

TECHNICKÁ UNIVERZITA V KOŠICIACH
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

**Popisné modely špecifických skupín pacientov vzhľadom
na ich kardiovaskulárne riziko**

Diplomová práca

2022

Bc. Adam Pavlíček

TECHNICKÁ UNIVERZITA V KOŠICIACH
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

**Popisné modely špecifických skupín pacientov vzhľadom
na ich kardiovaskulárne riziko**

Diplomová práca

Študijný program: Hospodárska informatika
Študijný odbor: Informatika
Školiace pracovisko: KKUI
Školiteľ: prof. Ing. Ján Paralič, PhD.
Konzultanti: Ing. Zuzana Pella
Ing. Oliver Lohaj

Abstrakt v SJ

Táto diplomová práca sa zaoberá problematikou aplikovania popisných modelov na špecifickú skupinu pacientov Východoslovenského ústavu srdcových a cievnych chorôb. Pri predkladanej diplomovej práci možno definovať hlavný cieľ ako návrh, implementáciu a následné overenie aplikácie pre podporu výskumu kardiológov na báze popisných modelov. Samotnú diplomovú prácu možno rozdeliť na dva samostatné celky, a teda teoretickú, resp. analytickú a praktickú časť. V prvej z nich sa zameriavame na teoretický rozbor skúmanej problematiky, analýzu súčasného stavu, pričom rozoberáme aj štúdie, ktoré mapujú využitie deskriptívnych modelov práve v oblasti medicíny a zdravotnej starostlivosti. Súčasťou teoretickej časti diplomovej práce je taktiež fáza návrhu softvérového riešenia, ktorého primárny cieľ sme charakterizovali vyššie. Praktická časť spočíva v implementácii systémového riešenia v programovacom jazyku R. Aplikácia pozostáva z troch hlavných častí, ktorých zameranie je na hľadanie asociačných pravidiel, aplikovanie metód zhukovania a hľadanie asociačných pravidiel v rámci definovaných zhukov. Záver diplomovej práce je venovaný používateľskej štúdii vykonanej na skupine lekárov, ktorí sú potenciálni používatelia implementovanej aplikácie.

Kľúčové slova v SJ

Kardiovaskulárne ochorenie, Deskriptívne modelovanie, Asociačné pravidlá, Zhukovanie, R Shiny

Abstrakt v AJ

This diploma thesis deals with the application of descriptive models to a specific group of patients from the East Slovak Institute of Cardiovascular Diseases. In the presented diploma thesis, the main objective can be defined as designing, implementing and subsequent verification of the application to support the research of cardiologists based on descriptive models. The diploma thesis itself can be divided into two separate parts - theoretical, or analytical, and practical part. In the first mentioned, we focus on a theoretical analysis of the researched issues, an analysis of the current state, while we also examine studies that map the use of descriptive models in the field of medicine and health care. One section of theoretical part of the thesis also describes the phase of designing software solution, of which primary goal was aforementioned. The practical part consists of the system solution implementation in the R programming language. The application consists of three main parts, that focus on the defining of association rules, application of clustering methods and defining association rules within defined clusters. The concluding part of the diploma thesis is dedicated to a user study performed on a group of doctors who are the potential users of the implemented application.

Klíčové slova v AJ

Cardiovascular diseases, Descriptive modelling, Association rules, Clustering, R Shiny

TECHNICKÁ UNIVERZITA V KOŠICIACH
FAKULTA ELEKTROTECHNIKY A INFORMATIKY
Katedra kybernetiky a umelej inteligencie

ZADANIE
DIPLOMOVEJ PRÁCE

Študijný odbor: **Informatika**
Študijný program: **Hospodárska informatika**

Názov práce:

Popisné modely špecifických skupín pacientov vzhľadom na ich kardiovaskulárne riziko

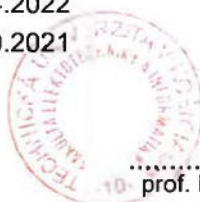
Descriptive models of specific groups of patients with respect to their cardiovascular risk

Študent: **Bc. Adam Pavlíček**
Školiteľ: **prof. Ing. Ján Paralič, PhD.**
Školiace pracovisko: **Katedra kybernetiky a umelej inteligencie**
Konzultant práce: **Ing. Zuzana Pella, Ing. Oliver Lohaj**
Pracovisko konzultanta: **Katedra kybernetiky a umelej inteligencie**

Pokyny na vypracovanie diplomovej práce:

1. Naštudovať a podať stručný prehľad o problematike popisných modelov analýzy dát v medicíne.
2. Analyzovať súčasný stav výskumu v oblasti využitia popisných modelov dátovej analýzy vo vybranej medicínskej oblasti - výskum rizika kardiovaskulárnych ochorení.
3. Navrhnuť, implementovať a vhodne overiť aplikáciu pre podporu výskumu kardiológov na báze popisných modelov.
4. Realizovať používateľskú štúdiu na skupine lekárov - potenciálnych používateľov vytvoreného systému a vhodne vyhodnotiť získané výsledky.
5. Vypracovať dokumentáciu podľa pokynov katedry a vedúceho práce (hlavná časť práce v rozsahu 50-70 strán, prílohy - používateľská a systémová príručka).

Jazyk, v ktorom sa práca vypracuje: **slovenský**
Termín pre odovzdanie práce: **22.04.2022**
Dátum zadania diplomovej práce: **29.10.2021**



prof. Ing. Liberios Vokorokos, PhD.

dekan fakulty

Čestné vyhlásenie

Vyhlasujem, že som celú diplomovú prácu vypracoval samostatne s použitím uvedenej odbornej literatúry.

Košice, 22. apríla 2022

.....

vlastnoručný podpis

Pod'akovanie

Týmto by som chcel vyjadriť svoje pod'akovanie prof. Ing. Jánovi Paraličovi, PhD. za jeho cenné rady, veľkú ochotu pri odbornom vedení a konzultáciách. Taktiež chcem poďakovať za podnetné pripomienky a usmernenie vo všetkých ohľadoch. Počas tvorby diplomovej práce ma taktiež usmerňovali, radili a pomáhali pani konzultantka, Ing. Zuzana Pella a pán Ing. Oliver Lohaj, ktorým tiež patrí veľké poďakovanie.

Obsah

Zoznam obrázkov	10
Zoznam tabuliek	12
Zoznam symbolov a skratiek	13
Úvod	14
1. Teoretický rozbor skúmanej problematiky	16
1.1. Charakteristika popisných modelov	16
1.2. Asociačné pravidlá.....	16
1.2.1. Algoritmus Apriori	18
1.2.2. Algoritmus FP-Growth	19
1.2.3. Algoritmus Eclat	19
1.3. Zhlukovanie	20
1.3.1. Hierarchické zhlukovanie	21
1.3.2. Metódy zhlukovania založené na výbere reprezentantov zhlukov	23
1.3.3. Zhlukovanie založené na hustote	24
1.3.4. Optimálny počet zhlukov.....	24
1.3.5. Validáčné kritéria	25
1.3.6. Miery podobnosti	26
2. Analýza súčasného stavu.....	32
2.1. Kardiovaskulárne ochorenie.....	32
2.2. Aplikačné oblasti popisných modelov	34
2.3. Vývoj používania popisných modelov v oblasti medicíny	36
2.4. Asociačné pravidlá a ich využitie v oblasti chronických ochorení	39
2.5. Metódy zhlukovania a ich využitie v oblasti chronických ochorení.....	41
3. Návrh systémového riešenia	44
3.1. Model životného cyklu	44
3.2. Koncept softvéru	46
3.3. Analýza používateľských požiadaviek.....	46
3.3.1. Cieľoví používatelia, ich roly a právomoci	47
3.3.2. Požiadavky používateľov podľa priorit	48
3.4. Diagram prípadov použitia	49
3.5. Architektúra navrhovaného systémového riešenia	50
3.6. Základný opis fungovania softvéru.....	52
3.6.1. Načítanie dát	53

3.6.2.	Hľadanie asociačných pravidiel	53
3.6.3.	Aplikovanie metód zhukovania	53
3.6.4.	Hľadanie asociačných pravidiel v rámci definovaných zhukov	54
4.	Charakteristika implementovaného systému	55
4.1.	Načítanie dát	56
4.2.	Hľadanie asociačných pravidiel	59
4.2.1.	Vstupy.....	60
4.2.2.	Výstupy.....	62
4.3.	Aplikovanie metód zhukovania	66
4.3.1.	Vstupy.....	66
4.3.2.	Výstupy.....	67
4.4.	Hľadanie asociačných pravidiel v rámci definovaných zhukov	72
4.4.1.	Vstupy.....	72
4.4.2.	Výstupy.....	73
5.	Testovanie implementovanej aplikácie	75
5.1.	Použitie prvej časti testovania na základe scenára	76
5.2.	Vyhodnotenie druhej časti testovania na základe dotazníka	76
	Záver	84
	Zoznam použitej literatúry	85
	Prílohy	91

Zoznam obrázkov

Obrázok 1 Využitie rôznych prístupov riešenia data mining úloh, v šiestich definovaných aplikačných oblastiach [7].....	38
Obrázok 2 Evolučný model životného cyklu vývoja softvéru [48]	45
Obrázok 3 Diagram prípadov použitia pri navrhovanej aplikácii.....	50
Obrázok 4 Architektúra navrhovaného systémového riešenia	52
Obrázok 5 Používateľské rozhranie aplikácie	55
Obrázok 6 Záložky hlavného menu	56
Obrázok 7 Panel určený pre načítavanie dát.....	56
Obrázok 8 Načítanie vlastných dát.....	57
Obrázok 9 Transformácia numerických atribútov na kategorické	58
Obrázok 10 Načítanie dát z aplikácie	59
Obrázok 11 Modálne okno s upozornením	59
Obrázok 12 Dátová tabuľka s možnosťou filtrovania	61
Obrázok 13 Ukážka vygenerovaných asociačných pravidiel s dodatočnými informáciami.....	63
Obrázok 14 Bodový graf využitý pri asociačných pravidlách.....	64
Obrázok 15 Vizualizácia vzťahov medzi hodnotami atribútov a asociačnými pravidlami.....	65
Obrázok 16 Vizualizácia znázorňujúca odporúčaný počet zhlukov	67
Obrázok 17 Analýza stability zhlukov prostredníctvom metódy siluety	68
Obrázok 18 Výsek rozdelenia hodnôt jednotlivých atribútov naprieč všetkými zhlukmi podľa zvolených parametrov.....	69
Obrázok 19 Ukážka súradnicového grafu v implementovanej aplikácii.....	70
Obrázok 20 Dátová množina s informáciou o zaradení záznamov do zhlukov	73
Obrázok 21 Unikátne hodnoty vyskytujúce sa na ľavej strane asociačných pravidiel	74
Obrázok 22 Výber zhuku záznamov využitých pri vizualizácii asociačných pravidiel	74
Obrázok 23 Vyhodnotenie otázky č. 1.....	77
Obrázok 24 Vyhodnotenie otázky č. 2.....	77
Obrázok 25 Vyhodnotenie otázky č. 3.....	78
Obrázok 26 Vyhodnotenie otázky č. 4.....	79
Obrázok 27 Vyhodnotenie otázky č.5	79
Obrázok 28 Vyhodnotenie otázky č.6	80
Obrázok 29 Vyhodnotenie otázky č. 7.....	80
Obrázok 30 Vyhodnotenie otázky č. 8.....	81
Obrázok 31 Vyhodnotenie otázky č. 9.....	81

Obrázok 32 Vyhodnotenie otázky č. 10.....	82
---	----

Zoznam tabuliek

Tabuľka 1 Tabuľka asociácie.....	27
Tabuľka 2 Príklad pre výpočet Gowerovej vzdialenosti	30

Zoznam symbolov a skratiek

VÚSCH	Východoslovenský ústav srdcových a cievnych chorôb
KVO	kardiovaskulárne ochorenie
OECD	Organizácia pre hospodársku spoluprácu a rozvoj (z angl. Organisation for Economic Co-operation and Development)

Úvod

Dáta, údaje či štatistiky sú v dnešnom modernom svete plnom technológií neodmysliteľnou súčasťou takmer každého odvetvia. Každým dňom vznikajú nové a nové záznamy, ktoré sú plné informácií. Využitím dátovej analytiky je možné z týchto záznamov extrahovať dôležité a užitočné znalosti, ktoré sú využiteľné v prospech ľudskej populácie. Oblasť medicíny nie je výnimkou. Aplikovaním popisných modelov je možné identifikovať rôzne faktory ovplyvňujúce závažnosť chorôb, no taktiež definovať jedinečné skupiny pacientov vzhľadom na ich výsledky lekárskeho vyšetrení.

Z dostupných informácií je možné konštatovať, že najčastejšiu príčinou úmrtí z celosvetového hľadiska predstavujú kardiovaskulárne ochorenia. Na základe štatistík Svetovej zdravotníckej organizácie až 31% globálnych úmrtí je možné pripísať ochoreniam a ťažkostiam súvisiacich so srdcom a jeho cievny systémom [1]. Na základe dát z roku 2019 jasne vyplýva, že na Slovensku je o takmer 100% vyššia úmrtnosť na kardiovaskulárne ochorenia v porovnaní s OECD priemerom a o 300% vyššia než je vo Francúzsku, Holandsku a Taliansku [3]. Zaujímavou, no zároveň alarmujúcou informáciou taktiež je, že na Slovensku umiera ročne na kardiovaskulárne ochorenia, t.j. je najmä ischemickú chorobu srdca a náhlu cievnu mozgovú príhodu, približne 23 tisíc ľudí, čo predstavuje približne 40% úmrtí u mužov a 50% úmrtí u žien [3]. Z globálneho hľadiska je najzasiahnutejšou krajinou Čína, pri ktorej sa odhaduje, že každý piaty dospelý obyvateľ krajiny trpí kardiovaskulárnym ochorením [2]. Negatívne svetové prvenstvo však patrí Spojeným štátom americkým, kde takmer polovica, konkrétne 48% dospeléj populácie má tento typ ochorenia [4].

Predkladanú diplomovú prácu možno rozdeliť na dve hlavné časti, a to teoretickú, resp. analytickú a praktickú. V prvej časti sme sa zamerali na teoretický rozbor skúmanej problematiky, kde sme bližšie charakterizovali deskriptívne modelovanie a konkrétne algoritmy, ktoré je možné do tejto oblasti zaradiť. Konkrétne máme na mysli algoritmy pre hľadanie asociačných pravidiel, či metódy a modely aplikovateľné v procese zhľukovania. Súčasťou teoretickej časti našej diplomovej práce je aj analýza súčasného stavu. V rámci tejto kapitoly sme spracovali informácie podložené štúdiami o využití popisného modelovania v oblasti medicíny a zdravotnej starostlivosti, no primárne v oblasti kardiovaskulárnych ochorení. Taktiež sme sa zamerali na analýzu konkrétnych vedeckých prác a z nich získaných výsledkov, ktorých obsah spočíva v aplikovaní popisných modelov na dáta v už spomínanej oblasti.

Náplňou praktickej časti je implementácia navrhovaného systémového riešenia pre podporu výskumu kardiológov na báze popisných modelov. Aplikácia pozostáva z troch hlavných celkov, ktoré slúžia na hľadanie asociačných pravidiel, aplikovanie metód zhľukovania a na hľadanie

asociačných pravidiel v rámci definovaných zhlukov. Počas implementácie aplikácie sme využili dátový súbor, ktorý je tvorený informáciami z lekárskych záznamov skutočných pacientov Východoslovenského ústavu srdcových a cievnych chorôb (VÚSCH). Používateľ má ale aj možnosť nahráť do softvéru svoju vlastnú dátovú množinu, čím sa aplikácia stáva univerzálnejšou a využiteľnou v širšom spektre aplikačných oblastí. Pre hľadanie asociačných pravidiel sme využili algoritmus *Apriori*, pričom získané výsledky sú prezentované prostredníctvom tabuľkovej formy, no taktiež vo forme interaktívnych vizualizácií. V rámci druhej časti aplikácie, a teda pri zhlukovaní sme využili metódy hierarchického zhlukovania a taktiež metódy segmentačného zhlukovania, konkrétne algoritmy *PAM* a *k-Means*. Aplikovanie týchto modelov je v súlade s cieľom našej diplomovej práce.

Záverečná časť diplomovej práce je venovaná používateľskej štúdií vykonanej na skupine lekárov, ktorí sú potenciálni používatelia aplikácie. Význam štúdie spočíva v otestovaní funkčnosti a overení dostatočne intuitívnej orientácie v aplikácii ako aj jej použiteľnosti niekoľkými testujúcimi a následnom vyhodnotení získaných výsledkov.

1. Teoretický rozbor skúmanej problematiky

Prístupov k riešeniu úloh dátovej analytiky, resp. dolovania dát je hneď niekoľko. Prediktívny prístup možno označiť za veľmi populárny a využiteľný v rôznych aplikačných oblastiach. Avšak deskriptívny, popisný prístup, má taktiež svoje jasné opodstatnenie a využitie v širokom spektre oblastí, vrátane tej medicínskej [8], ktoré si vyžadujú konanie podložené dôkladnou analýzou historických dát.

1.1. Charakteristika popisných modelov

Aplikovanie modelov na už spracované údaje je jedným z kľúčových krokov v procese dolovania znalostí a potenciálne užitočných informácií z dostupných dát [5]. Ako už bolo povedané, vzhľadom na tému našej diplomovej práce sú kľúčové práve popisné modely.

Deskriptívny prístup, resp. deskriptívne modelovanie, možno definovať ako proces identifikovania opakujúcich sa vzorov a podobnosti medzi vstupnými dátami [6]. Pre tento typ prístupu je charakteristických niekoľko faktov. Je relevantné tvrdiť, že práve vďaka popisným modelom je možné analyzovať to, čo sa stalo v minulosti a pochopiť tak príčiny a dôsledky konaní, alebo iných faktorov [6]. Zatiaľ čo pri prediktívnych modeloch je výstupom odhad toho, čo môže v budúcnosti nastať [5], popisné modely jasne definujú opakujúce sa vzory a podobnosti vo vstupnej dátovej množine [6]. Je teda možné tvrdiť, že deskriptívny prístup dátovej analytiky je zameraný na analýzu, sumarizáciu a následnú interpretáciu získaných informácií a znalostí. Rovnako dôležitým aspektom, ktorý odlišuje deskriptívnu analytiku od tej prediktívnej je cieľový atribút, ktorý pri deskriptívnych metódach nemusí byť nutne definovaný. Do značnej miery je tento pohľad porovnateľný s delením strojového učenia na kontrolované a nekontrolované. Práve pri kontrolovanom učení, inak označovanom aj učenie s učiteľom, je hodnota cieľového atribútu známa a využiteľná pri tréňovaní modelov [7]. Pri nekontrolovanom učení, ktoré je z hľadiska problematiky našej diplomovej práce kľúčové, cieľový atribút nemusí byť známy [7].

Do kategórie popisných modelov patrí niekoľko metód. V nasledujúcich podkapitolách sa zameriame na niektoré z nich.

1.2. Asociačné pravidlá

Asociačné pravidlá sa radia medzi tie najpopulárnejšie metódy popisného prístupu k dolovaniu dát. Ich hlavným cieľom je identifikovanie často sa opakujúcich a relevantných vzťahov medzi jednotlivými atribútmi vstupnej dátovej množiny [39]. Úspešne sa využívajú v rôznych aplikačných oblastiach, akými sú napríklad analýza nákupného košíka, analýza finančných dát, no svoje jednoznačné opodstatnenie majú aj v oblasti medicíny [8], resp. zdravotnej starostlivosti. Práve

v tejto, pre nás dôležitej oblasti, sú asociačné pravidlá využívané pre definovanie vzťahov medzi ochoreniami, alebo zdravotným stavom pacienta a určitými príznakmi [39]. Ako už bolo povedané, asociačné pravidlá nám umožňujú získať akýsi obraz o vzťahoch medzi jednotlivými prvkami v danej dátovej množine, s čím súvisí aj ich veľká výhoda, ktorou je relatívne jednoduchá interpretácia. Konštrukcia získaných pravidiel je totiž v tvare „*IF podmienka/konjunkcia podmienok - THEN dôsledok/konjunkcia dôsledkov*“ [9]. Konkrétny príklad vygenerovaného asociačného pravidla z oblasti analýzy medicínskych dát je možné uviesť nasledovne:

- $\{O_Hypertenzia, LDL = 4.1<\} \Rightarrow \{Chol = 6<\}, \text{spoľahlivosť} = 99\%, \text{podpora} = 10\%.$

Toto pravidlo vyjadruje z dát získanú znalosť, že hodnota celkového cholesterolu pacientov, ktorí trpia hypertenziou, teda vysokým krvným tlakom, pričom ich hodnota LDL cholesterolu presahovala hodnotu 4.1 mmol/l, s 99% pravdepodobnosťou presiahne hodnotu 6 mmol/l, pričom 10% záznamov pacientov z pôvodnej dátovej množiny obsahovalo množinu údajov „O_Hypertenzia“, „LDL = 4.1<“ a „Chol=6<“.

Práve spomínané percentuálne hodnoty predstavujú dôležité metriky asociačných pravidiel, na základe ktorých je možné interpretovať ich kvalitu, resp. silu, významnosť či mieru zaujímavosti. Jedná sa o *podporu*, *spoľahlivosť* a tzv. *lift*. Metriku podpory asociačného pravidla [10] v tvare $X \Rightarrow Y$, kde X a Y sú množiny položiek, možno definovať ako hodnotu, ktorá hovorí o tom, že dané pravidlo má podporu s , ak s percent transakcií v databáze obsahuje $X \cup Y$. Podpora, z angl. *support* asociačného pravidla teda reprezentuje frekvenciu výskytu danej množiny položiek v databáze. Druhá, rovnako dôležitá metrika je tzv. *spoľahlivosť* [10], z angl. *confidence*. Pre vysvetlenie tejto metriky opäť využijeme všeobecný zápis asociačného pravidla, a teda $X \Rightarrow Y$. Hodnota spoľahlivosti daného pravidla v číselnom vyjadrení definuje relatívne množstvo transakcií v databáze, ktoré obsahujú množinu položiek X , aj Y voči tým, ktoré obsahujú iba X . Spoľahlivosť asociačného pravidla teda interpretuje pravdepodobnosť výskytu pravej strany pravidla za podmienky výskytu ľavej strany pravidla. Ďalšia z najčastejšie používaných metrík, ktorá sa využíva pri charakteristike asociačných pravidiel, je tzv. *lift* [10], či miera zaujímavosti. Táto metrika definuje, koľkokrát častejšie sa množiny položiek X a Y vyskytujú spolu, než by to bolo v prípade, že sú štatisticky nezávislé. Ako konkrétny príklad uvádzame, že v prípade, ak hodnota *lift* pravidla $X \Rightarrow Y$ je väčšia než 1, znamená to, že množiny položiek X a Y sa vyskytujú častejšie spolu ako samostatne.

Cieľom tohto typu analýzy je nájsť všetky tie pravidlá, ktoré nadobúdajú stanovenú minimálnu mieru podpory a spoľahlivosti. Množina položiek, ktorá spĺňa kritérium minimálnej podpory je taktiež označovaná ako frekventovaná, či často sa opakujúca [10].

1.2.1. Algoritmus Apriori

Typickým a veľmi často využívaným algoritmom pre generovanie asociačných pravidiel je tzv. algoritmus *Apriori*. Podstata algoritmu spočíva v hľadaní frekventovaných množín položiek, ktoré predstavujú kombinácie hodnôt jednotlivých atribútov, pričom musia byť splnené dané metriky, ktoré sme definovali v prechádzajúcej podkapitole.

Algoritmus *Apriori* [5] generuje kombinácie hodnôt jednotlivých atribútov do šírky, čo znamená, že najskôr sú vygenerované kombinácie dĺžky jedna, následne dĺžky dva pokračujúc až po stanovený limit, resp. obmedzenie dĺžky kombinácie. Dôležité taktiež je, že sa v jednotlivých kombináciách atribúty nesmú opakovať a položky sú usporiadané lexikograficky [5]. Pri generovaní pravidiel sa využíva vlastnosť frekventovaných položkových množín, ktorá hovorí, že každá podmnožina frekventovanej množiny položiek musí byť taktiež frekventovaná [5] - toto pravidlo má za následok podstatnú redukciu prehľadávacieho priestoru a umožňuje definovanie pravidiel vo výpočtovo prijateľnom čase.

Hľadanie frekventovaných množín položiek je vo všeobecnosti možné rozdeliť do nasledujúcich krokov [12]:

1. Prvý krok spočíva v generovaní množiny kandidátov C_k (množiny položiek veľkosti k) na základe L_{k-1} , kde L predstavuje frekventované množiny položiek veľkosti $k-1$. Tento krok možno označiť ako „spájací“.
2. V rámci druhého kroku sa vykoná odstránenie tých množín z množiny kandidátov C_k , ktorých podmnožina sa nenachádza v L_{k-1} , čo možno interpretovať ako proces orezávania.
3. V poslednom, treťom kroku sa z množiny kandidátov C_k do L_k zaradia také množiny, ktoré spĺňajú stanovenú minimálnu mieru podpory.

Po nájdení často sa opakujúcich množín položiek, a teda tých kombinácií, ktoré dosahujú hodnotu stanovenej minimálnej miery podpory, sa z nich vytvárajú pravidlá, ktoré spĺňajú definovanú minimálnu úroveň spoľahlivosti.

V rámci zhodnotenia algoritmu *Apriori* je možné povedať, že v odvetví generovania asociačných pravidiel sa z hľadiska použiteľnosti jedná o relatívne jednoduchý a veľmi obľúbený algoritmus. Získané výsledky sú ľahko čitateľné, čo znamená, že ich komunikácia koncovému používateľovi nie je náročná. Za hlavnú nevýhodu však možno považovať jeho efektivitu, vzhľadom na to, že pri každom cykle je nevyhnutné pre nájdenie všetkých frekventovaných množín danej veľkosti prejsť celou databázou, čo je výpočtovo náročné [12].

1.2.2. Algoritmus FP-Growth

Algoritmus *FP-Growth* (frequent pattern growth) taktiež možno zaradiť medzi najpoužívanejšie algoritmy pre generovanie asociačných pravidiel. Kľúčovou vlastnosťou tohto algoritmu je, že identifikuje frekventované množiny položiek bez prvého generovania kandidátov vzhľadom na definovanú minimálnu úroveň podpory [16]. Zatiaľ čo algoritmus *Apriori* prechádza pri hľadaní frekventovaných množín položiek celou databázou niekoľkokrát, čo možno definovať ako výpočtovo náročné a taktiež čas potrebný pre generovanie asociačných pravidiel sa zvyšuje exponenciálne so zväčšovaním databázy, algoritmus *FP-Growth* prechádza celou databázou len dvakrát [13].

Celý proces fungovania algoritmu je možné rozdeliť do niekoľkých častí [12]. V rámci prvej algoritmus prejde celou databázou, pričom identifikuje a vytvorí množinu obsahujúcu početnosti výskytu jednotlivých prvkov a ich podporu. Následne sú z tejto množiny vyradené všetky tie, ktoré nespĺňajú používateľom definovanú minimálnu mieru podpory. V ďalšom kroku sa zo zvyšných položiek, ktoré spĺňajú stanovenú minimálnu mieru podpory vytvorí zoznam, v rámci ktorého sú zoradené zostupne na základe vyjadrenej podpory. Následne sa frekventované položky v jednotlivých transakciách triedia práve na základe vytvoreného zoznamu, čo umožní vygenerovanie dátovej štruktúry, taktiež nazývanej aj strom frekventovaných vzorov. V druhej časti je frekventovaná množina položiek vyselektovaná práve z danej dátovej štruktúry [12].

Ako už bolo povedané, medzi hlavné výhody algoritmu *FP-Growth* patrí výpočtová efektivita, tým pádom sa vyžaduje menej času pre vygenerovanie asociačných pravidiel. Za obmedzenie je možné považovať samotné generovanie dátovej štruktúry, resp. stromu frekventovaných vzorov, a to v prípade, ak operačná pamäť počítača nie je dostatočná na spracovanie väčšej dátovej množiny [12].

1.2.3. Algoritmus Eclat

Hlavný rozdiel medzi algoritmami Eclat a Apriori spočíva v tom, akým spôsobom sú tvorené kombinácie a vypočítavaná podpora jednotlivých vzorov, resp. ich početnosť. Tento algoritmus zhlukuje množiny kombinácií položiek, z ktorých je následne budovaná logická hierarchická štruktúra založená na princípe ekvivalencie tried. Časté vzory sú následne vyhľadávané v tomto type hierarchickej štruktúry [14].

Množinu dostatočne frekventovaných kombinácií položiek dĺžky je k označme nasledovne:

$$X = \{x_1, \dots, x_k\}.$$

Predpokladajme, že táto množina L_k je usporiadaná lexikografickým spôsobom. Potom pre $k \geq 2$ je možné L_k rozdeliť do tried ekvivalencie, pričom dve kombinácie položiek sú ekvivalentné za predpokladu, že sa zhodujú prefixy dĺžky $k - 1$. Triedu ekvivalencie je možné definovať nasledujúcim spôsobom:

$$[a] = \{b \in L_k \mid a = b[1 : k - 1]\}$$

Rovnako ako algoritmus *Apriori*, aj algoritmus *Eclat* počíta množinu L_k z množiny L_{k-1} , pričom triedy ekvivalencie sú vypočítané v medzikroku. Z týchto tried je v ďalšom kroku vypočítaná množina potenciálne frekventovaných vzorov L'_k , a to spojením všetkých možných dvojíc jednej triedy s identifikátorom triedy ako prefixom. Na začiatku, všetky položky vyskytujúce sa v záznamoch patria do množiny L'_1 . V množine L_1 ostávajú tie položky, ktorých výskyt je dostatočne častý. Pre $k = 2$ už je teda možné vytvoriť znalosť o ekvivalenciách položiek. Na záver je dôležité zdôrazniť, že s narastajúcou hodnotou k sa zvyšuje presnosť rozdelenia do tried ekvivalencie [14].

Výhoda algoritmu *Eclat* oproti algoritmu *Apriori* spočíva vo fakte, že transakčnú databázu prechádza iba raz a v ďalšom kroku už vykonáva prieniky. Vďaka tejto vlastnosti algoritmus *Eclat* menej výpočtovo náročný. Na druhej strane je nutné poznamenať, že *Eclat* je schopný generovať viac kandidátov častých množín než *Apriori*. Je to z dôvodu, že v momente rozhodovania nedisponuje podporou všetkých podmnožín [15].

1.3. Zhlukovanie

Zhlukovanie možno charakterizovať ako metódu pre hľadanie vzťahov a súvislostí medzi jednotlivými objektami dátovej množiny a ich následne zatriedenie do skupín na základe podobných vlastností. Tieto skupiny nesú taktiež názov zhľuky, alebo klastre. Objekty v rámci jedného zhľuku vykazujú medzi sebou oveľa väčšiu vzájomnú podobnosť, než s objektami z iných zhľukov [16]. Zhľukovanie radíme medzi metódy nekontrolovaného učenia, ktoré je charakteristické tým, že nie sú k dispozícii žiadne preddefinované triedy, a teda nie je možné určiť ani presný počet hľadaných tried [7]. Dôležité je však poznamenať, že v niektorých prípadoch je pri metódach zhľukovania potrebné zvoliť odhadovaný počet klastrov.

Analýza zhľukovania má využitie v širokom spektre odvetví [17]. Typickými príkladmi sú segmentácia zákazníkov, či používateľov na základe podobných demografických údajov, alebo vzorov nákupu pre ciele marketingové kampane. Taktiež identifikovanie nezvyčajného správania subjektov, označovaným aj ako detekcia anomálií, ktoré nespádajú do žiadnych známych klastrov [17]. Vo všeobecnosti môžeme povedať, že zhľukovanie je relevantné a využiteľné práve vtedy, keď dátová množina, s ktorou je potrebné pracovať, pozostáva z veľkého množstva údajov, pričom kvôli

lepšiemu pochopeniu a interpretácií sú nevyhnutné menšie homogénne skupiny, ktoré pozostávajú z podobných subjektov [17].

V nasledujúcich podkapitolách sa venujeme charakteristike jednotlivých prístupov k zhľukovaniu, ktoré sú taktiež využívané v oblasti medicíny a zdravotnej starostlivosti.

1.3.1. Hierarchické zhľukovanie

Metódy hierarchického zhľukovania patria medzi často využívané a populárne pri riešení rôznych otázok a nastolených problémov. Výhodou tohto prístupu k zhľukovaniu je, že na začiatku samotného procesu nie je potrebné definovať počet zhľukov. Algoritmy tohto typu [16] rozdeľia jednotlivé objekty hierarchickým spôsobom, čo znamená, že rozdelenie prebieha buď prístupom zdola nahor, alebo opačne, a teda zhoda nadol. Na základe tejto informácie je možné zaradiť jednotlivé algoritmy hierarchického zhľukovania do dvoch hlavných skupín, a to aglomeratívnej a divizívnej.

Aglomeratívne zhľukovanie je charakteristické tým, že na začiatku procesu je každý objekt v jednom, samostatnom zhľuku a následne sú postupne spájané do väčších zhľukov, až do momentu, kým nie je splnená stanovená podmienka ukončenia, alebo kým nie sú všetky objekty spojené v jednom veľkom zhľuku [16]. Ako koncovú podmienku možno využiť maximálnu možnú vzdialenosť, pri ktorej je ešte možné dva zhľuky zlúčiť, alebo minimálnu hranicu na celkový počet zhľukov, ktoré vznikli práve zlučovaním. Proces postupného spájania zhľukov je vo všeobecnosti možné charakterizovať, ako proces spájania dvoch množín bodov, resp. skupín objektov, ktoré sú si najbližšie [18]. Spôsobom ako merať vzdialenosti dvoch množín bodov je niekoľko. V nasledujúcej podkapitole si bližšie priblížime niektoré z nich.

Divizívne hierarchické zhľukovacie metódy sú naopak typické tým, že predstavujú rozklad množiny objektov, a to postupným delením zhľukov, ktoré už existujú. Delenie trvá dovtedy, kým opäť nie je splnená ukončovacia podmienka alebo každý objekt nie je zaradený do jedného zhľuku, čo značí, že ďalšie delenie už nie je možné. Tento typ postupu je ideálny v prípade menšieho počtu objektov [16].

Charakteristickou vizualizáciou hierarchického zhľukovania je tzv. dendrogram, ktorý tvorí hierarchiu zhľukov, taktiež nazývanú strom zoskupení. Každý uzol zhľuku obsahuje podradené zhľuky, pričom „súrodenecké“ zhľuky delia body, na ktoré sa vzťahuje ich spoločný „rodič“ [16].

1.3.1.1. Metódy aglomeratívneho zhukovania

Medzi najznámejšie metódy aglomeratívneho zhukovania možno zaradiť *metódu najbližšieho suseda*, *metódu najvzdialenejšieho suseda*, *metódu priemernej vzdialenosti*, ale taktiež *Wardovu metódu*.

Postup *metódy najbližšieho suseda* spočíva v minimálnej vzdialenosti. V prvom kroku sa identifikujú dva objekty, ktoré sú oddelené najkratšou vzdialenosťou a následne sa zlúčia do zhuku. Ďalší zhuk je vytvorený pridaním tretieho najbližšieho suseda. Tento proces sa opakuje dovtedy, pokiaľ nie sú všetky objekty v jednom spoločnom zhuku. Vzdialenosť medzi dvoma zhukmi je možné definovať ako najkratšiu vzdialenosť ktoréhokoľvek bodu v zhuku voči ľubovoľnému bodu v inom zhuku. Nevýhodou tejto metódy je tzv. efekt reťazenia, pri ktorom dochádza k spájaniu zhukov, ktorých dva objekty sú si najbližšie, ale vzhľadom k väčšine ostatných objektov nejde o najbližšie zhuky [19].

Metóda najvzdialenejšieho suseda je podobná tej prechádzajúcej, avšak kritérium je definované maximálnou a nie minimálnou vzdialenosťou. Vzdialenosť medzi objektami, resp. zhukmi je definovaná ako najväčšia vzdialenosť medzi bodmi týchto skupín. Táto metóda sa taktiež nazýva ako metóda úplného prepojenia, a to práve z dôvodu, že všetky objekty v zhuku sú navzájom prepojené pri maximálnej vzdialenosti, resp. minimálnej podobnosti [19].

Metóda priemernej vzdialenosti je charakteristická tým, že jednotlivé objekty, resp. zhuky sú spájané na základe vzdialenosti D , ktorá je vyjadrená pomocou priemernej vzdialenosti medzi objektami z daných zhukov. Na začiatku je každý objekt vo svojom vlastnom zhuku, a teda hodnota parametra D je nulová. Postupným zlučovaním jednotlivých zhukov hodnota parametru D narastá, pričom algoritmus končí v momente, keď hodnota spomínaného parametru dosiahne určitú vopred definovanú hodnotu [20].

Pri *Wardovej metóde* aglomeratívneho zhukovania je možné konštatovať, že podobnosť objektov, resp. zhukov je vyjadrená ako suma štvorcov medzi objektami z dvoch zhukov, sčítaná cez všetky atribúty daných objektov. Táto metóda disponuje kľúčovou vlastnosťou, a tou je minimalizácia sumy rozptylov cez všetky novovzniknuté zhuky. To značí, že na každej úrovni sú vytvárané zhuky tak, aby bola zachovaná čo najväčšia súdržnosť v rámci jednotlivých zhukov [20].

Vnútrozhuková variabilita VSS je definovaná nasledujúcim vzťahom:

$$VSS = \sum_{j=1}^m \sum_{i=1}^k (x_{ij} - \bar{x}_j)^2$$

pričom

$$\bar{x}_j = \frac{1}{k} \sum_{i=1}^k x_{ij}$$

1.3.2. Metódy zhlukovania založené na výbere reprezentantov zhlukov

Pri metódach zhlukovania, ktoré sú založené na výbere reprezentantov zhlukov je vo všeobecnosti možné povedať, že vytvorené zhluky sú rozdelením bodov do skupín, ktoré netvorí hierarchiu, pričom každá táto skupina je definovaná svojim reprezentantom. Tito reprezentanti môžu byť fiktívne vypočítaní, a teda sa môže jednať o priemer alebo medián dátových bodov v danom zhluku, no taktiež môžu byť vybraní z existujúcich dátových bodov zhluku. V stave, keď už sú reprezentanti zhlukov zvolení, prideliť ďalšie body z dátovej množiny do zhlukov je triviálne, vzhľadom na to, že každý takýto bod je pridelený k najbližšiemu reprezentantovi [18]. Taktiež je stanovených niekoľko podmienok. Maximálny počet prvkov v dátovej množine je n . Na základe tejto hodnoty je možné odvodiť maximálny počet zhlukov k , ktorý musí byť buď menší, alebo nanajvýš rovný celkovému počtu prvkov n . Počet zhlukov k určuje používateľ. Dôležité je taktiež uviesť, že vzniknuté zhluky musia pozostávať minimálne z jedného prvku, pričom každý prvok sa môže vyskytnúť len v jednom zhluku [16].

Princíp vykonávania tohto prístupu k zhlukovaniu spočíva v tom, že jednotlivé zhluky sú spočiatku vytvorené náhodne a následne sa v nich premiestňujú jednotlivé prvky s cieľom vylepšenia klastrov. Kritériom pre definovanie toho, či daný prvok patrí alebo nepatrí do daného zhluku je jeho vzdialenosť od stredu samotného klastra. Na základe určenia ťažiska centroidu a miery podobnosti sa do skupiny metód zhlukovania radia algoritmy ako *k-Means* alebo *k-Medoids* [18].

Algoritmus *k-Means* [18] možno zaradiť medzi najpopulárnejšie metódy pre zhlukovanie, ktoré sú využiteľné v širokom spektre aplikačných oblastí. Ako funkcia vzdialenosti je využitý *kvadrát Euklidovskej vzdialenosti*, ktorý je definovaný nasledovne:

$$Dist(\bar{X}_i, \bar{Y}_j) = ||\bar{X}_i - \bar{Y}_j||_2^2$$

V tomto prípade je možné výraz pre vzdialenosť chápať ako štvorec chyby vznikajúcej aproximáciou daného bodu reprezentantom príslušného zhluku. Je možné konštatovať, že optimálny reprezentant zhluku v každej iterácii algoritmu *k-Means* je aritmetický priemer dátových bodov zhluku. Aj keď algoritmus *k-Means* nie je možné aplikovať na kategorické atribúty, má svoje jasné opodstatnenie pre tie numerické.

Charakteristické pre algoritmus *k-Medoids* [18] je, že reprezentanti zhlukov sú vždy konkrétne dátové body z pôvodnej dátovej množiny. Reprezentanti teda nemôžu byť fiktívne vypočítaní

v podobe mediánu alebo aritmetického priemeru, ako to je v prípade algoritmu *k-Means*. Taktiež je odolný voči odľahlým, resp. extrémnym hodnotám.

Na veľmi podobnom princípe funguje taktiež algoritmus *PAM (Partitioning Around Medoids)* [18], pri ktorom proces zhlukovania začína nájdením reprezentatívneho súboru k objektov. Platí, že prvý objekt z reprezentatívneho súboru má najkratšiu vzdialenosť k ostatným objektom, čo znamená, že predstavuje stred. Možné alternatívy polohy k objektov sú vyberané iteračným spôsobom, kde algoritmus vyhľadáva nezaradené objekty a premiestňuje ich tak, aby sa hodnota celkovej vzdialenosti znižovala.

1.3.3. Zhlukovanie založené na hustote

Metódy zhlukovania založené na hustote je možné považovať v medicínskom prostredí za dôležité a efektívne. Dané zhľuky sú úzko späté s najbližšími susedmi bodu. Metódy hierarchického a segmentačného zhlukovania dokážu identifikovať len zhľuky sférického tvaru. Avšak zhľuk, ktorý je definovaný ako prepojený hustý komponent, rastie v ľubovoľnom smere, ktorým daná hustota smeruje. Metódy zhlukovania založené na hustote je preto možné využiť pri identifikácii zhľukov ľubovoľného tvaru [16].

Typickou metódou zhlukovania založené na hustote je tzv. *DBSCAN*. Táto metóda definuje hustotu dátového bodu, a to na základe jeho okolia *Eps*, vrátane daného bodu samotného. Na základe sféricky stanoveného okolia dátových bodov sú následne body rozčlenené do troch hlavných skupín. Prvou z nich je skupina bodov jadra, ktorá vo svojom okolí *Eps* obsahuje aspoň T bodov. Druhou vrstvou bodov sú okrajové body, ktoré napriek tomu, že vo svojom okolí *Eps* obsahujú menej ako T bodov, ale aspoň jeden z nich je bod jadra. Tretiu skupinu bodov tvoria tie body, ktoré nie sú ani bodom jadra a ani okrajovým bodom. Túto skupinu bodov je taktiež možné nazvať ako šum. Výhodou metódy *DBSCAN* je, že dokáže odhaliť zhľuky akéhokoľvek tvaru, a taktiež to, že nie je potrebné na vstupe definovať počet zhľukov. Nevýhodou však je, že táto metóda je citlivá na premenlivú lokálnu hustotu zhľukov [18].

1.3.4. Optimálny počet zhľukov

Ako sme naznačili v predchádzajúcej podkapitole, pri niektorých algoritmoch je potrebné identifikovať a definovať optimálny počet zhľukov vzhľadom na používanú dátovú množinu. V ideálnom prípade je možné túto hodnotu odvodiť z externých faktorov, prípadne využiť znalosť minulých analýz, avšak nie vždy je to možné. Práve v takých situáciách je vhodné využiť niektorú zo štandardných a často využívaných metód, akými sú tzv. „metóda lakťa“ (z angl. „*Elbow method*“) alebo „metóda siluety“ (z angl. „*Silhouette method*“).

Princíp fungovania „*Metódy lakťa*“ [22] spočíva v postupnom iterovaní, kedy sa začína na počte zhlukov rovným dvom a postupne sa v každom ďalšom kroku navýši ich počet o jeden, počítajúc zhluky a výslednú sumu reziduálnych štvorcov, prípadne iné vhodné kritérium. Prvé zoskupenia interpretujú veľké množstvo informácií, pričom značne znižujú sumu rezíduí. Avšak pri určitej hodnote reprezentujúcej počet zhlukov hodnota sumy reziduálnych štvorcov rapídne klesá, čo sa na výsledný graf premietne vo forme vizualizácie v tvare lakťa. Práve v tomto bode je definovaný optimálny počet zhlukov, ktorý sa taktiež nazýva „*laktové kritérium*“ (z angl. „*elbow criterion*“). Výstupnú hodnotu z „*laktovej metódy*“, a teda optimálny počet zhlukov je možné využiť až pri aplikovaní algoritmu zhlučovania [19].

Druhá menovaná metóda je tzv. „*Metóda siluety*“ [23]. Jedná sa o analýzu s cieľom definovať, ako podobný je objekt voči iným objektom v rámci rovnakého zhuku, v porovnaní s objektami nachádzajúcich sa v iných zhukoch. Výsledné hodnoty danej validačnej metriky nadobúdajú hodnoty z intervalu -1 až 1. Vysoká kladná hodnota siluety naznačuje, že objekt, resp. záznam sa nachádza v správnom zhuku. Ak je hodnota siluety nízka alebo záporná, tak to značí, že objekt nie je vhodne prispôbený vlastnému zhuku. Hodnota siluety objektu i je definovaná nasledovne [18]:

$$S_i = \frac{Dmin_i^{out} - Davg_i^{in}}{\max \{Dmin_i^{out}, Davg_i^{in}\}}$$

- kde $Davg_i^{in}$ predstavuje priemernú vzdialenosť bodu $\bar{X}_i$ od ostatných bodov v danom zhuku,
- kde $Dmin_i^{out}$ možno charakterizovať ako minimum priemerných vzdialeností bodu $\bar{X}_i$ od ostatných zhlukov.

Celkový *koeficient silhouette* [18] je potom priemer všetkých bodových hodnôt, ktoré tento koeficient nadobudol. Vysoké kladné hodnoty blížiac sa k hodnote 1 značia dobre odseparované zhluky, zatiaľ čo negatívne hodnoty zasa zhluky, ktoré sa navzájom prekrývajú.

1.3.5. Validáčné kritéria

Validácia identifikovaných zhlukov je podstatným krokom v rámci každej analýzy zhlučovania. Práve vďaka interným a externým validačným kritériám je možné vyhodnotiť získané výsledky objektívnym spôsobom.

Externé validačné kritériá [18] používajú oproti tým interným informácie, ktoré sa v dátach nenachádzajú. Dôležité je taktiež poznamenať, že tieto kritériá sa dajú využiť iba v tom prípade, ak je k dispozícii informácia o skutočnom zaradení do zhlukov, čo však v reálnych dátach väčšinou nie je možné. Je to však možné pri dátových množinách, ktoré sú generované synteticky. V reálnych

dátach sa však k splneniu tohto cieľa dá priblížiť, a to využitím označení tried v prípade, ak sú k dispozícii. Externé validačné kritériá sú preferované voči tým interným, a to aj z dôvodu, že sa vedia vyhnúť preferovaniu určitého typu algoritmov.

Medzi typické externé validačné kritériá patrí tzv. *čistota zhlučkov* či *entropia* [18]. Obidve tieto kritériá sú zamerané na *čistotu* zhlučkov vzhľadom k daným označeniam triedy. Správne určený zhluček by mal pozostávať z dátových bodov prevažne z jednej triedy. Vo všeobecnosti platí, že nižšia hodnota *entropie*, resp. vyššia hodnota *čistoty zhlučkov* značí lepší výkon zoskupenia.

Ďalším externým validačným kritériom je *Gini index* [18], ktorý poskytuje rýchly a intuitívny spôsob pre vyhodnotenie stupňa heterogenity dát v rámci jednotlivých klastrov. Oproti metrike hovoriacej o *čistote zhlučkov* má výhodu v tom, že *Gini index* neberie do úvahy len dominantný zhluček, ale taktiež aj distribúciu dátových bodov z ostatných zhlučkoch. Pre dobre odseparovaný zhluček nadobúda *Gini index* hodnotu blízku nule.

Interné validačné kritériá [18] sa využívajú najmä v prípade, ak pre vyhodnotenie kvality zhlučkov nie sú k dispozícii žiadne externé validačné kritériá. V rámci týchto typov validačných kritérií sa však väčšinou jedná o kritériálnu funkciu, ktorá je optimalizovaná určitým algoritmom zhlučkovania. Medzi tieto kritériá možno zaradiť sumu štvorcov vzdialeností od centroidov, pomer vnútrozhlučkovej a medzizhlučkovej vzdialenosti, či koeficient *silhouette*, ktorého využitie a postup výpočtu sme charakterizovali v predchádzajúcej podkapitole.

Čo sa týka procesu výpočtu sumy štvorcov vzdialeností od centroidov [18], prvý krok spočíva v určení centroidov jednotlivých zhlučkov, pričom následne sa vyjadří hodnota súčtu kvadrátov odchýlok jednotlivých bodov zhlučkov od centroidov príslušných zhlučkov. Nižšia hodnota tejto metriky značí lepšie definovaný zhluček.

V rámci pomeru vnútrozhlučkovej a medzizhlučkovej vzdialenosti [18] sa v prvom kroku vyberie náhodná vzorka r párov dátových bodov. Páry, ktoré pochádzajú z jedného zhlučku sú priradené do množiny P , a naopak tie dvojice dátových bodov, ktoré pochádzajú z rôznych zhlučkov sú zaradené do množiny Q . Práve tieto množiny obsahujúce páry dátových bodov sú využité pri vyjadrení pomeru vnútrozhlučkovej a medzizhlučkovej vzdialenosti. Malé hodnoty tejto metriky značia lepšie výsledky zhlučkovania.

1.3.6. Miery podobnosti

Za základné kritérium pri tvorbe zhlučkov možno považovať podobnosť medzi objektami. Klastrová analýza vytvára zhlučky objektov, ktoré sú si navzájom podobné. Medziobjektovú podobnosť [24] je možné merať rôznymi spôsobmi, ktoré sa dajú zaradiť do jednej z troch

základných skupín, a to miery asociácie, miery korelácie a miery vzdialenosti. Každá z týchto skupín predstavuje špecifický pohľad na podobnosť, ktorá sa odvíja od objektov a typov dát. Miery vzdialenosti je možné charakterizovať ako miery nepodobnosti, zatiaľ čo miery asociácie a korelácie ako miery podobnosti.

1.3.6.1. Miery asociácie

Koeficienty, ktoré predstavujú miery asociácie [24] sú primárne určené na hodnotenie vzťahov podobnosti daných objektov charakterizovaných dichotomickými znakmi. Asociácia týchto objektov, a teda objektov x_i a x_j je charakterizovaná asociačnou tabuľkou, v ktorej sú znázornené všetky možné kombinácie počtu znakov pre dané objekty x_i a x_j . Medzi základné miery asociácie možno zaradiť *Hamannov koeficient asociácie*, *Jaccardov koeficient* či *Sörensenov koeficient asociácie*.

Tabuľku asociácie (Tabuľka 1) spoločne s vysvetlením jednotlivých hodnôt, ako aj vzorce pre výpočet vyššie spomenutých mier uvádzame v nasledujúcej časti.

Tabuľka 1 Tabuľka asociácie

Asociácia		x_i	
		1	0
x_j	1	a	b
	0	c	d

Hodnoty a , b , c a d možno interpretovať nasledovne:

- znak a : predstavuje počet znakov, kde majú objekty hodnotu 1, resp. pozitívnu hodnotu,
- znak b : predstavuje počet znakov, kde má objekt x_i hodnotu 0 a objekt x_j hodnotu 1,
- znak c : predstavuje počet znakov, kde má objekt x_i hodnotu 1 a objekt x_j hodnotu 0,
- znak d : predstavuje počet znakov, kde majú objekty nulovú hodnotu, resp. zápornú, negatívnu zhodu.

Vzorec pre výpočet *Hamannovho koeficientu asociácie* [25] možno definovať ako:

$$S_s = \frac{2a}{2a + b + c}$$

Jaccardov koeficient [25] je daný vzorcom:

$$S_{rt} = \frac{a + b}{a + 2b + 2c + d}$$

Sörensenov koeficient asociácie [25] je možné vyjadriť nasledovne:

$$S_{rr} = \frac{d}{a + b + c + d}$$

1.3.6.2. Miery korelácie

Výsledné hodnoty korelačných koeficientov [24] sa nachádzajú v intervale $<-1, 1>$, pričom platí, že čím je daná hodnota vyššia, tým väčšia je podobnosť objektov. Ak je hodnota nulová, vzorky, resp. objekty sú vzájomne nezávislé. K najznámejším korelačným koeficientom sa radí *Pearsonov korelačný koeficient* a *Spearmanov korelačný koeficient*. Prvý z nich je možné len na vyjadrenie sily lineárneho vzťahu medzi veličinami. Spearmanov korelačný koeficient zachytáva taktiež monotónne vzťahy, čo znamená, že zachytáva vzťahy, ktoré vo všeobecnosti buď rastú alebo klesajú.

Pearsonov korelačný koeficient [25] možno vyjadriť nasledovne:

$$r_{xy} = \frac{S_{xz}}{S_x S_y} = \frac{\sum_{i=1}^n x_i y_i - n \bar{x} \bar{y}}{\sqrt{(\sum_{i=1}^n x_i^2 - n \bar{x}^2)(\sum_{i=1}^n y_i^2 - n \bar{y}^2)}}$$

Spearmanov korelačný koeficient [25] je definovaný vzťahom:

$$r_s = 1 - \frac{6}{n(n^2 - 1)} \sum_{i=1}^n (x_i - y_i)^2$$

1.3.6.3. Miery vzdialenosti

Vzdialenosť medzi objektami A a B možno označiť ako $d_{A,B}$, pričom miery vzdialenosti disponujú nasledujúcimi vlastnosťami [25]:

- Vzdialenosť medzi dvoma objektami je nezáporná, čiže $d_{A,B} \geq 0$. V prípade, ak je medziobjektová vzdialenosť nulová, hodnoty premenných, ktoré charakterizujú tieto dva objekty sú rovnaké.
- Vzdialenosť medzi objektmi A a B sa rovná vzdialenosti B a A , čiže $d_{A,B} = d_{B,A}$.
- Musí platiť trojuholníková nerovnosť, čiže $d_{A,B} \leq d_{A,i} + d_{B,j}$.

Podstata mier vzdialeností spočíva v prezentácii objektov v priestore, ktorého súradnice sú tvorené jednotlivými premennými. Platí, že čím je medziobjektová vzdialenosť menšia, tým sú si tieto objekty navzájom podobnejšie a čím je vzdialenosť medzi objektami väčšia, tým sú si objekty menej podobné [25].

Medzi najznámejšie metriky pre vyjadrenie vzdialenosti dvoch bodov x_i a x_j v m -rozmernom priestore možno zaradiť *Euklidovskú vzdialenosť*, *štvorcovú Euklidovskú vzdialenosť*, *váženú Euklidovskú vzdialenosť*, *Manhattanskú vzdialenosť*, *Mahalanobisovu vzdialenosť* či *Gowerovu vzdialenosť*.

Euklidovská vzdialenosť, ktorá sa taktiež nazýva geometrickou metrikou, predstavuje dĺžku prepony pravouhlého trojuholníka a jej výpočet je založený na Pytagorovej vete. Vzorec pre výpočet *Euklidovskej vzdialenosti* znie [25]:

$$D_E(x_i, x_j) = \sqrt{\sum_{l=1}^m (x_{il} - x_{jl})^2}$$

Štvorcová Euklidovská vzdialenosť predstavuje základ *Wardovej metódy* zhlukovacej analýzy, ktorá je aplikovateľná v rámci hierarchického zhlučovania. Zápis vzorca pre výpočet tejto vzdialenosti vyzerá nasledovne [25]:

$$D_{ES}(x_i, x_j) = \sum_{l=1}^m (x_{il} - x_{jl})^2$$

Váženú Euklidovskú vzdialenosť možno charakterizovať ako variantu euklidovskej vzdialenosti, pri ktorej je možné jednotlivým premenným pripisovať rôzne váhy a tým zohľadňovať ich význam. Vzorec pre výpočet tejto metriky vzdialenosti znie [25]:

$$D_{EW}(x_i, x_j) = \sqrt{\sum_{l=1}^m w_l (x_{il} - x_{jl})^2}$$

Manhattanská vzdialenosť, taktiež nazývaná ako vzdialenosť mestských blokov, alebo taxi metrika nesie tento názov preto, lebo je to najkratšia vzdialenosť, akú môže taxík prejsť medzi mestskými blokmi na Manhattane, pričom je potrebné rešpektovať štruktúru mestských blokov. Vzorec pre výpočet tejto vzdialenosti je možné nájsť nižšie [25]:

$$D_B(x_i, x_j) = \sum_{l=1}^m |x_{il} - x_{jl}| = |x_i - x_j|$$

Mahalanobisova vzdialenosť je náročnejšia na výpočet, no disponuje výhodou, že potláča vplyv rozdielu vo variabilite premenných a tiež vplyv premenných, ktoré sú korelované. *Mahalanobisovu*

vzdialenosť možno zjednodušene označiť ako Euklidovskú vzdialenosť pre neštandardizované premenné. Vzorec pre výpočet vzdialenosti znie [25]:

$$D_M(x_i, x_j) = (x_i - x_j)^T \cdot S^{-1} \cdot (x_i - x_j)$$

Gowerova vzdialenosť [26] slúži na meranie rozdielov medzi dvoma záznamami dátovej množiny. Výhodou tohto prístupu však je, že dané záznamy môžu obsahovať kombináciu logických, kategorických, číselných či textových údajov. Princíp výpočtu *Gowerovej vzdialenosti* pozostáva vo vzájomnom porovnaní záznamov dátovej množiny.

Ak sú hodnoty atribútu, ktoré dané záznamy tvoria numerické, čiastkovú výslednú hodnotu možno vyjadriť ako rozdiel medzi týmito dvoma hodnotami v absolútnej hodnote, ktorý je vydelený rozdielom najväčšej a najmenšej hodnoty vyskytujúcej sa pri danom numerickom atribúte [26]. Vzorec pre výpočet tohto čiastkového výsledku je možné definovať nasledovne:

$$D_{ijk} = \frac{|x_{ik} - x_{jk}|}{R_k}$$

V prípade, ak hodnoty atribútu nie sú numerické, pričom porovnané výrazy sa zhodujú, čiastková vzdialenosť je vyjadrená číslom 1. Ak sa však porovnávané výrazy nezhodujú, čiastkový výsledok je vyjadrený nulovou hodnotou [26]. Vzorec odpovedajúci tejto definícii znie:

$$D_{ijk} = 1\{x_{ik} = x_{jk}\}$$

Gowerovu vzdialenosť je možné vyjadriť ako súčet čiastkových vzdialeností, ktorý je vydelený počtom atribútov, ktoré tvoria porovnávané záznamy.

Pre vysvetlenie, uvažujme nasledujúci príklad. Majme dátovú množinu (Tabuľka 2), ktorá pozostáva zo štyroch lekárskech záznamov pacientov, ktoré sú tvorené piatimi atribútmi, a to *Pohlavie*, *Váha*, *Výška*, *FBG* (hladina fibrinogénov) a *O_ICHS* (výskyt ischemickej choroby srdca v minulosti pacienta). Obsah dátovej množiny vyzerá nasledovne:

Tabuľka 2 Príklad pre výpočet Gowerovej vzdialenosti

	Pohlavie	Váha	Výška	FBG	O_ICHS
1	Žena	75	153	3.40	FALSE
2	Muž	91	163	3.45	FALSE
3	Žena	89	169	3.70	TRUE
4	Muž	94	168	7.40	TRUE

V rámci vzorového príkladu je naším cieľom vypočítať *Gowerovu vzdialenosť* medzi prvým a druhým záznamom. Na základe postupu, ktorý sme charakterizovali vyššie je možné vypočítať požadované hodnoty nasledujúcim spôsobom:

- *Pohlavie*: porovnané hodnoty nie sú rovnaké, takže výsledná hodnota je 0.

- *Váha*: $(|75 - 91| / 19) = 0.842$

Hodnota 75 je váhou v prípade prvého záznamu, zatiaľ čo hodnota 91 v prípade druhého záznamu. Deliteľ, ktorého hodnota je 19 sme získali ako rozdiel najväčšej a najmenej hodnoty, ktoré sa nachádzajú pri atribúte „*Váha*“, a teda $94 - 75 = 19$.

- *Výška*: $(|153 - 163| / 16) = 0.625$

V tomto prípade, no taktiež v prípade nasledujúcim budeme postupovať identickým spôsobom, aký sme využili aj pri predchádzajúcom čiastkovom výpočte.

- *FBG*: $(|3.40 - 3.45| / 4) = 0.012$

- *O_ICHS*: porovnané hodnoty sú rovnaké, takže výsledná hodnota je 1.

Vypočítané čiastkové výsledky využijeme na získanie výslednej *Gowerovej vzdialenosti*, ktorá existuje medzi prvým a druhým záznamom našej dátovej množiny. Výpočet možno zapísať nasledovne:

$$\frac{0 + 0.842 + 0.625 + 0.012 + 1}{5} = \frac{2.479}{5} = 0.4958$$

Gowerova vzdialenosť medzi prvým a druhým záznam je 0.4958.

2. Analýza súčasného stavu

Aplikačná oblasť, ktorá je podstatná z hľadiska témy našej diplomovej práce je medicína, konkrétne problematika kardiovaskulárnych ochorení. Práve od týchto oblastí a od popisných modelov sa budú odvíjať ďalšie kapitoly a aktivita vykonávaná v rámci záverečnej práce.

V nasledujúcich podkapitolách sa zameriame na základnú charakteristiku kardiovaskulárneho ochorenia, na kľúčové rizikové faktory spôsobujúce toto ochorenie, ale taktiež na možné spôsoby prevencie. Nasledovať bude analýza štúdií, v rámci ktorých boli využité modely deskriptívnej dátovej analytiky v oblasti medicíny, špecificky v oblasti kardiovaskulárnych ochorení, ale tiež sa bližšie pozrieme na celkový vývoj využívania technológií a princípov dátovej analytiky v tejto sfére.

2.1. Kardiovaskulárne ochorenie

Kardiovaskulárne, resp. srdcovo-cievne ochorenia (KVO) možno v širšom kontexte charakterizovať ako všetky ochorenia srdca a ciev, a naopak v užšom význame ako skupinu ochorení, ktoré sú spôsobené aterosklerotickými degeneratívnymi zmenami. Do skupiny KVO je možné zaradiť choroby srdca a srdcových ciev (napríklad hypertenzné choroby, ischémia srdca, zápaly srdcového svalu, srdcové zlyhanie, infarkt myokardu a mnohé ďalšie), choroby pľúcnych ciev (napríklad pľúcna embólia), cievne choroby mozgu (napríklad mozgové krvácanie, mozgový infarkt, náhla cievna mozgová príhoda), choroby tepien (napríklad ateroskleróza a aneuryzma) a taktiež choroby žíl (napríklad zápal žíl, žilové embólie a trombózy, či kŕčové žily) [27].

Svetová zdravotnícka organizácia WHO definovala sedem hlavných rizikových faktorov spôsobujúcich KVO, a to [28]:

- vysoký krvný tlak,
- fajčenie,
- zvýšená hladina celkového cholesterolu v krvi,
- nadváha, resp. obezita,
- nedostatočná fyzická aktivita,
- nedostatočná konzumácia ovocia a zeleniny,
- nadmerná konzumácia alkoholu.

Rizikové faktory kardiovaskulárnych príhod však možno rozdeliť aj dvoch skupín, a to na základe toho, či sú alebo nie sú modifikovateľné. K časti nemodifikovateľným rizikovým faktorom možno zaradiť vek, pohlavie, genetickú predispozíciu, či príslušnosť k danému etniku pacienta. Naopak do skupiny modifikovateľných rizikových faktorov sa radia všetky tie, ktoré sme uviedli v bodoch vyššie, pričom je možné ju rozšíriť o nízky HDL cholesterol, zvýšenú hladinu cukru v krvi, fibriláciu predsiení

a už spomínané faktory životného štýlu, ktorými okrem iného sú aj pracovné podmienky, finančný príjem či vzdelanie [28].

Liečbu a prevenciu voči vzniku KVO [28] možno taktiež rozdeliť do samostatných skupín, a to:

- primárna prevencia,
- sekundárna prevencia.

Pri primárnej prevencii sa jedná najmä o identifikáciu rizikových faktorov, ktoré pacient môže zmeniť, o návrh plánu zmeny životného štýlu, no taktiež o kontrolné vyšetrenia s vysvetlením zmeny stavu a postupnou zmenou životného štýlu. Do kategórie kontrolných vyšetrení sa radia úkony ako fyzikálne vyšetrenie vrátane základnej antropometrie, spresnenie anamnézy zamerané na KVO a ich výskyt v rodine pacienta, EKG vyšetrenie, vyšetrenie tuku v organizme, rôzne sonografické vyšetrenie a mnoho ďalších [28].

V rámci sekundárnej prevencie je možné hovoriť o včasnej diagnostike ohrozených osôb, o účinných diagnostických a liečebných metódach u jedincov trpiacich artériovou hypertenziou, diabetes mellitus 2. typu či metabolickým syndrómom, a v neposlednom rade o využívaní moderných neinvazívnych, ale aj invazívnych liečebných metód [28].

Z globálneho hľadiska je možné povedať, že KVO predstavujú najčastejšiu príčinu smrti. Na základe štatistík Svetovej zdravotníckej organizácie, až 31% všetkých úmrtí sú spôsobené práve ochoreniami, ktoré súvisia so srdcom a jeho cievny systémom. Z nich, až 85% je zapríčinené infarktom alebo mŕtvicou [1]. Medzi najviac postihnuté krajiny rozhodne patrí Čína a Spojené štáty americké, kde v prípade Číny sa odhaduje, že každý piaty dospelý obyvateľ trpí KVO [2]. USA drží negatívne prvenstvo, keďže na základe štatistík z roku 2019 až 48% obyvateľstva v dospelom veku má tento typ ochorenia [4]. Alarmujúce sú taktiež výsledky hovoriace o situácii na Slovensku [3], na základe ktorých je možné konštatovať, že na Slovensku je o viac než 100% vyššia úmrtnosť na KVO v porovnaní s OECD priemerom a o 300% vyššia v porovnaní so štátmi ako Francúzsko, Holandsko či Taliansko. Až 40% úmrtí pri mužskom a 50% pri ženskom pohlaví predstavujú úmrtia spôsobené práve srdcovo-cievny ochorením.

KVO predstavujú vážny problém, a to z globálneho, celosvetového hľadiska. V nasledujúcich podkapitolách sa zameriame na analýzu štúdií, v ktorých boli využité modely deskriptívnej dátovej analytiky pre analýzu a riešenie určitých problémov týkajúcich sa práve oblasti medicíny a špecificky oblasti kardiovaskulárnych ochorení.

2.2. Aplikačné oblasti popisných modelov

Popisné modely využívané pri dolovaní znalostí z dát majú svoje opodstatnenie v širokom spektre aplikačných oblastí [8]. Relevantné informácie o tejto problematike poskytuje vedecká publikácia autorov Parvez Ahmad a kol. [8]. V rámci nej je možné nájsť mnoho referencií na ďalšie odborné publikácie, ktoré sa zaoberajú využitím algoritmov popisných modelov pri riešení konkrétnych situácií a problémov v oblasti medicíny a zdravotnej starostlivosti. Práve v tejto podkapitole sa zameriame na tie najdôležitejšie z nich, spoločne s problematikou, ktorou sa daný výskumný tím zaoberal.

Ako prvú uvádzame štúdiu vedeckého tímu Patil a kol. [29], ktorý sa v rámci svojej odbornej práce venoval hľadaniu asociačných pravidiel pomocou algoritmu *Apriori* a ich následnej analýze. Pravidlá boli získané z dátovej množiny, ktorú tvorilo 625 lekárskeho záznamov pacientiek nie mladších než 21 rokov. Množina pozostávala z jedného kategorického a zo šiestich numerických atribútov popisujúcich ich vek, BMI, to, koľkokrát boli počas života tehotné, hodnotu ich diastolického krvného tlaku a plazmatickej koncentrácie glukózy, no taktiež pravdepodobnosť výskytu cukrovky na základe rodinnej anamnézy a odhad, či sa pri jednotlivých pacientkach do piatich rokov prejaví ochorenie *diabetes mellitus*. Numerické atribúty boli transformované na kategorický tvar, a to využitím diskretizácie na intervaly rovnakej šírky, pričom približná šírka týchto intervalov bola definovaná na základe posudku a názoru lekárov. Taktiež z nej boli odstránené všetky záznamy, ktoré obsahovali minimálne jednu nulovú hodnotu. Na základe nájdených asociačných pravidiel je možné povedať, že faktory, ktoré smerujú k indikácii cukrovky v časovom horizonte piatich rokov, sú najmä vysoká hladina plazmatickej koncentrácie glukózy, obezita a vek pacientiek v intervale 40 až 59 rokov. Naopak významné faktory, ktoré neprispievajú k vzniku ochorenia cukrovky sú diastolický krvný tlak nižší než 80, vek v intervale 20 až 39 rokov, nízky počet otehotnení a nízka hodnota pravdepodobnosti výskytu cukrovky na základe rodinnej anamnézy.

Algoritmus *Apriori* bol taktiež použitý aj výskumným tímom Ilayaraja a kol. [30]. Ten v rámci svojej vedeckej činnosti analyzoval frekvenciu výskytu ochorení v rôznych geografických oblastiach v určitých časových úsekoch. Dátová množina, s ktorou výskumný tím pracoval, pochádza z roku 2012 a pozostáva z 1216 záznamov pacientov postihnutých rôznymi chorobami. Parametre, resp. metriky algoritmu *Apriori* boli v prípade minimálnej podpory určené hodnotou 0.35 a pri minimálnej akceptovanej spoľahlivosti hodnotou 0.9. Výslednú množinu tvorilo 49 nájdených asociačných pravidiel, pričom to najdlhšie z nich pozostávalo z štyroch rôznych hodnôt. Vďaka nájdeným asociačným pravidlám je možné konštatovať, že pacienti, ktorých lekárske záznamy tvorili využitú dátovú množinu, najčastejšie disponovali ťažkosťami v podobe srdcových ochorení a ochorenia pečene, pričom trpeli nadváhou a boli aktívni fajčiari.

Výskumný tím Nahar a kol. [31] využil taktiež algoritmus *Apriori* s cieľom identifikovať pravidlá pre pacientov s kardiovaskulárnymi ochoreniami. Súčasťou tohto výskumu bolo definovanie pravidiel zvlášť pre skupinu chorých a zvlášť skupinu zdravých ľudí. Vďaka získaným výsledkom bolo možné odhaliť faktory, ktoré spôsobovali srdcové ochorenia pri pacientoch mužského, ale aj ženského pohlavia. Výsledky naznačujú, že pri mužskom pohlaví sa jedná najmä o hypertrofiu komory. Faktory, ktoré indikujú KVO pri mužskom aj ženskom pohlaví je asymptomatická bolesť na hrudníku, no taktiež bolesť na hrudníku vyskytujúca sa v dôsledku fyzickej aktivity, resp. cvičenia.

Nasledujúca vedecká práca, na ktorej sa podieľal výskumný tím Wang a kol. [33], sa zamerala na identifikovanie frekvencovaných vzorov v rámci viactriednej dátovej množiny algoritmom *FP-Growth*. Dátová množina pozostáva z viac než 270 atribútov, ktoré pochádzajú z vyšetrenia EKG. Cieľom samotnej štúdie bolo nájsť vo vygenerovaných asociačných pravidlách vzory, ktoré je možné overiť lekársym posúdením, pričom tieto pravidlá môžu poslúžiť pre lepšie pochopenie problematiky a ako základ pre model viactriednej klasifikácie. Ako sa ukázalo, využitie algoritmu *FP-Growth* bolo dobrou voľbou. Na základe tohto algoritmu sa podarilo získať niekoľko smerodajných záverov, ako napríklad, že arytmia je úzko prepojená s ischémiou myokardu a rovnako tak s reperfúznym poškodením. K reperfúznemu poškodeniu [36] dôjde vtedy, keď sú určité tkanivá poškodené náhlým nedostatkom kyslíka, pričom následne dôjde k obnoveniu jeho prísunu. Príčinou je narušenie krvného obehu, zvyčajne ide o dôsledok akútneho tepnového uzáveru. Tkanivá sú náhle poškodené nedostatkom kyslíka, no taktiež živín a cieľom ošetrojúcich lekárov je, prirodzene v čo najkratšej dobe, obnoviť krvný obeh a zachrániť ischemické tkanivo. To však vedie k spomínanému reperfúznemu poškodeniu.

Na ďalšej ukážke vedeckej práce participoval výskumný tím Belciug a kol. [32]. Ich primárnym zámerom bolo využitie aglomeratívneho hierarchického prístupu k zhľukovaniu pre zoskupenie pacientov vzhľadom na ich dĺžku pobytu v nemocnici, s cieľom vypracovať analýzu pre zabezpečenie efektívneho využitia zdravotníckych a medicínskych zdrojov a poskytnúť tak pacientom kvalitnejšie služby. Dátová množina, s ktorou počas výskumu pracovali, pozostávala z viac než 100 tisíc záznamov pacientov, ktorí utrpeli cievnu mozgovú príhodu a boli hospitalizovaní v nemocniciach, v Anglicku, a to v období od 1. apríla 1994 do 31. marca 1995. Atribúty, ktoré sa v tejto dátovej množine nachádzali popisovali pohlavie a vek pacienta, spôsob prijatia do nemocničného zariadenia, pod aký okresný a regionálny zdravotnícky úrad daná nemocnica spadala, či hospitalizácia prebehla cez víkend, ale aj informáciu o tom, v akom ročnom období bol pacient hospitalizovaný. Dôležitým atribútom je taktiež ten, ktorý popisuje dĺžku pobytu v nemocnici jednotlivých pacientov. Tu je vhodné uviesť, že priemerný počet dní, počas ktorých boli pacienti hospitalizovaní, je štrnásť, pričom štandardná odchýlka je 52 a rozsah hodnôt pri tomto atribúte je

0 až 4907. Pre výpočet vzdialenosti medzi jednotlivými záznamami bola využitá metrika, ktorá sa nazýva *Euklidovská vzdialenosť* a ktorú sme bližšie charakterizovali v predchádzajúcej kapitole. Zo získaných výsledkov vyplýva, že využitie daného prístupu k zhľukovaniu môže skutočne viesť k návrhu a zlepšeniu podmienok a efektívnejšiemu využívaniu nemocničných zdrojov.

Dôležitá, z hľadiska medicíny a zdravotnej starostlivosti, je aj vedecká práca výskumného tímu Bertsimas a kol. [34]. Tí sa v rámci svojej odbornej činnosti zamerali na predikciu nákladov na zdravotnú starostlivosť, na základe dátovej množiny obsahujúcej informácie o viac než 800 000 poistných udalostí, ktoré nastali v horizonte troch rokov. Pre predikciu boli využité prístupy klasifikačných stromov a zhľukovacích algoritmov. Ako sa na záver štúdie ukázalo, vyžitie kombinácie prediktívnych a popisných modelov bolo veľmi efektívne, čo viedlo k vytvoreniu nástroja s vysokou presnosťou predikcie.

Zaujímavá je taktiež štúdia výskumného tímu, v rámci ktorého spolupracovali Ogbubaor G. a kol. [35]. Tento tím sa v rámci svojej vedeckej činnosti zameral na otestovanie dvoch algoritmov zhľukovania, ktorými sú *k-Means* a *DBSCAN*. Pre stanovenie optimálneho počtu zhľukov využili tzv. metódu siluety, ktorú sme špecifikovali v predchádzajúcich kapitolách. Dátová množina, s ktorou v rámci výskumu pracovali, pozostávala z údajov o fyzickej aktivite jednotlivých respondentov nameranej pomocou senzorov v mobilných zariadeniach. Po spracovaní údajov, následnej selekcii atribútov a aplikovaní algoritmov zhľukovania na dátovú množinu bolo možné pozorovať, že výsledné klastre obidvoch algoritmov majú silnú vnútroskupinovú súdržnosť a rovnako tak aj dostatočnú medzishľukovú vzdialenosť. Na základe čoho je možné usúdiť, že pre danú dátovú množinu je možné využiť obidva algoritmy. Rozdielne však bola výsledné hodnoty metriky siluet. Pri algoritme *k-Means* sa jednalo o hodnotu 0.97, zatiaľ čo pri algoritme *DBSCAN* o 0.91. *k-Means* bol taktiež oveľa efektívnejší aj z hľadiska výpočtového času.

Príkladov oblastí využitia popisných modelov je mnoho. Na základe doterajších tvrdení je možné povedať, že deskriptívne dolovanie v dátach, resp. popisné modely sú jednoznačne relevantné a využívané v rôznych odvetviach medicíny, či zdravotnej starostlivosti. V nasledujúcich podkapitolách sa zameriame na detailné interpretovanie tých štúdií, ktoré sú z hľadiska témy našej diplomovej práce relevantné a využitie techník a prístupov v nich je teoreticky aplikovateľné aj v našom prípade. Taktiež sa zameriame na samotný vývoj používania popisných modelov v oblasti medicíny a zdravotnej starostlivosti.

2.3. Vývoj používania popisných modelov v oblasti medicíny

Využitie spôsobov dolovania dát v jednotlivých aplikačných oblastiach je zväčša rôzne. V niektorých prípadoch dominujú prediktívne modely, v iných naopak deskriptívne [37]. V rámci

tejto podkapitoly sme sa zamerali na analýzu tých vedeckých publikácií, ktoré mapovali vývoj prístupov k dolovaniu dát v oblasti medicíny, či zdravotnej starostlivosti. Vzhľadom na to, že pre potreby našej diplomovej práce sú relevantné práve popisné modely, bude na ne kladený väčší dôraz.

Štúdia autorov Neesha Joth a kol. [38] sa zamerala na mapovanie vývoja spôsobov dolovania dát v medicíne na základe literatúry, vedeckých článkov a štúdií z rokov 2005 až 2015. Práve toto obdobie je dôležité, a to najmä z dôvodu, že sa v zdravotníckom priemysle postupne formovali nové rozšírenia prístupov k dolovaniu dát, ktoré zohrávajú významnú úlohu najmä pri vývoji metodík zberu údajov z online databáz. Samotné vyhľadávanie prebiehalo na základe kľúčového výrazu „*application of data mining techniques in healthcare*“, a to vo vedeckých prácach, ktoré je možné dohľadať v databázach *ScienceDirect*, *Springerlink* a *IEEE Xplore*. Počiatočná dátová množina bola tvorená z 3840 záznamov, avšak po vyfiltrovaní tém a redukciách sa tento počet znížil na konečných 50. Ako ukazujú výsledky, vo väčšine vedeckých článkov nachádzajúcich sa vo finálnej dátovej množine boli využité prediktívne modely, a to v pomere 47 ku 3 deskriptívnym.

Vzhľadom na fakt, že z hľadiska cieľa našej diplomovej práce sú podstatné práve popisné modely, uvádzame zastúpenie jednotlivých algoritmov spadajúcich do tejto kategórie: asociačné pravidlá 4-krát, zhľukovanie 2-krát a detekcia anomálií 1-krát.

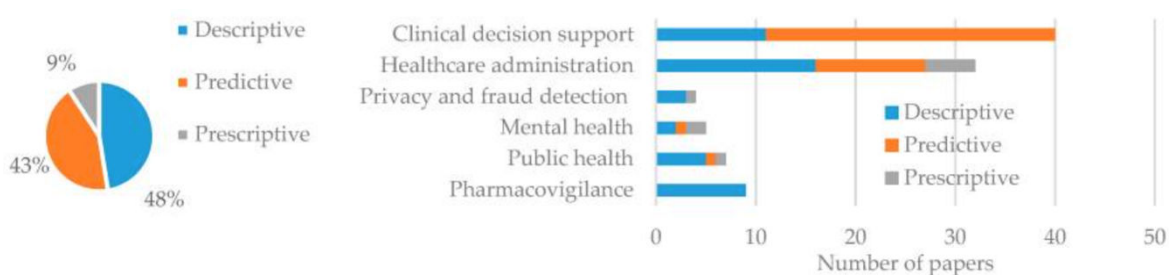
Štúdia autorov Md Saiful Islam a kol. [37] sa taktiež zamerala na analýzu využitia jednotlivých prístupov dolovania dát v oblasti medicíny a zdravotnej starostlivosti.

Primárnym cieľom tohto výskumu bolo zmapovanie prístupov k úlohám dolovania dát v oblasti medicíny a zdravotnej starostlivosti na základe publikovaných štúdií, odborných článkov a literatúry, z rokov 2005 až 2016. Vyhľadávanie týchto publikácií sa uskutočnilo v databázach a webových katalógoch, akými sú *Web of Science*, *PubMed* či *Google Scholar*. Vyhľadávanie prebiehalo v dvoch fázach. V prvej z nich boli využité nasledujúce kľúčové výrazy a rôzne kombinácie týchto výrazov: „*healthcare*“, „*health care*“ a „*data analysis*“. V druhej fázy sa kľúčové slová a slovné spojenia obmenili na „*healthcare*“, „*health care*“, „*disease*“, „*genomics*“, „*data mining*“ a „*big data*“. Prvotná dátová množina pozostávala z 6084 záznamov, avšak po dodatočných redukciách sa tento počet znížil na finálnych 117.

Prvotné štatistické analýzy boli zamerané na samotné štúdie, a to z pohľadu roku ich publikácie, pričom najviac ich bolo publikovaných v rokoch 2015 a 2011, no taktiež z pohľadu vedeckých časopisov, ktoré dané práce uverejnili. Z hľadiska témy našej diplomovej práce sú však podstatné tie štatistické analýzy, ktoré interpretujú informácie z medicínskeho hľadiska, resp. z prostredia medicíny či zdravotnej starostlivosti. Prvá dátová analýza tohto typu bola zameraná na

identifikovanie rôznych prístupov k riešeniam úloh dolovania dát. Z uvedených výsledkov vyplýva, že v 48% vedeckých publikácií nachádzajúcich sa v konečnej dátovej množine, boli využité algoritmy popisných modelov. Ďalej nasledujú algoritmy prediktívnych modelov so zastúpením na úrovni 43% a preskriptívny prístup na úrovni 9%. Ďalšia analýza bola zameraná práve na identifikovanie konkrétnych algoritmov deskriptívneho dolovania dát. Výsledky uvádzajú výrazné zastúpenie algoritmov asociačných pravidiel a zhukovania, konkrétne v pomere 21 a 18, pričom len v jednej štúdii bol využitý algoritmus na detekciu anomálií.

Veľmi dôležitá je taktiež analýza, ktorej výsledky hovoria o zastúpení jednotlivých prístupov v šiestich definovaných aplikačných oblastiach (Obrázok 1).



Obrázok 1 Využitie rôznych prístupov riešenia data mining úloh, v šiestich definovaných aplikačných oblastiach [7]

Ako je možné vidieť na grafe (Obrázok 1), najväčšie zastúpenie popisných modelov je práve v aplikačnej oblasti *Healthcare administration*, ktorej hlavný cieľ je zabezpečenie rovnakej dostupnosti zdravotnej starostlivosti pre všetky skupiny obyvateľstva.

Druhá v poradí nasleduje aplikačná oblasť *Clinical decision support*, ktorá, ako už názov napovedá, je venovaná analýze, extrakcii a interpretácii získaných znalostí o rôznych ochoreniach a rizikách, s cieľom poskytnúť zdravotníckemu personálu podporu pri rozhodovaní v dôležitých okamihoch.

Výraznú prevahu využitia popisných modelov môžeme sledovať aj v odvetví *Privacy and fraud detection*, ktorá je zameraná na detekciu podvodov, neoprávnenej či podozrivej aktivity, no taktiež na ochranu osobných údajov pacientov, ktorých údaje sa nachádzajú v databázach. Podobne je tomu aj v aplikačnej oblasti *Public health*, ktorá sa venuje analýze tých problémov, ktoré ovplyvňujú masovú populáciu, región alebo krajinu. Odvetvie *Mental health*, ktorej procesy možno považovať za podporné pri analytickom rozhodovaní pri pacientoch psychiatrie alebo pacientoch s duševnými poruchami, je zastúpené takmer rovnomerne všetkými tromi prístupmi k dolovaniu dát.

Špecifický prípad je aplikačná oblasť *Pharmacovigilance*, pri ktorej môžeme sledovať 100% zastúpenie popisných modelov. Pre vysvetlenie, procesy tejto aplikačnej oblasti sú zamerané na

analýzu a monitorovanie nežiadúcich prejavov a reakcií na lieky následne po tom, ako sú uvedené na trh.

Výsledky analyzovaných štúdií sa v značnej miere líšia. Zatiaľ čo tá prvá [38] interpretuje prediktívne modely ako jasne dominujúce, v rámci druhej štúdie [37] je pomer prediktívnych a deskriptívnych modelov vyrovnaný. Tento rozdiel možno vysvetliť viacerými faktormi. Prvým z nich je veľkosť konečnej dátovej množiny, ktorá bola v druhom prípade dvojnásobne väčšia, a teda aj spektrum prístupov k úlohám dolovania dát bolo pestrejšie. Dôležitý faktor sú taktiež kľúčové výrazy a ich kombinácie, ktoré boli využité pri vyhľadávaní vedeckých prác a odborných textov. Okrem toho, do druhej štúdie bola zaradená literatúra publikovaná až do roku 2016, taktiež výskum bol publikovaný v rôznych databázach a vedeckých časopisoch čo taktiež môže prispievať k rozdielu.

Zo získaných informácií je však zjavné, že popisné modely majú v sfére medicíny a zdravotnej starostlivosti svoje jasné opodstatnenie. Dokonca, v niektorých aplikačných oblastiach je tento prístup zastúpený v oveľa väčšom rozsahom.

2.4. Asociačné pravidlá a ich využitie v oblasti chronických ochorení

Zvyčajne existuje veľa vzťahov medzi ochoreniami, alebo zdravotným stavom pacienta a určitými príznakmi [39]. Objavenie takýchto vzťahov pomáha zdravotníkom lepšie porozumieť celej situácii a efektívnejšie postupovať pri liečbe.

V rámci výskumu vedeného Nahar a kol. [31] sa zamerali na hľadanie asociačných pravidiel, ktoré hovoria o symptómoch spôsobujúcich srdcovocievne ochorenia a extrakcii asociačných pravidiel, ktoré možno definovať ako „zdravé“, t.j. kombinácie hodnôt rizikových faktorov, ktoré tieto ochorenia nespôsobujú. Pri hľadaní asociačných pravidiel bol využitý algoritmus *Apriori*. V rámci výskumu bola využitá verejne dostupná dátová množina *UCI Heart Disease Data Set*, ktorá pozostáva z atribútov charakterizujúcich zdravotný stav pacientov. Finálna dátová množina pozostávala zo štrnástich atribútov, medzi ktoré patria informácie o veku, pohlaví, krvnom tlaku nameranom v pokoji, hladine cholesterolu, hladine cukru v krvi, stave srdca či niekoľkých ďalších parametrov jednotlivých respondentov. Taktiež je v nej ale možné nájsť cieľový atribút, ktorý môže nadobúdať päť rôznych hodnôt, a teda, či je respondent zdravý (hodnota 0), alebo sa u neho prejavili nejaké srdcové ochorenia (hodnoty 1 až 4, podľa závažnosti ochorenia). Hodnoty tohto atribútu však boli upravené na binárny tvar, a to na triedy „zdravý“ a „chorý“.

V rámci prvej analýzy boli využité všetky dáta. Analýza spočívala v extrakcii asociačných pravidiel, ktoré hovoria o symptómoch spôsobujúcich srdcovocievne ochorenia a analogicky extrakcii asociačných pravidiel, ktoré možno definovať ako „zdravé“, t.j. kombinácie hodnôt rizikových

faktorov, ktoré tieto ochorenia nespôsobujú. Vzhľadom na fakt, že v medicínskej oblasti sa zistilo, že pohlavie človeka je dôležitý aspekt ovplyvňujúci riziko výskytu srdcovocievnych chorôb, druhý experiment sa zameral na definovanie asociačných pravidiel osobitne pre mužskú a ženskú časť respondentov. Vygenerované výsledky priniesli mnohé zaujímavé zistenia. Ukázalo sa, pri mužskom pohlaví je významným faktorom pre predikciu ochorenia srdca hypertrofia komory, ktorú je možné identifikovať EKG vyšetrením. Medzi významné faktory, ktoré indikujú kardiovaskulárne ochorenie rovnako pri oboch pohlaviach možno zaradiť asymptomatickú bolesť na hrudníku, no taktiež bolesť na hrudníku, ktorá je spôsobená fyzickou aktivitou, resp. cvičením.

Algoritmus *Apriori* je možné využiť aj v spojitosti s iným chronickým ochorením, ako je napríklad ochorenie COVID-19. Pri ochorení Covid-19 bola identifikovaná široká škála príznakov, od mierneho nachladnutia až po závažné zdravotné komplikácie, ktoré môžu viesť aj k úmrtiu. Práve pochopenie často sa opakujúcich vzorov, v tomto prípade symptómov, v rôznych skupinách pacientov, v značnej miere zvyšuje účinnosť liečby. Tejto problematike sa venoval výskumný tím *Meera Tandan a kol.* [39]. Hlavným cieľom štúdie teda bolo identifikovať vzory pri symptómoch u pacientov s ochorením Covid-19 rozdelených do skupín podľa veku, pohlavia, výskytu chronických ochorení a podľa toho, či pacient chorobu prekonal, alebo jej podľahol. Dáta pre potreby tejto štúdie boli extrahované 2.6.2020 z platformy „*Wolfram Data Repository*“, pričom posledný záznam pochádza z 27.5.2020. Vzhľadom na pôvod dát, nie je možné určiť spôsob ich zhromažďovania. Je však možné predpokladať, že informácie o jednotlivých pacientoch pochádzajú z lekárskeho záznamov, a teda z dokumentov. Dátová množina pozostávala z 1783 záznamov, pričom každý záznam tvorili hodnoty piatich atribútov. Jednalo sa o informácie ako vek a pohlavie pacienta, príznaky ochorenia COVID-19, výskyt akýchkoľvek iných chronických ochorení a informáciu o tom, či daný pacient ochorenie prekonal alebo mu podľahol. V rámci štúdie sa v prvom kroku aplikoval algoritmus *Apriori* na spracovanú dátovú množinu, čoho výsledkom bolo identifikovanie prvotných pravidiel. V druhom kroku boli odstránené redundantné pravidlá a následne sa z nich pomocou Fisherovho exaktného testu vybrali len tie skutočne významné. Vďaka zloženiu dátovej množiny bolo možné identifikovať vzory pri symptómoch v jednotlivých kategóriách, resp. skupinách. Pre identifikovanie zriedkavých vzorov pri symptómoch bola zvolená nízka miera podpory a vysoká miera spoľahlivosti. Táto myšlienka bola prebraná z inej štúdie, ktorá sa ale taktiež venovala dolovaniu znalostí z medicínskych dát. V prípade, že symptóm, ktorý sa vyskytuje len zriedkavo a je silne prepojený s iným zriedkavým symptómom, je nevyhnutné ponechať pravidlá charakterizujúce tieto symptómy. Práve takéto pravidlá poskytujú lekárom a zdravotníckemu personálu cenné informácie o nových ochoreniach, akým je aj koronavírus. V rámci analýz, ktoré boli vykonané na tejto dátovej množine sa zistilo, že najčastejšie príznaky ochorenia boli horúčka (69%), kašeľ (37%),

malátnosť / bolesť tela (11%), zápal pľúc (11%) a bolesť hrdla (8%). Bolesť hlavy, nevoľnosť, hnačka, syndróm respiračnej tiesne a septický šok sa vyskytol pri 1% - 5% pacientov. Príznaky ako infarkt myokardu, zlyhanie srdca a ochorenie obličiek bolo evidovaných u menej ako 1% pacientov. Najvýznamnejšie pravidlá tvorené zo symptómov ochorenia COVID-19 boli definované nasledovne:

- pacienti mladší ako 20 rokov = [zápal očných spojiviek, silná nádcha],
- pacienti vo veku 20 až 45 rokov = [sucho v ústach, bolesť hrdla],
- pacienti vo veku 45 až 65 rokov = [nevoľnosť, slabosť],
- pacienti vo veku 65 rokov a viac = [anorexia, horúčka].

U pacientov starších ako 45 rokov tvorili väčšinu pravidiel príznaky asociované so srdcom, a teda zlyhanie srdca, srdcová arytmia, infarkt myokardu a dýchacími ťažkosťami, ako zápal pľúc alebo bolesť hrdla.

Súčasťou tejto štúdie bolo taktiež porovnanie výpočtového času troch algoritmov pre hľadanie asociačných pravidiel, a to *Apriori*, *FP-Growth* a *Eclat*. V rámci testovania bola využitá dátová množina, ktorú sme bližšie charakterizovali v predchádzajúcej časti. Pre všetky tri algoritmy boli definované rovnaké hodnoty metrík, t.j. minimálna podpora a minimálna spoľahlivosť. Zo získaných výsledkov je možné konštatovať, že *Apriori* bol rýchlejší než zvyšné dve metódy, rozdiel bol však veľmi malý. Výpočtový čas algoritmu *Apriori* bol 0.03 sekundy, zatiaľ čo pri *FP-Growth* to bolo 0.05 sekundy a pri algoritme *Eclat* 0.04 sekundy. Dôvod, prečo bol algoritmus *FP-Growth* pomalší než *Apriori* môže byť ten, že prvý menovaný je optimalizovaný najmä pre veľké množstvo dát.

2.5. Metódy zhlukovania a ich využitie v oblasti chronických ochorení

Autori Matthew W. Segar a kol. [40] sa v rámci svojej vedeckej práce zamerali na identifikáciu rizikových skupín obyvateľstva práve pomocou metód zhlukovania. Definovanie týchto skupín na základe širokého spektra atribútov by výrazným spôsobom dopomohlo k efektívnejšej a cielenejšej zdravotnej pomoci, čo by sa v konečnom dôsledku odzrkadlilo aj na nepriaznivom trende úmrtí v jednotlivých krajoch USA. V rámci štúdie boli využité dáta z databázy „*Country Health Ranking (CHR)*“ a zdravotníckeho centra „*Centers for Disease Control and Prevention Wide-Ranging Online Data for Epidemiologic Research (CDC WONDER)*“. Databáza nadácie *CHR* pozostáva z údajov pochádzajúcich z rokov 2016 až 2018 a obsahuje informácie týkajúce sa zdravia z viac než 3 000 krajov v USA. Je tvorená z údajov, ktoré by sa dali začleniť do štyroch hlavných skupín, ktoré sú: prístup k zdravotnej starostlivosti, sociálne a ekonomické faktory, životospráva, vplyv životného priestoru (pozn. - tu sa myslia faktory ako miera kriminality či miera sociálneho združovania sa). Databáza zdravotníckeho centra *CDC WONDER* poskytla informácie o počte a príčinách úmrtí na úrovni jednotlivých krajov v USA. Finálna dátová množina bola tvorená z 37 atribútov a 2676

záznamov. Pre identifikáciu jednotlivých zhlukov bol využitý model nekontrolovaného hierarchického zhlukovania hlavných komponentov. Ide o proces, pri ktorom dáta prechádzajú aglomeračným hierarchickým zhlukovaním. Keďže sa v databáze vyskytli aj extrémne hodnoty, ktoré môžu skresliť výsledky zhlukovania, bolo potrebné vykonať analýzu hlavných komponentov, aby sa zredukovala dimenzionalita kovariančných dát bez straty významných rozdielov medzi dôležitými atribútmi. Na analýzu trendu úmrtí spôsobených KVO, akými sú napríklad ischemická choroba srdca, či zlyhaním srdca, ale aj celkového počtu úmrtí, vzhľadom na vek pre všetky definované skupiny poslužil algoritmus zmiešaného lineárneho modelu.

Výsledky tejto štúdie priniesli veľmi dôležité zistenia. V rámci analýzy boli identifikované štyri rôzne skupiny obyvateľstva, odlišujúce sa v mnohých parametroch. Je teda možné tvrdiť, že vďaka zhlukovaniu demografických údajov a údajov o správaní a živote obyvateľstva je možné identifikovať ich jedinečné skupiny s rozdielnou mierou úmrtnosti spôsobenej KVO a rozdielnou mierou celkovej úmrtnosti vyjadrenej v percentách. Identifikovanie týchto skupín môže výrazne pomôcť s cieľovou a efektívnou pomocou.

Druhá vedecká práca zaradená do tejto časti bola uskutočnená vedeckým tímom Guo Q. a kol. [41]. Tí sa v rámci svojho výskumu zamerali na identifikovanie a následnú analýzu jedinečných skupín pacientov trpiacich hypertenziou, pomocou algoritmu zhlukovania *k-Means*. Samotná dátová množina bola tvorená lekárskymi záznamami pacientov pochádzajúcich z obdobia rokov 2012 až 2014, ktorých hodnoty boli získané pomocou špeciálneho zariadenia na monitorovanie krvného tlaku. Toto zariadenie daní respondenti nosili na svojej ruky po dobu 24 hodín. Okrem toho boli súčasťou dátovej množiny taktiež identifikačné atribúty a atribúty týkajúce sa životného štýlu pacientov. Počiatočný počet záznamov bol 1740, avšak po vyradení tých pacientov, ktorí netrpeli hypertenziou a nemali kompletne lekárske záznamy sa ich počet zredukoval na 791. Z tejto vzorky boli následne odstránené všetky tie záznamy pacientov, ktoré buď nespĺňali ďalšie podmienky, a teda ich vek bol buď pod hranicou 18 alebo nad hranicou 90 rokov, ale vyradení boli taktiež tí, ktorí trpeli závažnou poruchou spánku, pracovali v nočných zmenách, pacientky - ženy boli počas daného obdobia tehotné a ďalší, ktorí nezapadali do požadovanej vzorky. Finálna dátová množina pozostávala z 513 záznamov. Pri zhlukovaní bol využitý obľúbený a často využívaný algoritmus *k-Means*, ktorého procesy prebiehali v jeho štandardných krokoch. V rámci tejto analýzy boli identifikované štyri zhluky pacientov. Ich charakteristické črty je možné zhrnúť nasledovne:

- Prvý zhluk bol tvorený 172 pacientami, prevažne mladými mužmi - fajčiarmi s najvyšším systolickým aj diastolickým krvným tlakom zo všetkých skupín, najnižšou hladinou HDL, avšak bez diagnostikovanej ischemickej choroby srdca.

- Druhá skupinu tvorilo 70 pacientov, jednalo sa najmä o ženy vo vyššom veku, s nízkym diastolickým krvným tlakom, ale s najvyššou mierou hladiny glukózy a cholesterolu.
- V rámci tretieho zhľuku bolo identifikovaných 144 pacientov, nefajčiarov, prevažne žien s nízkym systolickým krvným tlakom, pričom daní pacienti netrpeli cukrovkou ani koronárnou chorobou srdca, čo potvrdzuje fakt, že je to skupina s najväčším zastúpením pacientov s normálnou hrúbkou krčných tepien.
- Poslednú, štvrtú skupinu tvorilo 127 pacientov, pre ktorých bol charakteristický zvýšený krvný tlak v nočných hodinách oproti denným, najvyššia hladina HDL a taktiež to, že všetkým z nich bola diagnostikovaná ischemická choroba srdca.

Analýza zhľukovania jasne preukázala heterogenitu hypertenzie. Rovnako tak preukázala jedinečné charakteristické vlastnosti definovaných skupín pacientov, ktorí týmto ochorením trpia. Aj práve vďaka identifikovaným klastrom môžu byť prístup k pacientom efektívnejší a oveľa viac cielenejší.

3. Návrh systémového riešenia

Obsahom tretej kapitoly našej diplomovej práce je návrh systémového riešenia, v podobe aplikácie pre využitie popisných modelov aplikovaných na dátovú množinu pozostávajúcu zo zdravotných údajov skutočných pacientov hospitalizovaných vo VÚSCHu. Cieľom taktiež je aby bolo možné do softvéru nahráť a ďalej pracovať s akoukoľvek inou vstupnou dátovou množinou.

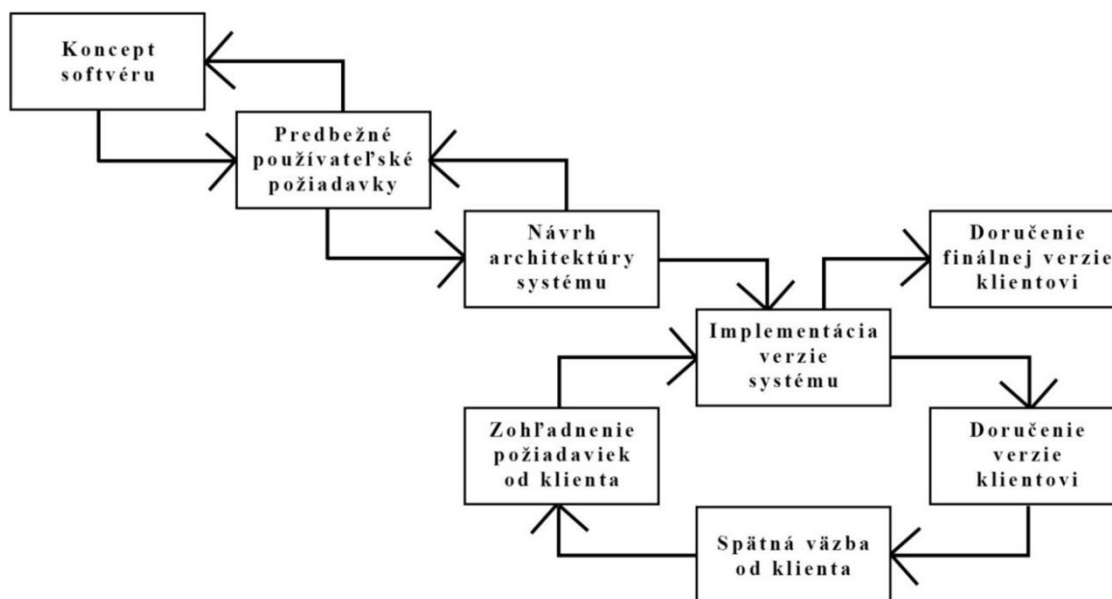
V prvej časti tejto kapitoly sa zameriame na výber vhodného modelu životného cyklu, ktorým definujeme spôsob a kroky tvoriace proces realizácie navrhovaného systémového riešenia. V druhom kroku navrhujeme koncept softvérového riešenia. Následne vypracujeme analýzu používateľských požiadaviek, v rámci ktorej taktiež charakterizujeme cieľových používateľov, ich role a právomoci, ktorými disponujú. Túto analýzu doplníme návrhom UML diagramu prípadov použitia, pričom vytvoríme aj používateľské scenáre, ktoré odzrkadľujú činnosti vykonateľné v navrhovanej aplikácii. Na záver sa zameriame na vytvorenie, resp. vizuálnu a slovnú prezentáciu architektúry tohto systému, návrh vizuálnych prototypov a základný opis fungovania softvéru.

3.1. Model životného cyklu

Model životného cyklu predstavuje základný a jeden z najdôležitejších prvkov pri návrhu a následnej realizácii akéhokoľvek softvérového riešenia. Práve daný model definuje postupnosť krokov, resp. fázy procesu formulujúce vývoj daného produktu, v našom prípade aplikácie. Modelov životného cyklu je hneď niekoľko, pričom každý z nich je využiteľný a vhodný pre inú situáciu.

Pre potreby našej diplomovej práce sme zvolili tzv. evolučný model životného cyklu vývoja softvérového, resp. systémového riešenia [42]. Ten je možné charakterizovať ako model postupných, na seba nadväzujúcich krokov, alebo tiež fáz, ktoré po celý čas podliehajú spätnej väzbe od klienta, a teda zadávateľa požiadaviek. V kontexte našej diplomovej práce je možné povedať, že klient reprezentuje skupinu osôb, ktorá je zložená z vedúceho, konzultantov záverečnej práce a doménového experta. Vývojový tím, ktorý sa podieľa na implementácii navrhovaného systémového riešenia musí byť flexibilný a schopný včas a vhodne reagovať na prípadné zmeny požiadaviek či zadania. Celý proces sa teda vyvíja v čase. Evolučný model životného cyklu je taktiež charakteristický rozdelením práce na niekoľko menších parciálnych častí, ktoré sú jednotlivo po vypracovaní dodávané zadávateľovi úloh, resp. klientovi. Ten je tak stálym účastníkom celého vývoja, má prehľad, je informovaný o aktuálnej situácii a svoje požiadavky dokáže meniť včas tak, aby bol konečný výsledok čo najuspokojivejší. Tento fakt výrazne zvyšuje dôveru klienta v samotný vývoj softvérového riešenia.

Na obrázku nižšie (Obrázok 2) sme znázornili jednotlivé fázy evolučného modelu.



Obrázok 2 Evolučný model životného cyklu vývoja softvéru [48]

Prvou fázou je tzv. „*Koncept softvéru*“. Ten je zameraný na definovanie základných náležitostí a pravidiel, akými sú výber programovacieho jazyka, forma, resp. spôsob prevedenia navrhovaného softvéru, charakteristika procesov, ktoré je nutné naplniť pre plynulý prechod na ďalšie úrovne a mnohé iné.

Druhá fáza nesie názov „*Predbežné používateľské požiadavky*“. Pod tým si je potrebné si predstaviť niekoľko samostatných, ale medzi sebou prepojených častí, kde je možné zaradiť identifikovanie cieľových používateľov, definovanie ich rolí a právomocí, ktorými disponujú, či stanovenie používateľských požiadaviek na navrhované systémové riešenie. Avšak, okrem základných používateľských požiadaviek je nutné definovať aj tzv. funkčné a mimofunkčné požiadavky. Pre vysvetlenie, jedná sa o požiadavky, ktoré predstavujú konkrétne procesy a funkcie, ale aj spôsob, akým budú navrhovaným systémovým riešením prezentované a ponúkané. Každá takáto požiadavka musí mať jasne pridelenú prioritu. Jedným zo spôsobov, akým je možné prioritu jednotlivých požiadaviek definovať, je tzv. *MoSCoW metóda*. Celou touto témou sa budeme zaoberať v podkapitole tejto 3.3 tejto diplomovej práce.

Tretiu fázou reprezentuje návrh architektúry systému. Tá definuje, z akých úrovní bude navrhované softvérové riešenie pozostávať. Taktiež to, aké technológie, programové prostriedky a metódy budú na jednotlivých úrovniach použité a v neposlednom rade, ako budú jednotlivé úrovne medzi sebou komunikovať a spolupracovať.

Nasleduje v poradí štvrtá, a to konečná fáza, ktorá je zároveň časovo najnáročnejšia. Jedná sa o samotnú implementáciu navrhovaného softvérového riešenia. Po tom, ako vývojový tím vypracuje prvotnú verziu softvéru, nasleduje jej doručenie klientovi, resp. zadávateľovi požiadaviek. Ten poskytne spätnú väzbu, prípadne návrhy na vykonanie zmien, na základe ktorých vývojový tím implementuje ďalšiu verziu softvéru. Tento proces sa opakuje dovtedy, pokiaľ výsledný produkt, resp. aplikácia nie je v požadovanom stave. Tu je dôležité podotknúť, že nie je možné vykonať návrat z fázy „*Implementácia verzie systému*“ do fáz predošlých, a teda do „*Koncept softvéru*“, „*Predbežné používateľské požiadavky*“ či „*Návrh architektúry systému*“. Nie je tak možné učiniť z dôvodu možného ohrozenia stability realizácie celého projektu.

Opäť je vhodné zdôrazniť, že klient, a teda zadávateľ úloh je stálou súčasťou jednotlivých fáz vývoja navrhovaného softvéru.

Tak, ako je definované v evolučnom modeli životného cyklu, a teda vypracovaním jednotlivých fáz, budeme postupovať aj v našom prípade.

3.2. Koncept softvéru

Detailný technický popis jednotlivých častí aplikácie a funkcionality, ktoré sú jej súčasťou sme podrobne charakterizovali v prílohe D, a teda v systémovej príručke. Architektúru navrhovaného softvérového riešenia je možné nájsť v podkapitole 3.5.

3.3. Analýza používateľských požiadaviek

V nasledujúcej podkapitole sa zameriame na definovanie požiadaviek, ktoré zišli zo spoločných diskusií s klientom. Ako sme vysvetlili už vyššie, v kontexte našej diplomovej práce reprezentuje klient skupinu osôb, ktorej súčasťou je vedúci a konzultanti záverečnej práce. Súčasťou tejto podkapitoly bude taktiež identifikácia cieľových používateľov navrhovaného systému, ich rolí a právomocí, ich potrieb a predstáv o jednotlivých funkciách, či procesoch, ale aj o vizuálnej stránke používateľského prostredia softvéru.

Pre definovanie a priradenie priority jednotlivým používateľským požiadavkám využijeme už vyššie spomenutú, tzv. *MoSCoW metódu* [43]. Tá je využívaná najmä v oblasti projektového manažmentu, obchodných analýz, no taktiež v sfére vývoja systémov či softvérov. Skratka *MoSCoW* je odvodená od prvého písmena každej zo štyroch určených priorít. Dve samohlásky „o“ sú pridané do skratky len pre lepšiu výslovnosť. Jednotlivé stupne dôležitosti je možné zhrnúť do nasledujúcich bodov:

1. **Must Have** - musí mať,
2. **Should Have** - malo by mať,
3. **Could Have** - bolo by dobré, ak by malo,

4. **Won't Have This Time** - zatiaľ nebude mať.

Stupne dôležitosti budú uvedené v zátvorke pri jednotlivých používateľských požiadavkách.

3.3.1. **Cieľoví používatelia, ich roly a právomoci**

Cieľových používateľov navrhovaného softvéru možno rozdeliť do dvoch hlavných celkov. Prvý z nich tvoria lekári, resp. ľudia z lekárskeho prostredia. Opäť je dôležité spomenúť, že v kontexte našej diplomovej práce sa jedná najmä o lekárov v oblasti kardiológie, avšak pri definovaní cieľových používateľov je vhodné zostať v rovine *lekár*.

Druhá skupina je tvorená z používateľov, ktorí nedisponujú znalosťami z oblasti medicíny, avšak majú dostatočné vzdelanie v oblasti dátovej analytiky. V konečnom dôsledku môže ísť o vysokoškolských pedagógov, a taktiež o študentov, pričom je nevyhnutné, aby títo študenti disponovali dostatočným vzdelaním. Preto je možné túto podskupinu ešte viac špecifikovať, a to na študentov s ukončeným bakalárskym vzdelaním v technickej oblasti, ideálne v oblasti dátovej analytiky. Vzhľadom na fakt, že rozsah právomocí sa pri členoch oboch skupín v zásade nelíši, je možné ich zovšeobecniť pod jednu persónu, a to *používateľ*. Jednotlivé časti systému sú navrhované na základe spoločných diskusií so zástupcami oboch vyššie definovaných skupín.

Špecifická persóna, ktorá má taktiež v kontexte navrhovaného systémového riešenia svoju rolu a disponuje výraznými právomocami je práve *administrátor*.

Právomoci jednotlivých persón, a teda *používateľa* a *administrátora* sme zhrnuli do nasledujúcich zoznamov nižšie.

Administrátor:

- spravuje celý softvér,
- má prístup k dátovej množine nahranej a umiestnenej v aplikácií (dátová množina obsahujúca údaje o reálnych pacientoch VÚSCH),
- má možnosť nahráť vlastné dáta do softvéru,
- má možnosť využívať všetky funkcionality, ktoré systémové riešenie ponúka,
- má možnosť si získané výsledky, ktoré sú výstupom jednotlivých funkcionalít, stiahnuť do svojho osobného počítača,
- má možnosť analyzovať a prezerať získané výsledky a vizualizácie.

Používateľ:

- má prístup k dátovej množine nahranej a umiestnenej v aplikácií (dátová množina obsahujúca údaje o reálnych pacientoch VÚSCH),

- má možnosť nahráť vlastné dáta do softvéru,
- má možnosť využívať všetky funkcionality, ktoré systémové riešenie ponúka,
- má možnosť si získané výsledky, ktoré sú výstupom jednotlivých funkcionalít, stiahnuť do svojho osobného počítača,
- má možnosť analyzovať a prezerať získané výsledky a vizualizácie.

3.3.2. Požiadavky používateľov podľa priorít

V nasledujúcej podkapitole sa zameriame na definovanie potrieb a predstáv cieľových používateľov, ktorí budú navrhované systémové riešenie využívať. Nejedná sa pritom len o jednotlivé funkcionality, či funkčnosť systému ako takého, ale taktiež o vizuálne prevedenie jeho používateľského rozhrania, či spôsob vizualizácie získaných výsledkov. Požiadavky používateľov rozdelíme do dvoch samostatných celkov, a to podľa toho, či ide o funkčné alebo mimofunkčné požiadavky.

3.3.2.1. Funkčné požiadavky

Funkčné požiadavky možno definovať ako konkrétne procesy či funkcionality v navrhovanom systémovom riešení.

Základné požiadavky:

- systém by mal byť interaktívny, a teda používateľ by mal mať plnú kontrolu nad dátami, ktoré vizualizuje (*Must Have*),
- systém by mal vedieť aplikovať všetky dostupné modely na akékoľvek vstupné dáta (*Must Have*),
- systém by mal vedieť vizualizovať výsledky zo všetkých dostupných modelov, ktoré boli aplikované na akékoľvek dáta (*Must Have*),
- používateľ by mal mať možnosť si získané výsledky, ktoré sú výstupom aplikovaných modelov, stiahnuť do svojho osobného počítača (*Must Have*),
- používateľ by mal mať možnosť nahráť do systému svoju vlastnú dátovú množinu (*Should Have*),
- používateľovi by malo byť umožnené upraviť nahratú dátovú množinu priamo v prostredí aplikácie (pod úpravou máme na mysli zatriedenie numerických atribútov do intervalov, či zmeniť dátový typ jednotlivých atribútov) (*Could Have*),
- v prípade, ak aplikovanie niektorého z modelov trvá príliš dlho, používateľ by mal byť schopný zastaviť tento proces, čo najjednoduchším spôsobom (napríklad kliknutím na príslušné tlačidlo) (*Won't Have This Time*).

3.3.2.2. Mimofunkčné požiadavky

Mimofunkčné požiadavky charakterizujú, akým spôsobom budú jednotlivé procesy a funkcionality ponúkané navrhovaným systémovým riešením.

Bezpečnosť:

- používatelia môžu pristupovať len k tým častiam systému, ku ktorým majú zo svojej pozície, a teda z pohľadu svojej roly oprávnenie (*Should Have*).

Dostupnosť:

- navrhované systémové riešenie by malo byť dostupné v on-line forme, a to v podobe webovej aplikácie (*Won't Have This Time*),
- systém by mal byť umiestnený, resp. nahratý na dostatočne výkonnom, rýchlom a kvalitnom serveri (*Won't Have This Time*),
- systém by mal byť dostupný z vývojového prostredia programovacieho jazyka R (*Must Have*).

Grafické rozhranie:

- používateľské rozhranie aplikácie by malo byť dostatočne intuitívne (*Should Have*),
- časť grafického rozhrania určenej pre vizualizáciu získaných výsledkov musí byť dostatočne veľká (*Should Have*),
- možnosti výberu pre hodnoty parametrov vstupujúcich do daných popisných modelov by mali byť implementované v prehľadom paneli (*Should Have*),
- vizualizácie by mali byť interaktívne (*Could Have*).

Spoľahlivosť:

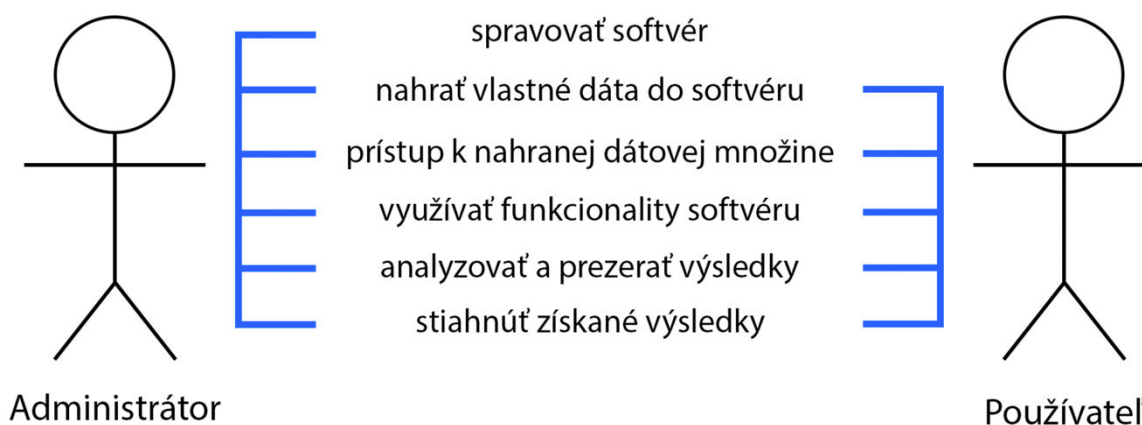
- systém by mal byť schopný pracovať s akoukoľvek vstupnou dátovou množinou, avšak podmienka je, že vstupný súbor musí byť vo formáte .csv, .xls alebo .xlsx (*Must Have*).

3.4. Diagram prípadov použitia

Diagram prípadov použitia [44] možno charakterizovať ako formu vizualizácie charakteristiky správania implementovaného systému a taktiež jednotlivých procesov z pohľadu používateľov. Taktiež definuje, ktoré typy, alebo teda skupiny používateľov s daným systémom pracujú a aké činnosti v rámci neho vykonávajú. Okrem už spomenutého je tiež možné načrtnúť funkčné požiadavky na navrhované softvérové riešenie tak, že popisuje interakciu medzi ním a danými používateľmi.

O diagrame prípadov použitia je taktiež možné povedať, že sa jedná o typ diagramu v jazyku *UML* (anglicky *Unified Modeling Language*) [45]. Jazyk *UML* sa v softvérovom inžinierstve definuje ako grafický jazyk využívaný najmä na vizualizovanie, zaznamenávanie, navrhovanie a dokumentovanie jednotlivých častí programových systémov.

Na nasledujúcom obrázku (Obrázok 3) je znázornený diagram použitia, v rámci ktorého je možné identifikovať dvoch hlavných aktérov navrhovaného systémového riešenia, a to *používateľa* a *administrátora*.



Obrázok 3 Diagram prípadov použitia pri navrhovanej aplikácii

Aktér *používateľ* má po spustení systému možnosť využívať ponúkané funkcionality, nahráť vlastné dáta do softvéru, stiahnuť si získané výsledky do svojho osobného počítača, analyzovať a prezerať si ich v prostredí aplikácie, no taktiež má prístup k dátovej množine, ktorá je do systému nahratá. Druhým aktérom je *administrátor*. Medzi ním a aktérom *používateľ* je tzv. generalizačný vzťah. To znamená, že aktér *administrátor* má právomoc vykonávať všetko to, čo môže vykonávať aktér *používateľ*, avšak okrem toho má jednu dodatočnú právomoc, a to správa celého systému. Pod správou celého systému je nutné si predstaviť úkony, ako úprava programovej logiky jednotlivých funkcií, ako aj celej aplikácie, ale tiež možnosť zasahovať do vizuálnej podoby používateľského rozhrania softvéru.

3.5. Architektúra navrhovaného systémového riešenia

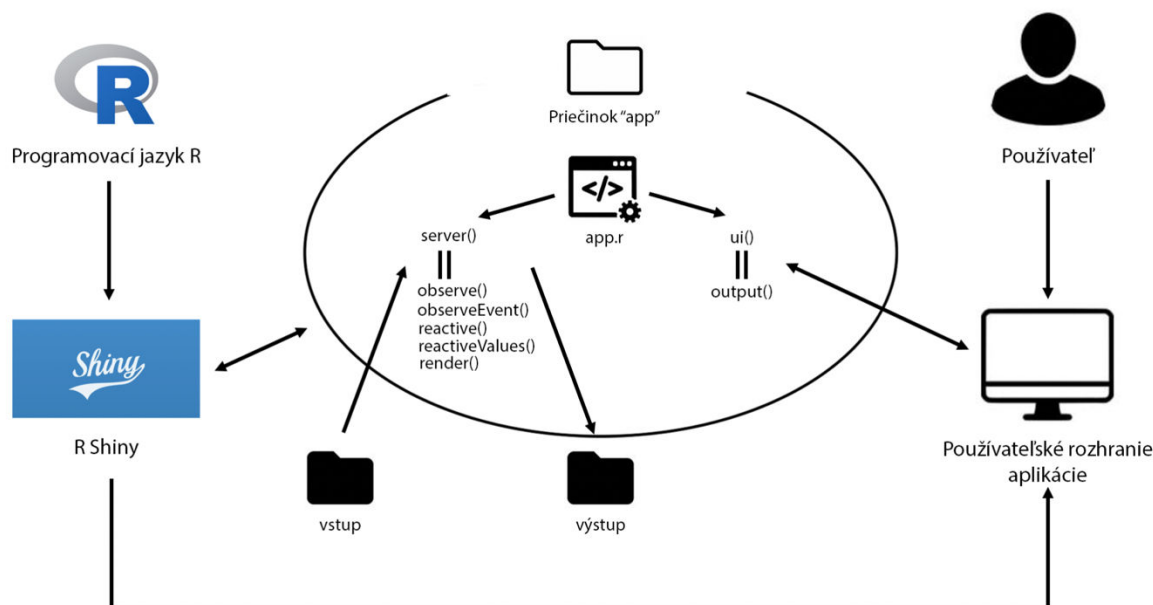
V nasledujúcej podkapitole sa zameriame na opis a vizualizáciu architektúry navrhovaného systémového riešenia, a teda aplikácie pre využitie popisných modelov aplikovaných na vstupné dáta.

Aplikácia bude pozostávať z dvoch hlavných častí, a to *ui().r* a *server().r*, ktorých implementácia sa nachádza v súbore, resp. v skripte *app.r*. V časti *server().r* sa nachádza celá programová logika aplikácie, pričom jej súčasťou sú *observe()*, *observeEvent()*, *reactive()*, *reactiveValues()*, *render()* funkcie, ktoré sú pre aplikáciu absolútne kľúčové. Podstata druhej časti, a teda časti *ui().r* spočíva v implementácii používateľského rozhrania aplikácie. Pre ňu sú typické funkcie *output()*, ktorými sú špecifikované rôzne výstupy definované podľa potreby a uváženia vývojového tímu, no rovnako tak podľa zadania a spätnej väzby od klienta. Taktiež je tu možné nájsť mnoho vstupných elementov, vďaka ktorým je používateľ schopný vybrať vstupnú hodnotu (pozn. - forma, ako je vstupná hodnota volená sa prirodzene líši od typu elementu), ktorá je následne využitá v procesoch, v časti *server().r*. Jedná sa napríklad o výber parametrov jednotlivých modelov, či filtračných kritérií, na základe ktorých prebieha selekcia záznamov. Medzi typické vstupné elementy, ktoré je možné využiť aj pri implementácii nášho softvérového riešenia, môžeme zaradiť tlačidlá, zaškrŕavacie polia, numerické posúvače, alebo aj rozbaľovacie menu s možnosťou výberu jednej či viacerých hodnôt. Časť *ui().r* je však rovnako dôležitá práve z toho hľadiska, že tvorí celú konštrukciu používateľského rozhrania aplikácie. Tá je implementovaná prostredníctvom kľúčových výrazov ako *navbarPage()*, *tabPanel()*, *fluidRow()*, *sidebarPanel()*, *mainPanel()* a niekoľko ďalších. Tie definujú jednotlivé časti používateľského rozhrania, akými sú napríklad bočný či hlavný panel, ktorých súčasťou sú práve už spomínané výstupy *output()* alebo vyššie vymenované vstupné elementy.

Pre potreby našej diplomovej práce sme nevyužili službu *shinyapps.io*, ktorá poskytuje aplikačný *R Shiny* server a ktorá je štandardne využívaná práve pre *R Shiny* aplikácie. Naopak, aplikácia je spúšťaná lokálne, a to na osobnom počítači priamo z programovacieho prostredia R Studio. Dôvod je ten, že výpočtová rýchlosť a sila serveru online služby *shinyapps.io*, ktorá je štandardne pre *R Shiny* aplikácie využívaná, nebola dostatočná. Bližšie technické špecifikácie týkajúce sa navrhovaného a implementovaného systémového riešenia je možné nájsť v systémovej príručke.

Náš softvér je charakteristický taktiež tým, že dokáže pracovať s akoukoľvek nahranou dátovou množinou, ktorá však musí byť vo formáte *.csv*, *.xls* alebo *.xlsx* súboru. Používateľ má však na výber aj možnosť pracovať s množinou údajov, ktoré je už vopred v aplikácii nahratá.

Na nasledujúcom obrázku (Obrázok 4) sme vizualizovali architektúru navrhovaného systémového riešenia.



Obrázok 4 Architektúra navrhovaného systémového riešenia

Pod pojmom „*vstup*“ máme na mysli hodnoty jednotlivých vstupných elementov, ktoré sme taktiež spomenuli vyššie a ktoré sú kľúčové pre správny chod implementovaných funkcionalít. Práve v časti *server()* prebieha proces, ktorý je priamo odvodený od činnosti a výsledku, ktorý chce používateľ získať. Tu je potrebné si predstaviť základné procesy akými sú filtrovanie údajov, resp. prispôbovanie obsahu vstupnej dátovej množiny používateľom, ale taktiež aj tie zložitejšie procesy, ako napríklad aplikovanie modelov na dáta. „*Výstup*“, ako názov napovedá, reprezentuje výstupy v rôznych formách. Tie môžu byť vo forme vizualizácií, a teda grafov, tabuliek či v textovej forme. Výstup sa následne zobrazí v používateľskom rozhraní aplikácie.

3.6. Základný opis fungovania softvéru

V aktuálnej podkapitole sa zameriame na základný opis fungovania softvéru. Vzhľadom na to, že naša aplikácia pozostáva z troch hlavných častí, a to:

- časť pre hľadanie asociačných pravidiel,
- časť pre aplikovanie metód zhukovania,
- časť pre hľadanie asociačných pravidiel v rámci definovaných zhukov,

sme sa rozhodli vypracovať stručný opis pre každú z nich zvlášť. Týmto častiam však ešte predchádza časť pre načítanie dát, a práve tou je vhodné začať.

Podrobnej charakteristike funkcionalít, vstupných elementov, výstupov ale taktiež vizuálnej štruktúre aplikácie sa venujeme v štvrtej kapitole našej diplomovej práce, ale rovnako tak aj v používateľskej príručke, ktorá je súčasťou príloh.

Obsahom tejto kapitoly bude skutočne len stručný náčrt fungovania jednotlivých častí nášho softvéru. Podrobnú charakteristiku jednotlivých elementov a vizualizácií je možné nájsť v systémovej a používateľskej príručke, no taktiež v podkapitole 3.5 a v kapitole číslo 4.

3.6.1. Načítanie dát

Prvý krok, ktorý používateľ musí v aplikácii vykonať, je načítanie dát. Má na výber z dvoch možností, a to využiť dáta, ktoré už v aplikácii nahraté, alebo nahrať svoje vlastné. V prípade, ak využije prvú možnosť, smie využívať ďalšie funkcionality aplikácie, ktoré stručne charakterizujeme v nasledujúcich podkapitolách. Ak sa používateľ rozhodne načítať svoje vlastné dáta, musí vykonať ďalší dodatočný úkon. Jedná sa o definovanie toho, akým spôsobom budú jednotlivé numerické atribúty, ktoré sú súčasťou načítavanej dátovej množiny, pretransformované na kategorický atribút. Výber prebieha pre každý numerický atribút zvlášť. V prípade, ak načítavaná dátová množina neobsahuje žiadne numerické atribúty, proces úpravy sa logicky nevykonáva. Ďalší krok spočíva už v samotnom načítaní dátovej množiny do aplikácie.

3.6.2. Hľadanie asociačných pravidiel

Po úspešnom načítaní dát má používateľ na výber niektorú z troch vyššie spomenutých sekcií. V prípade tej prvej, a teda časti určenej pre hľadanie asociačných pravidiel má používateľ k dispozícii niekoľko vstupných elementov, ktoré vie využiť. Tie sú umiestnené v bočnom ale taktiež v hlavnom paneli používateľského rozhrania, pričom sú určené pre prispôbenie obsahu vstupnej dátovej množiny. Používateľ je taktiež schopný definovať hraničné hodnoty vstupných parametrov modelu pre hľadanie asociačných pravidiel, akými sú minimálna podpora, spoľahlivosť či dĺžka ľavej strany pravidiel. Taktiež vie obmedziť výskyt zvolených atribútov na pravej strane asociačných pravidiel. K dispozícii má niekoľko výstupov, ktoré sú umiestnené v hlavnej časti používateľského rozhrania. Medzi výstupy je možno zaradiť nájdené asociačné pravidlá prezentované vo forme tabuľky, či interaktívne vizualizácie, vďaka ktorým je možné viac a lepšie pochopiť obsah získaných výsledkov.

3.6.3. Aplikovanie metód zhlukovania

Druhá časť je určená pre aplikovanie metód zhlukovania na vstupné dáta. Používateľ má na výber medzi tromi možnosťami, a to medzi hierarchickým zhlukovaním a zhlukovaním pomocou algoritmov *PAM* a *k-Means*. Rovnako ako v predchádzajúcom prípade, používateľ si môže prispôbiť obsah vstupnej dátovej množiny či vykonať analýzy pre odhad optimálneho počtu zhlukov. V prípade hierarchického zhlukovania má taktiež možnosť zvoliť si konkrétnu metódu tohto prístupu a okrem toho aj vykresliť špecifickú vizualizáciu, ktorou je dendrogram. Výstupov má k dispozícii opäť niekoľko. Konkrétne sa jedná o výsledky jednotlivých validačných kritérií,

vizualizáciu rozloženia bodov v zhlukoch, výslednú dátovú množinu s informáciou o tom, ktorý záznam zo vstupnej dátovej množiny do akého zhuku patrí, či informáciu o tom, ktoré záznamy sú si v jednotlivých zhukoch najviac, resp. najmenej podobné.

3.6.4. Hľadanie asociačných pravidiel v rámci definovaných zhukov

Posledná, tretia, časť slúži pre hľadanie asociačných pravidiel v rámci definovaných zhukov. Ak je používateľ spokojný s výsledkami zhukovania, má možnosť pre každý jeden zhuk nájsť asociačné pravidlá. Vstupy a výstupy sú identické s tými, ktoré sme definovali v podkapitole 3.7.2. (53), avšak s tým rozdielom, že v tejto časti aplikácie používateľ dokáže zvoliť len tie zhuky, pre ktoré chce asociačné pravidlá nájsť.

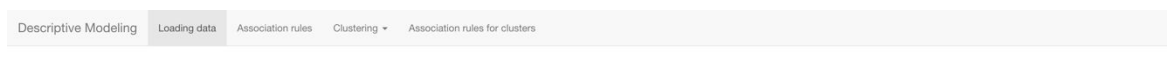
4. Charakteristika implementovaného systému

Implementácia navrhovaného systémového riešenia prebiehala v súlade so zvoleným a vyššie charakterizovaným modelom životného cyklu, a taktiež v súlade s definovanými používateľskými požiadavkami. Detailný technický popis, v ktorom sme charakterizovali jednotlivé kroky a časti, resp. fragmenty zdrojového kódu je možné nájsť v prílohe D, a teda v systémovej príručke našej diplomovej práce. Samotná implementácia nami navrhovaného systémového riešenia zahŕňa vývoj aplikácie pre využitie popisných modelov aplikovaných na vstupnú dátovú množinu.

V rámci implementácie nášho systému pre využitie popisných modelov v oblasti medicíny, presnejšie v oblasti kardiovaskulárnych ochorení s cieľom identifikovať jedinečné skupiny pacientov a vzťahy medzi faktormi vplyvujúcimi na zvýšené riziko kardiovaskulárnych ochorení sme využili *R* balík *R Shiny*, ktorý je vhodný pre tvorbu webových aplikácií. Používateľské rozhranie aplikácie je rozdelené na tri hlavné oddelenia. A to menu v hornej časti obrazovky, bočný panel na ľavej strane obrazovky a hlavná časť, ktorá pokrýva väčšinu zostávajúceho priestoru. Používateľské rozhranie aplikácie spoločne s vyznačenými oddeleniami je zobrazené na nasledujúcom obrázku (Obrázok 5).

Obrázok 5 Používateľské rozhranie aplikácie

Z funkčného hľadiska aplikácia ponúka tri hlavné prístupy popisných modelov v rámci troch samostatných záložiek hlavného menu, ktoré je možné vidieť aj na obrázku nižšie (Obrázok 6). Ide o hľadanie asociačných pravidiel, zhlukovanie a hľadanie asociačných pravidiel v rámci definovaných zhlukov.

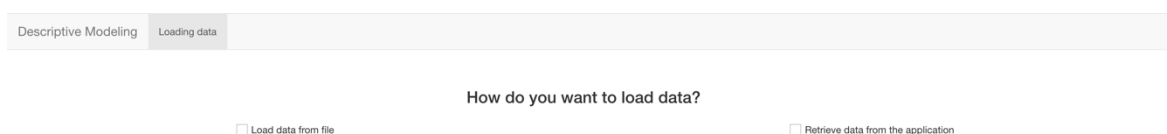


Obrázok 6 Záložky hlavného menu

Odlíšnou časťou aplikácie je časť pre načítavanie dát. Jej používateľské rozhranie neodpovedá tomu, ktoré sme spomenuli a znázornili vyššie. Tri hlavné funkčné časti spoločne s tou pre načítavanie dát do aplikácie budú podrobne vysvetlené v nasledujúcich štyroch podkapitolách.

4.1. Načítanie dát

Funkcionalita pre načítavanie dát je súčasťou panelu pomenovanom „*Loading Data*“, ktorý je dostupný ihneď po spustení našej aplikácie a ktorý je taktiež znázornený na nasledujúcom obrázku (Obrázok 7).



Obrázok 7 Panel určený pre načítavanie dát

V úvode, po spustení aplikácie používateľ vidí dve zaškrŕavacie polia, ktoré sú pomenované nasledovne:

- „*Load data from file*“,
- „*Retrieve data from the application*“.

Tieto dve možnosti určujú spôsoby, akými je možné načítať dáta do aplikácie, pričom sú znázornené na jednom z predchádzajúcich obrázkov (Obrázok 7).

V prípade, že sa používateľ rozhodne načítať svoje vlastné dáta, je potrebné aby označil prvú z ponúkaných možností, a teda tú, ktorá je pomenovaná ako „*Load data from file*“. Po jej označení sa zobrazí ďalšia časť používateľského rozhrania, ktorú je možné vidieť aj na nasledujúcom obrázku (Obrázok 8) a ktorá pozostáva zo štyroch elementov.

Obrázok 8 Načítanie vlastných dát

Prvý z nich slúži pre nahratie súboru, resp. dát, ktoré však musia byť vo formáte .csv, .xls alebo .xlsx. Proces prebieha štandardne, po kliknutí na tlačidlo „Browse“ sa v používateľskom prostredí zobrazí modálne okno v ktorom používateľ zo svojho adresára vyberie požadovaný súbor. Ďalšie dva elementy sú využiteľné len v prípade, ak je načítavaný súbor vo formáte .csv. Používateľ je tak schopný definovať, aký separátor bol v danom súbore použitý a taktiež to, či obsahuje hlavičku alebo nie. Po splnení týchto krokov a po kliknutí na tlačidlo „Load data“ sa dáta načítajú, avšak nie do aplikácie, ale len do medzipamäte. Následne sa načítané dáta vizualizujú prostredníctvom tabuľky a taktiež sa vygeneruje ďalšia časť používateľského rozhrania, ktorá je určená pre posledný a zároveň veľmi dôležitý proces. Ním je transformácia numerických atribútov na kategorické. Používateľ má k dispozícii dve varianty úprav, medzi ktoré patrí priame pretypovanie numerických atribútov na kategorické, čo je využiteľné v prípade, ak dané čísla reprezentujú inú hodnotu, napríklad kategóriu. Druhou možnosťou je diskretizácia, a to na intervaly rovnakej šírky či hĺbky. Je nutné definovať úpravu pre každý numerický atribút zvlášť. Výber úpravy prebieha tak, že po kliknutí na tlačidlo „Show me options“, ktoré sa vygenerovalo spoločne s tabuľkou s načítanými dátami, sa zobrazí posledná časť používateľského rozhrania, ktorá pozostáva z niekoľkých zoznamov. Ich množstvo je dané počtom numerických atribútov, ktoré sa v načítavanej dátovej množine nachádzajú. Presnejšie povedané, pre každý numerický atribút sa vygeneruje vlastný zoznam, resp. rozbaľovacie menu s možnosťou výberu jednej hodnoty. Tieto hodnoty predstavujú práve možnosti transformácie, ktoré sme spomenuli vyššie. V prípade, že sa používateľ rozhodne pre diskretizáciu, a teda vyberie možnosť, ktorá jej odpovedá, pod zoznamom numerického atribútu, ktorého sa daná úprava týka, sa vygeneruje ďalší zoznam. Ten obsahuje numerické hodnoty a používateľ prostredníctvom neho vyberie množstvo intervalov, do ktorých budú jednotlivé numerické hodnoty zadelené.

Tabuľku s načítanými dátami, rovnako ako aj tlačidlo s názvom „Show me options!“ a so spomínanými zoznamami je možné vidieť na nasledujúcom obrázku (Obrázok 9). Vzhľadom na to, že

do aplikácie môžu vstupovať akékoľvek dáta, nie len tie z lekárskeho prostredia, tak pre ilustráciu sme použili známu dátovú množinu *Iris*. Tá obsahuje štyri merateľné znaky pre každú z troch rôznych rastlín.

The loaded data looks like this!

Please check the loaded data carefully. If the loaded data suits you, you can continue.

Show entries

	Sepal_Length	Sepal_Width	Petal_Length	Petal_Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa

Showing 1 to 5 of 150 entries

Previous 2 3 4 5 ... 30 Next

How do you want transform your numeric data?

Click the button below to view the options!

Show me options!

Sepal_Length

Equal frequency binning

Choose number of intervals:

3

Sepal_Width

Index

Petal_Length

Index

Petal_Width

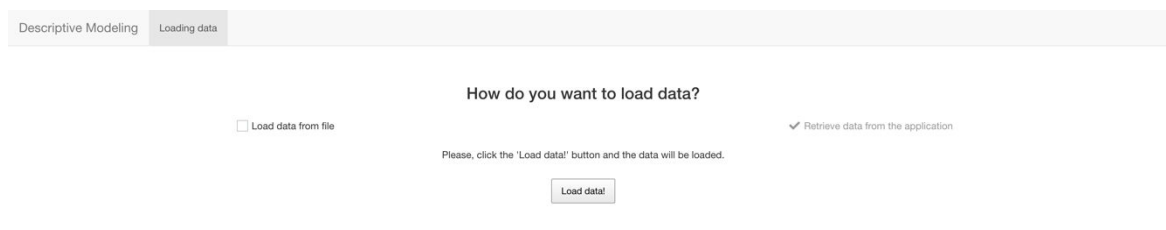
Index

Confirm selection

Obrázok 9 Transformácia numerických atribútov na kategorické

Po vybratí spôsobu úpravy pre každý numerický atribút a po kliknutí na tlačidlo „*Confirm selection*“ sa spracované dáta načítajú do hlavnej časti aplikácie.

Pri výbere druhej možnosti, teda možnosti „*Retrieve data from the application*“, bude v rámci aplikácie použitá dátová množina, ktorá je v nej už načítaná. Jedná sa o množinu údajov, ktorá obsahuje informácie o skutočných pacientoch VÚSCH. Táto dátová množina bude využitá v celej aplikácii, pri všetkých funkcionalitách softvéru. Špecifikáciou dátovej množiny sa nebude zaoberať, nakoľko náplňou predkladanej práce nebola práve analýza nad danou množinou, ale tvorba softvéru poskytujúca popisné modely pre prácu nad dátovými súbormi. Dátová množina bola dostatočne priblížená v diplomovej práci s názvom „*Metódy dátovej analytiky pre analýzu vplyvu faktorov na včasné zachytenie kardiovaskulárneho rizika*“ [46]. Po označení zaškrtnávacieho poľa sa zobrazí ďalšia časť používateľského rozhrania, ktorého súčasťou je správa v textovej forme a tlačidlo nesúce názov „*Load data!*“. Táto časť používateľského rozhrania je znázornená aj na nasledujúcom obrázku (Obrázok 10).

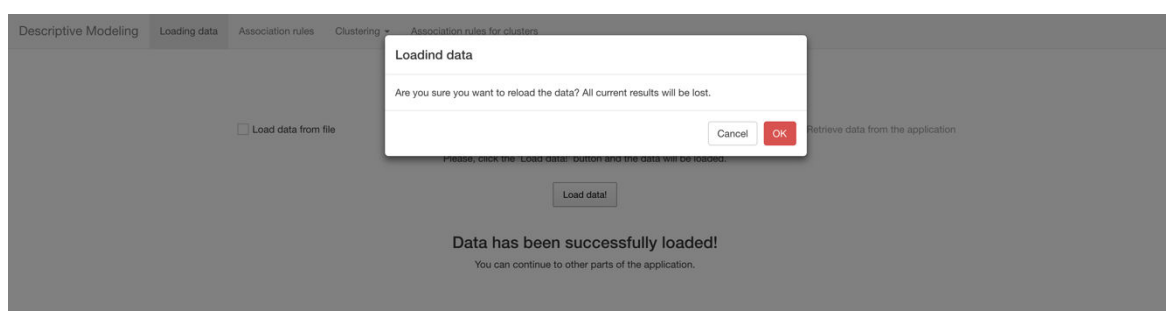


Obrázok 10 Načítanie dát z aplikácie

Po kliknutí na dané tlačidlo sa vykoná proces načítania dát a používateľovi sa sprístupnia ďalšie časti aplikácie.

Tu je dôležité podotknúť, že pokiaľ používateľ nenačíta dáta, či už prvým alebo druhým spôsobom, nebudú mu sprístupnené ďalšie časti softvéru. Sprístupnenie sa prejaví tak, že v lište, v hornej časti obrazovky, napravo od nápisu „*Loading data*“ sa zobrazia ďalšie názvy, ktoré reprezentujú hlavnú časť aplikácie.

V prípade, že sa už používateľ dostal do hlavnej časti aplikácie, no rozhodne sa vrátiť späť k časti pre načítavanie dát, zobrazí sa modálne okno. To obsahuje upozornenie, že ak sa rozhodne vrátiť, a teda potvrdí svoje rozhodnutie kliknutím na červené tlačidlo s nápisom „OK“, aplikácia sa reštartuje a tým pádom všetka vykonaná aktivita v hlavnej časti aplikácie bude stratená. V prípade, ak si svoje rozhodnutie rozmyslí a klikne na tlačidlo „Cancel“, prechod na časť aplikácie pre načítavanie dát nebude vykonaný a používateľ zostane v tej časti aplikácie, v ktorej sa aj nachádzal. Spomínané modálne okno je znázornené na obrázku nižšie (Obrázok 11).



Obrázok 11 Modálne okno s upozornením

4.2. Hľadanie asociačných pravidiel

Prvou z hlavnej časti aplikácie je sekcia pre hľadanie asociačných pravidiel. Pre tento účel sme využili algoritmus *Apriori*, ktorý sme bližšie charakterizovali v analytickej časti. Jednotlivé vstupy a výstupy sme popísali v nasledujúcich podkapitolách. Taktiež je dôležité zdôrazniť, že viac vizuálnych ukážok s ešte podrobnejším opisom jednotlivých krokov, ktoré je nutné vykonať v rámci jednotlivých častí aplikácie, je možné nájsť v používateľskej príručke.

4.2.1. Vstupy

Zmysel prvého vstupu, s ktorým používateľ príde do kontaktu, spočíva vo výbere atribútov z načítanej dátovej množiny, s ktorými bude v rámci tejto časti aplikácie ďalej pracovať. V skratke by sa dalo povedať, že sa jedná o voľbu zloženia dátovej množiny, ktorá poslúži pre ďalšie činnosti, resp. analýzy v danej časti aplikácie. Používateľ má na výber dve hlavné alternatívy. Prvá z nich využíva aktuálne zloženie dátovej množiny, a teda pracuje so všetkými atribútmi, ktoré sa v nej nachádzajú. V rámci druhej možnosti si používateľ môže podľa svojich požiadaviek zúžiť výber atribútov, ktoré budú algoritmom ďalej spracované. Výber jednej z týchto možností používateľ vykoná označením zaškrŕavacieho políčka, ktoré odpovedá jeho požiadavke. Je dôležité poznamenať, že možnosť práce s celou dátovou množinou je predvolená.

V prípade, že si vyberie druhú alternatívu, a teda voľbu zloženia vstupnej dátovej množiny, systém automaticky sprístupní ďalšiu časť v rámci tohto vstupu, ktorá už slúži priamo na výber, alebo redukciu počtu atribútov. Z technického hľadiska, resp. z hľadiska implementačnej logiky *R Shiny* sa jedná o tzv. *varSelectInput*. Ten možno charakterizovať ako interaktívny prvok pre výber premenných zo vstupného zoznamu, ktorý je možné implementovať v rámci používateľského rozhrania. V našom prípade bol tento prvok naplnený názvami atribútov vstupnej dátovej množiny. Používateľ je schopný si tak jednotlivé atribúty odstrániť alebo pridať. Taktiež sme implementovali funkcionality zabezpečujúcu dolné ohraničenie počtu zostávajúcich atribútov, a to na minimálny počet dva. Je to z toho dôvodu, aby používateľ nebol schopný odstrániť všetky atribúty.

Celá táto časť, zahŕňajúca zaškrŕavacie polia a element s výberom premenných zo zoznamu, je zaobalená do jednej sekcie, ktorú je možné zobrazíť a skryť na základe stlačenia tlačidla používateľom. Daná funkcionality je implementovaná z dôvodu šetrenia miesta a sprehľadnenia celého bočného panelu. Súčasťou časti tohto vstupu je aj informácia o tom, koľko atribútov sa nachádza vo vstupnej dátovej množine. Nakoľko je táto informácia dynamická, mení sa zakaždým akonáhle používateľ odstráni alebo pridá atribút v rámci *varSelectInput-u*, ktorý sme charakterizovali vyššie. V prípade, ak je používateľ spokojný so svojím výberom, potvrdí ho kliknutím na tlačidlo „*Show data*“, čím sa plynule dostáva k druhému vstupu.

Druhý vstup slúži na filtrovanie záznamov podľa ohraničení na hodnoty atribútov. V hlavnej časti, v dátovej tabuľke sa nachádzajú všetky záznamy a atribúty, ktoré možno nájsť vo vstupnej dátovej množine a ktoré odpovedajú zvolenej možnosti z predchádzajúceho vstupu. Dátová tabuľka je doplnená o funkcionality filtrovania záznamov. Samotný filter je umiestnený v hornej časti tabuľky, a to pod názvami jednotlivých atribútov. Selekciovú hodnotu pri spomínaných atribútoch,

resp. zúžením intervalu, v prípade, ak sa jedná o numerickú premennú, si používateľ vie z dátovej množiny vybrať len tie záznamy, ktoré sú pre neho podstatné.

2. Filtering data

You can use the filter below to select records. If you want to work with the whole data set, leave the filter empty. Confirm your decision with the 'Confirm' button.

Show entries

	Pohlavie	Vek	ESC	R_ICHS	R_CMP	R_IM	R_Hypertenzi
	All	All	All	All	All	All	All
1	Muz	[73,88]	[5,7]	true	false	true	false
2	Zena	[73,88]	[7,33]	true	false	true	false
3	Muz	[73,88]	[7,33]	false	false	false	false
4	Zena	[73,88]	[7,33]	false	false	false	false
5	Muz	[36,58]	[5,7]	false	false	false	false
6	Muz	[36,58]	[0,5]	false	false	false	false
7	Zena	[58,63]	[0,5]	false	false	false	false
8	Zena	[63,68]	[5,7]	false	false	false	false
9	Muz	[73,88]	[7,33]	true	false	false	false
10	Muz	[36,58]	[0,5]	false	false	false	false
11	Zena	[63,68]	[5,7]	false	true	false	false
12	Zena	[63,68]	[0,5]	false	false	false	false
13	Muz	[36,58]	[0,5]	true	false	true	false
14	Zena	[73,88]	[7,33]	true	false	true	false
15	Muz	[58,63]	[0,5]	false	false	false	false

Showing 1 to 15 of 764 entries

Previous 2 3 4 5 ... 51 Next

Obrázok 12 Dátová tabuľka s možnosťou filtrovania

Tak ako je možné vidieť na obrázku vyššie (Obrázok 12), súčasťou tohto vstupu je aj možnosť vymazať všetky používateľom zvolené filtrovacie kritériá. V situácií, ak by chcel používateľ filtrovať záznamy odznovu, nemusí tak aktuálne filtrovacie kritériá mazať po jednom, ale je schopný ich odstrániť všetky naraz.

Keď je používateľ spokojný so selekciou záznamov, potvrdí svoj výber kliknutím na príslušné tlačidlo „Confirm!“. Používateľ následne dostane niekoľko základných informácií o dátovej množine. Konkrétne sa jedná o informácie:

- ktoré atribúty, ak vôbec, boli odstránené z dátovej množiny z dôvodu, že obsahovali len jednu unikátnu hodnotu - takýto typ atribútu je pri generovaní asociačných pravidiel nepotrebný,
- aké kritériá boli využité pri filtrovaní záznamov,
- aký je aktuálny rozmer dátovej množiny, resp. koľko riadkov - záznamov a stĺpcov - atribútov obsahuje.

Tretí vstup, ktorý používateľ kontroluje, je voľba troch kľúčových parametrov využívaných pri generovaní asociačných pravidiel. Jedná sa o minimálnu podporu, minimálnu spoľahlivosť a minimálnu dĺžku ľavej strany asociačných pravidiel. Prvé dva parametre (podpora a spoľahlivosť) sme charakterizovali v podkapitole 1.2, v analytickej časti. Tretím parametrom môže používateľ stanoviť minimálnu dĺžku ľavej strany asociačných pravidiel, t.j. minimálny počet podmienok v konjunkcii. Rozsah intervalov pri jednotlivých parametroch, ako aj ich prednastavenú hodnotu, teda tú, ktorú sme nastavili na základe nášho uváženia a ktorá sa nachádza na jednotlivých numerických posúvačoch po spustení aplikácie, uvádzame v nasledujúcom zozname:

- minimálna podpora (rozsah intervalu: 0.05 až 0.5, prednastavená hodnota: 0.07),
- minimálna spoľahlivosť (rozsah intervalu: 0.5 až 1, prednastavená hodnota: 0.99),
- minimálna dĺžka ľavej strany asociačných pravidiel (rozsah intervalu: 3 až 10, prednastavená hodnota: 5).

Z hľadiska *R Shiny* elementov sú tieto metriky reprezentované prostredníctvom tzv. *sliderInput*-ov. Tie možno charakterizovať ako spôsob výberu numerickej hodnoty, ale taktiež ako spôsob voľby intervalu numerických hodnôt. V našom prípade je interval možné definovať posunom bodu po číselnej osi, pričom tento bod určuje začiatok intervalu a posledná hodnota na číselnej osi naopak určuje koniec intervalu. Z uvedeného vyplýva, že pravé ohraničenie intervalu je pevne dané, používateľ pracuje len s ľavým ohraničením.

Štvrtý vstup v tejto časti aplikácie, s ktorým používateľ prichádza do kontaktu, spočíva v obmedzení pravej strany asociačných pravidiel. Ak si je používateľ istý tým, že nechce, aby sa na pravej strane asociačných pravidiel, teda v dôsledkovej časti, vyskytli hodnoty nejakého atribútu, resp. atribútov, môže ich výberom obmedziť. Tak ako pri prvom vstupe, aj tu bol využitý element *varSelectInput*, a teda interaktívny prvok pre výber premenných zo vstupného zoznamu. Napĺňanie týchto vstupov, rovnako ako aj v prvom prípade, je plne dynamické, a teda vstupné hodnoty sa odvíjajú od načítaného súboru.

Posledný, piaty vstup je len akési potvrdenie tých predchádzajúcich, ktoré je vykonateľné kliknutím na príslušné tlačidlo „Render!“ nachádzajúce sa v dolnej časti bočného panelu. To slúži ako pokyn pre daný systém, pre spustenie algoritmu hľadania asociačných pravidiel.

4.2.2. Výstupy

Prvý, a zároveň hlavný výstup z tejto časti aplikácie sú samotné asociačné pravidlá. Tie sú vizualizované prostredníctvom tabuľky (Obrázok 13), pričom s niekoľkými ďalšími informáciami sú rozdelené do samostatných stĺpcov. Ide o ľavú stranu pravidiel, pravú stranu pravidiel (dôsledkovú

časť), podporu, spoľahlivosť, pokrytie, lift, počet a komentár. Posledný stĺpec, a teda stĺpec pre komentár, je špecificky určený pre samotného používateľa, ktorý si v rámci neho k príslušnému záznamu vie napísať poznámku, alebo informáciu o danom asociačnom pravidle. Počet nájdených asociačných pravidiel môže byť pri určitých nastaveniach vysoký, a preto je veľmi užitočná funkcia filtrovania nájdených asociačných pravidiel podľa používateľom zadaných podmienok. Pri prvom stĺpci - ľavej strane asociačných pravidiel je nutné, aby používateľ zadal, resp. napísal hľadaný výraz. Pri všetkých ostatných, hodnoty atribútov vyberá buď z rozbaľovacieho menu, alebo definuje rozsah intervalu pri numerických atribútoch. K funkcionalitám je možné taktiež zaradiť stiahnutie výslednej tabuľky, a to vo formátoch .xlsx a .csv, ale taktiež stiahnutie len vybraných záznamov, rovnako, do už spomínaných formátov.

Informations about association rules!

1. The number of association rules is: 25.
2. The following filter inputs were used to select the data used to generate the association rules: Pohlavie = Zena.
3. The unique values that appear on the LHS of association rules are: AST=[0.08-0.355]; AST=[0.355-0.42]; BMI=[18.2-29.1]; CL=[91.8-103]; CRP=[2.22-3.9]; CRP=[3.9-130]; ECHO_EF=51<=; ECHO_PH=0; EKG_FR=[72-130]; EKG_QRS=<=89; EKG_QRS=90-119; EKG_QT=[400-518]; EKG_Rytmus=0; EKG_T; ESC=[5-7]; Fajcenie=3; FBG=2-4; Cholesterol=4.2-6; Kreat=[6.6-73]; LDL=<=2.6; Na=[140-170]; Nalez=0; O_Hypertenzia; O_IJHS; Tlak=140-159; Tlak=160<=; Trigl=[1.21-1.72]; Trigl=[1.72-30]; Urea=[2.29-5.05].

Download table

Download selected rows

Deselect

Show 20 entries

LHS	RHS	Support	Confidence
All	All	All	All
13186 ESC=[5-7]; Trigl=[1.72-30]; FBG=2-4; EKG_Rytmus=0; ECHO_EF=51<=	ECHO_PH=0	0.083	
13965 O_Hypertenzia; Fajcenie=3; BMI=[18.2-29.1]; Kreat=[6.6-73]; ECHO_PH=0	ECHO_EF=51<=	0.071	
14735 Fajcenie=3; CRP=[2.22-3.9]; CL=[91.8-103]; EKG_Rytmus=0; ECHO_PH=0	ECHO_EF=51<=	0.074	
16126 O_Hypertenzia; Fajcenie=3; Tlak=160<=; EKG_QRS=<=89; Nalez=0	ECHO_EF=51<=	0.071	
16788 O_Hypertenzia; Fajcenie=3; Urea=[2.29-5.05]; Na=[140-170]; FBG=2-4	ECHO_EF=51<=	0.071	
17020 O_Hypertenzia; Fajcenie=3; Urea=[2.29-5.05]; ECHO_PH=0; Nalez=0	ECHO_EF=51<=	0.077	
17191 AST=[0.355-0.42]; Na=[140-170]; LDL=<=2.6; EKG_Rytmus=0; ECHO_PH=0	ECHO_EF=51<=	0.071	
17192 AST=[0.355-0.42]; Na=[140-170]; LDL=<=2.6; FBG=2-4; ECHO_PH=0	ECHO_EF=51<=	0.071	
18092 ESC=[5-7]; Tlak=140-159; Na=[140-170]; FBG=2-4; ECHO_EF=51<=	ECHO_PH=0	0.071	
18272 CL=[91.8-103]; EKG_FR=[72-130]; EKG_Rytmus=0; EKG_T; ECHO_EF=51<=	ECHO_PH=0	0.077	
18325 Kreat=[6.6-73]; Chol=4.2-6; FBG=2-4; EKG_FR=[72-130]; EKG_Rytmus=0	ECHO_PH=0	0.074	
18569 LDL=<=2.6; CL=[91.8-103]; FBG=2-4; EKG_T; ECHO_PH=0	ECHO_EF=51<=	0.093	
18685 O_Hypertenzia; Fajcenie=3; Na=[140-170]; CRP=[3.9-130]; EKG_Rytmus=0	ECHO_PH=0	0.071	

Obrázok 13 Ukážka vygenerovaných asociačných pravidiel s dodatočnými informáciami

Súčasťou tohto výstupu je aj niekoľko základných informácií o vygenerovaných asociačných pravidlách, menovite informácia o ich celkovom počte, aké filtračné kritériá boli využité pri selekcii nájdených asociačných pravidiel a aké unikátne hodnoty sa vyskytli na ľavej strane pravidiel, čo je viditeľné aj na obrázku vyššie (Obrázok 13). Práve tieto hodnoty je možné využiť pri už spomínanom filtrovaní záznamov podľa ľavej strany asociačných pravidiel, pri ktorých je nutné, aby používateľ zadal hľadaný výraz.

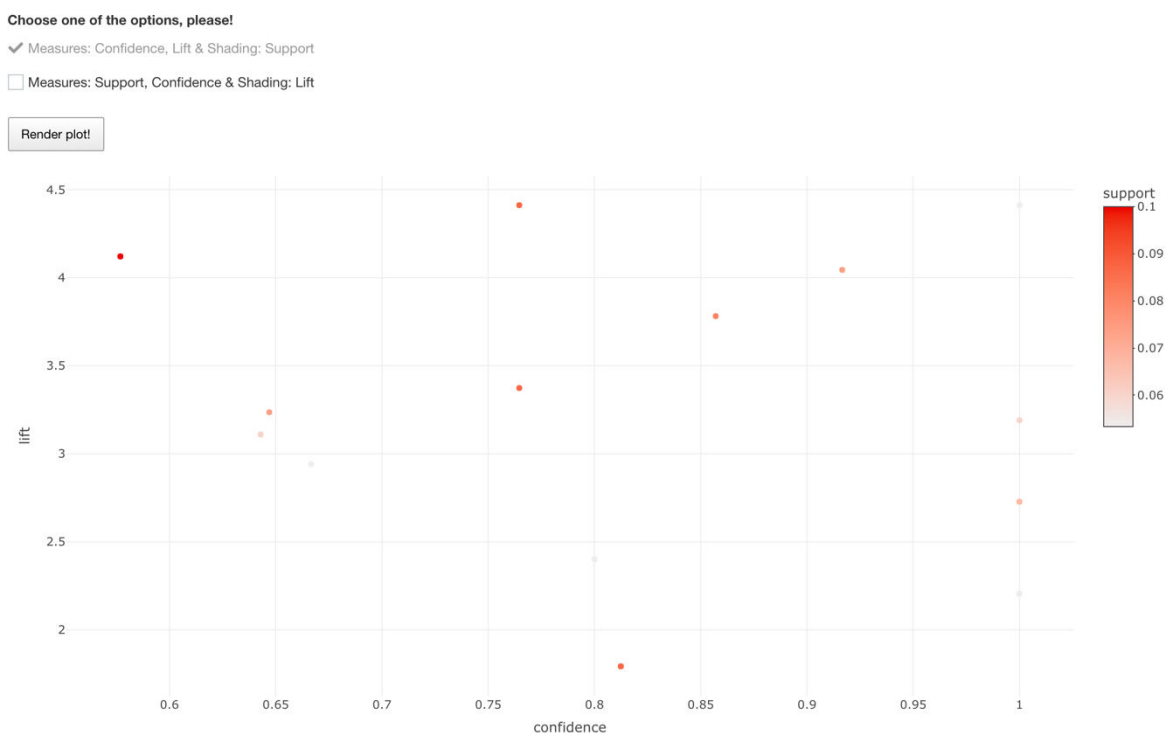
Druhým výstupom je vizualizácia pomocou bodového grafu. Ten sa v kontexte asociačných pravidiel využíva na znázornenie rozloženia jednotlivých asociačných pravidiel na základe metrík, akými sú podpora, spoľahlivosť a lift, ktoré sa v rôznych kombináciách môžu vyskytnúť na osiach x a y. Jedna z nich je vždy využitá ako miera, ktorá sa prejaví farebným odtieňom bodu zodpovedajúcemu danému asociačnému pravidlu. To znamená, že čím je hodnota zvolenej miery (podpora alebo lift) pri jednotlivých asociačných pravidlách vyššia, tým sýtejšie je zafarbenie bodu, ktoré odpovedá danému asociačnému pravidlu.

V našom prípade má používateľ na výber medzi dvoma možnosťami výberu, a to:

- pre osi x a y využiť metriky spoľahlivosť a lift, ako mieru použiť podporu,
- pre osi x a y využiť metriky podpora a spoľahlivosť, ako mieru použiť lift.

Používateľ vykoná výber označením začiarkavacieho políčka a následným kliknutím na príslušné tlačidlo. Podstatné taktiež je, že používateľ je schopný s danou vizualizáciou pracovať (približovať a oddiaľovať ju, posúvať ju, či vyberať len určitú časť grafu) a taktiež si ju stiahnuť vo formáte *.png*.

Pre jasné a praktické vysvetlenie tohto typu grafu uvažujme nasledujúcu vizualizáciu (Obrázok 14). Body vizualizované v grafe značia asociačné pravidlá. V tomto prípade, ich umiestnenie odpovedá hodnote liftu a spoľahlivosti (os x a y), ktoré dosiahli. Zafarbenie týchto bodov určuje hodnota miery spoľahlivosti.

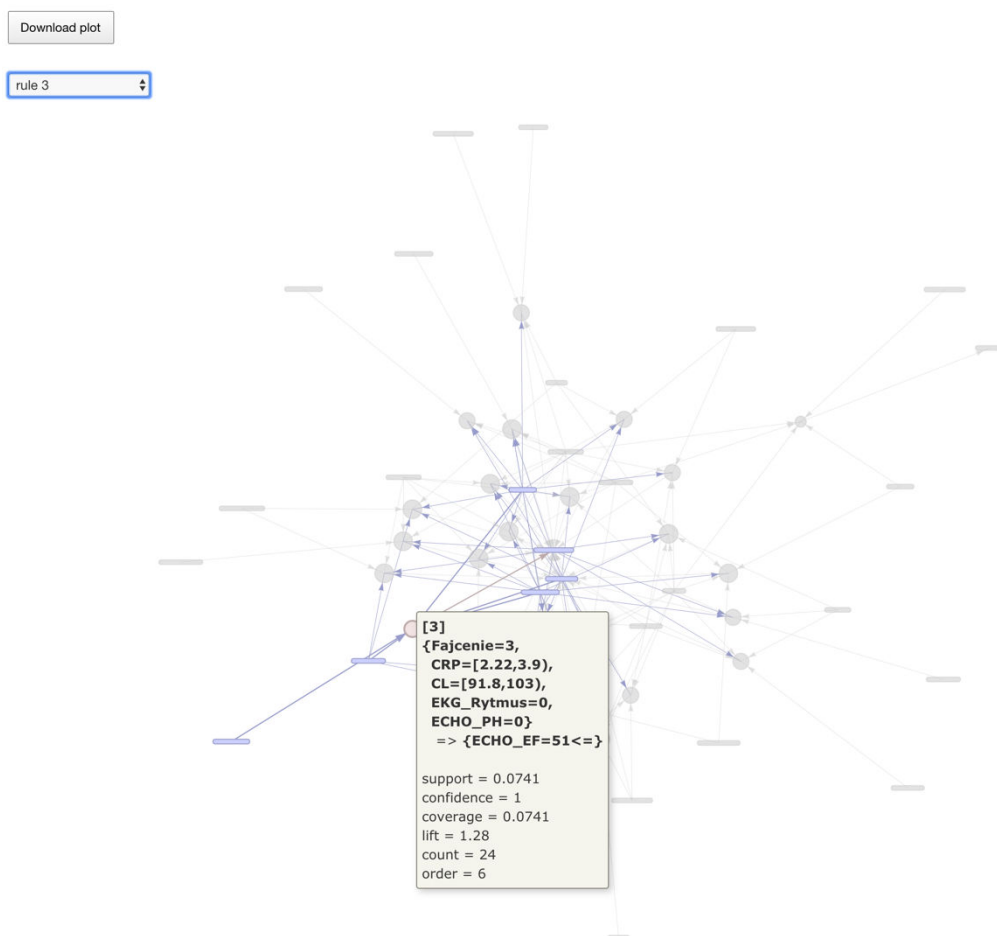


Obrázok 14 Bodový graf využitý pri asociačných pravidlách

Tretím a zároveň posledným výstupom tejto časti aplikácie je vizualizácia vzťahov medzi jednotlivými hodnotami atribútov, ktoré sa v asociačných pravidlách vyskytli a samotných asociačných pravidiel.

Vzhľadom na to, že tento typ vizualizácie má tendenciu byť neprehľadný, je schopný spracovať, a teda znázorniť maximálne 100 najlepšie hodnotených asociačných pravidiel. Hodnoty atribútov, ktoré sa v rámci asociačných pravidiel vyskytujú najčastejšie sú situované do stredu grafu. Vizualizácia, tak ako v predošlom prípade, je plne interaktívna, doplnená o možnosť zvýrazňovať špecifické časti daného grafu odpovedajúce jednotlivým asociačným pravidlám, alebo jednotlivým hodnotám atribútov (Obrázok 15). Pridali sme taktiež možnosť stiahnutia vizualizácie z prostredia aplikácie, a to vo formáte .html, čím sa zachovávajú jej interaktívne prvky.

V rámci praktickej ukážky sme si zvolili nasledujúci príklad, ktorý je zobrazený na obrázku nižšie (Obrázok 15). Konkrétne sa jedná o výsek grafu, na ktorom je možné vidieť zobrazené asociačné pravidlo číslo 3.



Obrázok 15 Vizualizácia vzťahov medzi hodnotami atribútov a asociačnými pravidlami

Zaoblené obdĺžniky predstavujú hodnoty atribútov, ktoré sa vyskytli v jednotlivých asociačných pravidlách. Krúžky, naopak, asociačné pravidlá. Čím je farba krúžkov sýtejšia, tým vyššiu hodnotu metriky lift dosiahli. V ľavej hornej časti vizualizácie sa nachádza rozbaľovacie menu, pomocou ktorého je používateľ schopný zvýrazňovať jej určité časti.

4.3. Aplikovanie metód zhlukovania

V rámci našej aplikácie sme implementovali tri metódy zhlukovania, avšak z hľadiska funkcionality a prevedenia, a teda vstupov a výstupov, s ktorými používateľ môže pracovať, sú totožné. Rozdiely je možné nájsť len pri používateľskom rozhraní, pri hierarchickom zhlukovaní a ktoré charakterizujeme v závere jednotlivých podkapitol. Okrem hierarchického zhlukovania sme taktiež využili algoritmy *PAM* a *k-Means*, ktoré sme priblížili v analytickej časti diplomovej práce. V nasledujúcej časti charakterizujeme jednotlivé kľúčové elementy časti pre aplikovanie metód zhlukovania.

4.3.1. Vstupy

Rovnako ako v časti pre hľadanie asociačných pravidiel, prvý vstup, s ktorým používateľ prichádza do kontaktu, je výber tých atribútov z načítanej dátovej množiny, s ktorými má záujem pracovať pri ďalších úkonoch. Tento vstupný element sme podrobne charakterizovali v predchádzajúcej podkapitole.

Význam druhého vstupu spočíva vo zvolení metódy pre odhad optimálneho počtu zhlukov, do ktorých sa skúmané dáta majú zaradiť. Používateľ má na výber z troch rôznych možností:

- metóda lakťa (angl. „*Elbow method*“),
- metóda siluety (angl. „*Silhouette method*“).
- analýza stability zhlukov prostredníctvom metódy siluety (angl. „*Silhouette analysis*“).

Z hľadiska R Shiny elementov tu bol využitý tzv. *selectInput*, ktorým možno charakterizovať rozbaľovacie menu, resp. zoznam hodnôt, z ktorého je používateľ schopný vybrať jednu alebo viac položiek. Výber je následne potrebné potvrdiť stlačením príslušného tlačidla „*Show me results!*“. Názornú ukážku vizualizácie odhadu optimálneho počtu zhlukov, ako aj výslednú vizualizáciu analýzy stability zhlukov prostredníctvom metódy siluety je možné nájsť v podkapitole 4.3.2. Vyššie spomínané metódy sme bližšie charakterizovali v analytickej časti.

Pri treťom vstupe používateľ už priamo volí počet zhlukov, do ktorých majú byť dáta zatriedené na základe techník odvíjajúcich sa od využitých metód zhlukovania. Používateľ má na výber celé čísla v rozsahu od 3 do 10. Samotný výber sa môže odvíjať od využitia metód pre odhad optimálneho počtu zhlukov, alebo od nadobudnutých znalostí, resp. úsudku používateľa. Z hľadiska

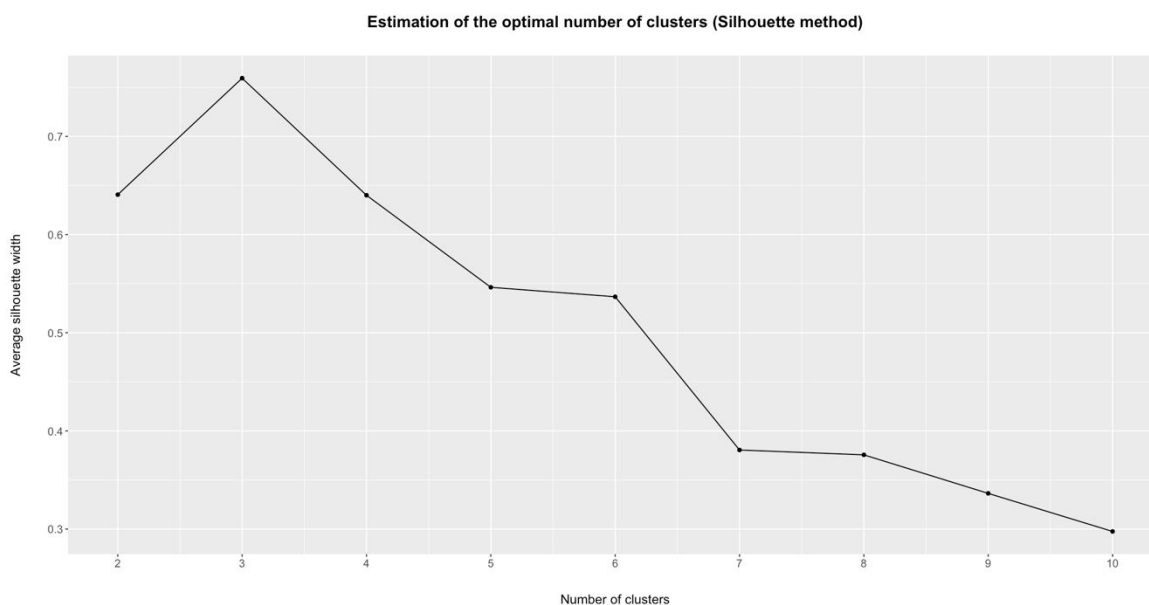
R Shiny elementov sme opäť využili *selectInput*, teda rozbaľovacie menu, alebo zoznam naplnený už spomínanými hodnotami.

Štvrtý, posledný vstup je tlačidlo pre potvrdenie predchádzajúcich vstupov a zároveň je taktiež indikátorom pre systém, aby bol zahájený proces zhukovania a používateľ tak dostal požadované výsledky.

Špecifické vstupy, ktoré uvádzame až na konci tejto podkapitoly súvisia a vyskytujú sa len v časti aplikácie pre aplikovanie hierarchického zhukovania. Z hľadiska vstupov sa jedná o výber konkrétnej metódy hierarchického zhukovania a tlačidla, ktorým je možné vykresliť vizualizáciu dendrogramu. Dendrogram je typ vizualizácie využívaný ku znázorneniu jednotlivých krokov analýzy zhukovania. Špecifická metóda hierarchického zhukovania je volená hneď po prispôbení si obsahu vstupnej dátovej množiny. Vykreslenie dendrogramu je možné po vykonaní tohto výberu.

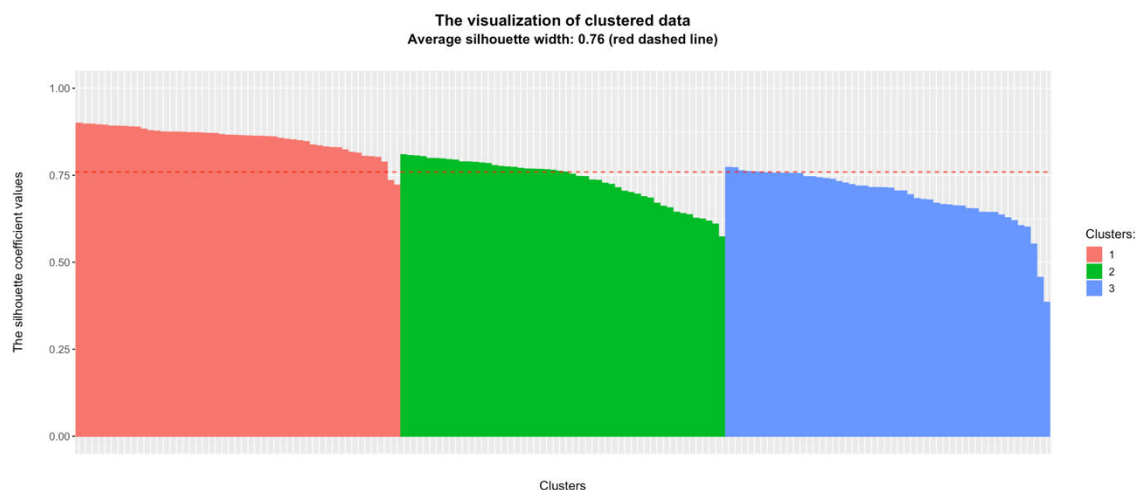
4.3.2. Výstupy

Prvý výstup, ktorý používateľ môže využiť, je odporúčaný počet zhukov na základe jednej z ponúkaných metód. Tá sa práve odvíja od voľby uskutočnenej pri druhom vstupe, ktorý sme bližšie charakterizovali v predchádzajúcej podkapitole. Odhad optimálneho počtu zhukov prostredníctvom metódy lakťa alebo metódy siluety je vizualizovaný pomocou čiarového grafu (Obrázok 16). Analýza stability zhukov pomocou metódy siluety je znázornená stĺpcovým grafom (Obrázok 17), pričom samotnú vizualizáciu charakterizujeme v nasledujúcej časti.



Obrázok 16 Vizualizácia znázorňujúca odporúčaný počet zhukov

V rámci praktickej ukážky sme zvolili vizualizáciu výsledkov optimálneho počtu zhlukov pri použití metódy siluety. Na vodorovnej osi, a teda na osi x sa nachádza informácia o počte zhlukov a na zvislej osi y číselné hodnoty koeficientu siluety. Pri tejto metóde sú ideálne vysoké hodnoty, resp. hodnoty blížiac sa k 1.



Obrázok 17 Analýza stability zhlukov prostredníctvom metódy siluety

Ako sme už naznačili, vizualizácia tejto analýzy je znázornená prostredníctvom stĺpcového grafu. Jednotlivé farebné skupiny stĺpcov znázorňujú získané zhluky. V prípade názornej ukážky sa teda jedná o tri zhluky. Červená prerušovaná čiara značí priemernú hodnotu siluety naprieč všetkými zhlukmi, pričom je ideálne aby všetky stĺpce boli práve nad touto linkou. Rovnako tak je dôležité, aby veľkosť zhlukov bola približne rovnaká, resp. aby rozdiely vo veľkosti zhlukov neboli príliš veľké. Spomenuté vizualizácie vie používateľ z aplikácie exportovať, a to formáte *.png*.

Druhý základný výstup je dátová množina, vizualizovaná v tabuľkovej forme, ktorá bola využitá pri všetkých ďalších analýzách spojených so zhlučovaním prostredníctvom danej metódy. Tento výstup dáva používateľovi možnosť nahliadnuť na finálnu podobu dát, s ktorými metóda zhlučovania pracovala.

V rámci tretieho výstupu používateľ dostane niekoľko súhrnných informácií, resp. výsledkov o samotnom zhlučovaní. Z hľadiska *R Shiny* elementov sú tieto informácie znázornené v textovej forme, pričom na výstup sme využili element *textOutput*. Práve ten je možné implementovať v používateľskom rozhraní aplikácie, pričom sa jedná o veľmi využívanú alternatívu textového výpisu.

Význam štvrtého výstupu spočíva v interpretácii rozdelenia hodnôt jednotlivých atribútov naprieč všetkými zhlukmi, no taktiež v štatistických charakteristikách ich centroidov. Údaje sú prezentované v tabuľkovej forme. Pre lepšie vysvetlenie použijeme nasledujúci príklad (Obrázok 18).

Information about table!

Rendering a results can take a few tens of seconds, be patient, please!

	1, N = 246	2, N = 113	3, N = 191	4, N = 214
Pohlavie				
Muz	202 / 246 (82%)	47 / 113 (42%)	191 / 191 (100%)	0 / 214 (0%)
Zena	44 / 246 (18%)	66 / 113 (58%)	0 / 191 (0%)	214 / 214 (100%)
Vek				
Median (IQR)	64 (58 - 71)	67 (60 - 72)	65 (58 - 72)	68 (60 - 73)
Mean (SD)	64 (9)	65 (9)	64 (10)	66 (10)
Range	36 - 86	38 - 88	40 - 87	37 - 85

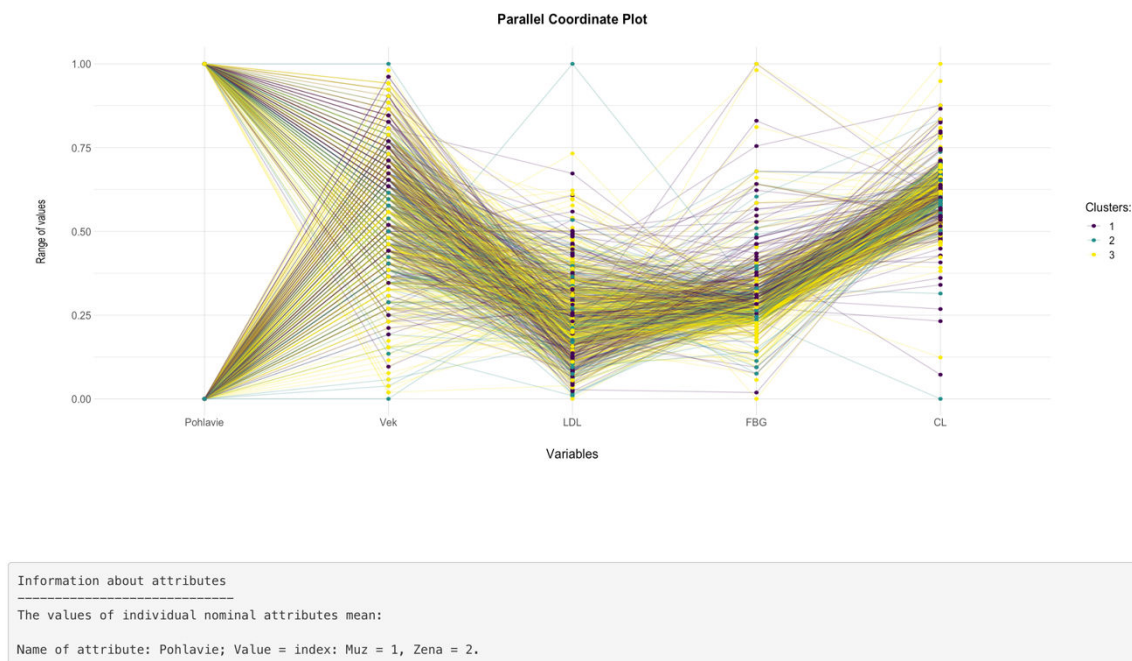
Obrázok 18 Výsek rozdelenia hodnôt jednotlivých atribútov naprieč všetkými zhukmi podľa zvolených parametrov

V hornej časti tabuľky sa nachádza informácia o tom, o aký zhuk v poradí sa jedná spoločne s informáciou, koľko záznamov je do jednotlivých zhukov priradených. V ďalších častiach tabuľky sú znázornené jednotlivé atribúty spoločne s hodnotami, ktoré nadobúdajú v jednotlivých zhukoch. Pre tieto hodnoty sú následne vypísané štatistické údaje, ktoré sa líšia vzhľadom na dátový typ jednotlivých atribútov. V prípade, ak je atribút dátového typu *factor*, alebo *character*, používateľovi je poskytnutá informácia o početnosti výskytu danej hodnoty atribútu v zhuku. V prípade, ak je atribút numerický, používateľ dostane informáciu o mediáne, priemernej hodnote a rozsahu hodnôt, ktoré sú vypočítané len z tých záznamov, ktoré prislúchajú danému zhuku.

Piaty výstup prezentuje vizualizáciu zhukovania v dvojrozmernom prevedení. Jedná sa o bodový graf, kde jednotlivé body predstavujú záznamy z dátovej množiny. Farba týchto bodov je definovaná na základe toho, do ktorého zhuku daný záznam, resp. bod patrí. Používateľ má tak predstavu o tom, či sú jednotlivé zhuky dostatočne odseparované a navzájom odlíšiteľné. Samotný výstup si používateľ môže stiahnuť vo formáte *.png*.

Súčasťou tejto sekcie aplikácie je taktiež výstup v podobe súradnicového grafu. Používateľ má niekoľko možností ako si prispôbiť obsah danej vizualizácie, a teda pracuje so vstupmi, avšak, táto časť používateľského rozhrania je vygenerovaná až po aplikovaní metódy zhukovania. Z tohto dôvodu vstupy súvisiace s týmto typom grafu, resp. analýzy, priblížime v aktuálnej podkapitole. Súradnicový graf možno charakterizovať ako prostriedok pre znázornenie rozloženia hodnôt v jednotlivých zhukoch naprieč zvolenými atribútmi. Taktiež je charakteristický tým, že dokáže

pracovať len s numerickými hodnotami. Pre lepšiu predstavu uvádzame príklad vizualizácie súradnicového grafu, ktorý je možné vidieť na obrázku nižšie (Obrázok 19).



Obrázok 19 Ukážka súradnicového grafu v implementovanej aplikácii

Prvý zo spomínaných vstupov, s ktorými používateľ má možnosť pracovať je voľba spôsobu škálovania numerických atribútov. Má na výber štyri možnosti, a to:

- neškálovať hodnoty numerických atribútov,
- škálovať hodnoty numerických atribútov na interval 0 až 1,
- normalizovať hodnoty jednorozmerne,
- štandardizovať a centrovať hodnoty numerických premenných.

Používateľ si môže zvoliť práve jednu možnosť, a to prostredníctvom označenia príslušného začiarkavacieho poľa. Druhý vstup spočíva vo výbere atribútov, ktoré budú súčasťou vizualizácie. Ich množstvo je ohraničené minimálne dvoma a maximálne desiatimi atribútmi, pričom odporúčaný počet premenných je 5, a to najmä kvôli prehľadnosti grafu. Možnosti, a teda názvy stĺpcov sa odvíjajú od predchádzajúcich úkonov, a teda, medzi možnosťami sú len tie názvy atribútov, ktoré používateľ vybral v rámci prvého kroku. Forma reprezentácie je totožná s tou, ktorú sme podrobne popísali v prechádzajúcej podkapitole - jedná sa o pole s možnosťou výberu viacerých hodnôt. Používateľ si taktiež vie určiť, ktoré nájdené zhľuky chce zahrnúť do výslednej vizualizácie. Na výber má možnosť vykresliť graf so všetkými zhľukami, alebo si zvoliť jeden konkrétny zhľuk. Tieto možnosti sú reprezentované prostredníctvom rozbaľovacieho menu. Na záver je potrebné potvrdiť zvolené hodnoty kliknutím na tlačidlo „Render plot!“.

Ako je možné vidieť aj na vizualizácii vyššie (Obrázok 19), na osi x sú nachádzajú zvolené atribúty, pričom na osi y je možné nájsť spektrum hodnôt týchto atribútov. Každá čiara predstavuje jeden záznam z výslednej dátovej množiny, pričom jej zafarbenie definuje zhuk do ktorého patrí. Táto čiara spája body, ktoré reprezentujú hodnoty vyskytujúce sa v danom zázname, pri jednotlivých zvolených atribútoch. Súčasťou výstupu je taktiež tabuľka, ktorú je možné vidieť aj na predchádzajúcom obrázku (Obrázok 19), v dolnej časti. Ako už bolo povedané, tento typ vizualizácie je schopný pracovať len s numerickými atribútmi, avšak aj tie nominálne je možné upraviť na požadovaný tvar. Možný spôsob, ktorý sme aj využili, je nahradenie jednotlivých hodnôt nominálneho atribútu celočíselnou hodnotou. Tým pádom aj takýto atribút je možné zahrnúť do spomínanej vizualizácie. Avšak, aby mal používateľ prehľad, aká celočíselná hodnota numerického atribútu odpovedá tej ktorej hodnote pôvodného atribútu, bolo vhodné túto informáciu umiestniť práve do danej tabuľky.

Používateľ má taktiež možnosť získať informáciu o tom, ktoré záznamy sú najviac a najmenej podobné v rámci jednotlivých zhukov. Vzhľadom na to, že pri výpočte vzdialenosti medzi záznamami využívame tzv. Gowerova vzdialenosť, informáciu o podobnosti, resp. vzdialenosti záznamov, určuje práve táto metrika. Pojem „Gowerova vzdialenosť“ sme priblížili v analytickej časti diplomovej práce. Čím je vzdialenosť nižšia, tým sú si záznamy viac podobné. Naopak, čím je táto hodnota vyššia, tým sú si záznamy menej podobné. Okrem záznamov, či už tých najviac, alebo najmenej podobných, ktoré sú prezentované v tabuľkovej forme, je používateľovi poskytnutá aj konkrétna hodnota vzdialenosti medzi danými záznamami.

Posledným výstupom je samotná dátová množina, ktorá predstavuje vstup do danej metódy zhukovania, avšak s pridanou informáciou o tom, ktorý záznam do ktorého zhuku patrí. Keďže sa jedná o dátovú množinu, v rámci aplikácie je prezentovaná v tabuľkovej forme. Opäť sme tu sprístupnili funkciu filtrovania, čo umožňuje používateľovi si môže vyselektovať len tie záznamy, ktoré sú pre neho relevantné, a to na základe ktorejkoľvek hodnoty, akéhokoľvek atribútu. Tieto záznamy, či už vyselektované, alebo nie, si používateľ môže stiahnuť v dvoch formátoch - .xlsx a .csv. Tento výstup je špeciálny, keďže zároveň predstavuje hlavný vstup pre poslednú, tretiu časť našej aplikácie, ktorá sa zameriava na hľadanie asociačných pravidiel v rámci definovaných zhukov.

Na záver tejto podkapitoly opäť uvádzame špecifický výstup, ktorý je typický pre hierarchické zhukovanie a ním je dendrogram. Dendrogram je možné nájsť v časti aplikácie, v ktorej je využívané práve hierarchické zhukovanie. Tento typ vizualizácie je využívaný ku znázorneniu jednotlivých krokov analýzy zhukovania. Súčasťou tohto výstupu je aj informácia o koeficiente zvolenej metódy

hierarchického zhukovania, ktorá definuje stabilitu zhukov. Čím je táto hodnota bližšie k 1, tým sú získané zhuky stabilnejšie. Výslednú vizualizáciu je možné exportovať vo formáte *.png*.

4.4. Hľadanie asociačných pravidiel v rámci definovaných zhukov

Tretia, posledná sekcia hlavnej časti aplikácie, je zameraná na hľadanie asociačných pravidiel v rámci definovaných zhukov. V nasledujúcich častiach opäť priblížime jednotlivé vstupy a výstupy, a teda funkcionality tejto časti aplikácie.

4.4.1. Vstupy

Ako už bolo naznačené, hlavný vstup pre túto časť aplikácie predstavuje jeden z výstupov z časti pre aplikovanie metód zhukovania. Konkrétne sa jedná o dátovú množinu pozostávajúcu z jednotlivých záznamov a atribútov, ktorých zloženie používateľ mohol, ale nemusel upraviť, no s pridanou informáciou o tom, do ktorého zhuku, ktorý záznam patrí. A to je práve kľúčové, keďže zmyslom, resp. cieľom tejto časti aplikácie je hľadanie asociačných pravidiel v rámci definovaných zhukov.

Dôležité je poznamenať, že v prípade, ak by aj používateľ záznamy v dátovej množine, v časti systému pre aplikovanie metód zhukovania selektoval, resp. filtroval, ako vstup budú využité vždy všetky záznamy, a teda celá dátová množina. Filtrovanie týchto záznamov slúži len pre jednoduchšiu orientáciu v množine údajov.

Druhý vstup, s ktorým používateľ prichádza do kontaktu, je výber zhukov, pre ktoré budú asociačne pravidlá vygenerované. Samotné prevedenie je rovnaké ako v prípade, v ktorom si používateľ vyberal zloženie atribútov dátovej množiny. Rozdiel je však v tom, že v tomto prípade *varSelectInput* nie je naplnený názvami atribútov, ktoré sa v danej dátovej množine vyskytujú, ale identifikátormi zhukov. Identifikátory zhukov sú prezentované celočíselnými hodnotami. Používateľ má opäť na výber medzi dvoma možnosťami, buď môže získať asociačné pravidlá pre všetky zhuky, alebo vybrať z nich len niektoré. Opäť sme však definovali podmienku, že celkový počet zhukov, pre ktoré budú asociačné pravidlá vygenerované, nesmie klesnúť pod dva. Ako už bolo povedané, grafické používateľské prevedenie je rovnaké ako pri selekcii atribútov, čiže používateľ pri výbere možností pracuje so zaškrtavajúcimi poľami.

Keďže sa opäť jedná o získavanie asociačných pravidiel, nasledujúce vstupy sú totožné s tými, ktoré sme bližšie charakterizovali pri prvej časti aplikácie 4.2. Konkrétne máme na mysli výber hodnoty minimálnej podpory, minimálnej spoľahlivosti a minimálnej dĺžky ľavej strany asociačných pravidiel. Rovnako tak sa jedná o obmedzenie výskytu hodnôt zvolených atribútov na pravej strane,

a teda v dôsledkovej časti asociačných pravidiel. Posledný vstup je taktiež rovnaký, pričom hovoríme o potvrdení zvolených možností a získaní požadovaných výsledkov.

4.4.2. Výstupy

Za prvý výstup, ktorý je používateľovi poskytnutý, možno považovať už samotnú načítanú dátovú množinu, ktorej zdrojom je práve využitá metóda zhukovania. Tento výstup je vizualizovaný v tabuľkovej forme (Obrázok 20), avšak možnosť filtrovania tentokrát už nie je dostupná.

Information about loaded dataset!

1. The number of clusters in the loaded dataset is: 3.
2. The number of rows in each cluster is: 1. = 276, 2. = 265, 3. = 223.

Detailed preview of data (optional)

To view a detailed preview of the data, click on the 'Detailed preview of data' button or on 'Detailed preview of data' panel.

Detailed preview of data

Show 15 entries

	Cluster	Pohlavie	Vek	ESC	R_ICHS	R_CMP
1	1	Muz	[73,88]	[5,7]	true	false
2	1	Muz	[36,58]	[5,7]	false	false
3	1	Muz	[36,58]	[0,5]	true	false
4	1	Zena	[58,63]	[5,7]	true	false
5	1	Muz	[63,68]	[0,5]	false	true
6	1	Muz	[36,58]	[0,5]	true	false
7	1	Muz	[63,68]	[5,7]	false	false
8	1	Muz	[63,68]	[5,7]	true	false
9	1	Muz	[36,58]	[0,5]	true	true
10	1	Muz	[58,63]	[7,33]	true	false

Obrázok 20 Dátová množina s informáciou o zaradení záznamov do zhukov

Okrem iného používateľ dostane informácie aj o tom, koľko rôznych zhukov sa v dátovej množine nachádza a koľko záznamov je do nich priradených.

Druhý výstup predstavuje možnosť detailného náhľadu na dáta nachádzajúce sa v jednotlivých zhukoch. Používateľ má možnosť si z rozbaľovacieho menu zvoliť požadovaný zhuk, svoj výber potvrdiť, čím dostane výsledné informácie. Jedná sa o tabuľku s dátovou množinou, ktorej záznamy odpovedajú len zvolenému zhuku, informácie o tom, ktoré atribúty budú v danom zhuku záznamov odstránené (pozn. - odstránené z dôvodu, že sa v nich nachádza len jedna unikátna hodnota, a teda takýto atribút v rámci získavania asociačných pravidiel nie je potrebný) a koľko záznamov a atribútov sa v dátovej množine nachádza (pozn. - už v rámci odstráneného atribútu, resp. atribútov, ak nejaké odstránené boli).

Výstupy v tejto časti aplikácie sú identické s tými, ktoré sú používateľovi dostupné v prvej sekcii hlavnej časti softvéru, pri hľadaní asociačných pravidiel (4.2). Napriek tomu je možné badať dva hlavné rozdiely. Prvým z nich je, že informácie o unikátnych hodnotách, ktoré sa vyskytli na ľavej strane asociačných pravidiel sú uvedené samostatne pre každý zo zhukov, tak ako to je viditeľné na obrázku nižšie (Obrázok 21). Zobrazia sa až po tom, ako používateľ klikne na tlačidlo „Show or hide unique values“. Kliknutím na rovnaké tlačidlo je túto časť možné taktiež skryť.

Information about association rules!

1. The number of association rules in each cluster is: 1. = 35, 2. = 20.
2. The unique values that appear on the LHS of association rules within the defined clusters are:

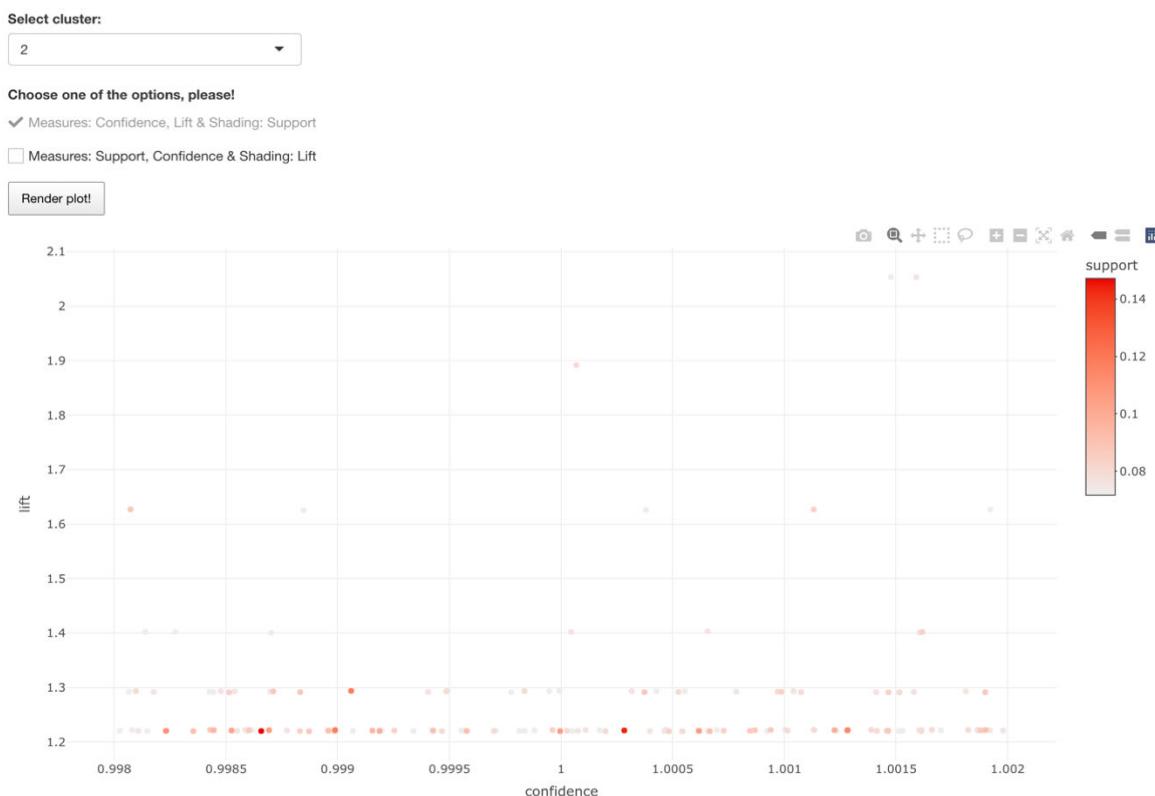
Show or hide unique values

Cluster 1: CL=[103-104]; CRP=[2.22-3.9]; CRP=[3.9-130]; ECHO_EF=51<=; ECHO_PH=0; EKG_FR=[45-64]; EKG_FR=[72-130]; EKG_PQ=[160-180]; EKG_QRS=90-119; EKG_QT=[40-380]; EKG_QT=[400-518]; EKG_Rytmus=0; EKG_STE; EKG_T; ESC=[0-5]; ESC=[5-7]; Fajcenie=3; FBG=2-4; HDL=[0.51-1.14]; HDL=[1.42-108]; Chol=<=4.2; Chol=4.2-6; K=[4.06-4.47]; Kreat=[73-89]; LDL=<=2.6; LDL=2.6-4.1; Na=[4.4-138]; O_CMP; O_DM; O_Hypertenzia; O_IJHS; O_IM; Pohlavie=Muz; Trigl=[1.72-30]; Urea=[5.05-6.47].

Cluster 2: CL=[104-111]; ECHO_EF=51<=; ECHO_PH=0; EKG_FR=[72-130]; EKG_PQ=[14-160]; EKG_QRS=<=89; EKG_QRS=90-119; EKG_Rytmus=0; EKG_T; Fajcenie=3; FBG=2-4; HDL=[1.42-108]; Chol=4.2-6; Kreat=[6.6-73]; LDL=<=2.6; LDL=2.6-4.1; Na=[138-140]; Na=[140-170]; Nalez=0; O_Hypertenzia; Pohlavie=Muz; Pohlavie=Zena; Trigl=[1.21-1.72]; Urea=[2.29-5.05].

Obrázok 21 Unikátne hodnoty vyskytujúce sa na ľavej strane asociačných pravidiel

Druhý rozdiel spočíva v prístupe pri interpretácii dvoch ďalších analýz. Používateľ si pri bodovom grafe a pri vizualizácii vzťahov medzi jednotlivými hodnotami atribútov pri vygenerovaných asociačných pravidlách a samotnými asociačnými pravidlami najprv zvolí zhuk záznamov. Následne, podľa jeho voľby, sú mu poskytnuté príslušné vizualizácie tak, ako to je viditeľné na nasledujúcom obrázku (Obrázok 22).



Obrázok 22 Výber zhuku záznamov využitých pri vizualizácii asociačných pravidiel

5. Testovanie implementovanej aplikácie

Testovanie aplikácie prebiehalo v časovom období medzi 30. novembrom 2021 a 7. decembrom 2021, pričom skupina respondentov, ktorá sa testovania zúčastnila, pozostávala zo siedmych členov. Konkrétne je možné hovoriť o lekároch pôsobiach na rôznych oddeleniach Univerzitnej nemocnice Louisa Pasteura (UNLP) v Košiciach a vo VÚSCHu, ktorý sídli taktiež v Košiciach. Respondenti boli rôzneho pohlavia, rovnako tak rôznych vekových kategórií a taktiež rozdielna bola dĺžka ich pôsobnosti v medicínskej sfére.

Samotné testovanie prebiehalo formou pripravených scenárov s úlohami a následným hodnotením aplikácie pomocou dotazníka. Scenár testovania pozostával z dvoch úloh, na ktoré musel respondent odpovedať výberom jednej z troch ponúkaných možností. Spôsob, akým bolo znenie úloh formulované, odpovedá obsadeniu primárnej cieľovej skupiny implementovanej aplikácie, ktorú tvoria už spomínaní lekári, resp. ľudia z lekárskeho prostredia. Taktiež je dôležité podotknúť, že dotazník svojim obsahom, a teda tým, ako boli dané úlohy postavené, pokrýval všetky dôležité časti aplikácie. A teda:

- sekciu pre hľadanie asociačných pravidiel,
- sekciu pre aplikovanie metód zhukovania,
- sekciu pre hľadanie asociačných pravidiel v rámci definovaných zhukov.

Úlohy, ktoré boli súčasťou prvej časti dotazníka boli navrhnuté v spolupráci s doktorom špecializovaného odboru kardiológie. Ten, po validácii získaných výsledkov z lekárskeho uhľa pohľadu navrhol znenie a formuláciu jednotlivých úloh, s cieľom doviesť testujúceho k záveru, či dané výsledky poukazujú na zvýšené kardiovaskulárne riziko. Celé znenie úloh spoločne s uvedenými možnými odpoveďami je možné nájsť v prílohe A.

Druhá časť testovania bola zameraná na vyhodnotenie použiteľnosti implementovanej aplikácie na základe dotazníka, pričom na tento účel sme využili rámec predefinovaných desiatich otázok, ktoré sú taktiež známe pod anglickým názvom *System usability scale*. Tento prístup poskytuje pohľad na subjektívne hodnotenie použiteľnosti aplikácie. Táto časť priamo nadväzuje na tú predchádzajúcu, vzhľadom na to, že testujúci hodnotí použiteľnosť aplikácie na základe používateľskej skúsenosti, ktorú získal pri plnení daných úloh. Dodatočné informácie, rovnako ako aj znenie desiatich hodnotiacich otázok je možné nájsť v prílohe B, ktorá je súčasťou tejto diplomovej práce.

Na záver je taktiež dôležité spomenúť, že aplikácia bola počas testovania spúšťaná lokálne, a to na osobnom počítači priamo z programovacieho prostredia *RStudio*. Dôvod je ten, že výpočtová

rýchlosť a sila serveru online služby *shinyapps.io*, ktorá je štandardne pre *R Shiny* aplikácie využívaná, nebola dostatočná.

5.1. Použitie prvej časti testovania na základe scenára

Cieľom prvej časti testovania bolo zabezpečiť, aby sa respondenti oboznámili a zvládli prácu so všetkými hlavnými časťami vytvorenej aplikácie. Vzhľadom na to, že pri testovaní aplikácie, a teda pri plnení daných úloh, ktorých znenie spoločne s možnými odpoveďami je možné nájsť v prílohe A, boli respondenti v prípade potreby usmernení, nie je relevantné vyhodnocovať ich výber odpovedí. Podstatné však bolo, že všetci respondenti prešli stanovenými úlohami a mali tak vlastnú skúsenosť práce so všetkými časťami aplikácie. Tým boli splnené predpoklady pre kvalifikované odpovede na otázky v druhej časti dotazníka, a teda tej o použiteľnosti testovanej aplikácie, ktorému sa budeme venovať v nasledujúcej podkapitole.

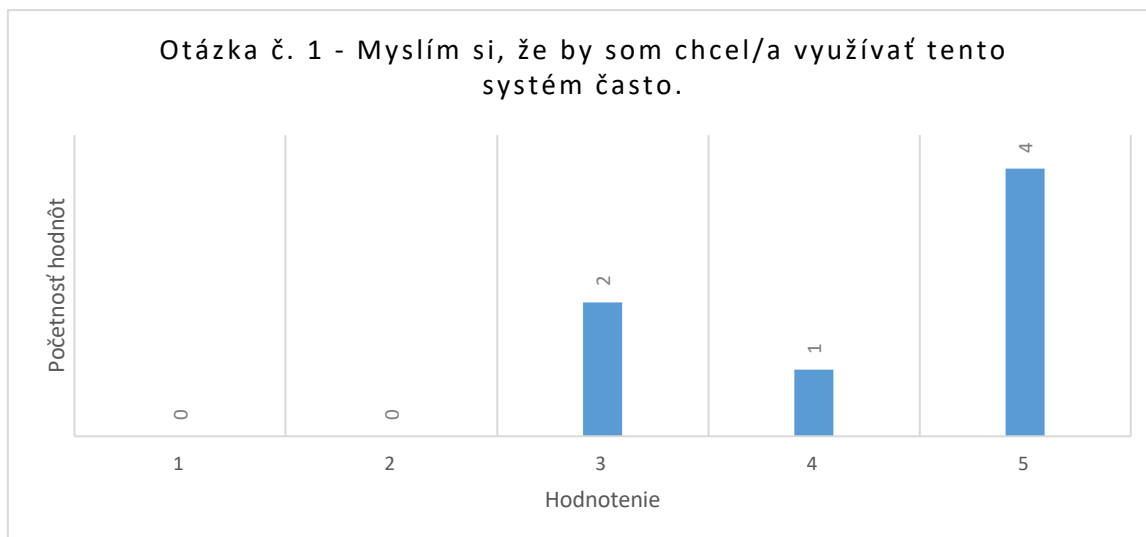
5.2. Vyhodnotenie druhej časti testovania na základe dotazníka

Ako už bolo naznačené, v rámci tejto podkapitoly sa zameriame na vyhodnotenie druhej časti testovania, ktorá bola zameraná práve na použiteľnosť testovanej aplikácie na základe pripraveného dotazníka. Každý respondent odpovedal na 10 rôznych otázok, pričom odpoveď spočívala vo výbere hodnoty z rozsahu od 1 po 5, kde hodnota 1 reprezentovala hodnotenie „vôbec nesúhlasím“ a naopak hodnota 5 hodnotenie „úplne súhlasím“. Táto logika, taktiež známa ako Likertova škála, bola aplikovaná pri všetkých otázkach.

V nasledujúcej časti sa zameriame na vyhodnotenie odpovedí, a to z pohľadu početnosti jednotlivých hodnôt. Vyhodnotenie bude vyjadrené prostredníctvom stĺpcových grafov a slovného komentáru.

Otázka č. 1 - Myslím si, že by som chcel/a využívať tento systém často.

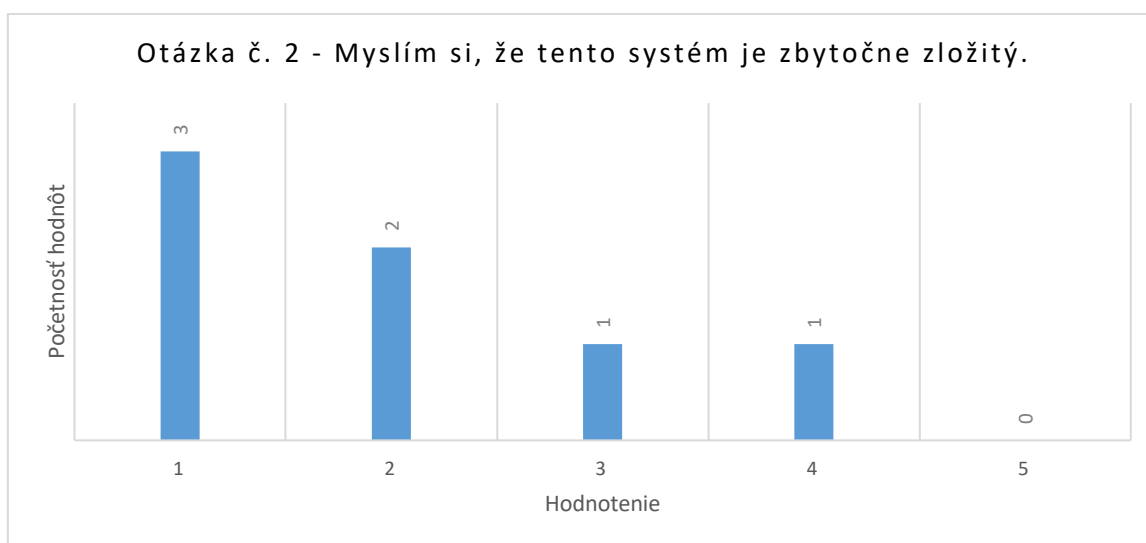
Odpovede na prvú otázku, ktorá bola zameraná najmä na celkový dojem z aplikácie, boli zväčša kladné. Všetky sa nachádzali buď na pomedzí alebo v druhej polovici hodnotiacej škály, ktorá vyjadruje súhlas s položenou otázkou. V kontexte tej prvej to znamená, že všetci používatelia sú naklonení k možnosti využívať implementovaný systém. Štyria z nich dokonca s daným výrokom súhlasia v plnom rozsahu.



Obrázok 23 Vyhodnotenie otázky č. 1

Otázka č. 2 - Myslím si, že tento systém je zbytočne zložitý.

Pri druhej otázke sa dvaja respondenti vyjadrili, že naša implementovaná aplikácia, resp. niektoré časti z nej sú zbytočne zložité. Avšak zvyšní piati s položenou otázkou nesúhlasia, z čoho je možné usúdiť, že daný systém nepokladajú za zložitý.

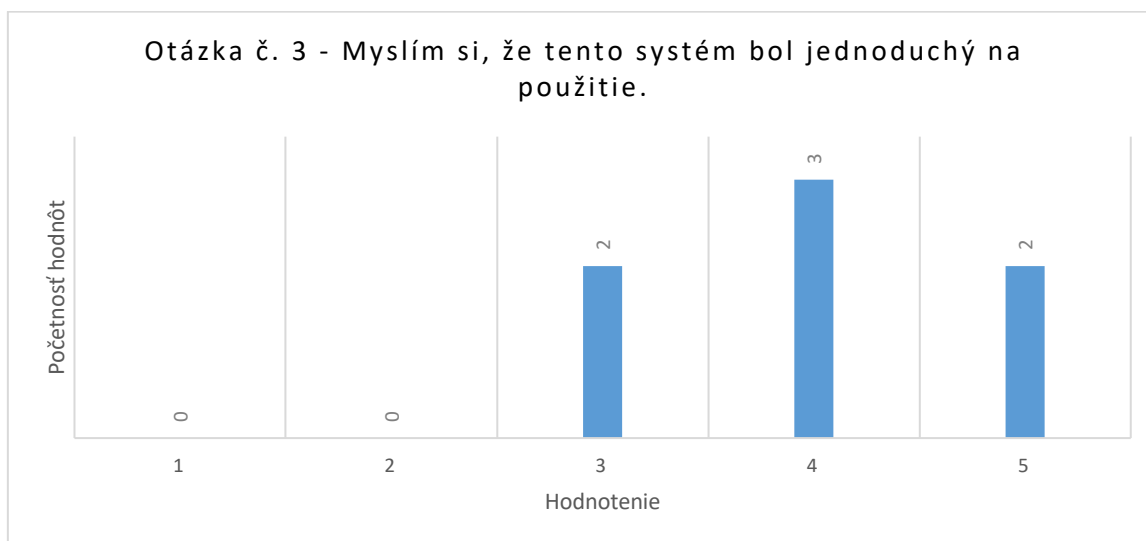


Obrázok 24 Vyhodnotenie otázky č. 2

Otázka č. 3 - Myslím si, že tento systém bol jednoduchý na použitie.

Tretia otázka nepriamo nadväzuje na tú predchádzajúcu, pričom respondenti boli konfrontovaní s otázkou, či majú z implementovanej aplikácie pocit, že je jednoduchá na použitie. Výsledky možno opäť vnímať pozitívne. Piaty zo siedmich testujúcich s tvrdením súhlasia, pričom odpovede zvyšných dvoch predstavujú hodnotu 3.

V kontexte druhej a tretej otázky môžeme teda zhodnotiť, že aj keď sa implementovaný systém javí miestami ako zložitý, stále je vysoká šanca, že po dôkladnom oboznámení sa s jednotlivými funkcionalitami ho nebude zložité používať.

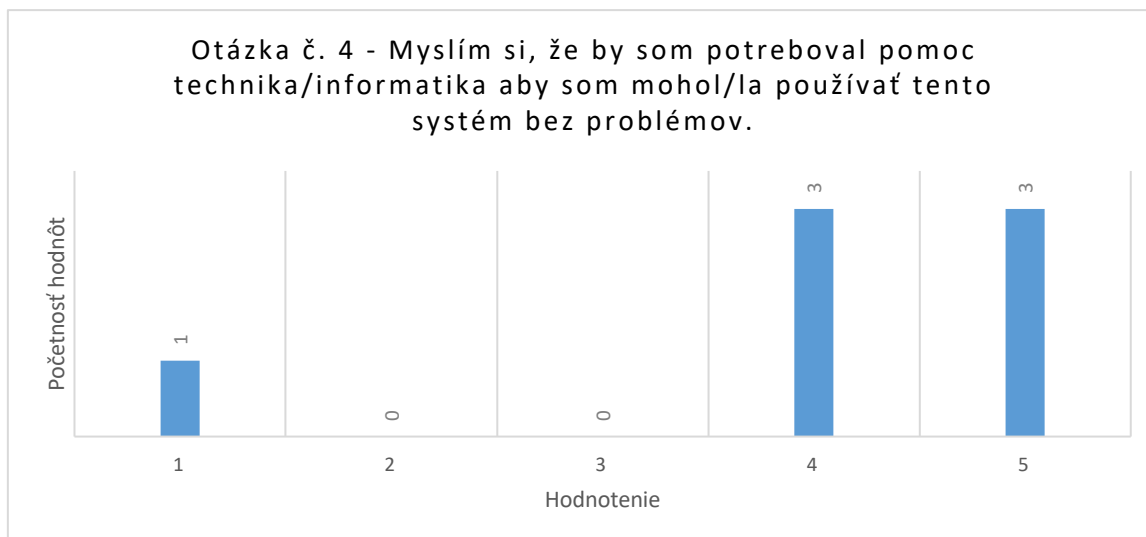


Obrázok 25 Vyhodnotenie otázky č. 3

Otázka č. 4 - Myslím si, že by som potreboval pomoc technika/informatika aby som mohol/la používať tento systém bez problémov.

Výsledky štvrtej otázky sú hodné diskusie. Väčšina respondentov, konkrétne šiesti zo siedmich sa vyjadrili, že pre bezproblémové používanie aplikácie by potrebovali pomoc informatika, či technika, pričom len jeden testujúci by o takúto pomoc nestál.

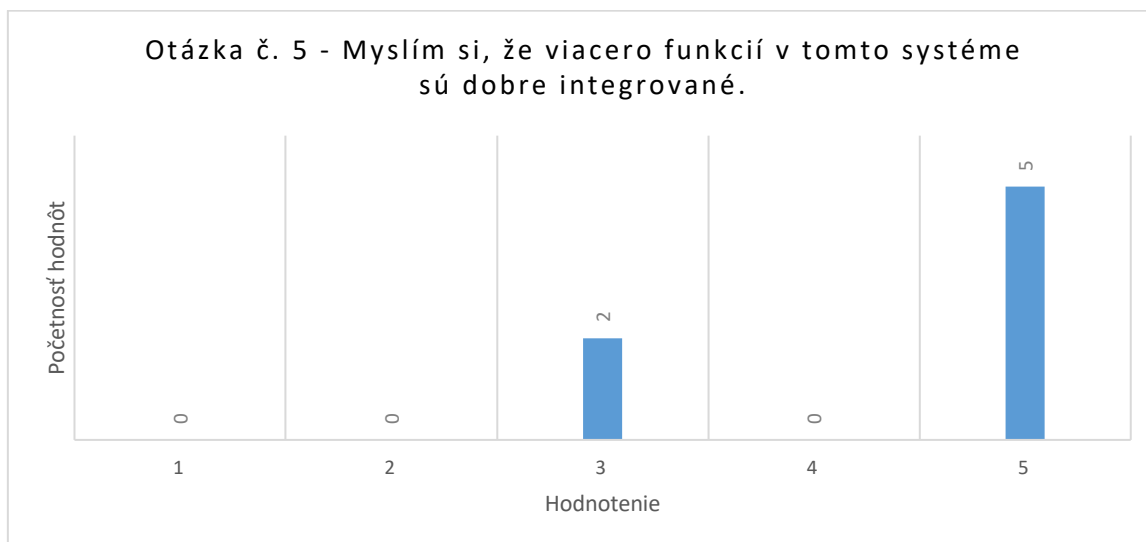
Opäť tu možno badať akúsi nadväznosť na predchádzajúce otázky a odpovede. Teda vo výsledku, napríklad ja s určitou pomocou by systém nepôsobil tak zložito a bol by jednoduchý na použitie.



Obrázok 26 Vyhodnotenie otázky č. 4

Otázka č. 5 - Myslím si, že viacero funkcií v tomto systéme sú dobre integrované.

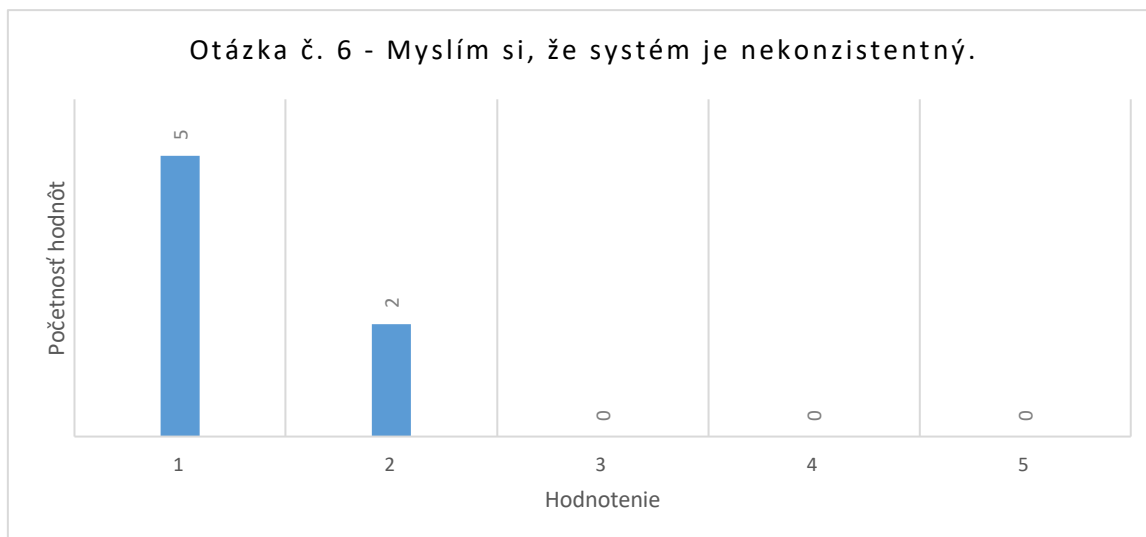
Výsledky piatej otázky možno hodnotiť opäť veľmi pozitívne. Piaty testujúci sa stotožňuje s tvrdením, že viacero funkcií, ktoré sú súčasťou našej aplikácie, sú integrované dobre. Zvyšní dvaja testujúci ako odpoveď zvolili hodnotu 3. Zníženie hodnotenia integrity funkcionality pripisujeme práve vyššej náročnosti na pochopenie jednotlivých funkcionality implementovaného systému.



Obrázok 27 Vyhodnotenie otázky č.5

Otázka č. 6 - Myslím si, že systém je nekonzistentný.

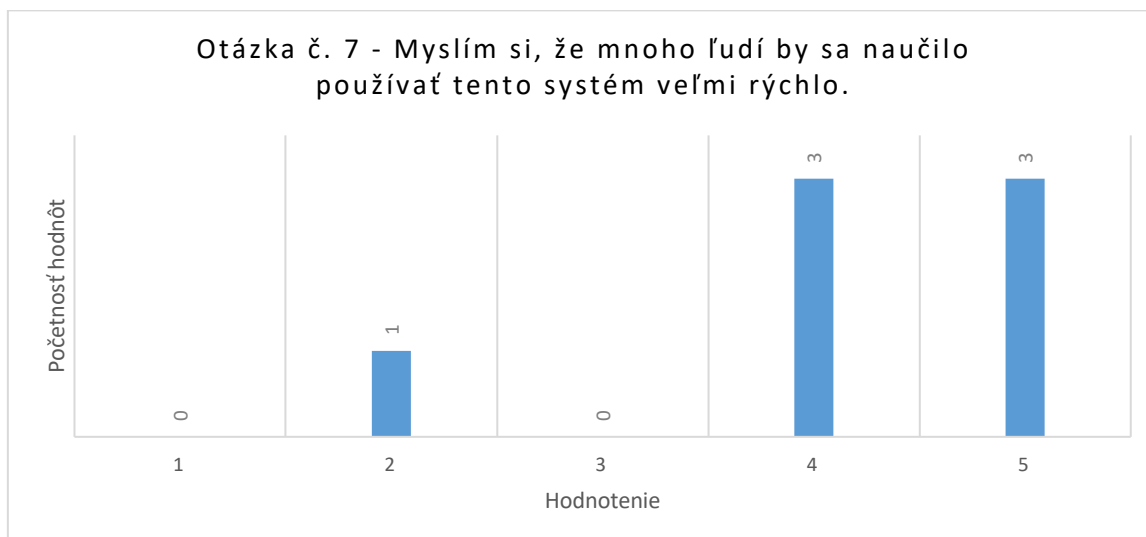
Potešujúce sú taktiež výsledky šiestej otázky, z ktorej je možné vydedukovať, že všetci používatelia nesúhlasia s tvrdením o nekonzistentnosti aplikácie, teda si myslia, že je konzistentná. Až piati respondenti s tvrdením o nekonzistentnosti vôbec nesúhlasia.



Obrázok 28 Vyhodnotenie otázky č.6

Otázka č. 7 - Myslím si, že mnoho ľudí by sa naučilo používať tento systém veľmi rýchlo.

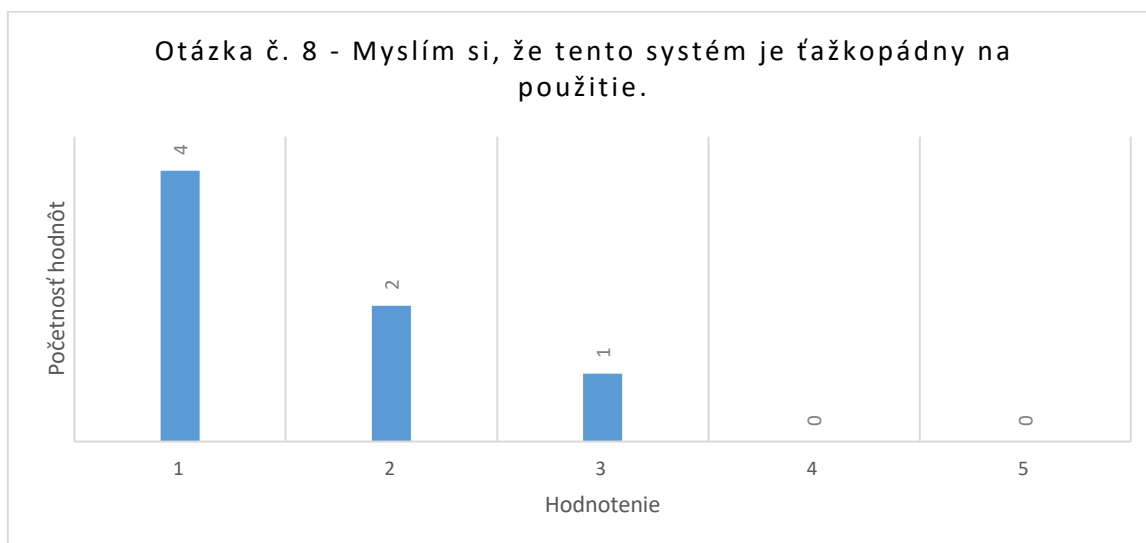
Pre pripomenutie, výsledky štvrtej otázky jasne hovoria o tom, že respondenti by potrebovali pomoc informatika či technika pre bezproblémový chod aplikácie. Odpovede na siedmu otázku však svedčia o tom, že testujúci taktiež súhlasia s tvrdením, že mnoho by sa náš systém naučilo používať veľmi rýchlo. Zaujímavosťou je, že rozloženie hodnôt pri odpovediach je totožné práve s tými pri otázke číslo 4.



Obrázok 29 Vyhodnotenie otázky č. 7

Otázka č. 8 - Myslím si, že tento systém je ťažkopádny na použitie.

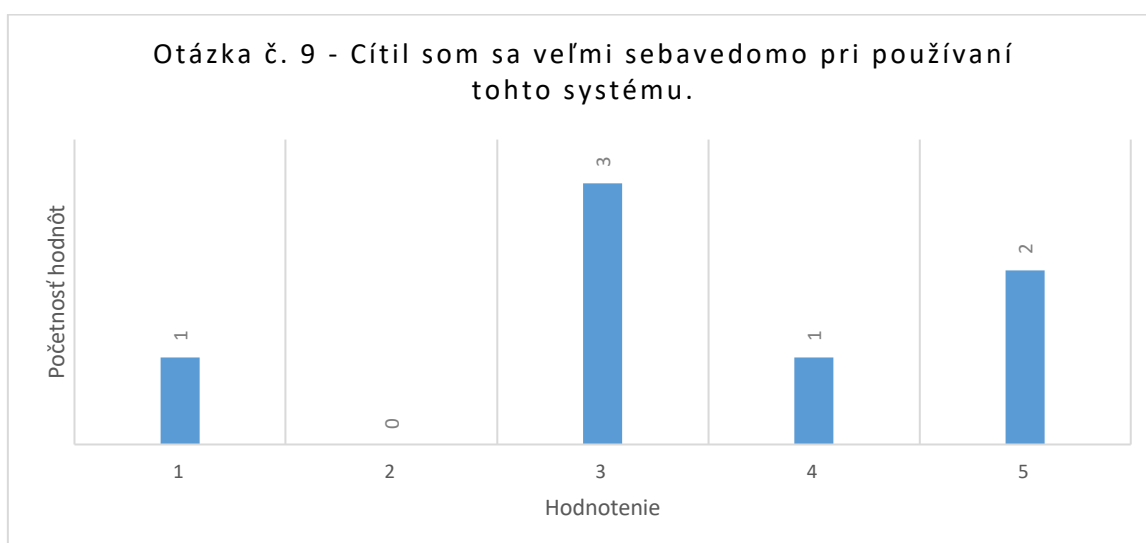
Respondenti taktiež nemali pocit, že testovaný systém bol ťažkopádny na použitie. Štyria z nich nesúhlasia s tvrdením o ťažkopádnosti v plnom rozsahu, dvaja sú taktiež naklonení k nesúhlasu, pričom len jeden testujúci odpovedal výberom hodnoty 3.



Obrázok 30 Vyhodnotenie otázky č. 8

Otázka č. 9 - Cítil som sa veľmi sebavedomo pri používaní tohto systému.

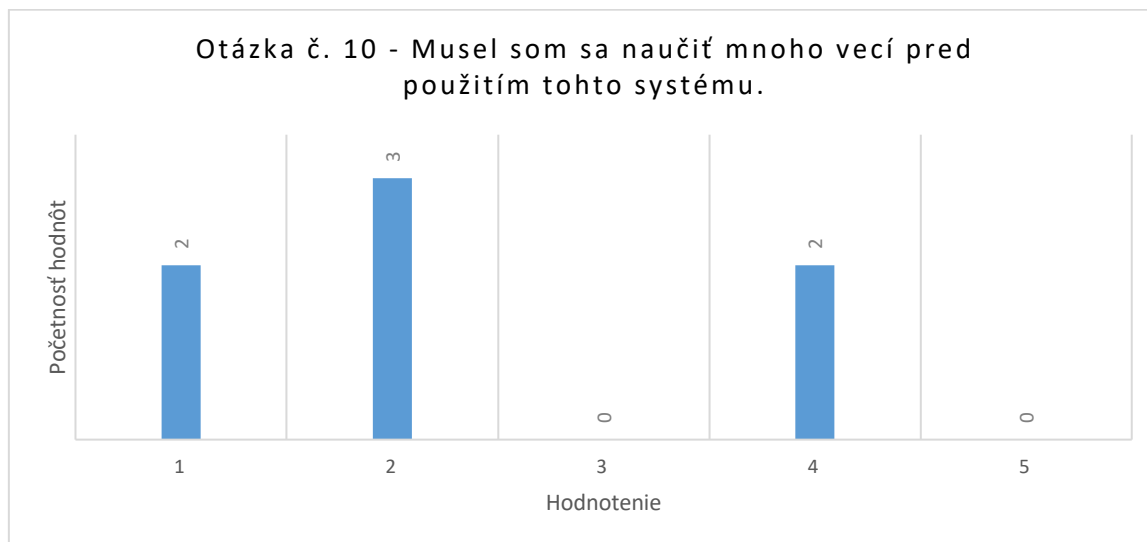
Pestré rozloženie hodnôt odpovedí je možné badať pri otázke číslo 9. Odpovede troch testujúcich sa nachádzajú v strede hodnotiacej škály, pričom ďalší traja súhlasia s tvrdením, že sa pri používaní aplikácie cítili veľmi sebavedomo. Naopak jeden respondent s týmto tvrdením vôbec nesúhlasí. Sebavedomosť pri používaní je možné taktiež previazať aj na prácu s pomocou technika alebo informatika.



Obrázok 31 Vyhodnotenie otázky č. 9

Otázka č. 10 - Musel som sa naučiť mnoho vecí pred použitím tohto systému.

Pri poslednej, desiatej otázke piati respondenti odpovedali, že z ich pohľadu nebolo nutné sa naučiť, resp. oboznámiť sa s mnohými vecami, bez ktorých by neboli schopní využívať systém bez akýchkoľvek problémov. Dvaja testujúci si však myslia opak a s uvedeným výrokom súhlasia.



Obrázok 32 Vyhodnotenie otázky č. 10

Z hľadiska rozloženia hodnôt odpovedí pri jednotlivých otázkach je možné tvrdiť, že získané výsledky plynúce z dotazníka o použiteľnosti aplikácie sú zväčša pozitívne. Samozrejme, je nevyhnutné správne pochopiť kontext položenej otázky, resp. tvrdenia a následných odpovedí.

Tento typ dotazníka je však potrebné vyhodnotiť aj špecifickým spôsobom. Pravidlá [47] pre vyhodnotenie prístupu „*System Usability Scale*“ sme zhrnuli do nasledujúcich bodov:

1. V prvom kroku je potrebné vyjadriť premennú X . Najprv je nutné zrátať jednotlivé body hodnotenia, a to pri všetkých nepárnych otázkach, a teda pri otázkach číslo 1, 3, 5, 7 a 9. Následne sa od tejto výslednej hodnoty odráta číslo 5 a výsledok je vyjadrením premennej X .
2. V druhom kroku je potrebné vyjadriť premennú Y . Najprv je nutné zrátať jednotlivé body hodnotenia, a to pri všetkých párnych otázkach, a teda pri otázkach číslo 2, 4, 6, 8 a 10. Výsledná hodnota sa odráta od hodnoty 25 a výsledok je vyjadrením premennej Y .
3. V poslednom kroku vyjadríme skóre tohto typu dotazníka. Vzorec pre jeho výpočet je nasledovný:
 - Skóre = $(X + Y) * 2.5$

Tento postup je potrebné aplikovať pre každý odovzdaný dotazník. Výsledne skóre pri jednotlivých odovzdaných dotazníkoch uvádzame v nasledujúcom zozname:

- $\text{Skóre}(\text{Dotazník_1}) = 80,$
- $\text{Skóre}(\text{Dotazník_2}) = 75,$
- $\text{Skóre}(\text{Dotazník_3}) = 90,$
- $\text{Skóre}(\text{Dotazník_4}) = 52.5,$
- $\text{Skóre}(\text{Dotazník_5}) = 92.5,$
- $\text{Skóre}(\text{Dotazník_6}) = 57.5,$
- $\text{Skóre}(\text{Dotazník_7}) = 75.$

Priemerné skóre vypočítané z vyššie uvedených hodnôt je 74.64. Z hľadiska hodnotiacej škály sa výsledná hodnota nachádza v intervale 68 až 80.3, čo je ohodnotenú známku B (pozn. - známky hodnotenia sú A, B, C, D, F a odvíjajú sa od jednotlivých bodových intervalov), pričom slovné hodnotenie nášho softvéru je možné vyjadriť ako „good“, a teda dobré. Z hľadiska použiteľnosti sa teda naša aplikácia javí ako dobre použiteľná, pričom niektoré menšie úpravy by vedeli jej úroveň zvýšiť a dopomôcť tak k jednoduchšiemu používaniu.

Záver

Využitie dátovej analytiky v oblasti medicíny možno považovať za prospešný a efektívny prístup. Získané informácie a znalosti majú široké spektrum využitia, či už pri včasnej diagnostike ochorení, ale taktiež pri ich liečení a prevencii.

Súčasťou tejto diplomovej práce je súhrn teoretických informácií o deskriptívnej dátovej analytike, ako aj o konkrétnych modeloch a algoritmoch, ktoré je možné do tejto oblasti zaradiť. Zamerali sme sa výhradne na hľadanie asociačných pravidiel a aplikovanie metód zhlukovania, pričom sme charakterizovali spôsoby vyhodnotenia získaných výsledkov. Taktiež sme sa zamerali na analýzu miery využitia deskriptívneho modelovania v oblasti medicíny a zdravotnej starostlivosti, no rovnako tak na analýzu konkrétnych prípadových štúdií, pri ktorých boli spomínane modely v tejto oblasti aplikované.

V rámci praktickej časti sme navrhli a implementovali aplikáciu pre podporu výskumu kardiológov práve na báze deskriptívnych modelov. Využili sme programovací jazyk *R*, konkrétne balík *Shiny*, ktorý je jednou z možností pre vytvorenie webových aplikácií. Implementovaný softvér pozostáva z troch hlavných častí, ktoré sa zameriavajú na je hľadanie asociačných pravidiel, aplikovanie metód zhlukovania a hľadanie asociačných pravidiel v definovaných zhlukoch. Vhodnou interpretáciou získaných výsledkov, či už v textovej podobe alebo vo forme interaktívnych vizualizácií, bolo možné overiť ich relevanciu z medicínskeho uhľa pohľadu, pričom na tomto úkone sa podieľal doktor špecializovaného odboru kardiológie. V spolupráci s ním boli navrhnuté dve úlohy, ktoré boli náplňou testovania implementovaného systému. Svojou formuláciou boli prispôbené pre našu cieľovú skupinu používateľov, ktorou sú najmä lekári, resp. ľudia pôsobiaci v medicínskom prostredí. Testovania softvéru sa zúčastnilo 7 respondentov - lekárov, ktorí pôsobia v UNLP v Košiciach alebo vo VÚSCHu. Po absolvovaní testovania každý z nich taktiež vyplnil dotazník hodnotiaci použiteľnosť testovaného softvéru. Získané výsledky možno hodnotiť ako pozitívne, pričom priemerné skóre, ktorým bola naša aplikácia ohodnotená je 74.64 bodov, čo zodpovedá známke B, ktorá sa dá interpretovať slovným hodnotením „good“, a teda ako „dobré“. Ako už bolo povedané, validácia získaných výsledkov z medicínskeho hľadiska prebehla úspešne. Je teda možné konštatovať, že z výsledkov extrahované znalosti môžu byť relevantným a nápomocným zdrojom pri výskume na báze analýzy dát pomocou popisných modelov.

Zoznam použitej literatúry

- [1] Who.int. 2021. Cardiovascular Diseases (Cvds). [online] Dostupné na internete: <[https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-\(cvds\)](https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds))>
- [2] World-heart-federation.org. 2020. Dostupné na internete: <https://world-heart-federation.org/wp-content/uploads/2017/05/Cardiovascular_diseases_in_China.pdf>
- [3] Vidinský, B., 2019. Slovensko - krajina s jednou z najväčších úmrtností na kardiovaskulárne ochorenia v rámci OECD. [online] Zalepsiezdravotnictvo.sk. Dostupné na internete: <<https://zalepsiezdravotnictvo.sk/blogy/36-slovensko-krajina-s-jednou-z-najvaecsich-umrtnosti-na-srdcove-ochorenia-v-ramci-oecd>>
- [4] American Heart Association News, 2019. Cardiovascular diseases affect nearly half of American adults, statistics show. [online] www.heart.org. Dostupné na internete: <<https://www.heart.org/en/news/2019/01/31/cardiovascular-diseases-affect-nearly-half-of-american-adults-statistics-show>>
- [5] Paralič, J., 2003. Objavovanie znalostí v databázach [online] people.tuke.sk. Dostupné na internete: <<http://people.tuke.sk/jan.paralic/knihy/ObjavovanieZnalostivDB.pdf>>
- [6] Agyapong, K. B., Hayfron-Acquah, Dr. J. B. and Asante, Dr. M.: An Overview of Data Mining Models (Descriptive and Predictive), IJournals: International Journal of Software & Hardware Research in Engineering (ISSN-2347-4890), Volume 4 Issue 5, May 2016 [online] ijournals.in. Dostupné na internete: <<https://ijournals.in/wp-content/uploads/2017/07/11.4517-Agyapong.compressed.pdf>>
- [7] Pilcher, G., 2019. Supervised vs. Unsupervised Machine Learning. [online] www.elderresearch.com. Dostupné na internete: <<https://www.elderresearch.com/blog/supervised-vs-unsupervised-machine-learning/>>
- [8] Parvez, A., Saqib, Q., Syed Q. and Afser R.: Techniques of Data Mining In Healthcare: A Review, International Journal of Computer Applications (0975 – 8887) - Volume 120 – No.15, June 2015. [online] Dostupné na internete: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.695.5904&rep=rep1&type=pdf>>.
- [9] Stankovičová, I.: Možnosti extrakcie asociačných pravidiel z údajov pomocou SAS Enterprise Miner. [Online] Dostupné na internete: <https://is.muni.cz/el/sci/podzim2012/M9DM2/um/Stankovicova_Zlin2009_asoc_analyza.pdf>

- [10] Garg, A.: Complete guide to Association Rules (1/2) [Online] Towards Data Science. Dostupné na internete: <<https://towardsdatascience.com/association-rules-2-aa9a77241654>>
- [11] Srikant, R., Agrawal, R. Mining generalized association rules, Future Generation Computer Systems. San Jose, CA, ISSN 0167-739X, Volume 13, Issues 2 – 3 (1997). [online] Dostupné na internete: <http://rakesh.agrawal-family.com/papers/vldb95tax_rj.pdf>
- [12] Kavitha, M., Tamil Selvi, S.T.: Comparative Study on Apriori Algorithm and Fp Growth Algorithm with Pros and Cons., International Journal of Computer Science Trends and Technology (IJST), Volume 4 Issue 4, Jul - Aug. 2016. [online] www.ijcstjournal.org. Dostupné na internete: <<http://www.ijcstjournal.org/volume-4/issue-4/IJCST-V4I4P28.pdf>>.
- [13] Mythili, M.S., Mohamed Shanavas, A.R.: Performance Evaluation of Apriori and FP-Growth Algorithms. Volume: 79, Issue: 10 (2013), International Journal of Computer Applications. [online] Dostupné na internete: <<https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.403.1361&rep=rep1&type=pdf>>
- [14] Briatková, M.: Dobývanie znalostí z e-learningových dát, diplomová práca, Masarykova univerzita, 2008, Brno, Česká republika. [Online] Dostupné na internete: <https://is.muni.cz/th/nn56a/DP_Briatkova.pdf>
- [15] Dzurovčin, A.: Dolovanie asociačných pravidiel pre potreby rozpoznávania pomenovaných entít, diplomová práca, Technická univerzita v Košiciach, 2013, Košice, Slovenská republika [Online] Dostupné na internete: <<https://opac.crzp.sk/?fn=detailBiblioForm&sid=24C6E28F89ED84B9286037F8EBFA&seo=CRZP-detail-kniha>>
- [16] Maimon, O., Rokach, L.: Data Mining and Knowledge Discovery Handbook. New York City, NY: Springer, ISBN: 978-0-387-25465-4 (2005).
- [17] Jain, A.K., Murty, M.N., Flynn, P.J.: *Data Clustering: A Review*. Volume: 31, Issue: 3, ACM Computing Surveys (1999). [Online] Dostupné na internete: <<https://dl.acm.org/doi/abs/10.1145/331499.331504>>
- [18] Paralič, J.: Analýza zhlukov [PDF dokument] Získané z <http://people.tuke.sk/jan.paralic/oz.html>
- [19] Gregorová, V.: Zhluková analýza v systémoch STATISTICA a MATLAB, diplomová práca, Masarykova univerzita, 2008, Brno, Česká republika. [Online] Dostupné na internete: <<https://is.muni.cz/th/qpzjr/>>

- [20] Trebuňa, P., Beres, M.: Klasifikácia metód zhlukovania a oblasti ich využitia, Transfer inovácií (2010). [Online] Dostupné na internete: <<https://www.sjf.tuke.sk/transferinovacii/pages/archiv/transfer/16-2010/pdf/31-34.pdf>>
- [21] Trebuňa, P., Halčinová, J.: Podstata nehierarchických metód zhlukovej analýzy, Transfer inovácií (2009). [Online] Dostupné na internete: <<https://www.sjf.tuke.sk/transferinovacii/pages/archiv/transfer/23-2012/pdf/138-140.pdf>>
- [22] Syakur, M.A., Khotimah, B.K., Rochman, E.M.S., Satoto, B.D.: Integration K- Means Clustering Method and Elbow Method For Identification of The Best Customer Profile Cluster. Volume 336, IOP Conference Series: Materials Science and Engineering (2018). [Online] Dostupné na internete: <<https://iopscience.iop.org/article/10.1088/1757-899X/336/1/012017>>
- [23] Dinh, D., Fujinami, T., Huynh, V.: Estimating the Optimal Number of Cluster in Categorical Data Clustering by Silhouette Coefficient. Communications in Computer and Information Science, vol 1103. Springer, Singapore (2019). Dostupné na internete: <https://link.springer.com/chapter/10.1007/978-981-15-1209-4_1#citeas>
- [24] Bednáriková, V.: Vytvorenie aplikácie s implementáciou vybraných algoritmov zhlukovania, diplomová práca, Technická univerzita v Košiciach, 2013, Košice, Slovenská republika [Online] Dostupné na internete: <<https://opac.crzp.sk/?fn=detailBiblioForm&sid=512CD14455EC7E23F05E0E245F73&seo=CRZP-detail-kniha>>
- [25] Trebuňa, P., Halčinová, J.: Princípy zhlukovacej analýzy, Trends and Innovative Approaches in Business Processes (2010). [Online] Dostupné na internete: <https://www.sjf.tuke.sk/umpadi/taipvpp/2010/index.files/clanky%20PDF/TREBUNA_HALCINOVA.pdf>
- [26] Divyanshu, A.: Gower's distance [Online] [www.medium.com](https://medium.com/analytics-vidhya/gowers-distance-899f9c4bd553). Dostupné na internete: <<https://medium.com/analytics-vidhya/gowers-distance-899f9c4bd553>>
- [27] Kardiovaskulárne choroby [online]. Encyclopaedia Beliana, ISBN 978-80-89524-30-3. [cit. 2022-04-04]. Dostupné na internete: <<https://beliana.sav.sk/heslo/kardiovaskularne-choroby>>.
- [28] Majerčák, I.: Kardiovaskulárne riziko po dovŕšení 40. roku života. [Online] Unilabs, Alpha medical. Dostupné na internete: <<https://www.unilabs.sk/clanky-invito/kardiovaskularne-riziko-po-dovrseni-40-roku-zivota>>

- [29] Patil, M., Joshi, R. C. and Toshniwal, D.: Association rule for classification of type-2 diabetic patients, Second International Conference on Machine Learning and Computing, (2010). [Online] Dostupné na internete: <<https://ieeexplore.ieee.org/document/5460715>>
- [30] Ilayaraja, M. and Meyyappan, T.: Mining Medical Data to Identify Frequent Diseases using Apriori Algorithm, Proceedings of the 2013 International Conference on Pattern Recognition, Informatics and Mobile Engineering, (2013) February 21-22. [Online] Dostupné na internete: <<https://ieeexplore.ieee.org/document/6496471>>.
- [31] J. Nahar, T. Imam, K. S. Tickle and Y. P. Chen: Association rule mining to detect factors which contribute to heart disease in males and females, Expert Systems with Applications, vol. 40, pp. 1086-1093, (2013) [Online] Dostupné na internete: <<https://www.sciencedirect.com/science/article/pii/S095741741200989X>>
- [32] S. Belciug: Patients length of stay grouping using the hierarchical clustering algorithm, Annals of University of Craiova, Math. Comp. Sci. Ser., ISSN: 1223-6934, vol. 36, no. 2, (2009), pp. 79-84. [Online] Dostupné na internete: <<http://inf.ucv.ro/~ami/index.php/ami/article/view/288>>.
- [33] D. Wang, J. Ge, L. Wu and X. Song: Mining Frequent Patterns for ECG Multi-label Data by FP-Growth Algorithm Based on Spark, 2019 7th International Conference on Information, Communication and Networks (ICICN), 2019, pp. 171-174, doi: 10.1109/ICICN.2019.8834931. [Online] Dostupné na internete: <<https://ieeexplore.ieee.org/document/8834931>>
- [34] Bertsimas, M. V. Bjarnadóttir, M. A. Kane, J. C. Kryder, R. Pandey, S. Vempala and G. Wang: Algorithmic prediction of health-care costs, Oper. Res., vol. 56, no. 6, (2008), pp. 1382-1392. [Online] Dostupné na internete: <<https://pubsonline.informs.org/doi/abs/10.1287/opre.1080.0619?journalCode=opre>>
- [35] Godwin Ogbuabor, & Ugwoke, F. N. (2018). Clustering Algorithm for a Healthcare Dataset Using Silhouette Score Value. International Journal of Computer Science & Information Technology (IJCSIT), 10(2), 27–37. <http://doi.org/10.5281/zenodo.1248795> [Online] Dostupné na internete: <<https://aircconline.com/ijcsit/V10N2/10218ijcsit03.pdf>>
- [36] Pecháň, I.: Niektoré aspekty energetického metabolizmu myokardu. 3. časť. Reperfúzne poškodenie myokardu, Slovenský ústav srdcových a cievnych chorôb, Bratislava. [Online] Dostupné na internete: <<http://www.cardiology.sk/casopis/100/05/main.htm>>
- [37] Md Saiful Islam, Md Mahmudul Hasan, Xiaoyi Wang, Hayley D. Germack and Md Noor-E-Alam: A Systematic Review on Healthcare Analytics: Application and Theoretical Perspective of Data Mining, Healthcare (Basel). 2018 Jun; 6(2): 54 Dostupné na internete: <<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6023432/>>.

- [38] Neesha, J., Nur'Aini Abdul R. and Wahidah H.: Data Mining in Healthcare – A Review, *Procedia Computer Science*, Volume 72, 2015, Pages 306-313, ISSN 1877-0509. [online] Dostupné na internete: <<https://www.sciencedirect.com/science/article/pii/S1877050915036066>>.
- [39] Tandan, M., Acharya, Y., Pokharel, S. and Timilsina, M.: Discovering symptom patterns of COVID-19 patients using association rule mining. *Computers in Biology and Medicine*, Volume 131, 2021, 104249, ISSN 0010-4825. [online] Dostupné na internete: <<https://www.sciencedirect.com/science/article/pii/S0010482521000433>>.
- [40] Matthew W. Segar, Shreya Rao, Ann Marie Navar, Erin D. Michos, Alana Lewis, Adolfo Correa, Mario Sims, Amit Khera, Amy E. Hughes, Ambarish Pandey, County-level phenomapping to identify disparities in cardiovascular outcomes: An unsupervised clustering analysis: Short title: Unsupervised clustering of counties and risk of cardiovascular mortality, *American Journal of Preventive Cardiology*, Volume 4, 2020, 100118, ISSN 2666-6677, <https://doi.org/10.1016/j.ajpc.2020.100118>. Dostupné na internete: <<https://www.sciencedirect.com/science/article/pii/S2666667720301185>>
- [41] Guo, Q., Lu, X., Gao, Y., Zhang, J., Yan, B., Su, D., Song, A., Zhao, X., & Wang, G. (2017). Cluster analysis: a new approach for identification of underlying risk factors for coronary artery disease in essential hypertensive patients. *Scientific reports*, 7, 43965. <https://doi.org/10.1038/srep43965>
- [42] Software Engineering, Evolutionary Model (2019) [Online] www.geeksforgeeks.org. Dostupné na internete: <<https://www.geeksforgeeks.org/software-engineering-evolutionary-model/>>.
- [43] Moscow Prioritisation [Online] www.agilebusiness.org. Dostupné na internete: <https://www.agilebusiness.org/page/ProjectFramework_10_MoSCoWPrioritisation>.
- [44] What is Use Case Diagram? [Online] www.visual-paradigm.com. Dostupné na internete: <<https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/>>.
- [45] Unified Modeling Language (2019) [Online] www.geeksforgeeks.org. Dostupné na internete: <<https://www.geeksforgeeks.org/unified-modeling-language-uml-introduction/>>.
- [46] Lohaj, O.: Metódy dátovej analytiky pre analýzu vplyvu faktorov na včasné zachytenie kardiovaskulárneho rizika, diplomová práca, Technická univerzita v Košiciach, 2013, Košice, Slovenská republika [Online] Dostupné na internete: <<https://opac.crzp.sk/?fn=detailBiblioForm&sid=85C8718C4BC2C28F5F436CCAA842&seo=CRZP-detail-kniha>>

- [47] How To Use The System Usability Scale (SUS) To Evaluate The Usability Of Your Website
[Online] www.usabilitygeek.com. Dostupné na internete: <
<https://usabilitygeek.com/how-to-use-the-system-usability-scale-sus-to-evaluate-the-usability-of-your-website/>>.
- [48] Pavlíček, A.: Analýza tém, použitých metod a dosiahnutých výsledkov záverečných prác,
diplomová práca, Technická univerzita v Košiciach, 2013, Košice, Slovenská republika
[Online] Dostupné na internete: <
<https://opac.crzp.sk/?fn=detailBiblioForm&sid=AFB64E0160B4F4E92249D19F9648&seo=CRZP-detail-kniha>>

Prílohy

Príloha A: Dotazník s úlohami pre testovanie aplikácie na podporu výskumu pre lekárov

Príloha B: Dotazník s úlohami pre vyhodnotenie použiteľnosti aplikácie

Príloha C: Používateľská príručka

Príloha D: Systémová príručka

Príloha E: CD médium

TECHNICKÁ UNIVERZITA V KOŠICIACH
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

POPISNÉ MODELY ŠPECIFICKÝCH SKUPÍN PACIENTOV
VZHĽADOM NA ICH KARDIOVASKULÁRNE RIZIKO
Dotazník s úlohami pre testovanie aplikácie na podporu výskumu
pre lekárov

Príloha A

Študijný program: Hospodárska informatika
Študijný odbor: Informatika
Školiace pracovisko: KKUI
Školiteľ: prof. Ing. Ján Paralič, PhD.
Konzultant: Ing. Zuzana Pella
Ing. Oliver Lohaj

Obsah

1. Základné informácie.....	94
2. Dotazník	95

1. Základné informácie

V rámci prvej kapitoly sa zameriame na niekoľko základných informácií o dotazníku, ktorý bol využitý pre potreby otestovania implementovanej aplikácie. V prvom rade je nutné poznamenať, že dotazník svojim obsahom, a teda tým, ako sú dané úlohy postavené, pokrýva všetky dôležité časti aplikácie. A teda:

- sekciu pre hľadanie asociačných pravidiel,
- sekciu pre aplikovanie metód zhlukovania,
- sekciu pre hľadanie asociačných pravidiel v rámci definovaných zhlukov.

Spôsob, akým je znenie daných úloh formulované, odpovedá obsadeniu primárnej cieľovej skupiny aplikácie, ktorú tvoria najmä lekári, resp. ľudia z lekárskeho prostredia. Dotazník obsahuje dve úlohy, pričom tá druhá je rozvetvená na dva celky. Testovaný ma na výber z pomedzi troch odpovedí, avšak len jedna z nich je správna.

2. Dotazník

Náplňou druhej kapitoly je celé znenie a formulácia dotazníku, ktorý bol využitý pri testovaní implementovanej aplikácie.

Dotazník:

Úlohy pre testovanie aplikácie na podporu výskumu pre lekárov

Náš súbor pozostáva z pacientov, ktorí boli hospitalizovaní na 1. kardiologickej klinike VÚSCH. Pacienti boli prijatí v plánovanom režime a všetci absolvovali okrem laboratórneho skríningu a EKG vyšetrenia aj selektívnu koronarografiu (invazívne zobrazovacie vyšetrenie koronárnych artérií). U niektorých pacientov sme doplnili aj ECHO vyšetrenie srdca. Zároveň sme sa zamerali aj na komplexný zber dát v zmysle anamnézy a fyzikálneho vyšetrenia. Do nášho súboru sme vyberali pacientov s podozrením na prítomnosť koronárnej choroby srdca na základe vybraných rizikových faktorov (modifikovateľných aj nemodifikovateľných).

Získaný súbor pacientov prešiel rôznymi formami spracovania – od anonymizácie pacientov (na základe pridelenia identifikačného čísla za účelom párovania údajov z ďalších možných vyšetrení), cez transformáciu údajov do štruktúrovanej podoby, čistenie a diskretizáciu až po vytváranie nových atribútov, dôležitých z kardiologického hľadiska a dopĺňanie chýbajúcich údajov.

Dáta popisujúce súbor pacientov môžeme následne analyzovať s cieľom získať nové korelácie a vzťahy známych rizikových faktorov kardiovaskulárnych ochorení (KVO) a zároveň zistiť aj sily významnosti jednotlivých sledovaných rizikových faktorov KVO (menej známe, nie často asociované s aterosklerózou).

Počas vykonávania analýz budeme pracovať výlučne s týmto dátovým súborom. Postupne prejdeme 3 typy úloh, ktoré ponúka testovaná aplikácia.

1. Generovanie asociačných pravidiel

V rámci prvej úlohy sa zameriame na generovanie a získanie asociačných pravidiel (AP).

Na úvod je potrebné načítať predpripravenú dátovú množinu. Ďalej budeme postupovať podľa nasledovných krokov:

1. Zachováme pôvodné zloženie atribútov, teda žiadne z nich nebudú odstránené.
2. Vyfiltrujeme si iba mužov v produktívnom veku (18-50 rokov), fajčiarov (kat. 1) a exfajčiarov (kat. 2), s hodnotou LDL nad 2.6 mmol/l a srdcovou frekvenciou (EKG_FR) nad 80.¹

3. Parametre algoritmu pre generovanie AP v rámci našej úlohy je nutné nastaviť na hodnoty:

- o minimálna podpora: 0.1,
- o minimálna spoľahlivosť: 0.95,
- o minimálna dĺžka ľavej strany pravidla: 5.

4. Taktiež musíme zabezpečiť, aby sa na pravej strane asociačných pravidiel (RHS) nevyskytovali atribúty popisujúce osobnú a rodinnú anamnézu (nakoľko ide o subjektívne zhodnotenie pacienta pri ich odbere), okrem osobnej anamnézy popisujúcej vysoký krvný tlak. Rovnako vylúčime atribúty popisujúce zlovyk pitia alkoholu a ECHO nálezov (jeden typ nadobúdanej hodnoty).

5. V rámci vygenerovaných AP vyberieme len tie, ktoré majú na pravej strane (RHS) výraz odpovedajúci vysokému kardiovaskulárnemu riziku (hodnoty ESC).

6. Zameriame sa na AP popisujúce prítomnosť ischemickej choroby srdca, prekonaného akútneho koronárneho syndrómu (IM) a dokumentovanú hypertenziu s možným predĺženým QRS komplexom.

Môžeme konštatovať, že získané frekventované množiny (LHS) poukazujú na zvýšené kardiovaskulárne riziko?

- a. Nie, hodnoty ESC sú pomerne nízke, teda poukazujú na nízke KV riziko.
- b. Hodnoty ESC sú mierne vyššie, avšak nie signifikantné.
- c. AP poukazuje na asociáciu frekventovaných množín a vysokého KV rizika.

2. Aplikovanie metódy zhlukovania s následným vytvorením AP na získaných zhlukoch

Náplň druhej úlohy spočíva v aplikovaní metódy zhlukovania na dáta.

Našou úlohou je opäť overenie známych RF aterosklerózy ako sú napr. vysoký tlak, vysoký cholesterol a taktiež aj možné nové RF (hladina fibrinogénu). Zároveň máme za cieľ zhodnotiť význam hladiny HDL cholesterolu, keďže doposiaľ v klinických štúdiách bolo nejednoznačné, či zvýšená hladina HDL cholesterolu má kardioprotektívny účinok. Postupovať budeme podľa nasledujúcich krokov:

1. Pracovať budeme len s určitým výberom atribútov, t.j. nakoľko rodinná a osobná anamnéza nie je vždy kompletná alebo presná, rozhodli sme sa ich vylúčiť z tejto analýzy. Naše rozhodnutie je podporené aj spôsobom odberu rodinnej

a osobnej anamnézy – na základe subjektívneho zhodnotenia pacienta. Podobne je to s udaním návyku na alkohol, kedy pacienti túto skutočnosť majú tendenciu zamlčať. Hodnoty sledovaných parametrov EKG rytmu a T vlny nadobúdajú v datasete prevažne jeden typ hodnoty, teda nie sú pre nás použiteľné. (Počet atribútov vo výbere bude 30)

2. Metódu hierarchického zhukovania nastavíme na „ward“.
3. Pre zistenie optimálneho počtu zhukov použijeme metódu siluety, kde najvyššie číslo udáva najvhodnejší počet zhukov pre daný dataset. (10)
4. Následne môžeme výsledok nechať vykresliť výsledok zhukovania.

Na základe výberu našich RF sme sa zamerali na zhuk, kde je najväčšia predispozícia na prítomnosť koronárnej choroby srdca – teda zameriavame sa na skupinu obéznych fajčiarov (kat. 1 pre atribút Fajcenie) s vysokým LDL cholesterolom a celkovým vysokým kardiovaskulárnym rizikom. (zhuk 7) Výsledok zhukovania odošleme k tvorbe AP na vytvorených zhukoch:

1. Nie je potrebné filtrovať žiadne zhuky.
2. Parametre pre AP určíme nasledovne:
 - o minimálna podpora: 0.09,
 - o minimálna spoľahlivosť: 0.95,
 - o minimálna dĺžka ľavej strany pravidla: 4.
3. Nakoľko pohlavie a vek patria medzi nemodifikovateľné RF, nie je pre nás podstatná ich účasť na dôsledkovej strane pravidiel (RHS). Rovnako nie je pre nás v tejto fáze zaujímavý ani výsledok EKG nálezu.
4. Z vyexportovaných pravidiel sa zameriame na tie, ktoré obsahujú na ľavej strane (LHS) výsledok koronarografického nálezu s hemodynamicky závažnou stenózou (kat. 4 pre atribút Nalez). Nakoľko sme od úvodu smerovali k skúmaniu kardioprotektívneho účinku HDL vzhľadom k ateroskleróze, zameriame sa na tie AP, ktoré na pravej strane (RHS) odkazujú na vysokú hladinu HDL cholesterolu.
5. Výsledkom zvolených krokov a nastavených parametrov je vyfiltrovanie 3 AP, ktoré popisujú kombináciu vysokého tlaku, obezity, vysokého LDL cholesterolu (pričom ide prevažne o skupinu fajčiarov). Zároveň títo pacienti mali významnú stenózu na koronárnych artériách, čím sa potvrdila správnosť RF aterosklerózy. Zároveň tieto pravidlá poukazujú na vyššie hodnoty HDL cholesterolu.

Môžeme v kontexte obsiahnutých frekventovaných množín (LHS) potvrdiť alebo vyvrátiť teóriu kardioprotektívneho účinku HDL cholesterolu vzhľadom k ateroskleróze? (Pri hodnotení sa zameriame aj na počty pokrytých prípadov).

Potvrdzujeme / Nie je možné potvrdiť / Vyvraciame teóriu kardioprotektívneho účinku HDL cholesterolu vzhľadom k ateroskleróze.

TECHNICKÁ UNIVERZITA V KOŠICIACH
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

POPISNÉ MODELY ŠPECIFICKÝCH SKUPÍN PACIENTOV
VZHĽADOM NA ICH KARDIOVASKULÁRNE RIZIKO
Dotazník s otázkami pre vyhodnotenie použiteľnosti aplikácie

Príloha B

Študijný program: Hospodárska informatika
Študijný odbor: Informatika
Školiace pracovisko: KKUI
Školiteľ: prof. Ing. Ján Paralič, PhD.
Konzultant: Ing. Zuzana Pella
 Ing. Oliver Lohaj

2022 Košice

Bc. Adam Pavlíček

Obsah

1. Základné informácie.....	101
2. Dotazník	102

1. Základné informácie

Pre vyhodnotenie použiteľnosti aplikácie sme využili rámec desiatich otázok, ktorý je taktiež známy pod anglickým názvom *System usability scale*. Tento prístup poskytuje pohľad na subjektívne hodnotenie použiteľnosti aplikácie.

Navrhol ho John Brooke v Digital Equipment Corporation v Spojenom Kráľovstve, v roku 1986 ako prostriedok na ohodnotenie použiteľnosti elektronických kancelárskych systémov.

Vďaka tomuto prístupu je možné ohodnotiť hneď niekoľko aspektov, medzi ktorými sú:

- efektívnosť („Dokážu používatelia úspešne dosiahnuť svoj cieľ?“),
- výkonnosť („Koľko úsilia a koľko zdrojov je potrebné vynaložiť pre dosiahnutie daného cieľa?“),
- spokojnosť („Bola skúsenosť, pri dosahovaní daného cieľa uspokojivá?“).

Spôsob, akým sa daný dotazník vyhodnocuje sme bližšie charakterizovali a vysvetlili v hlavnej časti diplomovej práce, v šiestej kapitole.

Niekoľko ďalších informácií, ako aj znenie otázok v anglickom jazyku je možné nájsť na nasledujúcom odkaze - <https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html>.

2. Dotazník

Náplňou druhej kapitoly je celé znenie a formulácia dotazníku, ktorý bol využitý pri testovaní použiteľnosti implementovanej aplikácie.

Dotazník:

Otázky ohľadom použiteľnosti aplikácie SPPRK.

1. Myslím si, že by som chcel/a využívať tento systém často.

Vôbec nesúhlasím

Úplne súhlasím

1	2	3	4	5

2. Myslím si, že tento systém je zbytočne zložitý.

Vôbec nesúhlasím

Úplne súhlasím

1	2	3	4	5

3. Myslím si, že tento systém bol jednoduchý na použitie.

Vôbec nesúhlasím

Úplne súhlasím

1	2	3	4	5

4. Myslím si, že by som potreboval pomoc technika/informatika aby som mohol/la používať tento systém bez problémov.

Vôbec nesúhlasím

Úplne súhlasím

1	2	3	4	5

5. Myslím si, že viacero funkcií v tomto systéme sú dobre integrované.

Vôbec nesúhlasím

Úplne súhlasím

1	2	3	4	5

6. Myslím si, že tento systém je nekonzistentný.

Vôbec nesúhlasím

Úplne súhlasím

1	2	3	4	5

7. Myslím si, že mnoho ľudí by sa naučilo používať tento systém veľmi rýchlo.

Vôbec nesúhlasím

Úplne súhlasím

1	2	3	4	5

8. Myslím si, že tento systém je ťažkopádny na použitie.

Vôbec nesúhlasím

Úplne súhlasím

1	2	3	4	5

9. Cítil som sa sebavedomo pri používaní tohto systému.

Vôbec nesúhlasím

Úplne súhlasím

1	2	3	4	5

10. Musel som sa naučiť mnoho vecí pred použitím tohto systému.

Vôbec nesúhlasím

Úplne súhlasím

1	2	3	4	5

TECHNICKÁ UNIVERZITA V KOŠICIACH
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

POPISNÉ MODELY ŠPECIFICKÝCH SKUPÍN PACIENTOV
VZHĽADOM NA ICH KARDIOVASKULÁRNE RIZIKO
Používateľská príručka

Príloha C

Študijný program: Hospodárska informatika
Študijný odbor: Informatika
Školiace pracovisko: KKUI
Školiteľ: prof. Ing. Ján Paralič, PhD.
Konzultant: Ing. Zuzana Pella
Ing. Oliver Lohaj

Obsah

1. Úvod do používateľskej príručky	107
2. Systémové požiadavky	108
3. Načítanie dát	109
4. Sekcia pre hľadanie asociačných pravidiel	110
4.1. Výber atribútov zo vstupnej dátovej množiny	110
4.2. Filtrovanie záznamov.....	111
4.3. Metriky pre hľadanie asociačných pravidiel.....	113
4.4. Obmedzenie pravej strany asociačných pravidiel	113
4.5. Potvrdenie všetkých úkonov	114
4.6. Prezентация výsledkov	115
5. Sekcia pre aplikovanie metód zhukovania	119
5.1. Výber atribútov zo vstupnej dátovej množiny	119
5.2. Výber metódy hierarchického zhukovania	119
5.3. Vygenerovanie dendrogramu.....	120
5.4. Odhad optimálneho počtu zhukov	120
5.5. Počet zhukov ako parameter metódy zhukovania	122
6.3. Výber v danom kroku nie je potrebné nijak potvrdiť.	122
5.6. Potvrdenie všetkých úkonov	122
5.7. Prezентация výsledkov	123
6. Hľadanie asociačných pravidiel v rámci definovaných zhukov.....	126
6.1. Podrobné informácie o zhukoch	126
6.2. Výber zhukov.....	127
6.3. Nadväzujúce úkony	127
6.4. Prezентация výsledkov	127

Zoznam obrázkov

Obrázok 1 Hlavné menu aplikácie	107
Obrázok 2 Výber atribútov zo vstupnej dátovej množiny	111
Obrázok 3 Filtrovanie záznamov	112
Obrázok 4 Metriky pre hľadanie asociačných pravidiel	113
Obrázok 5 Obmedzenie pravej strany asociačných pravidiel.....	114
Obrázok 6 Potvrdenie všetkých úkonov.....	114
Obrázok 7 Získané výsledky.....	115
Obrázok 8 Vygenerované asociačné pravidlá.....	115
Obrázok 9 Možnosti vizualizácie bodového grafu.....	116
Obrázok 10 Bodový graf	116
Obrázok 11 Používateľské rozhranie pri vizualizácii vzťahov medzi asociačnými pravidlami	117
Obrázok 12 Vizualizácia znázorňujúca vzťahy medzi asociačnými pravidlami	117
Obrázok 13 Výber metódy hierarchického zhľukovania.....	119
Obrázok 14 Generovanie dendrogramu.....	120
Obrázok 15 Výber metódy pre odhad optimálneho počtu zhľukov (1).....	121
Obrázok 16 Výber metódy pre odhad optimálneho počtu zhľukov (2).....	121
Obrázok 17 Výber počtu zhľukov	122
Obrázok 18 Potvrdenie všetkých úkonov.....	123
Obrázok 19 Odkazy na panely so získanými výsledkami	123
Obrázok 20 Používateľské rozhranie pre vizualizáciu „Parallel plot“	124
Obrázok 21 Zatriedené záznamy do zhľukov.....	125
Obrázok 22 Úvodné informácie o dátovej množine.....	126
Obrázok 23 Informácie o jednotlivých zhľukoch	127
Obrázok 24 Hodnoty vyskytujúce sa na ľavej strane asociačných pravidiel.....	128
Obrázok 25 Výber zhľuku asociačných pravidiel pre vizualizáciu	128

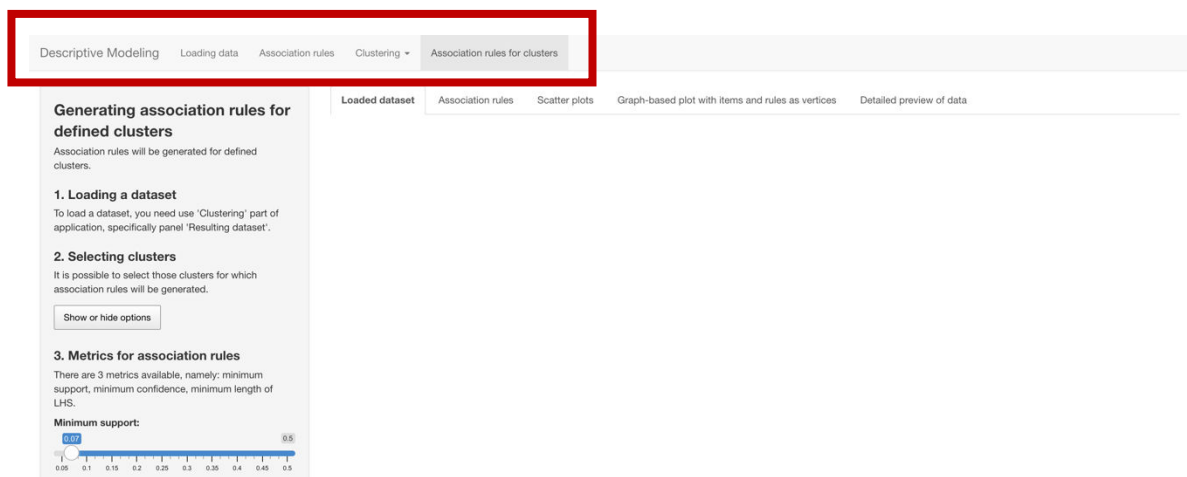
1. Úvod do používateľskej príručky

Úlohou nasledujúcej prílohy je používateľovi priblížiť a objasniť úkony a kroky, ktoré je potrebné vykonať pre správny chod implementovaného systému, a rovnako tak pre dosiahnutie požadovaných výsledkov.

Tak ako je uvedené v kapitole, ktorá je súčasťou hlavnej časti diplomovej práce, zameranej na charakteristiku jednotlivých vstupov a výstupov, naša aplikácia pozostáva z troch hlavných sekcií, a to:

- sekcia pre hľadanie asociačných pravidiel,
- sekcia aplikovanie metód zhukovania,
- sekcia pre hľadanie asociačných pravidiel v rámci definovaných zhukov.

Hlavnej časti aplikácie však ešte predchádza tá, ktorá je určená pre načítanie vstupných dát. Prechod medzi jednotlivými sekciami je uskutočňovaný prostredníctvom záložiek, ktoré sa nachádzajú v hornej časti používateľského rozhrania aplikácie (Obrázok 33).



Obrázok 33 Hlavné menu aplikácie

V jednotlivých podkapitolách používateľskej príručky sa teda zameriame na vyššie spomenutú charakteristiku, a to pre každú zo spomínaných častí zvlášť. Používateľské prostredie každej z týchto sekcií je totižto iné, s inými vstupmi a výstupmi, aj keď disponujú niekoľkými spoločnými elementami. Prezentovaný odporúčaný postup aplikáciou bude dopĺňaný taktiež obrázkami, pre lepšiu orientáciu a pochopenie. Je nutné podotknúť, že v používateľskej príručke nebudeme bližšie špecifikovať jednotlivé elementy, keďže tie sme charakterizovali v hlavnej časti diplomovej práce. Taktiež je dôležité poznamenať, že niektoré časti aplikácie nie je nutné spustiť v takom stave, v akom budú opísané. Niektoré časti aplikácie nie je potrebné spustiť vôbec, avšak to a všetky ostatné dôležité informácie uvedieme v nasledujúcich kapitolách a podkapitolách.

2. Systémové požiadavky

Druhá kapitola používateľskej príručky bude venovaná základným systémovým požiadavkám, ktoré sú potrebné pre plynulý a bezproblémový chod implementovanej aplikácie.

Aplikácia je spustiteľná lokálne, pomocou programu *R Studio*, pričom je potrebné, aby boli splnené nasledujúce minimálne požiadavky:

- počítač alebo notebook s vhodným operačným systémom (aplikácia je otestovaná na operačnom systéme *macOS Monterey*, verzia 12.2 a na operačnom systéme *Microsoft Windows 10*) a s dostatočnou operačnou pamäťou RAM (je potrebná operačná pamäť RAM s veľkosťou minimálne 4GB),
- programovací jazyk R (aplikácia bola vyvíjaná a otestovaná vo verzií 4.0.2),
- *R Studio* (aplikácia bola vyvíjaná a otestovaná vo verzií 1.3.1073),
- knižnice v *R Studio* (konkrétne sa jedná o knižnice *shiny*, *DT*, *readxl*, *openxlsx*, *arules*, *rCBA*, *dplyr*, *stringi*, *stringr*, *writexl*, *arulesViz*, *ggplot2*, *rapport*, *shinyjs*, *Hmisc*, *dplyr*, *shinycssloaders*, *shinyWidgets*, *clustMixType*, *cluster*, *factorextra*, *compareGroups*, *fpc*, *Rtsne*, *gtsummary*, *gt*, *mltools*, *data.table*, *visNetwork*, *plotly*, *hrbrthemes*, *patchwork*, *GGally*, *viridis*, *rvest*, *dendextend*, *ggdendro*),
- nainštalovaná správna verzia Java (32 alebo 64 bitová) a jej prislúchajúce knižnice v *RStudio*.

3. Načítanie dát

Podrobnej charakteristike časti aplikácie pre načítanie vstupných dát, ako aj jednotlivým krokom, ktoré je v nej nutné vykonať, sme sa venovali v hlavnej časti diplomovej práce, v podkapitole 4.1.

4. Sekcia pre hľadanie asociačných pravidiel

V rámci tretej kapitoly používateľskej príručky sa zameriame na charakteristiku jednotlivých krokov, ktoré je potrebné a možné vykonať v časti implementovanej aplikácie pre hľadanie asociačných pravidiel.

4.1. Výber atribútov zo vstupnej dátovej množiny

V prvom kroku sa používateľ stretáva s časťou používateľského prostredia, ktorá je zameraná na výber atribútov zo vstupnej dátovej množiny. Zvolené atribúty, resp. prispôbena dátová množina bude využitá pri všetkých nasledujúcich aktivitách v rámci danej sekcie.

Ako je možné vidieť na Obrázok 34, v tejto časti sa okrem iných elementov, nachádzajú aj dve tlačidlá, ktoré sú prezentované textom „*Show or hide options*“ a „*Show data*“.

Prvé z nich, „*Show or hide options*“ slúži na odkrytie, resp. zakrytie niekoľkých ďalších elementov, konkrétne dvoch zaškrŕavacích polí a jedného poľa slúžiaceho pre výber možností z ponúkaného zoznamu. Používateľ si vie vybrať, či chce pracovať s celou dátovou množinou („*All dataset!*“), a teda so všetkými atribútmi, ktoré sa v nej nachádzajú, alebo si zvoliť len niektoré z nich („*Select attributes!*“). Pri prvom scenári nemusí robiť nič, nakoľko možnosť výberu práce s celou dátovou množinou je prednastavená. V prípade, že sa však rozhodne vybrať si len istú podmnožinu atribútov, resp. dát, je potrebné aby používateľ vybral druhé zaškrŕavacie pole, ktoré nesie názov „*Select attributes!*“. Po jeho výbere sa sprístupní pole s názvami atribútov, ktoré sa v načítanej dátovej množine nachádzajú. Z neho si používateľ môže vybrať len tie, ktoré sú pre neho dôležité a ktoré by chcel využívať aj pri ďalších činnostiach v rámci danej sekcie. Jednotlivé bunky reprezentujúce názvy stĺpcov z dátovej množiny, je možné odstrániť pomocou spätnej klávesy, resp. klávesy „*backspace*“. Výber už odstráneného prvku je realizovaný kliknutím na prázdne miesto daného poľa a zvolením hodnoty z ponúkaného rozbaľovacieho menu. Daná funkcia bola prispôbena tak, aby nebolo možné zvoliť si menej než dva atribúty, teda menej než dva rôzne prvky z poľa.

Associations rules

The Apriori algorithm will be used to generate association rules.

1. Choosing dataset

It is possible to choose input attributes or work with the whole dataset.

Show or hide options

☒ All dataset!

☐ Select attributes!

Attributes (the minimum number of selected attributes is 2):

Pohlavie Vek ESC R_ICHS R_CMP R_IM
 R_Hypertenzia R_DM R_AoS O_ICHS O_CMP
 O_IM O_Hyperchol O_Hypertenzia O_DM
 O_AoS Fajcenie Alkohol BMI Tlak Urea Kreat
 AST Na K Chol Trigl HDL LDL CRP CL
 FBG EKG_FR EKG_Rytmus EKG_PQ
 EKG_QRS EKG_QT EKG_BPTR EKG_KES
 EKG_SVES EKG_STD EKG_STE EKG_T
 ECHO_EF ECHO_PH Nalez

Show data

Data filtering | Association rules | Scatter plots | Graph-based plot with items and rules as vertices | Selected data

Obrázok 34 Výber atribútov zo vstupnej dátovej množiny

Ak je používateľ stotožnený so zložením atribútov danej dátovej množiny, svoj výber potvrdí kliknutím na tlačidlo „Show data“, vďaka čomu sa presunie do ďalšej časti aplikácie. Ako už bolo spomenuté, možnosť pre prácu s celou dátovou množinou je prednastavená, čo znamená, že ak by sa používateľ rozhodol rovno kliknúť na tlačidlo „Show data“, aplikácia ho automaticky presunie do ďalšej časti s obsahom dátovej množiny v pôvodnom zložení, v akom bola do aplikácie načítaná.

4.2. Filtrovanie záznamov

Po výbere atribútov zo vstupnej dátovej množiny sa dostávame k ďalšiemu kroku, v rámci ktorého dochádza k filtrovaniu samotných záznamov. V prípade, ak by používateľ chcel pri nasledujúcich činnostiach v danej sekcii aplikácie pracovať len so špecifickou podmnožinou záznamov, ich výber je možný vykonať práve v tomto bode.

2. Filtering data

You can use the filter below to select records. If you want to work with the whole data set, leave the filter empty. Confirm your decision with the 'Confirm' button.

Show 15 entries

	Pohlavie	Vek	ESC	R_ICHS	R_CMP	R_IM	R_Hypertenzi
		All	All	All	All	All	All
1	Muz	[73,88]	[5,7]	true	false	true	false
2	Zena	[73,88]	[7,33]	true	false	true	false
3	Muz	[73,88]	[7,33]	false	false	false	false
4	Zena	[73,88]	[7,33]	false	false	false	false
5	Muz	[36,58]	[5,7]	false	false	false	false
6	Muz	[36,58]	[0,5]	false	false	false	false
7	Zena	[58,63]	[0,5]	false	false	false	false
8	Zena	[63,68]	[5,7]	false	false	false	false
9	Muz	[73,88]	[7,33]	true	false	false	false
10	Muz	[36,58]	[0,5]	false	false	false	false
11	Zena	[63,68]	[5,7]	false	true	false	false
12	Zena	[63,68]	[0,5]	false	false	false	false
13	Muz	[36,58]	[0,5]	true	false	true	false
14	Zena	[73,88]	[7,33]	true	false	true	false
15	Muz	[58,63]	[0,5]	false	false	false	false

Showing 1 to 15 of 764 entries

Previous
1
2
3
4
5
...
51
Next

Obrázok 35 Filtrovanie záznamov

Ako je možné vidieť na obrázku vyššie (Obrázok 35) filtrovanie prebieha prostredníctvom lišty, ktorá sa nachádza nad jednotlivými záznamami umiestnenými v tabuľke. Pre filtrovanie je potrebné kliknúť na prázdne pole pri príslušnom atribúte v spomínanej lište. Používateľ dostane na výber možnosti, ktoré sú vlastne triedy daného atribútu, a to vo forme elementov nachádzajúcich sa v rozbaľovacom zozname, čo je vidieť aj na obrázku vyššie. Množstvo tried, ktoré si je používateľ schopný vybrať nie je nijak obmedzené. V prípade, ak používateľ možnosť filtrovania nijak nevyužije, obsah dátovej množiny ostane nezmenený, a v takom zložení bude využitý pri ďalších činnostiach v sekcii pre hľadanie asociačných pravidiel.

Ak je používateľ stotožnený s výberom záznamov, svoje rozhodnutie potvrdí kliknutím na tlačidlo „Confirm!“, čím sa celý proces opäť posunie o krok vpred. V prípade, ak by používateľ chcel vybrané filtre odstrániť, môže tak urobiť kliknutím na spätnú klávesu, tzv. klávesu „backspace“, po označení zvoleného filtračného kritéria v poli pri príslušnom atribúte, alebo to môže jednoducho urobiť kliknutím na tlačidlo „Reset all filters!“, čím sa však vymažú všetky zvolené filtračné kritéria a tabuľka sa vráti do pôvodného stavu.

4.3. Metriky pre hľadanie asociačných pravidiel

Po potvrdení zloženia záznamov dátovej množiny, systém automaticky používateľovi ponúkne niekoľko základných informácií súvisiacich s danou množinou údajov. Tento výstup sme podrobne charakterizovali v hlavnej časti diplomovej práce. V rámci používateľskej príručky je však nutné podotknúť, že nie je nutné s informáciami nijak interagovať. Ak je však používateľ s nimi stotožnený, je možné pristúpiť k tretiemu kroku, ktorým je výber hodnôt metrík pre hľadanie asociačných pravidiel. Tak ako sme charakterizovali v hlavnej časti diplomovej práce, jedná sa o minimálnu podporu, spoľahlivosť a dĺžku ľavej strany pravidla.

3	[58,63]	[0,5]	false	false	false	false	false
4	[63,68]	[5,7]	false	false	false	false	false
5	[63,68]	[5,7]	false	true	false	false	false
6	[63,68]	[0,5]	false	false	false	false	false
7	[73,88]	[7,33]	true	false	true	false	false
8	[36,58]	[0,5]	false	false	false	true	true
9	[58,63]	[5,7]	false	false	false	false	false
10	[58,63]	[5,7]	true	false	true	true	false
11	[73,88]	[5,7]	false	false	false	false	false
12	[68,73]	[7,33]	false	false	false	false	false
13	[73,88]	[7,33]	false	false	false	false	false
14	[58,63]	[5,7]	false	false	false	false	false
15	[73,88]	[5,7]	false	false	false	false	false

Obrázok 36 Metriky pre hľadanie asociačných pravidiel

Ako je možné vidieť na Obrázok 36, spomínané metriky sú prezentované formou numerických posúvačov, „sliderov“, s jedným posuvným interval ohraničujúcim bodom. Platná je vždy tá časť numerickej mierky, ktorá je zvýraznená modrou farbou. Význam jednotlivých metrík sme charakterizovali v prvej kapitole našej diplomovej práce. Výber týchto hodnôt nie je potrebné nijak potvrdiť. Používateľ ich môže nechať v pôvodnom tvare, ktorý je možné vidieť aj na Obrázok 36, alebo ich upraviť podľa vlastného uváženia.

Ak je s používateľ s výberom stotožnený, je možné pristúpiť k nasledujúcej kroku.

4.4. Obmedzenie pravej strany asociačných pravidiel

V poradí štvrtý krok je zameraný na obmedzenie výskytu atribútov na pravej strane nájdených asociačných pravidiel. Ak si je používateľ istý tým, že nemá záujem o asociačné pravidlá, na pravej strane ktorých sa vyskytnú hodnoty špecifických atribútov, môže toto obmedzenie definovať práve v tejto časti. Výber atribútov funguje rovnako ako pri prvom kroku - odstrániť názvy stĺpcov je možné pomocou späťnej klávesy, tzv. klávesy „backspace“. Ich výber je realizovateľný výberom

z rozbaľovacieho menu, ktoré sa zobrazí po kliknutí do prázdnej časti daného elementu, poľa, ktoré znázorňuje Obrázok 37. Je však nutné pripomenúť, že používateľ má na výber len tie názvy stĺpcov, ktoré sa nachádzajú už v upravenej dátovej množine. Ak teda niektoré atribúty už používateľ odstránil v predchádzajúcich krokoch, logicky sa nevyskytnú ani v tejto časti. Pole môže ostať taktiež prázdne, v prípade, že používateľ nechce asociačné pravidlá nijak obmedzovať. Výber taktiež nie je nutné nijak potvrdiť.

3. Metrics for association rules
There are 3 metrics available, namely: minimum support, minimum confidence, minimum length of LHS.

Minimum support: 0.05 0.1 0.15 0.2 0.25 0.3 0.35 0.4 0.45 0.5

Minimum confidence: 0.5 0.55 0.6 0.65 0.7 0.75 0.8 0.85 0.9 0.95 1

Minimum length of LHS: 3 4 5 6 7 8 9 10

4. Restriction of RHS
The selected attributes will not appear on the right-hand-side of the association rules.

Attributes:
Vek Fajenie FBG EKG_Rythmus EKG_T

5. Get results!
You can get the results by clicking on the 'Render!' button. It can take seconds to minutes to generate association rules. Be patient, please!

Render!

3	[58,63]	[0,5]	false	false	false	false	false
4	[63,68]	[5,7]	false	false	false	false	false
5	[63,68]	[5,7]	false	true	false	false	false
6	[63,68]	[0,5]	false	false	false	false	false
7	[73,88]	[7,33]	true	false	true	false	false
8	[36,58]	[0,5]	false	false	false	true	true
9	[58,63]	[5,7]	false	false	false	false	false
10	[58,63]	[5,7]	true	false	true	true	false
11	[73,88]	[5,7]	false	false	false	false	false
12	[68,73]	[7,33]	false	false	false	false	false
13	[73,88]	[7,33]	false	false	false	false	false
14	[58,63]	[5,7]	false	false	false	false	false
15	[73,88]	[5,7]	false	false	false	false	false

Showing 1 to 15 of 324 entries

Previous 1 2 3 4 5 ... 22 Next

Obrázok 37 Obmedzenie pravej strany asociačných pravidiel

4.5. Potvrdenie všetkých úkonov

Na záver je potrebné potvrdiť zvolené kritéria a dať implementovanému systému pokyn pre vykonanie všetkých dostupných analýz v rámci danej sekcie aplikácie, a to kliknutím na tlačidlo „Render!“, ktoré je taktiež znázornené na obrázku nižšie (Obrázok 38).

3. Metrics for association rules
There are 3 metrics available, namely: minimum support, minimum confidence, minimum length of LHS.

Minimum support: 0.05 0.1 0.15 0.2 0.25 0.3 0.35 0.4 0.45 0.5

Minimum confidence: 0.5 0.55 0.6 0.65 0.7 0.75 0.8 0.85 0.9 0.95 1

Minimum length of LHS: 3 4 5 6 7 8 9 10

4. Restriction of RHS
The selected attributes will not appear on the right-hand-side of the association rules.

Attributes:
Vek Fajenie FBG EKG_Rythmus EKG_T

5. Get results!
You can get the results by clicking on the 'Render!' button. It can take seconds to minutes to generate association rules. Be patient, please!

Render!

3	[58,63]	[0,5]	false	false	false	false	false
4	[63,68]	[5,7]	false	false	false	false	false
5	[63,68]	[5,7]	false	true	false	false	false
6	[63,68]	[0,5]	false	false	false	false	false
7	[73,88]	[7,33]	true	false	true	false	false
8	[36,58]	[0,5]	false	false	false	true	true
9	[58,63]	[5,7]	false	false	false	false	false
10	[58,63]	[5,7]	true	false	true	true	false
11	[73,88]	[5,7]	false	false	false	false	false
12	[68,73]	[7,33]	false	false	false	false	false
13	[73,88]	[7,33]	false	false	false	false	false
14	[58,63]	[5,7]	false	false	false	false	false
15	[73,88]	[5,7]	false	false	false	false	false

Showing 1 to 15 of 324 entries

Previous 1 2 3 4 5 ... 22 Next

Obrázok 38 Potvrdenie všetkých úkonov

4.6. Prezentácia výsledkov

Po vykonaní všetkých dostupných analýz a úkonov má používateľ k dispozícii niekoľko výsledkov, ktoré sme bližšie charakterizovali v piatej kapitole diplomovej práce. Tu je však nutné zdôrazniť, že prechod medzi nimi prebieha výberom možností nachádzajúcich sa v lište, v hlavnej časti obrazovky, tak ako to je vizualizované na nasledujúcom obrázku (Obrázok 39).



Obrázok 39 Získané výsledky

Jednotlivé časti nájdených asociačných pravidiel je možné filtrovať identickým spôsobom, ktorý sme charakterizovali aj v podkapitole 4.2.2, v hlavnej časti diplomovej práce. Špecifický prípad je stĺpec obsahujúci ľavú stranu asociačných pravidiel (konkrétne máme na mysli stĺpec „LHS“, ktorý je možné vidieť aj na obrázku nižšie), pri ktorom filtrovaní je potrebné hľadaný výraz do príslušného poľa napísať.

Používateľ má taktiež možnosť vyexportovať všetky nájdené asociačné pravidlá z aplikácie, a to po kliknutí na tlačidlo „Download table“, ktoré je znázornené na Obrázok 40 a následným výberom z možností „Excel“ alebo „CSV“. Ako je zrejmé, jedná sa o formu výstupu, a teda výstup v podobe .xlsx alebo .csv súboru. Taktiež je dostupná možnosť stiahnuť len niektoré nájdené asociačné pravidlá. V prípade, ak má používateľ o túto funkciu záujem, je potrebné dané záznamy označiť. Označenie záznamov je možné realizovať spôsobom, pri ktorom používateľ drží klávesu „control“ a kliká na požadované asociačné pravidlá. Po úspešnom označení záznamov je potrebné kliknúť na tlačidlo „Download selected rows“ a ďalší postup je rovnaký ako v prvom prípade. Tlačidlo „Deselect“ slúži na zrušenie označenia používateľom zvýraznených a vybraných záznamov.

Download table		Download selected rows		Deselect	
Show 20 entries					
LHS	RHS	Support	Confidenc		
All	All	All	All		
13186 ESC=[5-7]; Trig=[1.72-30]; FBG=2-4; EKG_Rytmus=0; ECHO_EF=51<=	ECHO_PH=0	0.083			
13965 O_Hypertenzia; Fajcenie=3; BMI=[18.2-29.1]; Kreat=[6.6-73]; ECHO_PH=0	ECHO_EF=51<=	0.071			
14735 Fajcenie=3; CRP=[2.22-3.9]; CL=[91.8-103]; EKG_Rytmus=0; ECHO_PH=0	ECHO_EF=51<=	0.074			
16126 O_Hypertenzia; Fajcenie=3; Tlak=160<=; EKG_QRS=<=89; Nalez=0	ECHO_EF=51<=	0.071			
16788 O_Hypertenzia; Fajcenie=3; Urea=[2.29-5.05]; Na=[140-170]; FBG=2-4	ECHO_EF=51<=	0.071			

Obrázok 40 Vygenerované asociačné pravidlá

V časti „Scatter plots“, ktorej obsahom je vizualizácia prostredníctvom bodového grafu a ktorého význam sme definovali v hlavnej časti diplomovej práce, má používateľ taktiež na výber, ako budú

jednotlivé hodnoty pochádzajúce z nájdených asociačných pravidiel pri vizualizácii využité. Tak ako je znázornené na Obrázok 41, používateľ má k dispozícii dve rôzne možnosti prezentované formou zaškrtávacích polí, pričom zvoliť môže iba jednu z nich. Pre potvrdenie výberu a následné vygenerovanie vizualizácie je potrebné kliknúť na tlačidlo „Render plot!“.

Data filtering
Association rules
Scatter plots
Graph-based plot with items and rules as vertices
Selected data

Choose one of the options, please!

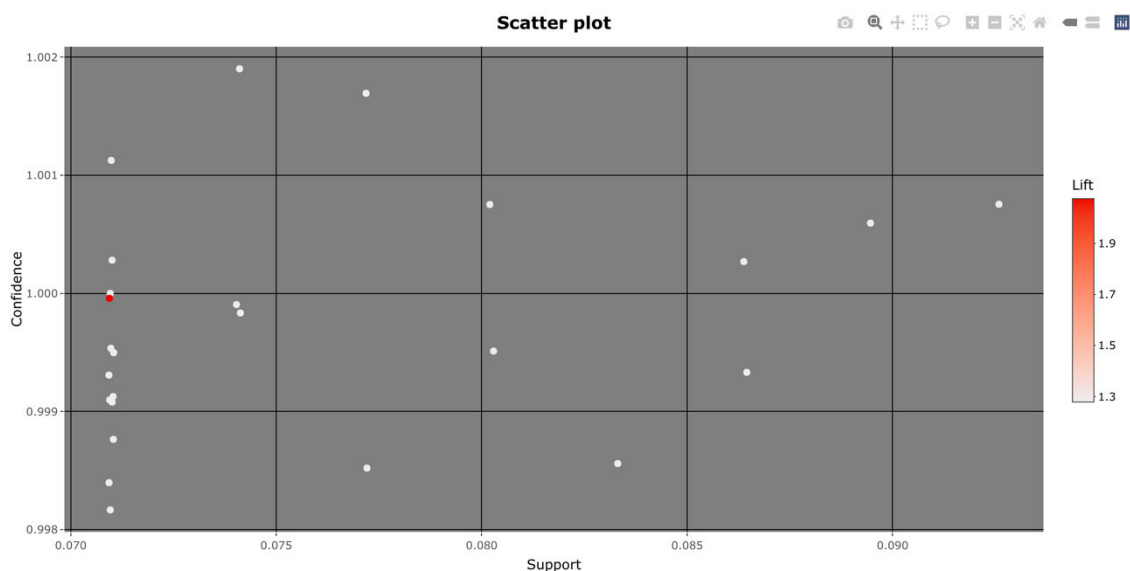
☒ Measures: Confidence, Lift & Shading: Support

☐ Measures: Support, Confidence & Shading: Lift

Render plot!

Obrázok 41 Možnosti vizualizácie bodového grafu

Typ výslednej vizualizácie - bodový graf obsahuje niekoľko interaktívnych prvkov. Tie je možné vidieť na nasledujúcom obrázku (Obrázok 42) v pravej hornej časti, ktoré sú zobrazené vo forme ikon. Pomocou nich, kliknutím na príslušnú ikonu, vie používateľ s výslednou vizualizáciou uskutočniť niekoľko úkonov. Medzi nich patrí priblíženie, resp. oddialenie grafu, výber určitej časti vizualizácie, uvedenie grafu do pôvodného stavu, či možnosť jeho stiahnutia vo forme .png súboru.



Obrázok 42 Bodový graf

Posledná vizualizácia, ktorej význam sme taktiež charakterizovali v hlavnej časti diplomovej práce, sa nachádza v časti pomenovanej ako „Graph-based plot with items and rules as vertices“.

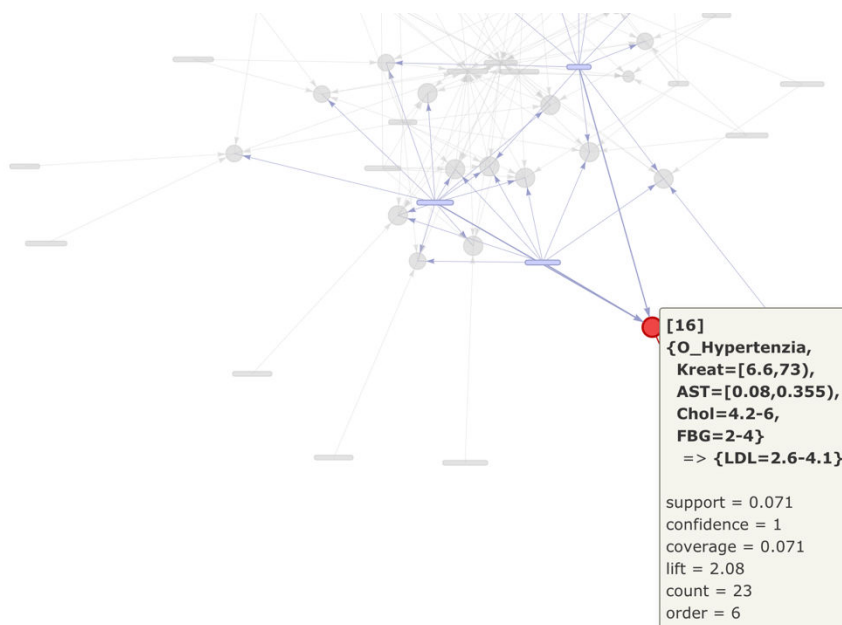
Na Obrázok 43 je možné vidieť dva hlavné elementy. Prvým z nich je tlačidlo „Download plot“, ktoré slúži na stiahnutie danej vizualizácie. Vzhľadom na to, že vizualizácia je interaktívna a znázorňuje

vzťahy medzi jednotlivými hodnotami atribútov, ktoré sa v asociačných pravidlách vyskytli a samotnými asociačnými pravidlami, stiahnutý súbor je vo formáte *.html*. Tým sa zabezpečí uchovanie interaktívnych prvkov, s ktorými je tak naďalej možné pracovať aj mimo prostredia implementovanej aplikácie.



Obrázok 43 Používateľské rozhranie pri vizualizácii vzťahov medzi asociačnými pravidlami

Druhý element, ktorého obsahom sú už spomínané hodnoty atribútov vyskytujúcich sa v nájdených asociačných pravidlách a samotné asociačné pravidlá, je prezentovaný vo forme zoznamu hodnôt umiestneného v rozbaľovacom menu. Používateľ si vie vybrať ktorúkoľvek z ponúkaných hodnôt, pričom tá časť grafu, ktorá s touto hodnotou sa zvýrazní sýtou farbou. Pre lepšiu predstavu a pochopenie uvádzame taktiež výsek zo spomínanej vizualizácie, ktorá je znázornená na Obrázok 44.



Obrázok 44 Vizualizácia znázorňujúca vzťahy medzi asociačnými pravidlami

Okrem spomenutých možností, používateľ taktiež dokáže získať podrobné informácie o tom, ktorý bod vizualizácie nesie akú informáciu. Je však potrebné premiestniť a ponechať kurzor na danom elemente spomínaného grafu.

5. Sekcia pre aplikovanie metód zhlukovania

V rámci štvrtej kapitoly používateľskej príručky sa zameriame na charakteristiku krokov, ktoré je potrebné a možné vykonať v sekcií implementovanej aplikácie pre aplikovanie metód zhlukovania. Jednotlivé kroky bližšie popíšeme v rámci používateľského prostredia pre aplikovanie metód hierarchického zhlukovania. Je to z dôvodu, že v tejto časti aplikácie sa nachádzajú elementy, ktoré sú špecifické len pre hierarchické zhlukovanie, a pri iných metódach zhlukovania sa nevyskytujú. Týmto výberom teda zabezpečíme, že každý z elementov bude v používateľskej príručke obsiahnutý.

5.1. Výber atribútov zo vstupnej dátovej množiny

Prvý krok v tejto časti aplikácie je totožný s tým, ktorý sme podrobne charakterizovali v podkapitole 3.1. používateľskej príručky. Jedná sa o výber atribútov zo vstupnej dátovej množiny, resp. prispôbenie dátovej množiny z hľadiska obsahu jej atribútov, ktorá bude využitá pri ďalších činnostiach v danej sekcií implementovanej aplikácie.

5.2. Výber metódy hierarchického zhlukovania

Ďalší, v poradí druhý krok je špecifický a vyskytuje sa len v rámci používateľského prostredia pri hierarchickom zhlukovaní. Jedná sa o výber metódy hierarchického zhlukovania, ktorá bude v danom procese aplikovaná. V kontexte používateľskej príručky je potrebné poznamenať, že dané možnosti sú používateľovi prezentované vo forme rozbaľovacieho menu tak, ako to je zobrazené aj na nasledujúcom obrázku (Obrázok 45).

Hierarchical clustering
The hierarchical clustering will be used for data clustering.

1. Choosing dataset
It is possible to choose input attributes or work with the whole dataset.
[Show or hide options](#)

Information about selected dataset:
Current number of attributes is: 46.

2. Method of hierarchical clustering
Select a method for calculating the optimal number of clusters.
Methods:
complete

3. Dendrogram
After selecting a clustering method, you can render a dendrogram by pressing the button below.
[Render dendrogram!](#)

4. Optimal number of clusters
The optimal number of clusters is displayed after clicking the 'Show me number!' button. The process can take several tens of seconds to a few minutes.
Methods:
Elbow method
[Show me results!](#)

Dendrogram | Optimal number of clusters | Silhouette analysis | Summary results | Clustering results | Clustering visualization | Parallel plot | Resulting dataset

(Dis)similar pairs of records | Selected dataset

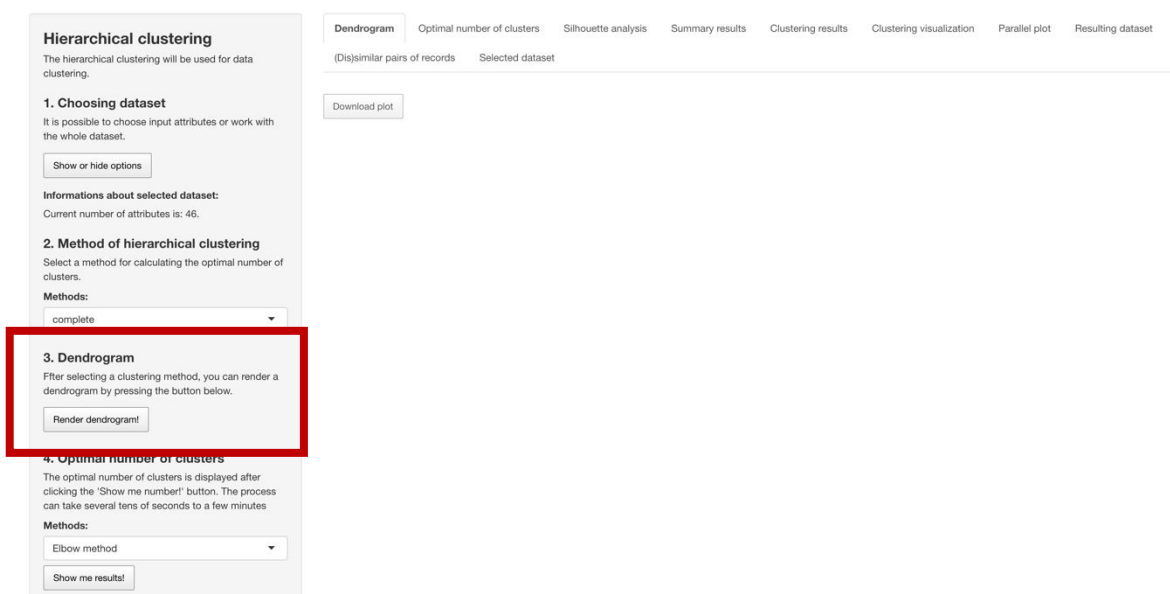
[Download plot](#)

Obrázok 45 Výber metódy hierarchického zhlukovania

Používateľ si musí vybrať práve jednu možnosť, pričom tento výber nie je potrebné potvrdiť žiadnym ďalším úkonom. Zvolená metóda bude využitá pri všetkých ďalších úkonov v rámci danej sekcie, vrátane vygenerovanie dendrogramu, či odhadu optimálneho počtu zhlukov.

5.3. Vygenerovanie dendrogramu

Tretí krok, a teda vygenerovanie dendrogramu je taktiež špecifický a vyskytuje sa len v rámci používateľského rozhrania pri hierarchickom zhlukovaní. Okrem iného, tento krok nie je nutné vykonať. Jedná sa len o analýzu, ktorá dokáže používateľovi napomôcť pri výbere optimálneho počtu zhlukov.



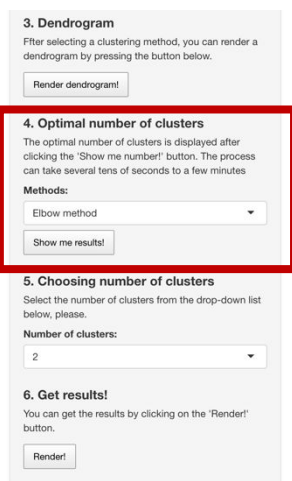
The screenshot shows a web application for hierarchical clustering. The left sidebar contains four steps: 1. Choosing dataset, 2. Method of hierarchical clustering, 3. Dendrogram (highlighted with a red box), and 4. Optimal number of clusters. The main area on the right shows the 'Dendrogram' tab selected, with a 'Download plot' button and a description of the visualization.

Obrázok 46 Generovanie dendrogramu

Vykreslenie tejto vizualizácie sa uskutoční po kliknutí na tlačidlo „Render dendrogram!“ tak, ako to je zobrazené na Obrázok 46.

5.4. Odhad optimálneho počtu zhlukov

Ďalším, v poradí štvrtým krokom je odhad optimálneho počtu zhlukov. Tento krok, ako aj ten predchádzajúci nie je potrebné vykonať pre správny chod aplikácie, avšak jedná sa o užitočnú analýzu, ktorá môže používateľovi napomôcť pri výbere počtu zhlukov. Jednotlivé metódy sme bližšie charakterizovali v hlavnej časti diplomovej práce. Možnosti výberu sú implementované vo forme rozbaľovacieho menu, pričom používateľ si môže zvoliť práve jednu hodnotu. Pre vygenerovanie grafu, ktorý vizualizuje ohodnotenie daného zhlukovania vzhľadom na počet zhlukov je potrebné kliknúť na tlačidlo „Show me results!“, ktoré je znázornené aj na Obrázok 47.



3. Dendrogram
After selecting a clustering method, you can render a dendrogram by pressing the button below.

Render dendrogram!

4. Optimal number of clusters
The optimal number of clusters is displayed after clicking the 'Show me number!' button. The process can take several tens of seconds to a few minutes

Methods:
Elbow method

Show me results!

5. Choosing number of clusters
Select the number of clusters from the drop-down list below, please.

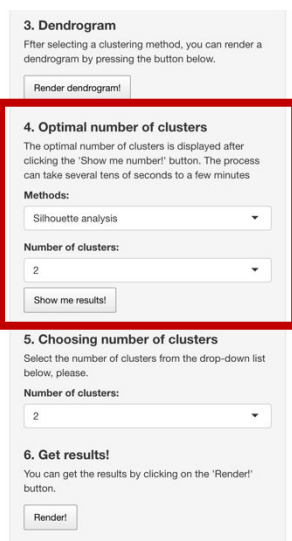
Number of clusters:
2

6. Get results!
You can get the results by clicking on the 'Render!' button.

Render!

Obrázok 47 Výber metódy pre odhad optimálneho počtu zhlukov (1)

Rozdielna situácia nastáva v prípade, ak používateľ z rozbaľovacieho menu vyberie možnosť „*Silhouette analysis*“. Keďže sa jedná o analýzu, v rámci ktorej je potrebné definovať aj počet zhlukov, systém automaticky vygeneruje ďalšiu časť používateľského rozhrania tak, ako to je znázornené na nasledujúcom obrázku (Obrázok 48).



3. Dendrogram
After selecting a clustering method, you can render a dendrogram by pressing the button below.

Render dendrogram!

4. Optimal number of clusters
The optimal number of clusters is displayed after clicking the 'Show me number!' button. The process can take several tens of seconds to a few minutes

Methods:
Silhouette analysis

Number of clusters:
2

Show me results!

5. Choosing number of clusters
Select the number of clusters from the drop-down list below, please.

Number of clusters:
2

6. Get results!
You can get the results by clicking on the 'Render!' button.

Render!

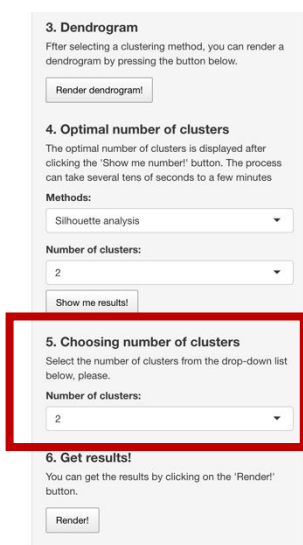
Obrázok 48 Výber metódy pre odhad optimálneho počtu zhlukov (2)

V novo vytvorenej časti používateľského rozhrania si používateľ zvolí hodnotu hovoriacu o počte zhlukov, ktorá bude využitá pri analýze ohodnotenia zhukovania prostredníctvom koeficientu siluety. Hodnoty reprezentujúce počet zhlukov sú taktiež znázornené vo forme prvkov v rozbaľovacom menu. Používateľ ma na výber hodnoty z intervalu od 2 po 10, pričom si musí zvoliť

práve jednu hodnotu. Potvrdenie výberu a následné vygenerovanie vizualizácií je uskutočniteľné kliknutím na už spomínané tlačidlo „Show me results!“.

5.5. Počet zhlukov ako parameter metódy zhlukovania

V piatom kroku dochádza k výberu počtu zhlukov, do ktorých chceme záznamy zo vstupnej dátovej množiny zadeliť. Používateľ si môže vybrať ktorúkoľvek z hodnôt z intervalu od 2 po 10, pričom si však musí vybrať práve jednu hodnotu. Tie sú reprezentované vo forme prvkov v rozbaľovacom menu tak, ako je zobrazené na obrázku nižšie (Obrázok 49).



3. Dendrogram
Filter selecting a clustering method, you can render a dendrogram by pressing the button below.
Render dendrogram!

4. Optimal number of clusters
The optimal number of clusters is displayed after clicking the 'Show me number!' button. The process can take several tens of seconds to a few minutes
Methods:
Silhouette analysis
Number of clusters:
2
Show me results!

5. Choosing number of clusters
Select the number of clusters from the drop-down list below, please.
Number of clusters:
2

6. Get results!
You can get the results by clicking on the 'Render!' button.
Render!

Obrázok 49 Výber počtu zhlukov

Výber v danom kroku nie je potrebné nijak potvrdiť.

5.6. Potvrdenie všetkých úkonov

Rovnako, ako v časti pre hľadanie asociačných pravidiel, na záver je potrebné potvrdiť zvolené kritéria a dať systému pokyn pre vykonanie všetkých analýz v rámci danej sekcie aplikácie, a to kliknutím na tlačidlo „Render!“, ktoré je taktiež znázornené na nasledujúcom obrázku (Obrázok 50).

3. Dendrogram
After selecting a clustering method, you can render a dendrogram by pressing the button below.

4. Optimal number of clusters
The optimal number of clusters is displayed after clicking the 'Show me number!' button. The process can take several tens of seconds to a few minutes.
Methods:

Number of clusters:

5. Choosing number of clusters
Select the number of clusters from the drop-down list below, please.
Number of clusters:

6. Get results!
You can get the results by clicking on the 'Render!' button.

Obrázok 50 Potvrdenie všetkých úkonov

5.7. Prezentácia výsledkov

Po úspešnom vykonaní všetkých dostupných analýz a úkonov má používateľ k dispozícii mnoho výsledkov, ktoré sme podrobne charakterizovali vo štvrtej kapitole hlavnej časti diplomovej práce (4). V rámci používateľskej príručky je dôležité zdôrazniť, že prechod medzi nimi opäť prebieha výberom možností nachádzajúcich sa v lište, v hlavnej časti obrazovky, tak ako to je vizualizované na obrázku nižšie (Obrázok 51).



Obrázok 51 Odkazy na panely so získanými výsledkami

Používateľ má rovnako ako v predchádzajúcej sekcii, možnosť stiahnuť si jednotlivé získané výsledky a vizualizácie. Tlačidlo pre tento úkon sa nachádza spravidla v ľavej hornej časti pre prezentáciu výsledkov, a teda priamo pod lištou, ktorá je znázornená na **Chyba! Nenašiel sa žiaden zdroj odkazov..** Tlačidlo je označené buď textom „Download plot“ alebo „Download summary results“. V paneli „Parallel plot“ sa po aplikovaní danej metódy zhukovania na dáta vygeneruje dodatočná časť používateľského prostredia, ktorá je znázornená na Obrázok 52.

Please, choose one of scaling options:☒ No scaling☐ Standardize to Min = 0 and Max = 1☐ Normalize univariately (subtract mean & divide by sd)☐ Standardize and center variables**Select attributes from used dataframe:***A maximum of 10 attributes can be selected. **Which clusters do you want to highlight on the graph?**

Obrázok 52 Používateľské rozhranie pre vizualizáciu „Parallel plot“

Podrobnú charakteristiku danej vizualizácie spoločne s vysvetlením jednotlivých možností, z ktorých si používateľ môže vybrať, sme definovali v hlavnej časti diplomovej práce. Avšak pre objasnenie ide o znázornenie rozloženia hodnôt v jednotlivých zhlukoch naprieč zvolenými atribútmi. Daná vizualizácia dokáže znázorniť len numerické hodnoty, od čoho sa odvíja aj prvá časť tohto používateľského rozhrania, v rámci ktorej je možné zvoliť si spôsob škálovania týchto hodnôt. Používateľ má na výber štyri možnosti, ktoré sú reprezentované vo forme zaškrtnutých polí, pričom si musí zvoliť práve jednu možnosť. V druhej časti si používateľ volí atribúty, ktoré chce zahrnúť do vizualizácie. Ich množstvo je ohraničené minimálne dvoma a maximálne desiatimi atribútmi, pričom odporúčaný počet premenných je päť, a to najmä kvôli prehľadnosti grafu. Možnosti, teda názvy stĺpcov sa odvíjajú od predchádzajúcich úkonov, konkrétne od úkonu popísaného v podkapitole 5.1. Medzi možnosťami sú len tie názvy atribútov, ktoré používateľ vybral v rámci prvého kroku. Forma reprezentácia je totožná s tou, ktorú sme podrobne popísali v podkapitole 4.2.1 - jedná sa o pole s možnosťou výberu viacerých hodnôt. Premennú je možné vymazať po kliknutí na ňu a následne stlačením spätnej klávesy, tzv. klávesy „backspace“. Pridanie novej hodnoty je vykonateľné výberom z rozbaľovacieho menu, ktoré sa zobrazí po kliknutí do prázdneho priestoru daného poľa. Pri poslednej možnosti si používateľ môže vybrať, ktoré zhľuky vo vizualizácii zahrnie. Môže zvoliť možnosť „All clusters“, čím do vizualizácie zahrnie všetky zhľuky, alebo zvoliť jeden konkrétny zhľuk. Množstvo možností sa samozrejme odvíja od výberu hodnoty, ktorý sme charakterizovali v podkapitole 5.5. Na záver používateľ danú vizualizáciu vygeneruje kliknutím na tlačidlo „Render

plot!". Taktiež má k dispozícii možnosť získaný graf vyexportovať vo forme *.png* súboru, a to kliknutím na tlačidlo „*Download plot*“.

Dôležitou časťou tejto sekcie, konkrétne časti pre prezentáciu výsledkov, je panel označený názvom „*Resulting dataset*“. V rámci neho sa nachádza sa nachádza tabuľka, obsahujúca záznamy z dátovej množiny, ktorej výsledná podoba sa odvíja od všetkých úkonov, bližšie popísaných v predchádzajúcich podkapitolách. K tabuľke je navyše pridaná informácia označenia zhuku, do ktorého daný záznam patrí. Používateľ si opäť môže túto tabuľku stiahnuť, a to vo forme *.xlsx* alebo *.csv* súboru, po kliknutí na príslušné tlačidlá „*Download as xlsx*“ alebo „*Download as csv*“. Zároveň je tento panel akýmsi prostriedkom prepojenia tejto a nasledujúcej, tretej, sekcie. Totižto, ak je používateľ s výsledkami zhukovania spokojný a chcel by pre záznamy v rámci jednotlivých zhukov nájsť asociačné pravidlá, môže tak urobiť. V prvom rade je však potrebné výslednú dátovú množinu, s označením do ktorého zhuku dané záznamy patria, preposlať do spomínanej tretej sekcie implementovanej aplikácie. Vykonať tak môže kliknutím na tlačidlo „*Send hierarchical clustering results!*“ (názov, ktorým je tlačidlo reprezentované sa odvíja od použitého prístupu k zhukovaniu), ktoré je taktiež znázornené na Obrázok 53. Systém sa automaticky presunie do ďalšej časti používateľského prostredia, ktoré bližšie charakterizujeme v nasledujúcej podkapitole.

Are you satisfied with results?

If so, use this dataset to generate association rules. It is necessary to click on the button below.

Show 15 entries

	Cluster	Pohlavie	Vek	ESC	R_ICHS	R_CMP	R_IM
	All	All	All	All	All	All	All
1	1	Muz	[73,88]	[5,7]	true	false	true
4	1	Zena	[73,88]	[7,33]	false	false	false
5	1	Muz	[36,58]	[5,7]	false	false	false
8	1	Zena	[63,68]	[5,7]	false	false	false
9	1	Muz	[73,88]	[7,33]	true	false	false
15	1	Muz	[58,63]	[0,5]	false	false	false
20	1	Muz	[63,68]	[0,5]	false	true	false
28	1	Muz	[63,68]	[5,7]	false	false	false

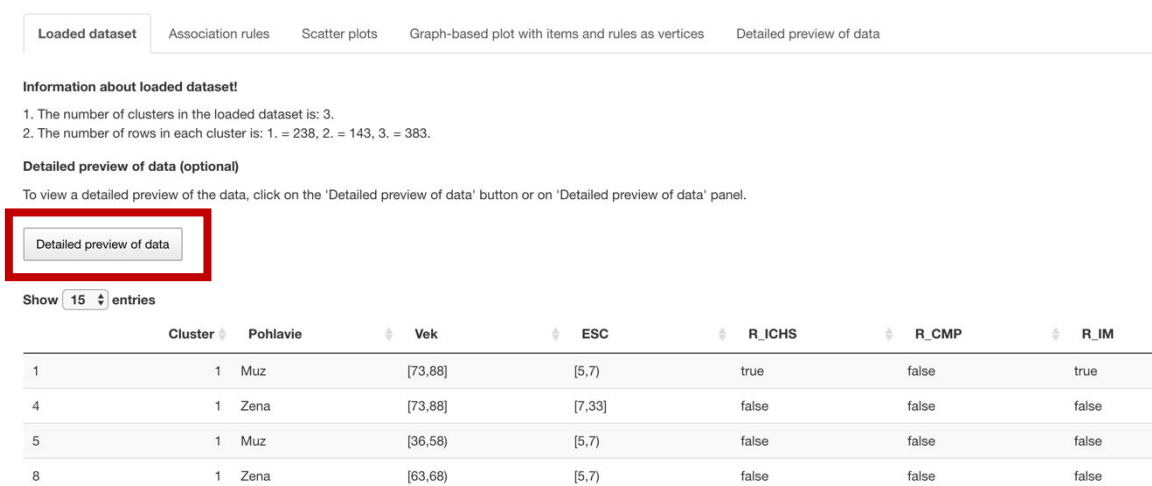
Obrázok 53 Zatriedené záznamy do zhukov

6. Hľadanie asociačných pravidiel v rámci definovaných zhlukov

V poslednej kapitole používateľskej príručky sa zameriame na charakteristiku krokov, ktoré je potrebné vykonať pre získanie asociačných pravidiel v rámci definovaných zhlukov, ktoré pochádzajú z druhej sekcie implementovanej aplikácie.

6.1. Podrobné informácie o zhluchoch

Za prvý krok možno považovať možnosť detailného náhľadu na dátovú množinu, ktorá bola načítaná z predošlej časti aplikácie (charakterizovanej v prechádzajúcej kapitole). Tento krok nie je nutné vykonať, avšak používateľovi poskytne bližší pohľad na dáta. K danej časti je možné sa dostať dvomi rôznymi spôsobmi, a to buď kliknutím na tlačidlo „*Detailed preview of data*“, ktoré sa nachádza v paneli „*Loaded dataset*“, tak ako je viditeľné na Obrázok 54, alebo priamym kliknutím na panel, ktorý nesie rovnaký názov ako tlačidlo.



Loaded dataset Association rules Scatter plots Graph-based plot with items and rules as vertices Detailed preview of data

Information about loaded dataset!

1. The number of clusters in the loaded dataset is: 3.
2. The number of rows in each cluster is: 1. = 238, 2. = 143, 3. = 383.

Detailed preview of data (optional)

To view a detailed preview of the data, click on the 'Detailed preview of data' button or on 'Detailed preview of data' panel.

Detailed preview of data

Show 15 entries

	Cluster	Pohlavie	Vek	ESC	R_ICHS	R_CMP	R_IM
1	1	Muz	[73,88]	[5,7]	true	false	true
4	1	Zena	[73,88]	[7,33]	false	false	false
5	1	Muz	[36,58]	[5,7]	false	false	false
8	1	Zena	[63,68]	[5,7]	false	false	false

Obrázok 54 Úvodné informácie o dátovej množine

Súčasťou používateľského prostredia v rámci časti „*Detailed preview of data*“ je rozbaľovacie menu, tlačidlo, text a tabuľka. Súčasťou rozbaľovacieho menu sú hodnoty, ktoré reprezentujú zhluky nachádzajúce sa v načítanej dátovej množine. Používateľ si musí zvoliť práve jednu hodnotu, pričom výber a pokyn pre vygenerovanie tabuľky a informácií o zhluke záznamov je nutné potvrdiť kliknutím na tlačidlo „*Render table!*“, ktoré je možné vidieť aj na Obrázok 55.

Loaded dataset Association rules Scatter plots Graph-based plot with items and rules as vertices **Detailed preview of data**

Select cluster:
1

Render table!

Information about selected data!

1. Attributes that contained only one unique value will be removed. These attributes are: R_AoS.
2. The number of rows of the selected dataset is: 238. The number of columns of the selected dataset is: 45.

Show 15 entries

	Pohlavie	Vek	ESC	R_ICHS	R_CMP	R_IM	R_Hypertenzi
1	Muz	[73,88]	[5,7]	true	false	true	false
4	Zena	[73,88]	[7,33]	false	false	false	false
5	Muz	[36,58]	[5,7]	false	false	false	false

Obrázok 55 Informácie o jednotlivých zhlukoch

6.2. Výber zhlukov

V druhom kroku si používateľ volí zhluky, v rámci ktorých chce hľadať asociačné pravidlá. Z hľadiska implementácie používateľského prostredia sa jedná o takmer identické riešenie, ktoré bolo využité aj v predchádzajúcich častiach, a ktoré sme podrobne charakterizovali v podkapitole 4.1. používateľskej príručky. Rozdiel je však v tom, že hodnoty, ktoré má používateľ na výber sú reprezentované označením jednotlivých zhlukov.

6.3. Nadväzujúce úkony

Kroky, ktoré je potrebné vykonať pri ďalších úkonoch, akými sú výber hodnôt metrík pre hľadanie asociačných pravidiel, výber premenných pre obmedzenie pravej strany asociačných pravidiel, či potvrdenie vykonaných rozhodnutí je totožné s tými, ktoré sme charakterizovali v podkapitolách 4.3, 4.4 a 4.5.

6.4. Prezентация výsledkov

Prístup k jednotlivým získaným výsledkom je takmer rovnaký, ako v prvej sekcii, ktorú sme charakterizovali v tretej kapitole, avšak možno tu pozorovať dva hlavné rozdiely. Prvým z nich je v časti, resp. paneli „*Association rules*“, kde hodnoty (Obrázok 56), ktoré sa vyskytli na ľavej strane nájdených asociačných pravidiel v rámci jednotlivých zhlukov je možné zobrazit a taktiež opätovne zakryť kliknutím na tlačidlo „*Show or hide unique values*“. Je to najmä z dôvodu, že týchto hodnôt je spravidla väčší počet a používateľské prostredie, ako celok môže pôsobiť pre používateľa mätúco. Týmto spôsobom má plnú kontrolu nad tým, kedy si tieto hodnoty zobrazí a kedy naopak nie.

Loaded dataset **Association rules** Scatter plots Graph-based plot with items and rules as vertices Detailed preview of data

Information about association rules!

- The number of association rules in each cluster is: 1. = 140, 2. = 169, 3. = 33.
- The unique values that appear on the LHS of association rules within the defined clusters are:

Show or hide unique values

Cluster 1: AST=[0.08-0.355]; BMI=[33.1-51.8]; CL=[103-104]; CL=[91.8-103]; CRP=[0.1-2.22]; CRP=[3.9-130]; ECHO_EF=31-50; ECHO_EF=51<; ECHO_PH=0; EKG_FR=[45-64]; EKG_FR=[64-72]; EKG_FR=[72-130]; EKG_PQ=[180-360]; EKG_QRS=<=89; EKG_QRS=90-119; EKG_QT=[40-380]; EKG_QT=[400-518]; EKG_Rytmus=0; EKG_STE; EKG_T; ESC=[0-5]; ESC=[5-7]; ESC=[7-33]; Fajcienie=3; FBG=2-4; FBG=4<; HDL=[0.51-1.14]; Chol=<=4.2; Chol=6<; K=[2.9-4.06]; K=[4.06-4.47]; K=[4.47-58.5]; Kreat=[6.6-73]; Kreat=[73-89]; Kreat=[89-735]; LDL=<=2.6; Na=[138-140]; Na=[140-170]; Na=[4.4-138]; Nalez=5; O_DM; O_Hypertenzia; O_IJCHS; O_IM; Pohlavie=Muz; Pohlavie=Zena; Tlak=140-159; Tlak=160<; Trigl=[1.21-1.72]; Trigl=[1.72-30]; Urea=[2.29-5.05]; Urea=[6.47-27.1]; Vek=[73-88].

Cluster 2: AST=[0.08-0.355]; AST=[0.355-0.42]; BMI=[18.2-29.1]; BMI=[29.1-33.1]; BMI=[33.1-51.8]; CL=[104-111]; CL=[91.8-103]; CRP=[0.1-2.22]; CRP=[3.9-130]; ECHO_EF=51<; ECHO_PH=0; EKG_FR=[45-64]; EKG_PQ=[180-360]; EKG_QRS=<=89; EKG_QRS=90-119; EKG_QT=[400-518]; EKG_Rytmus=0; EKG_T; ESC=[0-5]; ESC=[5-7]; ESC=[7-33]; Fajcienie=3; FBG=2-4; HDL=[0.51-1.14]; HDL=[1.42-1.08]; Chol=<=4.2; Chol=4.2-6; K=[4.06-4.47]; K=[4.47-58.5]; Kreat=[6.6-73]; Kreat=[73-89]; Kreat=[89-735]; LDL=<=2.6; LDL=2.6-4.1; Na=[138-140]; Na=[140-170]; Na=[4.4-138]; Nalez=0; O_DM; O_Hypertenzia; O_IJCHS; O_IM; Pohlavie=Muz; Pohlavie=Zena; R_IJCHS; R_IM; Tlak=140-159; Tlak=160<; Trigl=[0.46-1.21]; Trigl=[1.21-1.72]; Urea=[2.29-5.05]; Urea=[5.05-6.47]; Urea=[6.47-27.1]; Vek=[36-58].

Cluster 3: AST=[0.355-0.42]; BMI=[18.2-29.1]; CL=[104-111]; CL=[91.8-103]; CRP=[3.9-130]; ECHO_EF=51<; ECHO_PH=0; EKG_FR=[72-130]; EKG_PQ=[160-180]; EKG_QRS=<=89; EKG_QRS=90-119; EKG_QT=[380-400]; EKG_QT=[40-380]; EKG_Rytmus=0; EKG_T; ESC=[5-7]; Fajcienie=3; FBG=2-4; HDL=[1.14-1.42]; Chol=<=4.2; Chol=4.2-6; Kreat=[6.6-73]; Kreat=[73-89]; LDL=2.6-4.1; Nalez=0; O_Hypertenzia; Pohlavie=Muz; Pohlavie=Zena; Tlak=140-159; Trigl=[1.21-1.72]; Trigl=[1.72-30]; Urea=[2.29-5.05].

Download table Download selected rows Deselect

Show **20** entries

Cluster	LHS	RHS	Supp
All	All	All	All
36055 1	Chol=6<; Trigl=[1.72-30]; FBG=2-4; EKG_Rytmus=0; EKG_T	ECHO_PH=0	

Obrázok 56 Hodnoty vyskytujúce sa na ľavej strane asociačných pravidiel

Druhý rozdiel spočíva v tom, že pri generovaní vizualizácií v častiach, resp. paneloch „*Scatter plots*“, „*Graph-based plot with items and rules as vertices*“ je najprv potrebné zvoliť niektorý zo zhlukov, pre ktorý boli vygenerované asociačné pravidlá. Práve tie budú súčasťou daných vizualizácií. Výber zhluku prebieha štandardne prostredníctvom rozbaľovacieho menu, tak ako je znázornené na obrázku nižšie (Obrázok 57), pričom používateľ si musí zvoliť práve jednu možnosť. Ďalší postup je totožný s tým, ktorý sme detailne opísali v podkapitole 4.6 používateľskej príručky.

Loaded dataset Association rules **Scatter plots** Graph-based plot with items and rules as vertices Detailed preview of data

Select cluster:

1

Choose one of the options, please!

☒ Measures: Confidence, Lift & Shading: Support

☐ Measures: Support, Confidence & Shading: Lift

Render plot!

Obrázok 57 Výber zhluku asociačných pravidiel pre vizualizáciu

TECHNICKÁ UNIVERZITA V KOŠICIACH
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

POPISNÉ MODELY ŠPECIFICKÝCH SKUPÍN PACIENTOV
VZHĽADOM NA ICH KARDIOVASKULÁRNE RIZIKO
Systémová príručka

Príloha D

Študijný program: Hospodárska informatika
Študijný odbor: Informatika
Školiace pracovisko: KKUI
Školiteľ: prof. Ing. Ján Paralič, PhD.
Konzultant: Ing. Zuzana Pella
Ing. Oliver Lohaj

2022 Košice

Bc. Adam Pavlíček

Obsah

1.	Úvod do systémovej príručky.....	132
1.1.	Balíky využité pri implementácii softvéru	132
2.	Systémová analýza implementovaného softvéru	134
2.1.	Generovanie elementov pre výber metódy úpravy numerických atribútov	134
2.2.	Minimálny počet prvkov v elemente varSelectInput()	136
2.3.	Proces hľadania asociačných pravidiel	137
2.4.	Vizualizácia odhadu optimálneho počtu zhlukov	138
2.5.	Proces zhlučovania prostredníctvom algoritmu PAM	139

Zoznam obrázkov

Obrázok 1 Fragment kódu pre dynamické generovanie <i>selectInput</i> elementov na základe počtu numerických premenných v načítavanej dátovej množine	135
Obrázok 2 Vygenerované používateľské rozhranie	136
Obrázok 3 Element <i>varSelectInput()</i>	136
Obrázok 4 Fragment kódu zabezpečujúci minimálny počet prvkov v elemente <i>varSelectInput()</i> .	137
Obrázok 5 Proces hľadania asociačných pravidiel	138
Obrázok 6 Fragment kódu vizualizácie odhadu optimálneho počtu zhlukov	139
Obrázok 7 Fragment kódu znázorňujúci proces zhukovania prostredníctvom algoritmu PAM	140

1. Úvod do systémovej príručky

Obsahom tejto príručky bude opis hlavných funkcionalít aplikácie zo systémoveho, resp. technického uhľa pohľadu. V nasledujúcich podkapitolách sa taktiež zameriame na charakteristiku balíkov, ktoré boli počas implementácie softvéru využité. Súčasťou systémovej príručky je štandardne aj opis architektúry implementovaného systému, avšak tejto téme sa venujeme v hlavnej časti diplomovej práce, v tretej kapitole (3.5).

Pri realizácii praktickej časti našej záverečnej práce sme využili programovací jazyk *R*, ktorý je voľne dostupný a ktorý možno charakterizovať ako univerzálny nástroj vhodný pre spracovanie, analýzu dát a následnú prezentáciu získaných výsledkov. Jazyk *R* je taktiež vhodný pre implementáciu tzv. *dashboardov*, či aplikácií, ktoré disponujú mnohými interaktívnymi prvkami. Výhodou jazyka *R* taktiež je, že používateľ má možnosť definovať svoje vlastné funkcie, či prispôbovať si tie existujúce.

Počas implementácie softvéru sme využili množstvo balíkov. Balík v programovacom jazyku *R* možno charakterizovať ako funkcionalitu, resp. časť zdrojového kódu, ktorá je navrhnutá a vytvorená profesionálnymi vývojármi, ale aj bežnými používateľmi. Primárny cieľ balíka, resp. funkcionalít, ktoré sú jeho obsahom, je riešenie špecifického problému alebo situácie. Balík teda zahŕňa metódy a funkcie, ktoré nie sú základnou súčasťou vývojového prostredia. Tie, ktoré sme pri programovaní využili, priblížime v nasledujúcej podkapitole.

1.1. Balíky využité pri implementácii softvéru

Implementovaný softvér možno popísať ako aplikáciu pre podporu výskumu kardiológov na báze popisných modelov. Pri jeho tvorbe bolo nevyhnutné využiť niekoľko balíkov, vďaka ktorým sa bolo možné dopracovať k požadovaným výsledkom. Konkrétne sa jedná o balíky:

- *shiny* (verzia 1.6.0) - balík, vďaka ktorému je možné implementovať interaktívne webové aplikácie priamo z prostredia jazyka *R*, pričom ich je možné rozšíriť o *CSS*, *html* či *JavaScript* skripty,
- *readxl* (verzia 1.3.1) - balík, ktorý je svojou funkcionalitou určený pre načítavanie údajov z Excelu do prostredia jazyka *R*,
- *writexl* (verzia 1.4.0) - balík určený pre exportovanie dát z prostredia *R* do Excel súboru .xlsx,
- *DT* (verzia 0.15) - balík ponúkajúci funkcionalitu výpisu dát do tabuľkovej formy v prostredí *R Shiny*, ktorý taktiež ponúka funkcie pre filtrovanie, triedenie údajov a mnohé iné,
- *arules* (verzia 1.6.7) - balík zabezpečujúci funkcionalitu pre získanie, analýzu, spracovanie a prezentáciu frekventovaných množín a nájdených asociačných pravidiel,

- *dplyr* (verzia 1.0.7) - balík slúžiaci na rýchle a intuitívne spracovanie a úpravu dát, pričom sa jedná napríklad o filtrovanie záznamov alebo o ich zjednocovanie na základe zvolených atribútov,
- *stringi* (verzia 1.5.3) - balík obsahujúci funkcionality pre úpravu textových dát,
- *stringr* (verzia 1.4.0) - balík určený pre rýchle a konzistentné spracovanie znakov a súvislého textu, a to v akomkoľvek kódovaní,
- *arulesViz* (verzia 1.4.0) - balík ponúkajúci rôzne spôsoby vizualizácie nájdených asociačných pravidiel,
- *ggplot2* (verzia 3.3.5) - balík obsahujúci prostriedky pre vytváranie komplexných vizualizácií jednoduchých, viacrozmerných, numerických ale aj kategorických údajov,
- *shinyjs* (verzia 2.0.0) - balík umožňujúci využívať jazyk *JavaScript* v prostredí jazyka *R*,
- *shinycssloaders* (verzia 1.0.0) - balík, vďaka ktorému je možné pridať do aplikácie animácie znázorňujúce načítavanie tabuliek či grafov,
- *shinyWidgets* (verzia 0.6.0) - balík, pomocou ktorého je možné do softvéru implementovať rôzne vstupné elementy, ktoré sa oproti tým štandardným líšia iným grafickým prevedením,
- *cluster* (verzia 2.1.0) - balík poskytujúci množstvo funkcionalít a metód určených pre vykonávanie zhukovacích analýz,
- *fpc* (verzia 2.2.9) - balík slúžiaci pre vyhodnotenie výsledkov zhukovania, a to širokou škálou rôznych validačných metrík,
- *gtsummary* (verzia 1.4.2.9008) - balík umožňujúci používateľom vytvárať prehľadné tabuľky zobrazujúce napríklad výsledky zhukovania,
- *visNetwork* (verzia 2.1.0) - balík, vďaka ktorému je možné vytvárať interaktívne vizualizácie sieťového typu, pričom sú využívané prvky jazyka *JavaScript*,
- *plotly* (verzia 4.10.0) - balík určený pre vytváranie interaktívnych vizualizácií rôznych typov, a to vo vysokej kvalite rozlíšenia,
- *ggdendro* (verzia 0.1.23) - balík slúžiaci pre extrahovanie údajov zo stromových štruktúr a ich následne uloženie do dátových rámcov, ktoré je opätovne možné využiť na ich vizualizáciu, avšak už pomocou balíka *ggplot2*.

2. Systémová analýza implementovaného softvéru

V druhej kapitole systémovej príručky sa zameriame na technický popis častí a niektorých funkcionalít, ktoré sú súčasťou implementovaného systému.

Samotná aplikácia sa nachádza v súbore `app.r`, ktorý je možné rozdeliť na dve hlavné časti, ktorými sú `ui().r` a `server().r`. Obsah prvej z nich pozostáva zo štruktúry používateľského prostredia softvéru, ktoré sme vizuálne prispôbili pomocou mechanizmu `CSS`. V druhej časti, a teda v `server()` je definovaná celá programová logika aplikácie, teda všetkých funkcionalít, ktoré sú jej súčasťou.

Implementovaný softvér pozostáva z troch hlavných sekcií, a to:

- sekcia pre hľadanie asociačných pravidiel,
- sekcia pre aplikovanie metód zhukovania,
- sekcia pre hľadanie asociačných pravidiel,

pričom je dôležité poznamenať, že im predchádza časť pre načítanie vstupných dát. V nasledujúcich podkapitolách si rozoberieme niekoľko fragmentov zdrojového kódu, ktoré predstavujú dôležité aspekty našej aplikácie.

2.1. Generovanie elementov pre výber metódy úpravy numerických atribútov

Pri načítavaní dát do softvéru má používateľ na výber dve možnosti. Buď využije dátovú množinu, ktorá sa už v systéme nachádza, alebo si načíta svoje vlastné dáta. V prípade, ak sa rozhodne pre druhú možnosť, musí vykonať niekoľko úkonov, ktoré sme podrobne charakterizovali v hlavnej časti diplomovej práce, v piatej kapitole. Kľúčový krok však predstavuje úprava numerických atribútov na kategorický tvar. Používateľ má na výber dve možnosti transformácie, a to diskretizácia na rovnakú šírku alebo hĺbku a pretypovanie numerickej premennej na dátový typ *factor*. Tieto možnosti sa nachádzajú v rozbaľovacom menu, ktoré je však potrebné vygenerovať pre každý numerický atribút zvlášť. Práve tento problém rieši nasledujúci fragment kódu (Obrázok 58).

```

output$elements_loading_data = renderUI({
  tagList(
    lapply(1:length(numeric_colnames), function(i) {
      tags$div(id = paste0("id_div_", numeric_colnames[i]), checked = NA,
        selectInput(inputId = paste0("id_", numeric_colnames[i]), label = numeric_colnames[i],
          choices = c("As factor", "Discretize (method: frequency)", "Discretize (method: interval)", selected = "As factor")
        )
    })
  ),
  actionButton("confirm_my_options", label = "Confirm selection")
})

```

Obrázok 58 Fragment kódu pre dynamické generovanie *selectInput* elementov na základe počtu numerických premenných v načítavanej dátovej množine

Ako je možné vidieť, výstup je definovaný ako funkcia *renderUI()*, ktorá zabezpečuje dynamické generovanie elementov používateľského rozhrania na základe stanovených podmienok. V dátovej štruktúre vektor, ktorá je pomenovaná *numeric_colnames* sa nachádzajú názvy všetkých numerických atribútov, ktoré sa vyskytli v načítavanej dátovej množine. Cyklus pre generovanie používateľského rozhrania, v tomto prípade hovoríme o elementoch *selectInput*, ktoré predstavujú rozbaľovacie menu, sa zopakuje toľkokrát, koľko hodnôt sa nachádza vo vyššie spomenutom vektore *numeric_colnames*. Je vhodné zdôrazniť, že je to z dôvodu potreby vygenerovať tento element pre každý numerický atribút, ktorý sa nachádza vo vstupnej dátovej množine. ID, a teda identifikátor týchto elementov je reprezentovaný ako parameter funkcie *selectInput*, pričom je zapísaný v nasledujúcom tvare:

- „id_“ a aktuálny názov numerického atribútu, pre ktorý je element *selectInput* generovaný.

Identicky to prebieha aj pri popise elementu, ktorý je definovaný parametrom *label* a ktorý sa nachádza v tej istej funkcii. Parameter *label* možno definovať ako nadpis, ktorý je súčasťou elementu *selectInput*, pričom jeho hodnota odpovedá jednému z n numerických atribútov v načítavanej dátovej množine. Na záver sa taktiež vygeneruje tlačidlo, ktoré slúži pre potvrdenie zvolených metód úpravy týchto atribútov.

Časť používateľského rozhrania, ktorá je vygenerovaná fragmentu kódu uvedeného vyššie, je zobrazená na Obrázok 59. Pri načítavaní dát do systému sme využili dátovú množinu *Iris*, ktorá obsahuje 4 numerické atribúty.

Sepal_Length
As factor

Sepal_Width
As factor

Petal_Length
As factor

Petal_Width
As factor

Confirm selection

Obrázok 59 Vygenerované používateľské rozhranie

Ďalšia implementovaná funkcia zabezpečí, že ak si používateľ vyberie možnosť diskretizácie niektorého, resp. niektorých numerických atribútov, pod príslušným elementom *selectInput* sa vygeneruje ďalší *selectInput*, kde si vie používateľ vybrať počet intervalov, do ktorých chce hodnoty danej numerickej premennej začleniť. Po zvolení vhodných úprav je možné načítať dátovú množinu do ďalších častí aplikácie.

2.2. Minimálny počet prvkov v elemente *varSelectInput()*

V hlavnej časti aplikácie, a teda buď v sekcii pre hľadanie asociačných pravidiel alebo v sekcii pre aplikovanie metód zhlukovania, sa nachádza vstup, ktorý slúži pre prispôbenie obsahu vstupnej dátovej množiny. Používateľ si vie zvoliť len tie atribúty, s ktorými chce ďalej pracovať v danej sekcii softvéru. Výber týchto atribútov prebieha prostredníctvom elementu *varSelectInput()*, ktorý v používateľskom rozhraní aplikácie vyzerá tak, ako je zobrazené na obrázku nižšie (Obrázok 60).

Descriptive Modeling Loading data Association rules Clustering Association rules for clusters

Associations rules
The Apriori algorithm will be used to generate association rules.

1. Choosing dataset
It is possible to choose input attributes or work with the whole dataset.

Show or hide options

☐ All dataset!

☒ Selected attributes only

Attributes:
*The minimum number of selected attributes is 2.

Pohlavie Vek ESC R_ICH5 R_CMP R_IM
R_Hypertenzia R_DM R_AoS O_ICH5 O_CMP
O_IM O_Hyperchol O_Hypertenzia O_DM
O_AoS Fajcienie Alkohol BMI Tlak Urea Kreat
AST Na K Chol Trigl HDL LDL CRP CL
FBG EKG_FR EKG_Rythmus EKG_PQ
EKG_QRS EKG_QT EKG_BPTR EKG_KES
EKG_SVES EKG_STD EKG_STE EKG_T
ECHO_EF ECHO_PH Nalez

Show data

Data filtering Association rules Scatter plots Graph-based plot with items and rules as vertices Selected data

Obrázok 60 Element *varSelectInput()*

V danom elemente sa nachádzajú názvy stĺpcov, resp. atribútov, ktoré sú súčasťou načítanej dátovej množiny. Jedna zo stanovených podmienok, ktorú sme definovali je, že počet hodnôt

v danom elemente nesmie klesnúť pod dva. Práve túto situáciu rieši fragment kódu (Obrázok 61), ktorý je znázornený nižšie.

```
observeEvent(input$select_input_vars_alt, {
  if(length(input$select_input_vars_alt) == 2){
    val_inp = c()

    abc = unlist(input$select_input_vars_alt, use.names=FALSE)

    i = 1

    while (i <= length(input$select_input_vars_alt)) {
      val_inp = append(val_inp, as.character(abc[[i]]))

      i = i + 1
    }

    val_inp_df = as.data.frame(t(val_inp))
    colnames(val_inp_df) = val_inp

    input_val_react_alt$df = val_inp_df
  }

  if(length(input$select_input_vars_alt) == 1){
    df_toto = input_val_react_alt$df

    updateVarSelectInput(session, "select_input_vars_alt", selected = unique(colnames(df_toto)))
  }
})
```

Obrázok 61 Fragment kódu zabezpečujúci minimálny počet prvkov v elemente *varSelectInput()*

Logika riešenia spočíva v tom, že ak počet prvkov v elemente *varSelectInput()* klesne na počet dva, systém zaznamená, o aké prvky sa jedná, resp. zaznamená a do premennej uloží názvy dvoch zostávajúcich atribútov. Ak sa používateľ pokúsi odstrániť ďalší prvok, čím by celkový počet klesol na hodnotu 1, obsah element *varSelectInput()* sa automaticky aktualizuje na predchádzajúci stav, teda na stav, ktorý bol uložený v predchádzajúcom kroku - v momente, keď v elemente boli dva zostávajúce prvky.

2.3. Proces hľadania asociačných pravidiel

Nasledujúci fragment kódu, ktorý je zobrazený na obrázku nižšie (Obrázok 62), znázorňuje proces hľadania asociačných pravidiel prostredníctvom algoritmu *Apriori*.

```

data_process_alt = reactive({
  select_df = final_dataset_alt$df
  tran_data = as(select_df, "transactions")

  rules = apriori(tran_data, parameter = list(minlen = input$length_value_alt, maxlen = 11,
                                              support = as.numeric(input$support_value_alt), confidence = as.numeric(input$conf_value_alt)))

  if(length(rules) < 500000){
    rulesPar1 = rules[!is.redundant(rules)]

    limit_rhs_vector = as.vector(t(as.character(input$var_select_input_atts_alt)))

    i = 1

    if(length(limit_rhs_vector) != 0){
      i = 1

      while (i <= length(limit_rhs_vector)) {
        rulesPar1 = subset(rulesPar1, !(rhs %pin% limit_rhs_vector[i]))

        i = i + 1
      }
    }

    rules_reactive_val$rules = rulesPar1
    rules_subset = rulesPar1
  } else if(length(rules) >= 500000){
    rules_reactive_val$rules = "too_much"

    rules_subset = "too_much"
  }
})

```

Obrázok 62 Proces hľadania asociačných pravidiel

V prvom kroku je zo zvolenej dátovej množiny vygenerovaná transakčná matica, ktorá je následne využitá ako vstupná entita do funkcie *Apriori()*. Práve funkcia *Apriori()* zabezpečuje hľadanie asociačných pravidiel. Parametre tejto funkcie predstavujú hodnoty minimálnej miery podpory, minimálnej miery spoľahlivosti a minimálnej dĺžky asociačných pravidiel. Všetky tieto vstupy používateľ definuje pomocou numerických posúvačov nachádzajúcich sa v používateľskom rozhraní aplikácie. Taktiež sme stanovili podmienku, že ak počet nájdených asociačných pravidiel presahuje hranicu 500 tisíc, žiadne ďalšie úkony nebudú vykonané a používateľ dostane oznámenie o tom, že je potrebné zmeniť vstupné hodnoty metrík. Ak však počet záznamov túto hranicu nepresahuje, nasleduje odstránenie redundantných hodnôt, resp. asociačných pravidiel. Odstránené budú aj tie asociačné pravidlá, ktorých dôsledková časť je tvorená niektorou z hodnôt atribútov, ktoré používateľ zadefinoval, ako nechcené (pozn. - za predpokladu, že tak urobil, ak nie, tento krok prakticky nič nevykoná). Takto orezaná podmnožina asociačných pravidiel je následne využitá pri ďalších úkonoch v danej sekcii aplikácie.

2.4. Vizualizácia odhadu optimálneho počtu zhlukov

Fragment kódu zobrazený na Obrázok 63 názorňuje spôsob vizualizácie odhadu optimálneho počtu zhlukov, a to prostredníctvom čiarového grafu.

```

output$optimal_num_clus_pam = renderPlot({
  if (silh_anal_rv_btn$doTable == FALSE) return()

  isolate({
    disable("downloadPlot_optnumclas_pam")

    p_3 = opt_num_clust_pam$data_all +
      opt_num_clust_pam$ggtitle +
      opt_num_clust_pam$labs +
      geom_point() +
      geom_line() +
      scale_x_continuous(breaks = seq(2, 10, by = 1)) +
      theme(axis.title.y = element_text(margin = margin(t = 0, r = 25, b = 0, l = 0), hjust = 0.5, size = 13),
            axis.text.y = element_text(size = 11),
            axis.title.x = element_text(margin = margin(t = 25, r = 0, b = 0, l = 0), hjust = 0.5, size = 13),
            axis.text.x = element_text(size = 11),
            plot.title = element_text(margin = margin(t = 0, r = 0, b = 25, l = 0), face = "bold", size = 16, hjust = 0.5))

    opt_num_of_clusters_rv_plots$plot = p_3

    enable("downloadPlot_optnumclas_pam")

    print(p_3)
  })
})

```

Obrázok 63 Fragment kódu vizualizácie odhadu optimálneho počtu zhukov

Ako je možné vidieť na obrázku vyššie, funkciu zabezpečujúcu vykreslenie grafu možno definovať ako *renderPlot()*. Samotná vizualizácia je uložená do premennej *p_3*, pričom v poslednom kroku je daný príkaz *print()*, ktorý slúži pre jej vykreslenie. Výpočet odhadu optimálneho počtu zhukov je realizovaný v inej metóde, a rovnako v nej je definovaná aj funkcia *ggplot()*, ktorá je uložená do reaktívnej hodnoty *opt_num_clust_pam\$data_all*. Do tejto reaktívnej hodnoty sú uložené aj informácie o hlavnom nadpise a názvoch jednotlivých osí. Tieto hodnoty sú následne využité pri vizualizácii tak, ako to je znázornené na Obrázok 63. Pri vizualizácii taktiež definujeme veľkosť písma, jeho hrúbku a rozmery medzier medzi názvom osí a samotným grafom. Vo funkcii *renderPlot()* taktiež definujeme, kedy sa má sprístupniť tlačidlo pre stiahnutie vizualizácie.

2.5. Proces zhukovania prostredníctvom algoritmu PAM

Nasledujúci fragment kódu, ktorý je zobrazený na Obrázok 64 znázorňuje proces zhukovania, a to pomocou algoritmu *PAM*.

```
observeEvent(input$render_pam, {  
  if(input$checkbox_all_ds_pam == TRUE && input$checkbox_selected_ds_pam == FALSE){  
    final_ds_pam = rv_for_pam_clustering$df  
  }else if(input$checkbox_all_ds_pam == FALSE && input$checkbox_selected_ds_pam == TRUE){  
    final_ds_pam = rv_for_pam_clustering$df %>% dplyr::select(!!!input$select_input_vars_pam)  
  }  
  
  df_reactive_pam$dfr = final_ds_pam  
  df_output_pam = final_ds_pam  
  
  input_number_pam = as.numeric(input$num_clus_pam)  
  
  gower_dist = daisy(df_output_pam, metric = "gower")  
  pam_fit = pam(gower_dist, diss = TRUE, input_number_pam)  
  
  pam_algorithm$res = pam_fit  
  pam_algorithm_gower_dist$dist = gower_dist  
})
```

Obrázok 64 Fragment kódu znázorňujúci proces zhlukovania prostredníctvom algoritmu PAM

V prvom kroku sa prispôsobí obsah vstupnej dátovej množiny požiadavkám používateľa. Ten má na výber buď to pracovať s celou dátovou množinou, a teda so všetkými atribútmi, alebo si zvoliť len niektoré z nich. Do premennej „*input_number_pam*“ je priradená hodnota zvolená používateľom, ktorá definuje počet zhlukov, do akého sa majú záznamy zo vstupnej dátovej množiny rozdeliť. Pre výpočet vzdialenosti sme využili metriku, ktorá sa nazýva *Gowerova vzdialenosť* a ktorú sme bližšie charakterizovali v hlavnej časti diplomovej práce, v prvej kapitole. Následne je spustený proces zhlukovania, a to pomocou funkcie *pam()*. Výsledky zhlukovania, rovnako aj výsledky z funkcie *daisy()*, ktorá slúži práve na výpočet *Gowerovej vzdialenosti* sú uložené do reaktívnych hodnôt, keďže sú využívané v ďalších častiach aplikácie.