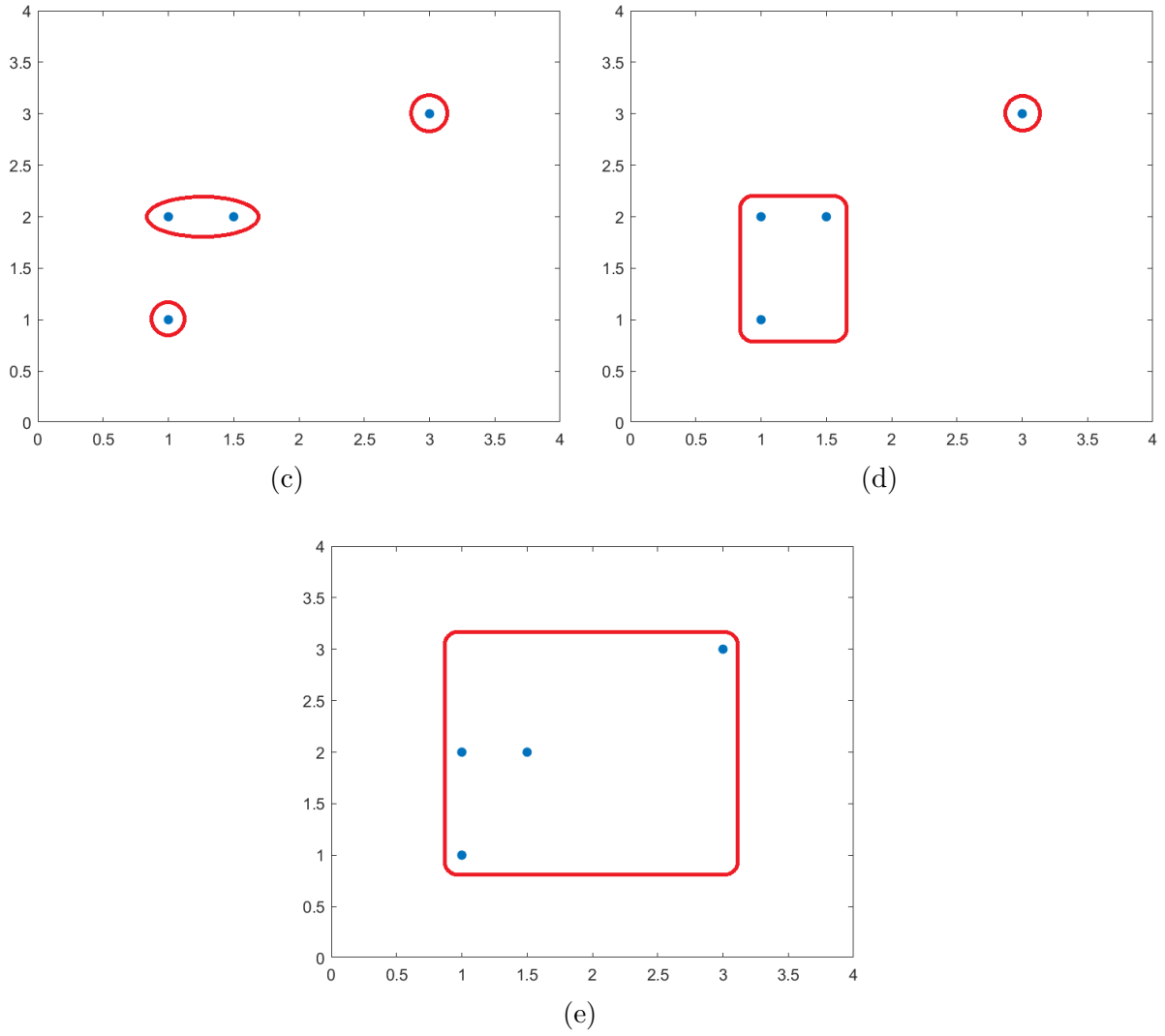
Ilustrujme si nyní aglomerativní shlukování krok po kroku na jednoduchém příkladu. Využijeme u toho metodu nejbližšího souseda a nepodobnosti bodů budeme měřit euklidovskou metrikou. Mějme v rovině dané body (1, 1), (1, 2), (3, 3), (1.5, 2). Na obrázku níže nejprve z každého bodu uděláme samostatný shluk a v každém dalším kroku pak sloučíme dva nejbližší shluky, dokud všechny body netvoří jediný shluk.



(a)



(b)



Obrázek 1.2: Příklad: Aglomerativní hierarchické shlukování

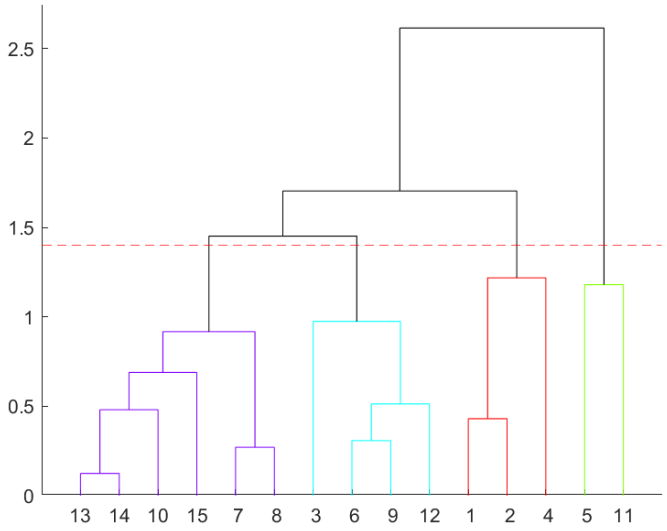
1.2.2. Divizní metody

Na rozdíl od aglomerativních metod, u nichž vytváříme hierarchický systém rozkladů množiny objektů postupným sjednocováním shluků, u divizních metod vzniká hierarchický systém postupným dělením shluků. Divizní procedura shlukování tedy na počátku pokládá celou množinu objektů $O = \{O_1, O_2, \ldots, O_n\}$ za jeden shluk. Dále se rozdělují existující shluky, dokud nejsou všechny shluky tvořeny jediným objektem.

Nalezení optimálního rozkladu množiny obsahující n objektů na dvě podmnožiny si žádá vyzkoušet všechny možné rozklady a to lze učinit $2^{n-1}-1$ způsoby. To je mnohokrát více než u aglomerativního přístupu, u kterého stačí v prvním kroku provést $n(n-1)/2$ výpočtů. Divizní procedury jsou tedy prakticky proveditelné pouze pro menší počet objektů. V této práci je jimi dále zaobírat nebudeme.

1.2.3. Dendrogram

Konečné výsledky všech metod hierarchického shlukování můžeme graficky vizualizovat stromovým diagramem, který nazýváme *dendrogram*. Určuje pořadí, v němž byly shluky sloučeny při aglomerativním shlukování nebo rozděleny při divizním shlukování. Na jedné ose jsou vyneseny indexy jednotlivých objektů. Druhá osa reprezentuje shlukovací hladiny udávající koeficient nepodobnosti shluků. Provedeme-li řez dendrogramem v nějaké shlukovací hladině, získáme konkrétní rozklad a uzly reprezentující jednotlivé shluky. Na obrázku 1.3 je znázorněn vodorovný řez hodnotou 1,4. Dostali jsme tak rozdělení 15 objektů do 4 shluků.



Obrázek 1.3: Dendrogram

1.3. Nehierarchické shlukování

Metody nehierarchického shlukování slouží k nalezení optimálního rozkladu množiny objektů do shluků. Na rozdíl od hierarchického shlukování dochází během shlukovacího procesu k přesouvání objektů mezi shluky.

Při hledání optimálního rozkladu množiny objektů je nutné stanovit, v jakém smyslu má rozklad být optimální. Kvalitu rozkladu obvykle udává *funkcionál kvality rozkladu*, pro který jsou určující jedna nebo více z následujících vlastností shluků tvořících rozklad:

- vzájemná podobnost objektů uvnitř shluku,
- míra separace shluků,
- homogenita rozložení objektů uvnitř shluků,
- rovnoměrnost rozložení objektů do různých shluků.

1.3.1. Problém počátečního rozkladu

Typické pro nehierarchické shlukování je, že na začátku postupu stanovíme nebo odvodíme počáteční rozklad na k shluků. Algoritmy poté při hledání optimálního rozkladu

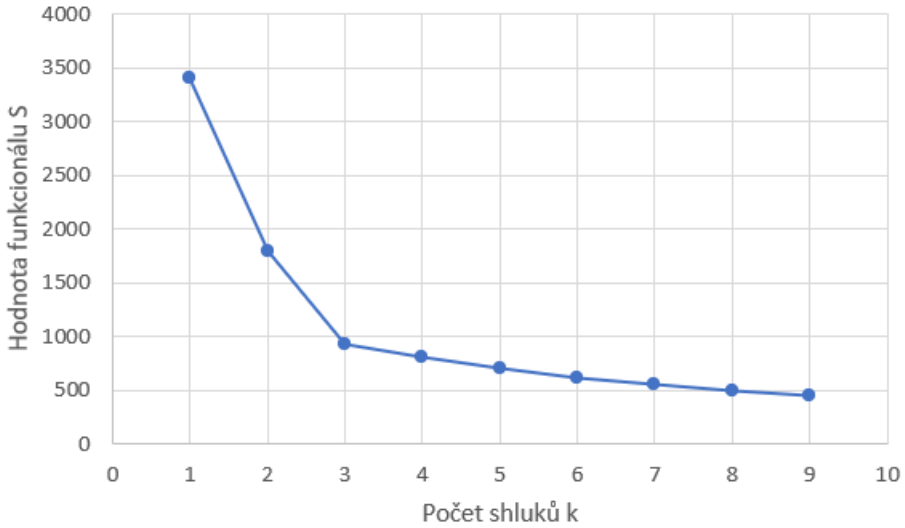
mohou buď zachovávat daný počet shluků, nebo v případě některých složitějších algoritmů v průběhu výpočtu počet shluků měnit v závislosti na určitých řídících parametrech.

Nejjednodušším způsobem volby počátečního rozkladu je náhodné vygenerování k bodů (tzv. centroidů). Všechny objekty pak přiřadíme k nejbližšímu (ve smyslu nepodobnosti objektů) centroidu. U tohoto přístupu však může nastat, že některý ze shluků v počátečním rozkladu nebude obsahovat žádné objekty. Tomu se vyhneme, pokud za centroidy náhodně vybereme k prvků z množiny objektů. Další možností je vybrat za počáteční centroidy objekty tak, aby jejich nepodobnosti byly maximální.

Optimální počet shluků

Především pro metody shlukování zachovávající předem daný počet shluků je počáteční rozklad důležitý, protože zahrnuje volbu počtu shluků k. K tomu využijeme buď některou z metod hledání optimálního počtu shluků, nebo k určíme díky znalosti povahy problému.

U metod s neměnným počtem shluků se běžně vhodný počet shluků zvolí tak, že se provede shluková analýza pro různý počet shluků a z výsledků se vybere nejlepší. Jednou z nejpopulárnějších metod, která navrhuje optimální počet shluků, je *elbow method* („loketní metoda“). Na obrázku 1.4 níže jsou vykresleny hodnoty funkcionálu, který bývá například sumou čtverců nepodobností objektů a těžišť příslušných shluků, pro různý počet shluků. Pozorujeme výrazný „loket“, když je počet shluků roven 3. Z toho pak vyplývá, že právě 3 je vhodný počet shluků. Chceme-li určit optimální počet shluků předem, využívá se různých indexů (Calinski-Harabascův index, C index, Goodman-Kruskal).



Obrázek 1.4: Elbow method

1.3.2. Metoda k-means

Metoda k-means je nejznámější metodou nehierarchického shlukování, také se někdy nazývá *MacQueenova k-průměrová* podle svého objevitele. Podstatou algoritmu je minima-

lizace sumy čtverců nepodobností objektů a těžišť shluků, do nichž patří. Minimalizujeme tedy funkcionál

$$SSE = \sum_{i=1}^{k} \sum_{O \in C_i} [d(O, c_i)]^2,$$

kde c_i je těžiště shluku C_i a d je nepodobnost objektů. Označení *SSE* pochází z anglického *sum of squared errors*. V dalších kapitolách budeme zkratku *SSE* používat.

Postup metody k-means lze shrnout následovně:

1. Určíme počet shluků k a náhodně vybereme k počátečních centroidů.
2. Přiřadíme všechny prvky z množiny objektů k nejbližšímu centroidu. Dostáváme tak počáteční rozklad.
3. Vypočítáme nová těžiště jednotlivých shluků a opět každý objekt přiřadíme do shluku, k jehož těžišti má nejblíže (ve smyslu nepodobnosti objektů).
4. Předchozí krok opakujeme, dokud nenastane situace, že se centroidy shluků nezmění.

Počet kroků závisí na počáteční volbě centroidů. Metoda konverguje pouze k lokálnímu minimu, je proto vhodné použít algoritmus pro různé počáteční rozklady a vybrat pak ten výsledný rozklad, pro nějž funkcionál *SSE* dosahuje minimální hodnoty.

K-means++

Metoda k-means++ vychází, jak už název napovídá, z metody k-means. Liší se počátečním rozkladem. Za první centroid počátečního rozkladu vybereme náhodně jeden z objektů. Druhým centroidem pak bude objekt nejvíce nepodobný s prvním centroidem. Další centroid poté zvolíme jako objekt co nejvzdálenější od předchozích centroidů a takto pokračujeme, dokud není zvoleno k centroidů. Takto upravená metoda dosahuje lepších výsledků než při použití zcela náhodného počátečního rozkladu.

1.3.3. Metoda k-medoids

Metoda k-medoids je obdobou metody k-means. Místo centroidu však shluk reprezentujeme tzv. medoidem, jímž je přímo jeden z objektů ve shluku. Jako medoid označíme ten objekt, pro který je součet nepodobností tohoto objektu s ostatními objekty ve shluku minimální.

Algoritmus tedy funguje následovně:

1. Určíme počet shluků k a vybereme k počátečních medoidů.
2. Přiřadíme všechny prvky z množiny objektů k nejbližšímu medoidu (ve smyslu nepodobnosti objektů). Dostáváme tak počáteční rozklad.
3. Vypočítáme nové medoidy jednotlivých shluků a opět každý objekt přiřadíme do shluku, k jehož medoidu má nejblíže.
4. Předchozí krok opakujeme, dokud nenastane situace, že se medoidy shluků nezmění.

2. Shlukování v matematických programech

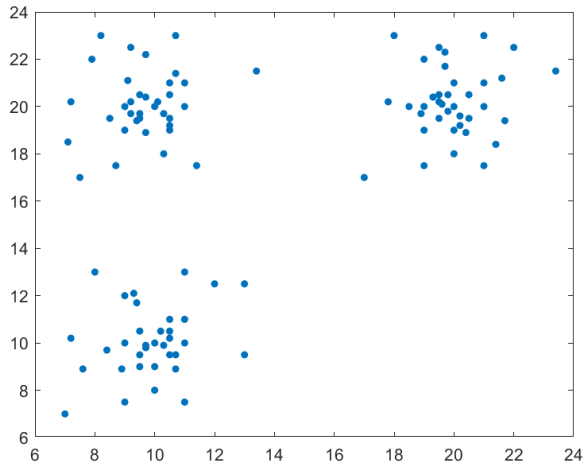
2.1. Programy

Metody shlukové analýzy jsou součástí různých matematických programů. V této kapitole je vyzkoušíme a porovnáme jejich funkčnost. Využívat budeme následujících programů:

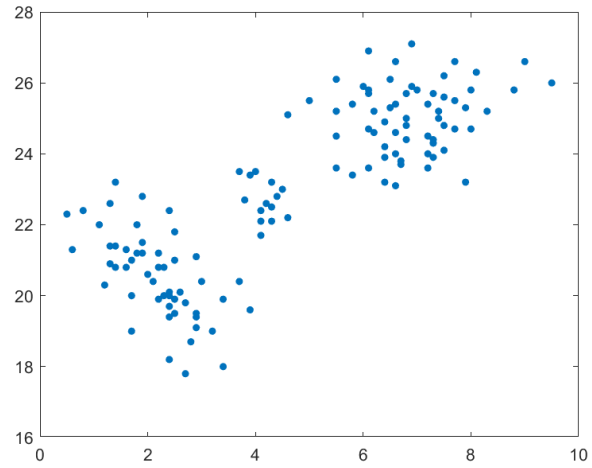
- MATLAB - Software MATLAB je vyvíjen společností MathWorks a je určen pro širokou škálu vědeckotechnických výpočtů. Je založen na počítání s maticemi, které jsou základním typem proměnné. Dokonce i obyčejné číslo je bráno jako matice rozměrů 1×1.
- R - Programovací jazyk R je určený k statistické analýze dat a jejich grafickému zobrazení. Práci s jazykem R umožňuje například vývojové prostředí RStudio.
- Minitab - Software Minitab obsahuje balík statistických metod určených k práci s daty.
- STATISTICA - Software STATISTICA byl vytvořen společností StatSoft a v současnosti vyvíjený společností TIBCO Software Inc. Program slouží pro práci s daty, jejich analýzu či dolování.

2.2. Data

Pro potřeby bakalářské práce byly vytvořeny fiktivní datové soubory, se kterými budeme pracovat. Z vizualizačních důvodů jsou veškerá využitá data dvourozměrná, lze je tedy zobrazit jako body v rovině.

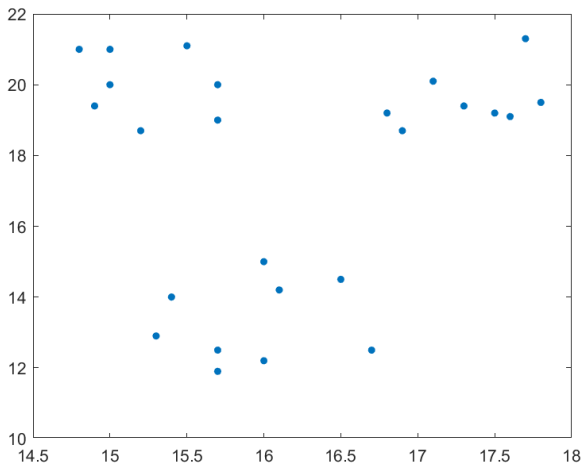


Data č. 1

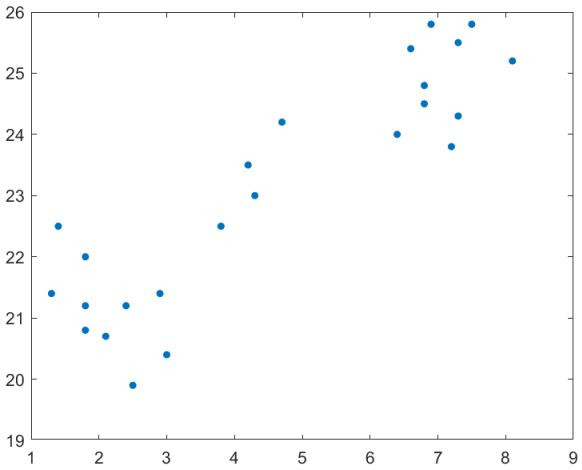


Data č. 2

U prvních dvou problémů je rozdělení do shluků zřejmé. Vlevo máme datový soubor se třemi zhruba stejně velkými shluky, vpravo je jeden malý shluk umístěn mezi dvěma velkými.

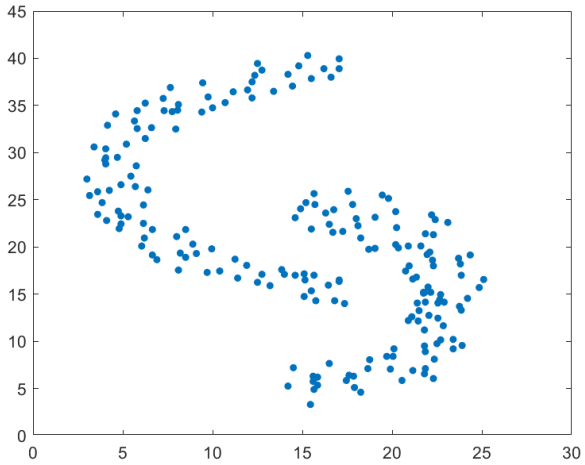


Data č. 3

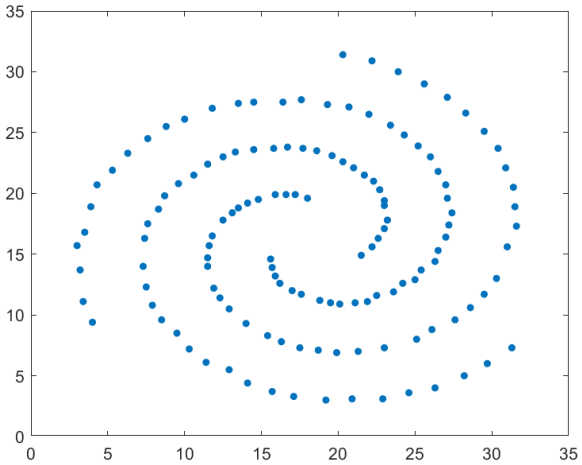


Data č. 4

Data č. 3 a 4 jsou podobná datům č. 1 a 2, jsou však několikanásobně menší. To nám zajistí, že hierarchickým shlukováním vznikne přehledný dendrogram.

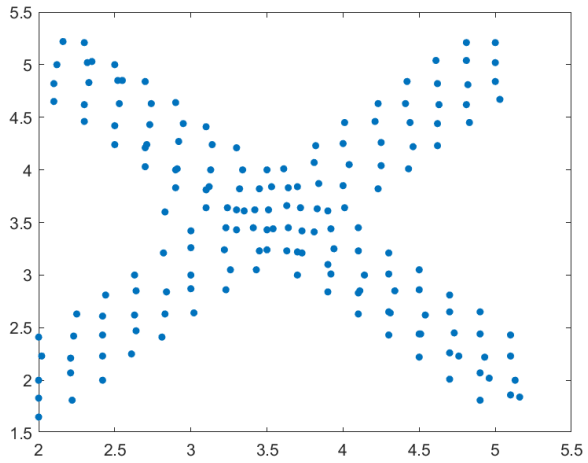


Data č. 5

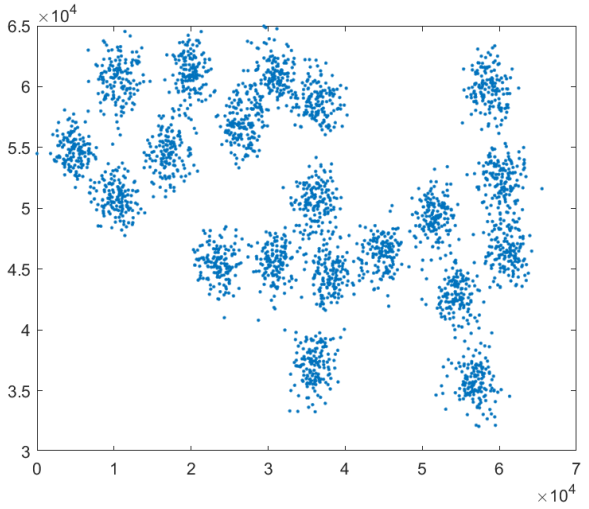


Data č. 6

Data vlevo jsou tvořena dvěma shluky zaseklými do sebe. Pro lidské oko je jednoduché rozklad do shluků vytvořit, pro mnohé shlukovací algoritmy je to však problematické. Podobně to je s daty tvaru spirály napravo.



Data č. 7



Data č. 8

Data č. 7 jsou v podstatě tvořena dvěma překrývajícími se shluky. Datový soubor vpravo (pochází z [3]) obsahuje přibližně 3000 bodů a 20 shluků, použijeme jej k testování časové náročnosti výpočtu.

Poznámka

Datové soubory jsme očíslovali od 1 do 8. Pomocí těchto čísel je budeme v této práci označovat. Bude-li například obrázek popsán jako *Dendrogram 4*, znamená to, že vznikl z dat č. 4.

V úlohách budeme dále určovat nepodobnosti objektů jakožto euklidovskou vzdálenost a při hierarchickém shlukování zavedeme koeficient nepodobnosti shluků metodou nejbližšího souseda.

Zkratkou SSE (*sum of squared errors*) budeme značit sumu čtverců nepodobností objektů a centroidů příslušných shluků.

2.3. MATLAB

MATLAB nabízí řadu známých shlukovacích algoritmů. K jejich aplikaci potřebuje uživatel základní znalosti práce v matlabovském prostředí. U hierarchického shlukování nabízí program všechny základní možnosti volby koeficientu nepodobnosti shluků: *single* pro metodu nejbližšího souseda, *complete* pro metodu nejvzdálenějšího souseda, *centroid* pro centroidní metodu a další metody. Zvolit si můžeme i způsob zavedení nepodobnosti objektů, například *euclidean* pro euklidovskou metriku nebo *cityblock* pro manhattanskou metriku. Na základě výsledků pak lze vykreslit dendrogram a určit rozklad do konkrétního počtu shluků.

Z metod nehierarchického shlukování jsou k dispozici k-means a k-medoids. Počáteční rozklad je přitom vytvářen metodou k-means++. Na počátku si musíme zvolit počet shluků. Výhodou je, že je možné nastavit, aby se algoritmus provedl mnohokrát pro různé počáteční rozklady a z výsledků se pak vybere ten s nejmenší hodnotou SSE.

2.3.1. Hierarchické shlukování

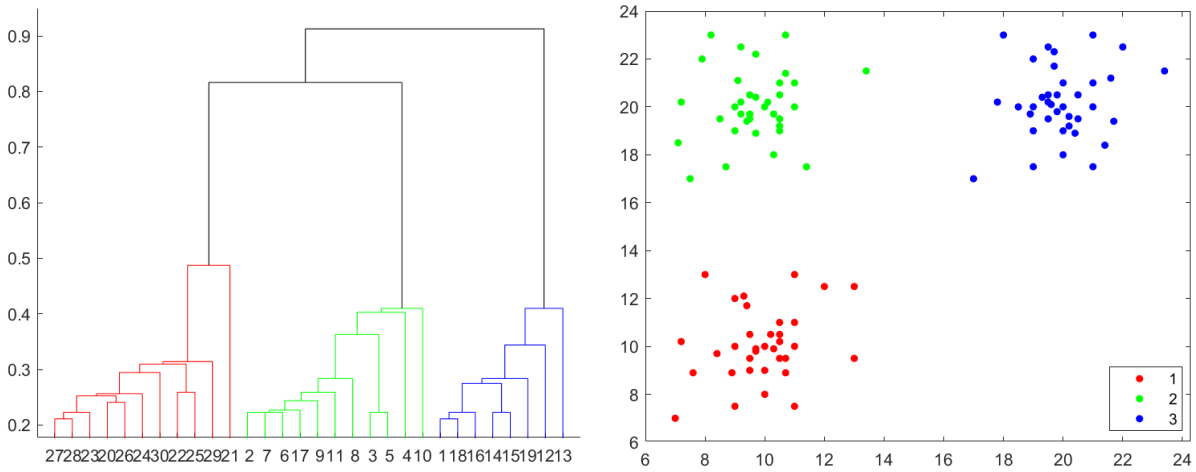
Ukázka kódu

```
Z = zscore(X);
vzd = pdist(Z,'euclidean');
Y = linkage(vzd,'single');

figure(1)
dendrogram(Y)

figure(2)
T = cluster(Y,'maxclust',3);
gscatter(X(:,1),X(:,2),T)
```

Obrázek 2.5: Kód - MATLAB (hierarchické shlukování)

Po úvodním načtení dat do matice X jsme využili funkci *zscore* ke standardizaci dat. Následně pomocí *pdist* spočítáme nepodobnosti objektů. V kódu jsme zvolili euklidovskou vzdálenost. Funkce *linkage* provede samotné aglomerativní hierarchické shlukování. Musíme přitom vybrat, který koeficient nepodobnosti shluků uvažujeme. Výsledky této procedury jsou zašifrovány v matici Y. Příkazem *dendrogram* pak můžeme výsledky zobrazit do podobnostního stromu. Chceme-li na základě hierarchického shlukování vytvořit rozdělení do konkrétního počtu shluků, využijeme funkci *cluster*. V kódu na obrázku 2.5 jsme tak vytvořili rozklad do 3 shluků. Vizualizovat tento rozklad pro původní matici dat X lze funkcí *gscatter*.

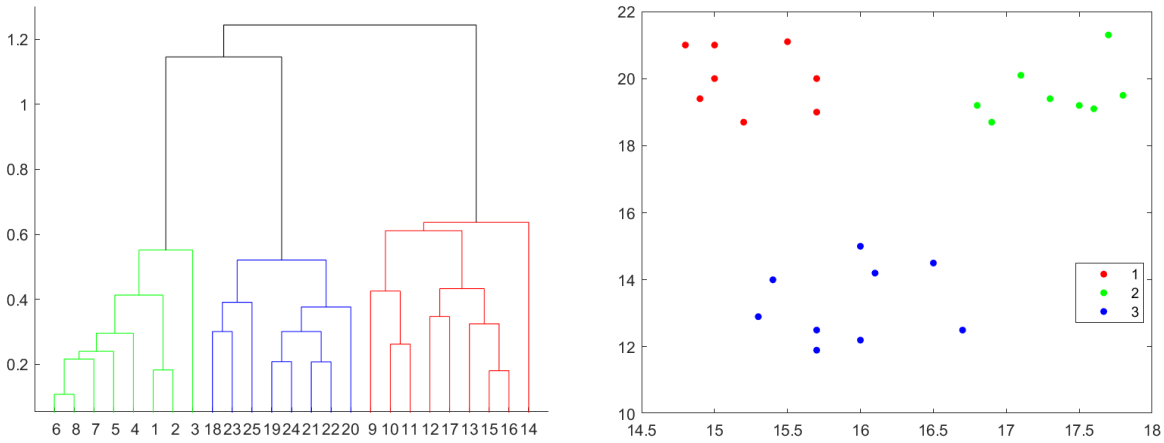
Výsledky



(a) Dendrogram 1

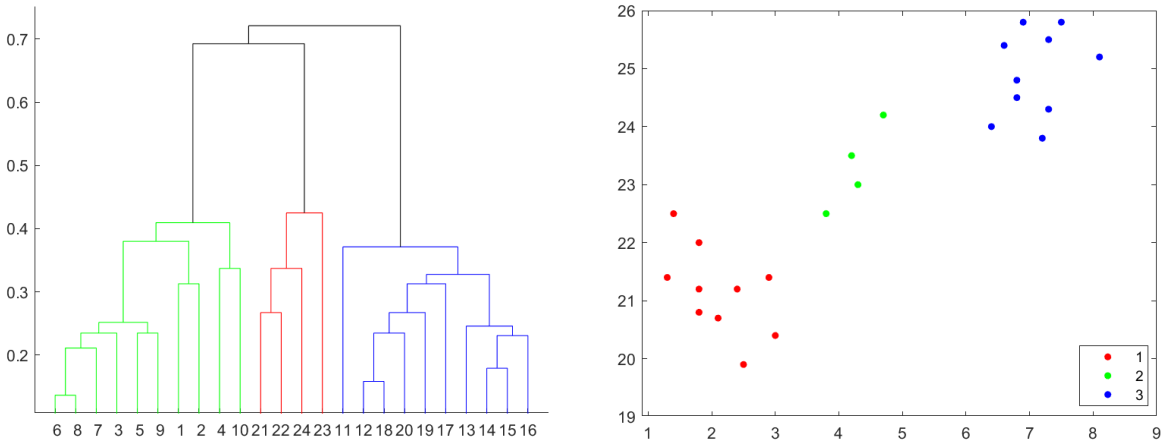
(b) Vizualizace shluků 1

Obrázek 2.6: Hierarchické shlukování 1 - MATLAB



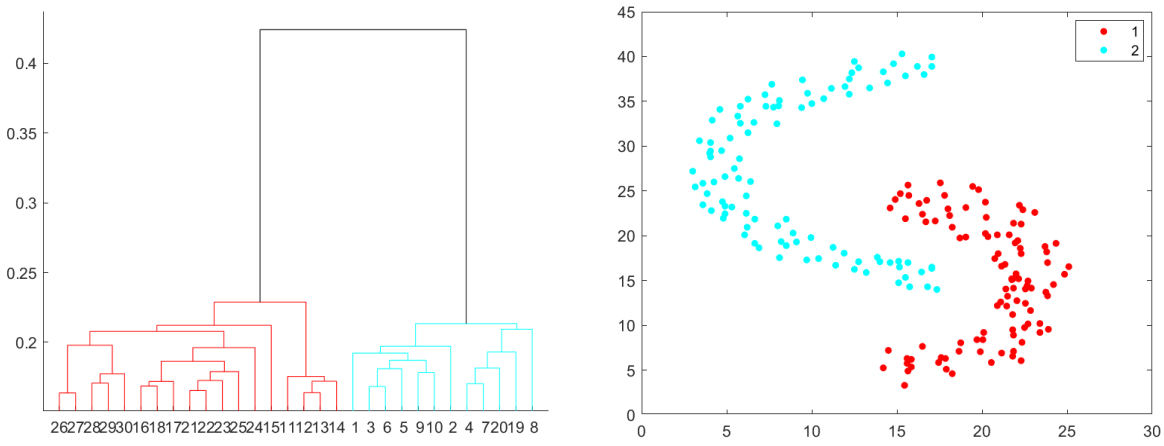
(a) Dendrogram 3 (b) Vizualizace shluků 3

Obrázek 2.7: Hierarchické shlukování 3 - MATLAB



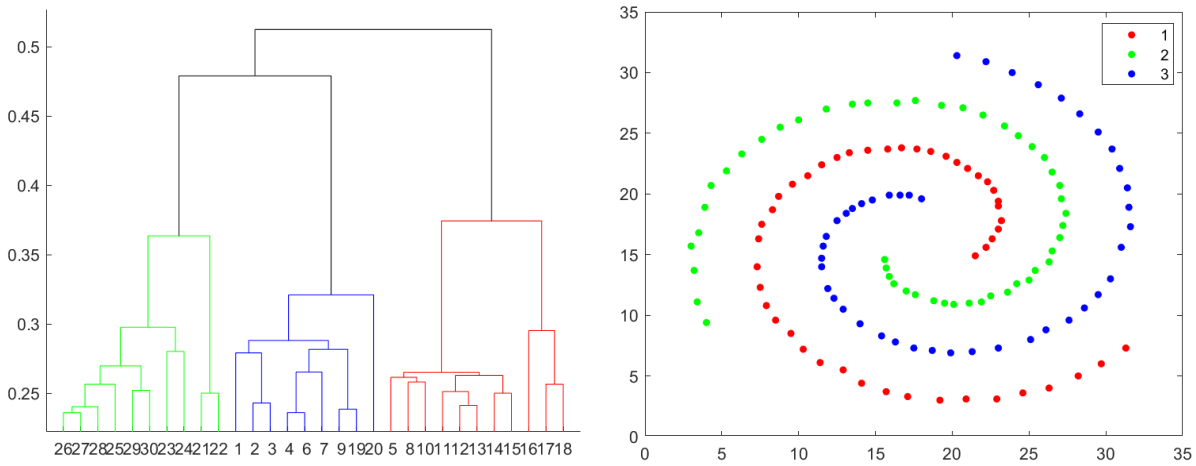
(a) Dendrogram 4 (b) Vizualizace shluků 4

Obrázek 2.8: Hierarchické shlukování 4 - MATLAB



(a) Dendrogram 5 (b) Vizualizace shluků 5

Obrázek 2.9: Hierarchické shlukování 5 - MATLAB



(a) Dendrogram 6

(b) Vizualizace shluků 6

Obrázek 2.10: Hierarchické shlukování 6 - MATLAB

2.3.2. Nehierarchické shlukování

Ukázka kódu na obrázku 2.11 je pro metodu k-means. Pokud bychom chtěli využít metodu k-medoids, pouze bychom v kódu zaměnili příkazy *kmeans* a *kmedoids*.

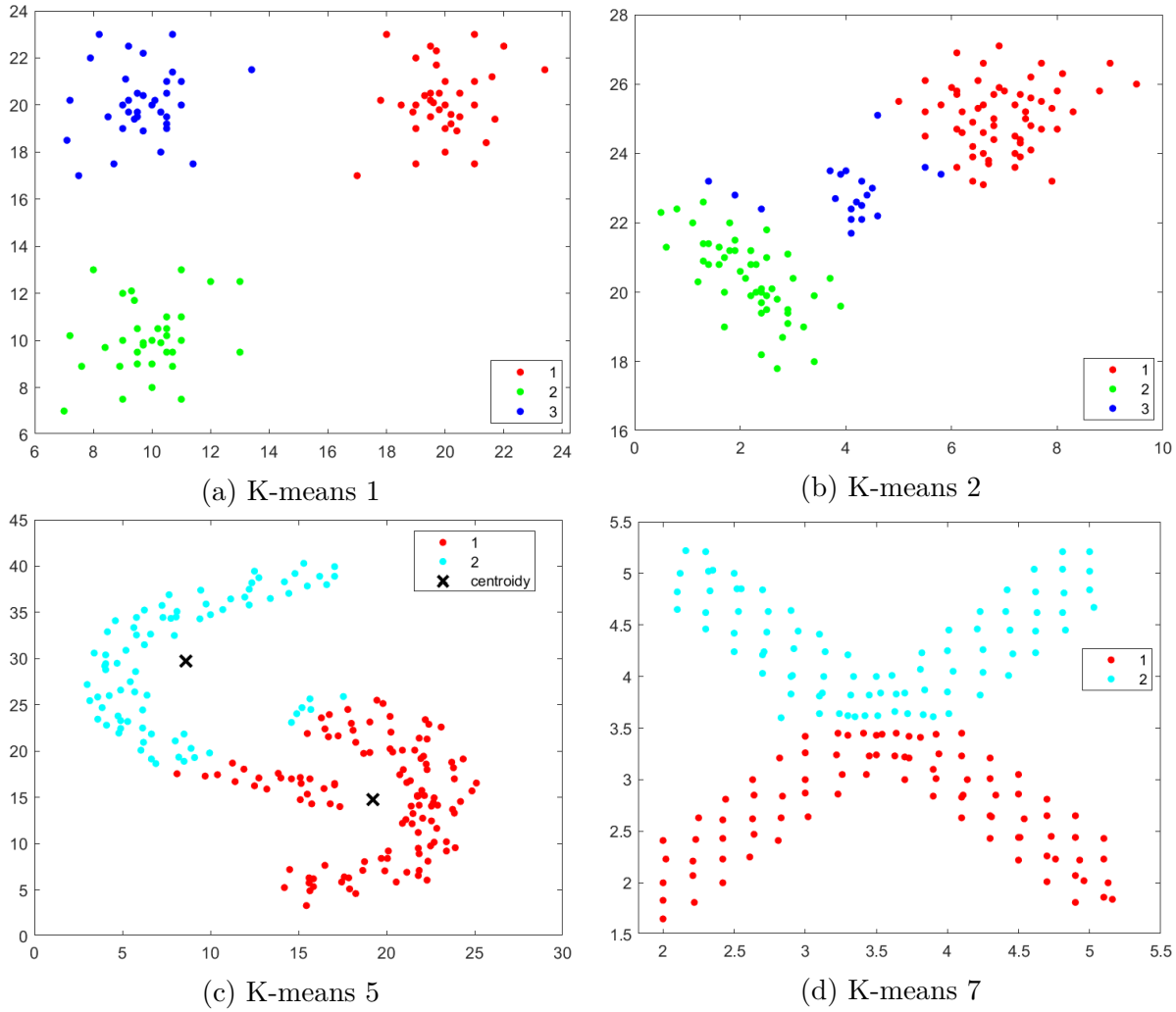
Ukázka kódu

```
[idx,C] = kmeans(X,20,'Distance','euclidean', 'Replicates',50);
gscatter(X(:,1),X(:,2),idx);
```

Obrázek 2.11: Kód - MATLAB (nehierarchické shlukování)

Data ke shlukování máme opět v matici X. Na rozdíl od hierarchického shlukování nemusíme standardizovat data, metody to udělají. Výsledky metody ukládáme v kódu výše do vektoru *idx*, jenž udává rozdělení objektů do shluků. Do matice *C* pak ukládáme souřadnice centroidů (medoidů v případě k-medoids) jednotlivých shluků. Pokud pro nás umístění centroidů není důležité, můžeme to z kódu vypustit. Prvním argumentem příkazu *kmeans* je matice dat *X*, dalším je volba počtu shluků (v našem případě 3). Pomocí *'Distance'* a následného *'euclidean'* jsme rozhodli, že k určení nepodobnosti objektů využijeme euklidovskou vzdálenost. Prostřednictvím příkazu *'Replicates', 50* se zajistí, že se shlukovací procedura provede 50krát. Ze získaných rozkladů se pak vybere ten s minimální hodnotou SSE. Vizualizaci provedeme stejně jako u hierarchického shlukování příkazem *gscatter*.

Výsledky



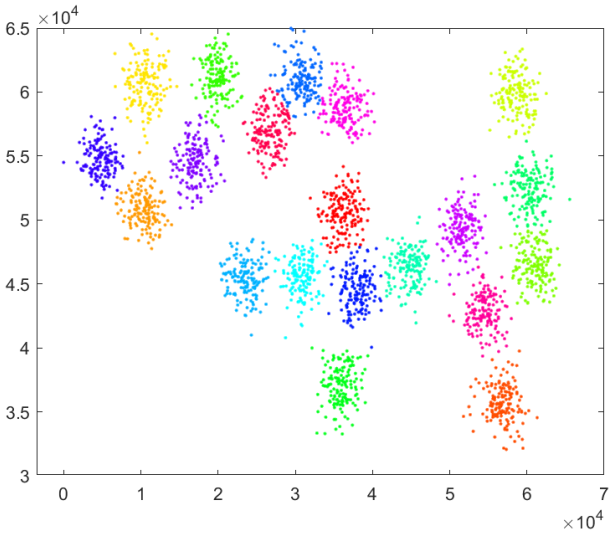
(a) K-means 1

(b) K-means 2

(c) K-means 5

(d) K-means 7

Obrázek 2.12: K-means - MATLAB



Obrázek 2.13: K-means 8 - MATLAB

Tento problém s 3000 objekty slouží k otestování časových nároků programu. Vykonat algoritmus k-means pro 50 různých počátečních rozkladů trvalo MATLABU 1,9 sekundy. Metoda k-medoids potřebovala zhruba čtyřikrát delší čas, a to 7,7 sekund.

2.4. R

Programovací jazyk R nabízí všechny nejznámější shlukovací algoritmy. Při hierarchickém shlukování můžeme využít mnoho různých způsobů zavedení nepodobnosti shluků (*single*, *complete*, *average*, *centroid*, *ward*, …). Před samotnou procedurou je vhodné standardizovat data, což je možné jediným řádkem kódu. Podle výsledků pak lze jednoduše vykreslit dendrogram a určit rozklad do libovolného počtu shluků. Z nehierarchických metod je k dispozici nejpopulárnější metoda k-means. Uživatel zvolí počet shluků, poté má možnost zvolit, pro kolik různých počátečních rozkladů chce algoritmus provést. Pokud je jich více než 1, je za konečný výsledek považován rozklad, u nějž je minimální hodnota SSE. Výchozím způsobem určování nepodobnosti objektů je euklidovská metrika, uživatel však může zvolit i jinou (*manhattan*, *maximum*, …). Metodu lze též omezit maximálním počtem iterací, lepších výsledků však bývá dosaženo, když je algoritmu umožněn libovolný počet iterací.

2.4.1. Hierarchické shlukování

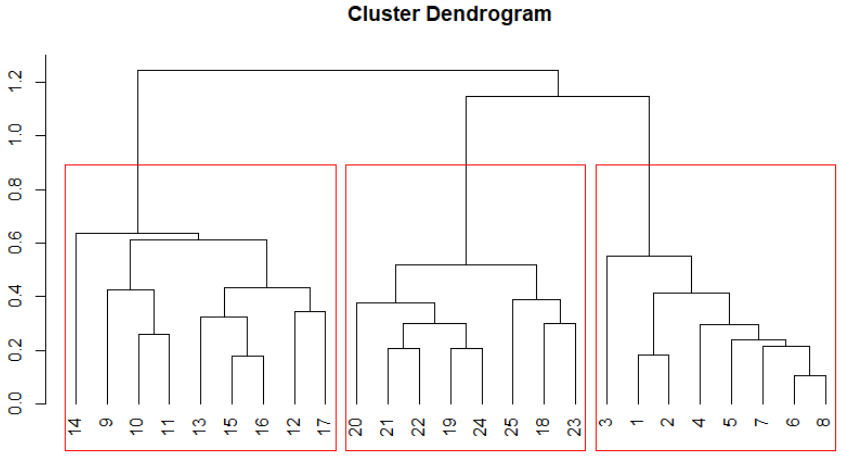
Ukázka kódu

```
mydatas <- scale(mydata)
vzd <- dist(mydatas)
dendrogram <- hclust(vzd,method='single')
plot(dendrogram)
rect.hclust(dendrogram, k=3, border="red")
```

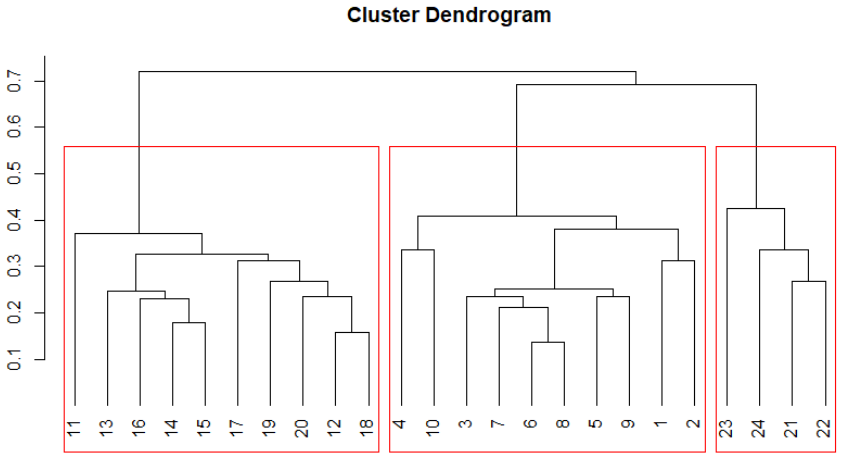
Obrázek 2.14: Kód - R (hierarchické shlukování)

Matici dat máme nazvanou *mydata*. Následně provedeme standardizaci pomocí *scale*, standardizovaná matici dat je pak nazvána *mydatas*. Dále spočítáme nepodobnosti všech objektů použitím *dist*, využije se při tom euklidovská metrika, pokud nespecifikujeme jinak. Příkaz *hclust* provede hierarchického shlukování, jehož výsledky uložíme do *dendrogram*. Vybrat při tom musíme, jaký koeficient nepodobnosti shluků chceme použít. Poté vykreslíme podobnostní strom. Poslední řádek kódu ve vykresleném dendrogramu zvýrazní vzniklé shluky červeným rámečkem.

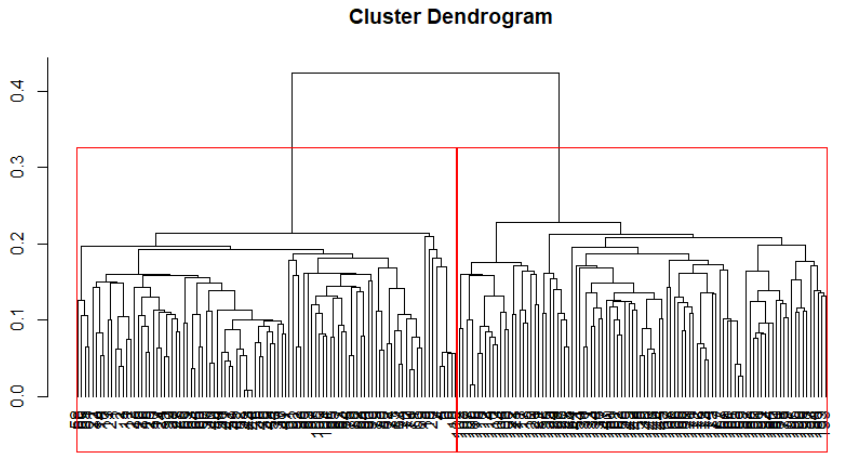
Výsledky



Obrázek 2.15: Dendrogram 3 - R



Obrázek 2.16: Dendrogram 4 - R



Obrázek 2.17: Dendrogram 5 - R

2.4.2. Nehierarchické shlukování

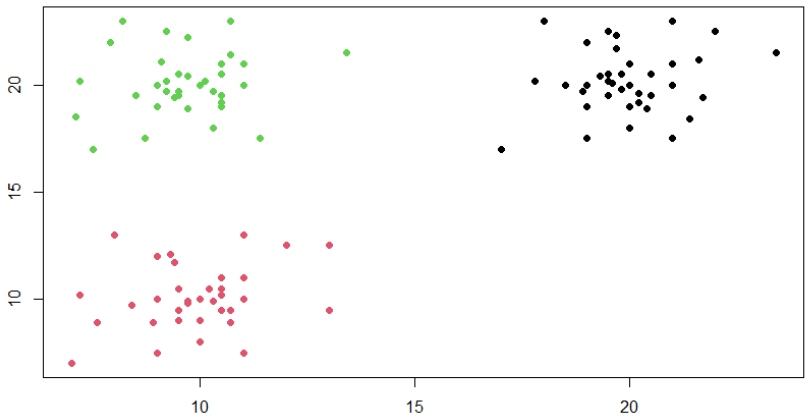
Ukázka kódu

```
kc<-kmeans(mydata,3, nstart = 50)
plot(mydata, col=kc$cluster, pch = 19)
kc$tot.withinss
```

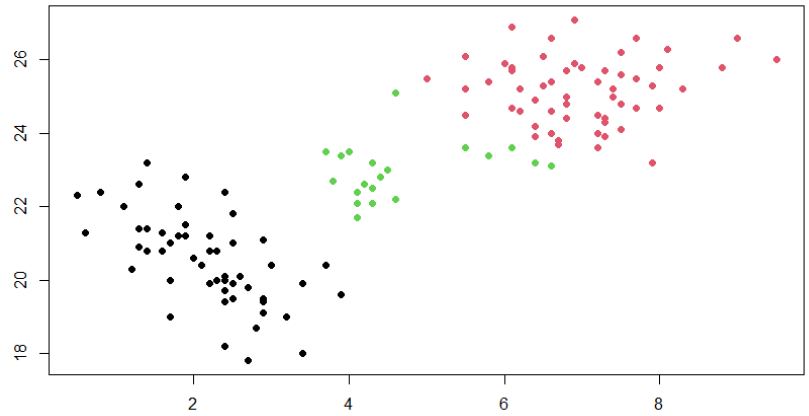
Obrázek 2.18: Kód - R (k-means)

V prvním řádku provedeme shlukování metodou k-means použitím funkce *kmeans*. V argumentech funkce zadáváme matici dat *mydata*, počet shluků rozkladu (v našem případě 3) a údajem *nstart = 50* požadujeme, aby se algoritmus provedl pro 50 různých počátečních rozkladů a z nich se pak vybral ten nejkvalitnější. Výsledky procedury ukládáme do *kc*. Na druhém řádku vykreslíme výsledky shlukování, *col=kc$cluster* určuje, že body v jednom shluku budou vykresleny stejnou barvou. Na posledním řádku je pouze ukázáno, jak lze přistupovat k údaji o hodnotě SSE u výsledného rozkladu.

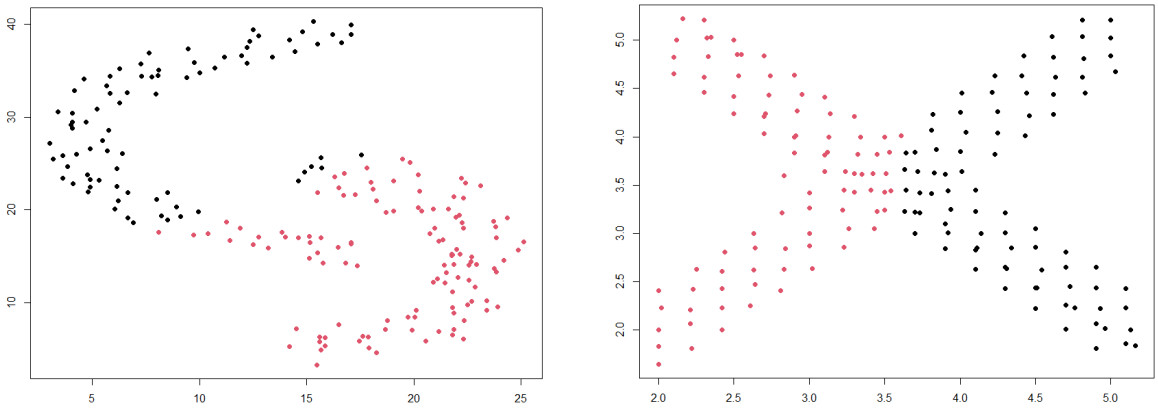
Výsledky



Obrázek 2.19: K-means 1 - R



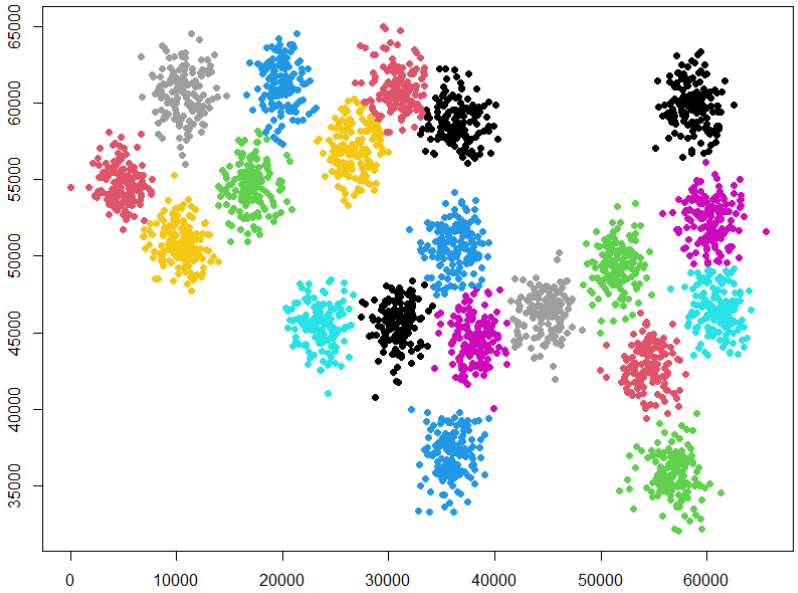
Obrázek 2.20: K-means 2 - R



(a) K-means 5

(b) K-means 7

Obrázek 2.21: K-means - R



Obrázek 2.22: K-means 8 - R

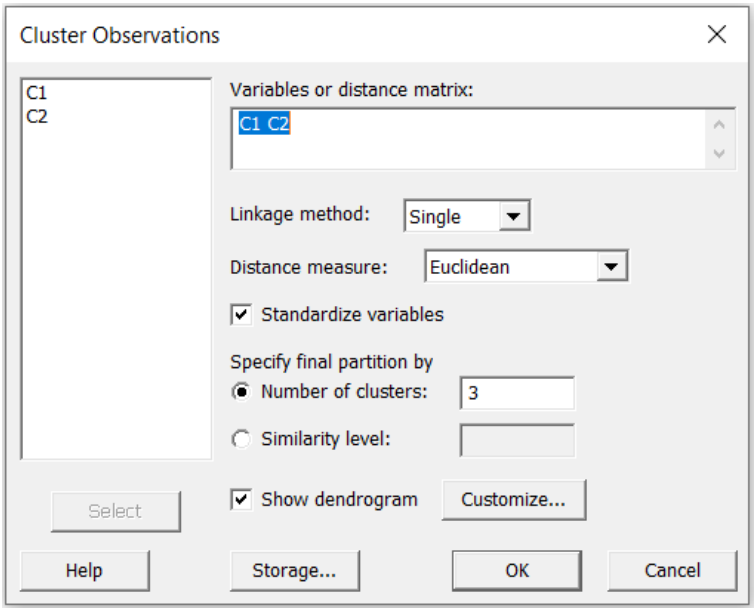
Provést k-means algoritmus pro 50 různých počátečních rozkladů trvalo programu pouze necelou sekundu.

2.5. Minitab

Minitab nabízí hierarchické shlukování s řadou nastavitelných možností. V menu se k němu dostaneme přes **Stat > Multivariate > Cluster Observations**. Z nehierarchických metod je dostupná metoda k-means. Nalezneme ji v nabídce pod **Stat > Multivariate > Cluster K-Means**.

2.5.1. Hierarchické shlukování

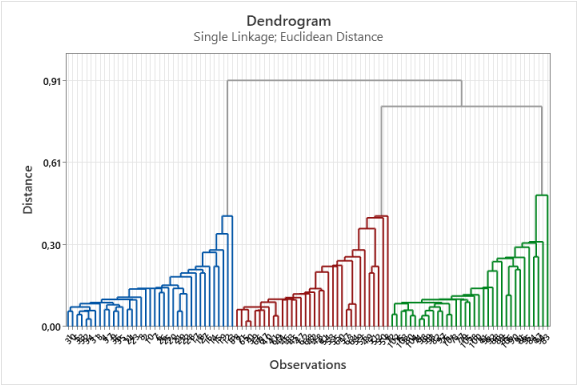
Ukázka prostředí



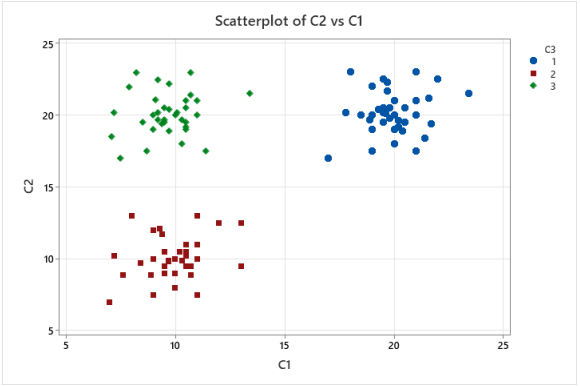
Obrázek 2.23: Minitab - možnosti hierarchického shlukování

Nejdříve vybereme proměnné, které ovlivní shlukování. V našem případě máme objekty popsány proměnnými *C1* a *C2*. Poté vybereme jednu ze sedmi metod zavedení nepodobnosti shluků (*Linkage method*). Na výběr je také pět způsobů výpočtu nepodobnosti objektů (*Distance measure*). Je vhodné zaškrtnout možnost standardizace dat. Rozklad do shluků může být určen buď podle počtu shluků, nebo podle hladiny dané nepodobností shluků. Výsledkem hierarchického shlukování jsou kromě dendrogramu také údaje o vytvořených shlucích (počet objektů ve shluku, poloha centroidů shluků a další). Pokud chceme shluky vizualizovat ve dvourozměrném grafu, je nutné to udělat zvlášť přes **Graph > Scatterplot**.

Výsledky

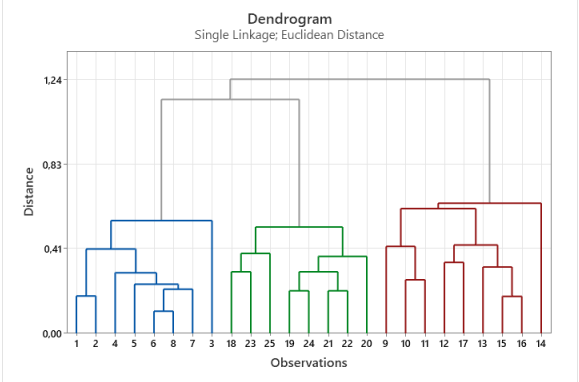


(a) Dendrogram 1

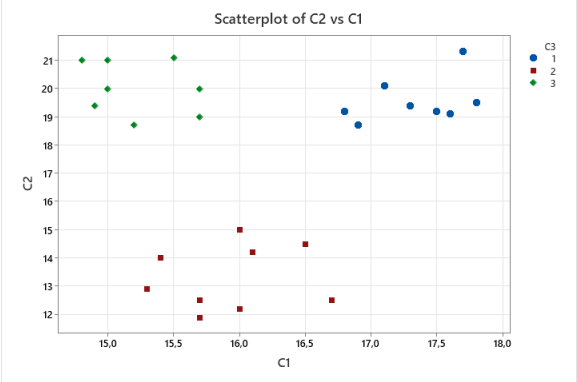


(b) Vizualizace shluků 1

Obrázek 2.24: Hierarchické shlukování 1 - Minitab

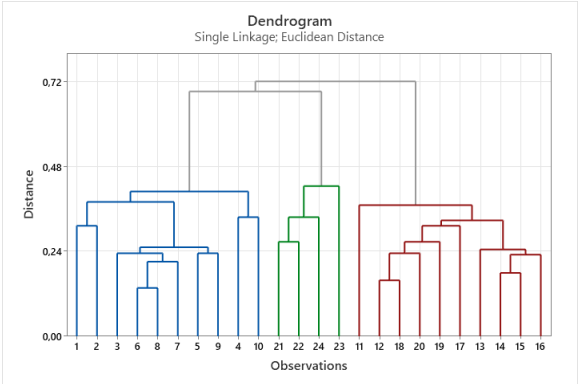


(a) Dendrogram 3

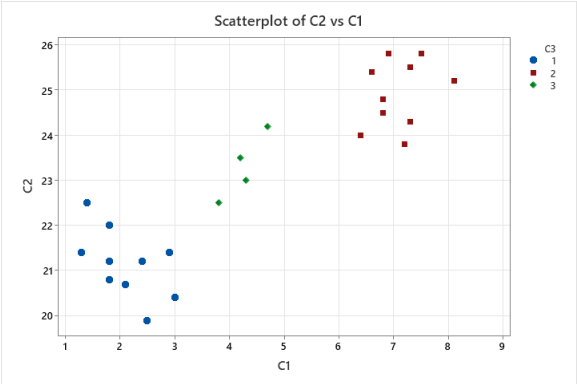


(b) Vizualizace shluků 3

Obrázek 2.25: Hierarchické shlukování 3 - Minitab



(a) Dendrogram 4

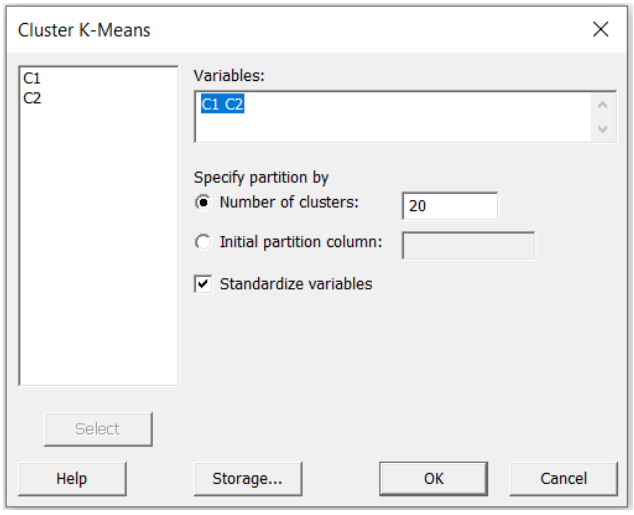


(b) Vizualizace shluků 4

Obrázek 2.26: Hierarchické shlukování 4 - Minitab

2.5.2. Nehierarchické shlukování

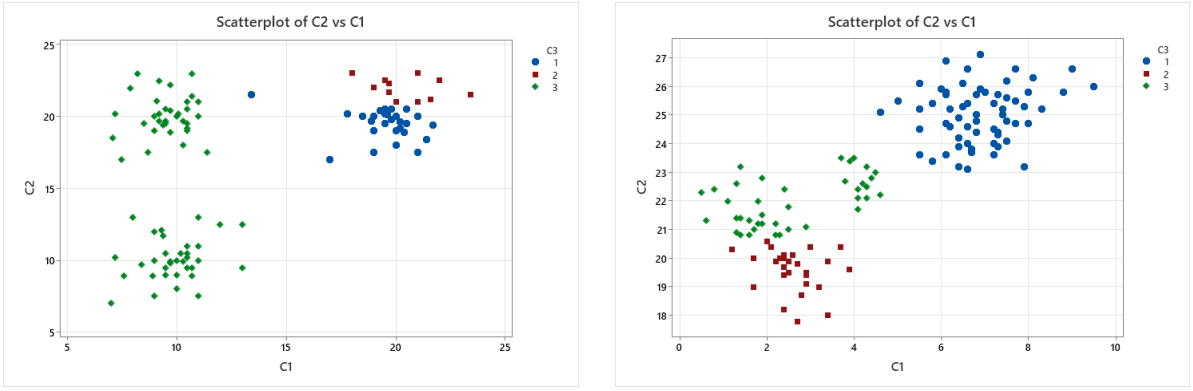
Ukázka prostředí



Obrázek 2.27: Minitab - možnosti k-means

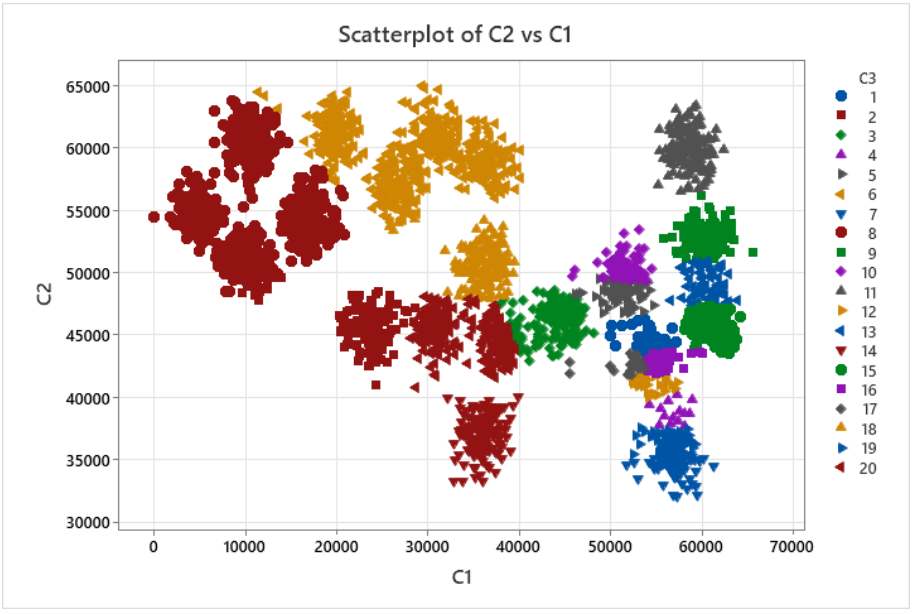
Nejdříve opět zvolíme proměnné, na základě kterých mají být objekty shlukovány (na obrázku výše jsou to *C1* a *C2*). Poté specifikujeme, kolik shluků má vzniknout. Po rozkliknutí *Storage* lze požadovat, aby do nové proměnné (například *C3*) byl každému objektu přiřazen index shluku, do nějž patří. Na základě toho pak můžeme shluky vykreslit přes **Graph > Scatterplot** v menu, protože výsledkem samotného shlukování jsou pouze údaje o shlucích bez vizualizace.

Výsledky



(a) K-means 1 (b) K-means 2

Obrázek 2.28: K-means - Minitab



Obrázek 2.29: K-means 8 - Minitab

K-means metoda se sice pro náš největší datový soubor provedla v pár desetinách sekundy, výsledek však nelze považovat za správný.

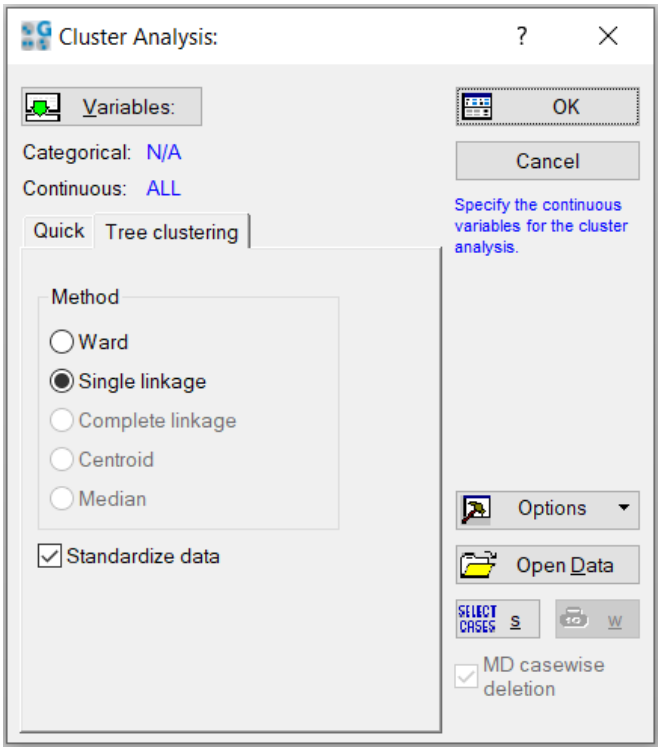
2.6. STATISTICA

Shlukovou analýzu v programu STATISTICA najdeme pod názvem **Cluster** v záložce **Data Mining**. Po rozkliknutí se nám zobrazí nabídka shlukovacích metod. K dispozici je metoda k-means, pod názvem *Tree Clustering* nalezneme hierarchické shlukování a použít lze i metodu EM, tou se však tato práce nezabývá.

Po zvolení metody nesmíme zapomenout ve *Variables* vybrat proměnné, které mají zasahovat do shlukové analýzy. Tyto proměnné označíme jako *Continuous*.

2.6.1. Hierarchické shlukování

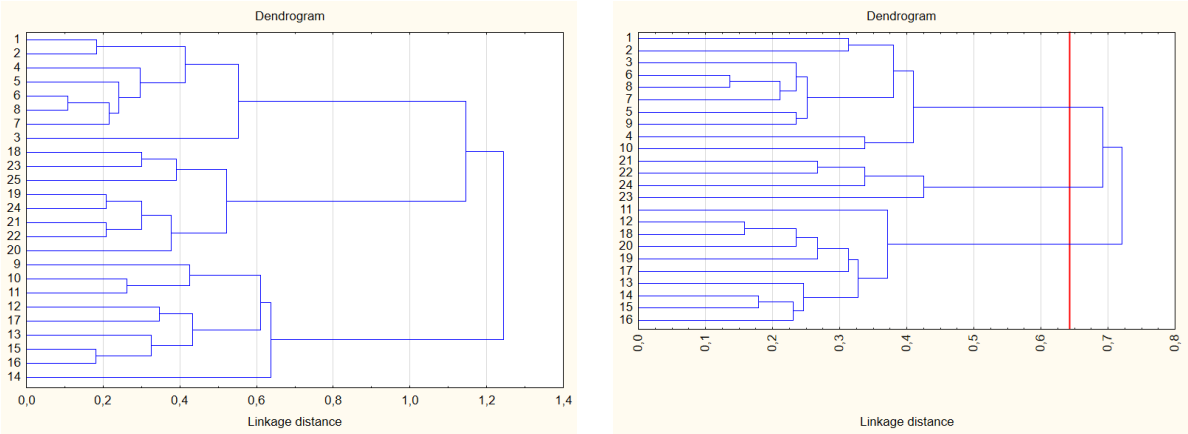
Ukázka prostředí



Obrázek 2.30: STATISTICA - možnosti hierarchického shlukování

Program umožňuje volbu koeficientu nepodobnosti shluků a standardizaci dat. Výsledkem procedury je pak horizontálně orientovaný dendrogram, jehož podstata je identická s dříve uvedenými dendrogramy. Dalším výstupem je řada statistik týkajících se vzniklých shluků.

Výsledky

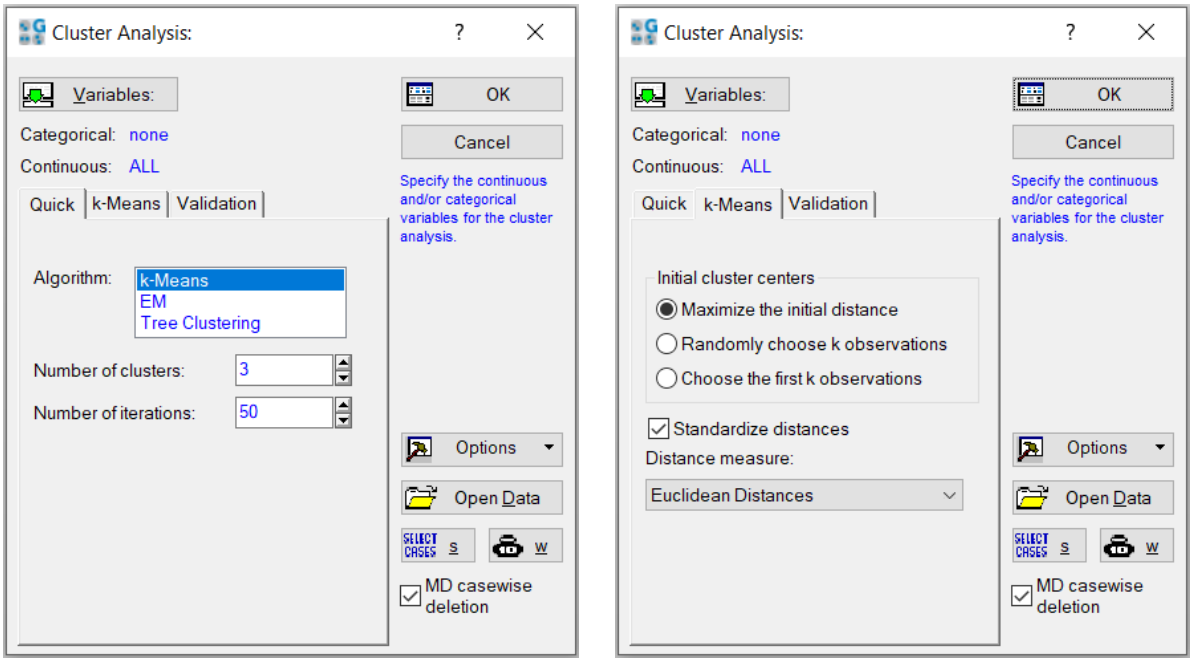


(a) Dendrogram 3 (b) Dendrogram 4

Obrázek 2.31: Hierarchické shlukování - STATISTICA

2.6.2. Nehierarchické shlukování

Ukázka prostředí

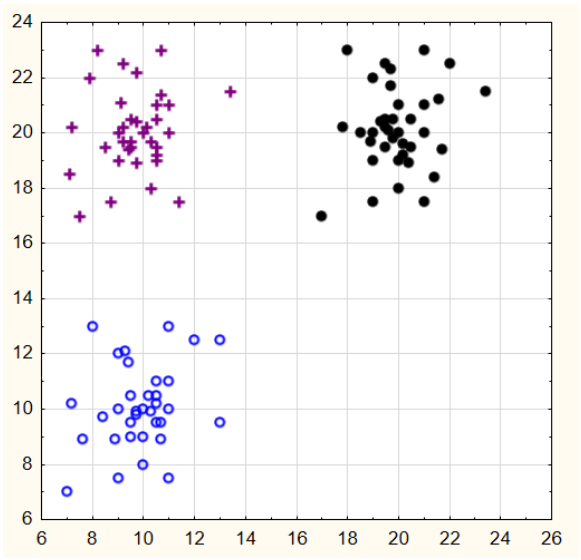


Obrázek 2.32: STATISTICA - možnosti metody k-means

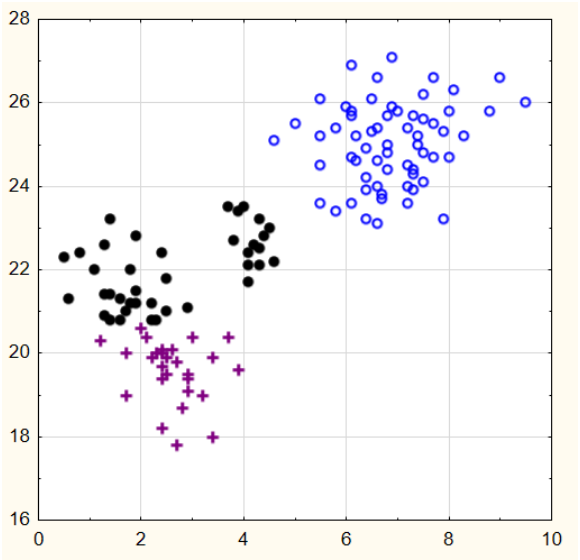
Nejprve určíme počet shluků (*Number of clusters*) a dostatečně velký maximální počet iterací (*Number of iterations*). Na výběr jsou tři možnosti vytvoření počátečního rozkladu. Můžeme požadovat, aby počáteční centroidy byly co nejdále od sebe (*Maximize the initial distance*), nebo lze náhodně zvolit k bodů (*Randomly choose k obvervations*) nebo vybereme prvních k objektů (*Choose the first k observations*). Nepodobnost objektů (*Distance*

measure) lze měřit čtyřmi různými způsoby. Výstupem procedury je pak rozklad objektů do shluků, který můžeme vizualizovat použitím **Graphs > Scatterplot**. Dále obdržíme statistické popisy vzniklých shluků (pozice centroidů shluků, analýza rozptylu bodů ve shluku, míra vlivu jednotlivých proměnných na shlukování a podobně).

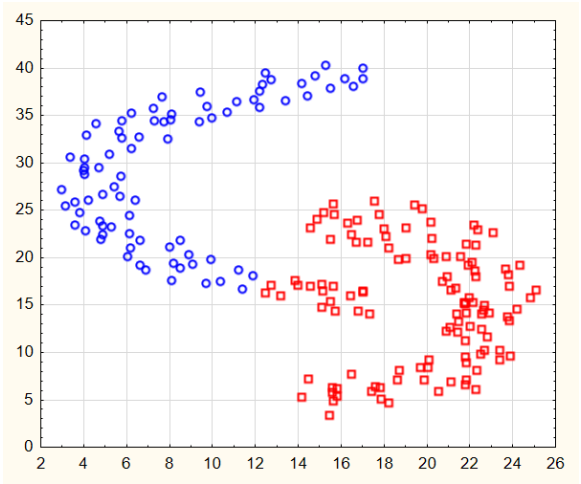
Výsledky



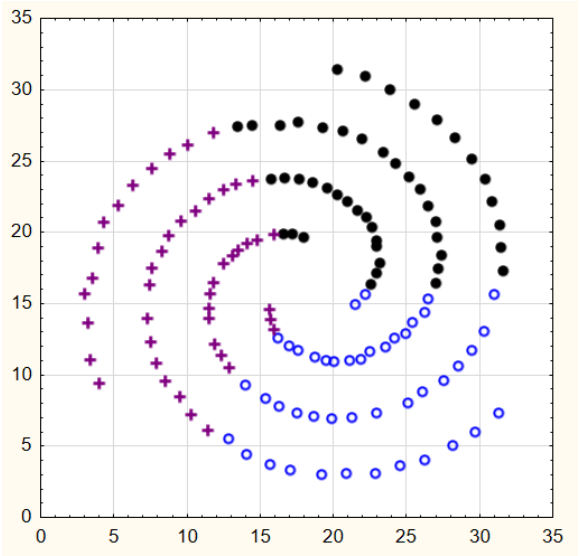
(a) K-means 1



(b) K-means 2

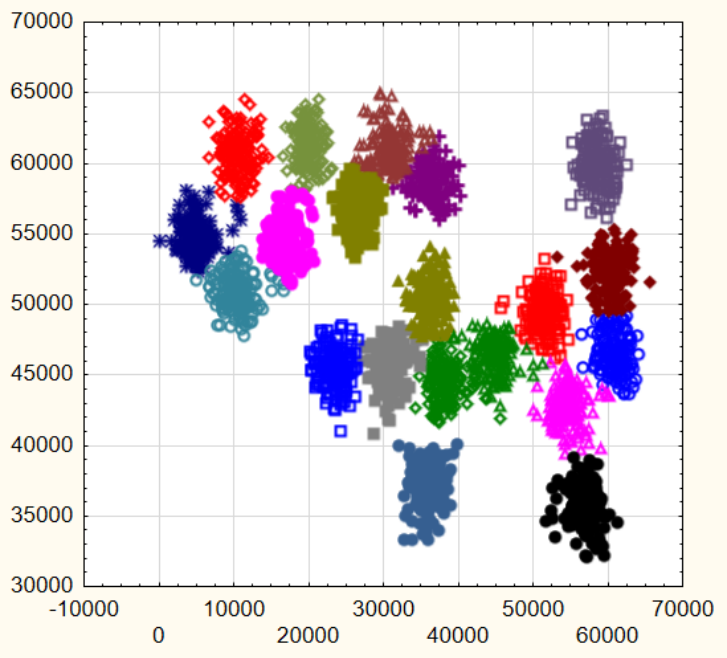


(c) K-means 5



(d) K-means 6

Obrázek 2.33: K-means - STATISTICA



Obrázek 2.34: K-means 8 - STATISTICA

Software si metodou k-means poradil s tímto datovým souborem téměř okamžitě.

Závěr

V bakalářské práci jsme v kapitole 1 zmínili pojmy nezbytné pro pochopení principů shlukové analýzy. Následoval popis základních shlukovacích metod. Představili jsme si algoritmus hierarchického shlukování, objasněn byl význam dendrogramu. Z metod nehierarchického shlukování jsme se zaměřili na populární metodu k-means a věnovali jsme se problému počátečního rozkladu.

V kapitole 2 jsme ve čtyřech různých programech vyzkoušeli aplikovat metody shlukové analýzy. Jednalo se o dva obecnější programy (MATLAB a R) a o dva softwary zaměřené na statistiku a analýzu dat (Minitab, STATISTICA).

Všechny tyto programy obsahují metody aglomerativního hierarchického shlukování a výsledky získané tímto přístupem jsou v programech identické. V MATLABu a R bylo nutné naprogramovat několikařádkový kód, zatímco v Minitabu a STATISTICE stačilo možnosti hierarchického shlukování navolit v menu. Uživatelsky nejpřívětivější byl Minitab, který umožňuje také pěknou vizualizaci výsledků. Subjektivně bylo nejjednodušší při hierarchickém shlukování pracovat právě s Minitabem. Nejpodrobnější statistický popis shluků zase poskytuje STATISTICA, jejíž grafické výstupy však jsou méně přívětivé. Ve všech programech se také ukázalo, že pro objemnější data vzniká nepřehledný dendrogram, řešením je nezobrazovat dendrogram celý, ale pouze jeho horní část, jako je tomu na obrázcích 2.6a či 2.9a.

Využité softwary také bez výjimky obsahují metodu k-means. V programu Minitab jsme pomocí této metody nedosáhli dobrých výsledků, což je patrné z obrázků 2.28a a 2.29. Výsledky v dalších programech byly příznivější, avšak odhalili jsme některá omezení metody k-means. Na obrázcích 2.12b, 2.20 a 2.33b vidíme, že se metodě nepodařilo rozeznat malý shluk mezi dvěma většími shluky v datovém souboru č. 2, metoda k-means totiž tíhne k vytváření stejně velkých shluků. Na obrázcích 2.12c, 2.21a, 2.33c nebo 2.33d se zase projevuje, že metoda předpokládá tvar shluků sférického charakteru, a nemůže tedy rozlišit shluky do sebe zaseknuté. Jako výhoda metody k-means se ukázala rychlost, když si s datovým souborem o 3000 objektech algoritmus poradil v krátkém čase. Menší prodlevu šlo pozorovat pouze u softwaru MATLAB. Předností R a MATLABu je možnost nechat algoritmus provést pro zvolené množství různých počátečních rozkladů a z výsledných rozkladů do shluků pak vybrat ten nejkvalitnější. Subjektivně působilo nehierarchické shlukování nejpříjemněji v MATLABu a v R, kde šlo provést jednoduchým kódem a kde vznikaly pěkné a přehledné vizualizace.

Literatura

[1] ANDERBERG, Michael R. *Cluster analysis for applications*. New York: Academic Press, 1973. ISBN 9780120576500.

[2] BERANOVÁ, Petra, Lenka BLAŽKOVÁ a Miloš ULDRICH. *Manuál k ovládání programu STATISTICA* [online]. StatSoft CR, 2012 [cit. 2021-5-19]. Dostupné z: http://www.statsoft.cz/file1/PDF/manualy/Manual_k_ovladani_programu_STATISTICA.pdf

[3] FRÄNTI, P. a S. SIERANOJA. K-means properties on six clustering benchmark datasets. *Applied Intelligence* [online]. Prosinec 2018, **48**(12), 4743-4759 [cit. 2021-5-19]. Dostupné z: http://cs.joensuu.fi/sipu/datasets/

[4] GAN, Guojun, Chaoqun MA a Jianhong WU. *Data clustering: theory, algorithms, and applications*. Philadelphia: SIAM, Society for Industrial and Applied Mathematics, c2007. ISBN 978-0-898716-23-8.

[5] GIORDANI, Paolo, Maria Brigida FERRARO a Francesca MARTELLA. *An Introduction to Clustering with R*. Springer Nature Singapore Pte, 2020. ISBN 978-981-13-0553-5.

[6] GUEVARA ALVEZ, Pamela Beatriz. *Inference of a human brain fiber bundle atlas from high angular resolution diffusion imaging* [online]. Université Paris Sud, dizertační práce, 2011 [cit. 2021-5-19]. Dostupné z: https://tel.archives-ouvertes.fr/tel-00638766

[7] KABACOFF, Robert I. Cluster Analysis. *Quick-R* [online]. [cit. 2021-5-19]. Dostupné z: https://www.statmethods.net/advstats/cluster.html

[8] LUKASOVÁ, Alena a Jana ŠARMANOVÁ. *Metody shlukové analýzy*. Praha: Státní nakladatelství technické literatury, 1985.

[9] TRIOLA, Mario F. *Minitab Manual for the Triola Statistics Series*. 12. Pearson, 2014. ISBN 978-0321833792.

[10] *MathWorks* [online]. [cit. 2021-5-19]. Dostupné z: https://www.mathworks.com/