****

##### Metody vyhledávání situací zvýšené fyzické aktivity u pacientů s diabetem

Semestrální práce

Studijní program:Biomedicínské inženýrství a informatika

Studijní obor: Biomedicínské inženýrství

Autor práce: Bc. Milan Poláček

Vedoucí práce: Ing. Anna Holubová

###### Zadání

Navrhněte algoritmy pro automatické vyhledávání situací zvýšené fyzické aktivity (počet kroků za minutu) spolu s poklesem glykémie (hodnoty v 5-ti minutových intervalech) u pacientů s onemocněním diabetes mellitus. Situace budou vyhledávány z již stávající databáze v rámci projektu Diani. Vytvořené algoritmy rovněž umožní sledovat, jaké má nastavení jednotlivých parametrů (práh pro počet kroků, velikost poklesu glykémie, minimální doba trvání poklesu) vliv na počet nalezených situací.

##### Abstrakt:

V současné době existuje řada softwarových řešení pro stahování a agregaci dat z mobilních zařízení pro podporu při léčbě pacientů s diabetem. Získaná data lze pak vyhodnotit pomocí aplikací třetích stran (např. diasend, NightScout). Tyto aplikace umožňují jen náhled na sbíraná data.

Účelem tohoto projektu bylo doplnit do již existující webové aplikace Diani modul pro vyhledávání segmentů s poklesy glykémie v období zvýšené fyzické aktivity v datech snímaných u pacientů s diabetem. Využitím technologií vyššího programovacího C# a dotazovacího jazyka LINQ tedy implementovat algoritmus, který podle zvolených kritérií nalezne tyto segmenty. Dále zobrazit podrobné informace o nalezených segmentech a možnost je zobrazit v přehledových grafech pro analyzování tohoto poklesu v širší souvislosti s dalšími měřenými veličinami.

Tento modul je součástí webové aplikace Diani, která umožňuje automatický přesun, uložení a hodnocení dat z různých elektronických přístrojů, jakými jsou např. hodnoty glykémie, krevního tlaku, počtu nachozených kroků, tepové frekvence aj. Kromě zobrazení grafů je možné vést i osobní kartu pacienta s informacemi jako je váha, výška, HbA1c (glykovaný hemoglobin), hodnoty sacharidů v jídle, dávky inzulínu, spánek/bdění aj.

Vyhledávání těchto incidentů by mělo přispět ke zkvalitnění self-monitoringu a edukaci pacientů s diabetem v situacích zvýšené fyzické aktivity. Uživatel má takto možnost rychle a jednoduše nalézt části dne, v nichž došlo k chybnému dávkování inzulínu či příjmu sacharidů v důsledku špatné reakce na zvýšenou fyzickou aktivitu. A to bez procházení velkého množství naměřených dat a jejich podrobné vizuální kontroly, která může být nespolehlivá a časově náročná.

Výsledkem této práce je formulář s přehledovou tabulkou nalezených segmentů. Tabulka zobrazuje jednotlivé dny, ve kterých byly incidenty nalezeny. V instancích (řádcích tabulky) jednotlivých dnů je odkaz na přehledové grafy pro vizuální analýzu problematických míst v daném dni a informace o jednotlivých nalezených segmentech (počátek, trvání, pokles glykémie a celkový počet kroků v daném období).

Obsah

[1. Úvod 3](#_Toc474080823)

[2. Přehled současného stavu 4](#_Toc474080824)

[3. Metody 5](#_Toc474080825)

[3.1. Sběr dat 5](#_Toc474080826)

[3.2. Tvorba algoritmů a jejich implementace 5](#_Toc474080827)

[3.3. Testování a datová analýza 9](#_Toc474080828)

[4. Výsledky 11](#_Toc474080829)

[5. Diskuse 13](#_Toc474080830)

[6. Závěr 14](#_Toc474080831)

[7. Reference 15](#_Toc474080832)

##### Seznam užitých zkratek a slovních spojení

|  |  |
| --- | --- |
|  |  |
| aktivity tracker | senzor pro měření aktivity a hloubky spánku (např. náramek Fitbit) |
| telemonitorovací systém | soubor technologií měřících biologické výstupy s vizualizační aplikací a možností jejich sledování na dálku |
| self-monitoring | pravidelné měření (např. glykémie) a reakce měřené osoby na změřenou veličinu |
| C# | aplikační webový framework (aplikační rámec) implementující návrhový vzor model-pohled-řadič (model-view-cotroller) |
| LINQ | Microsoft standardizovaný strukturovaný dotazovací jazyk (Structured Query Language) |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Úvod

Fyzická aktivita je jedním z hlavních parametrů, který významně ovlivňuje hladinu cukru v krvi u pacientů s diabetem. Mechanismus využívání glukózy k získávání energie v průběhu vykonávání fyzické aktivity je velmi komplexní a v porovnání s bez onemocnění diabetem se u těchto pacientů liší.

Zatímco u zdravých jedinců je regulace inzulínu v průběhu zvýšené fyzické aktivity zcela automatická (u aerobního pohybu je to snížení sekrece inzulínu a zvýšení kontraregulačních hormonů, které podpoří vyplavování zásobního glykogenu). Pacient, zejména s diabetem 1. typu, musí administraci inzulínu regulovat jeho subkutánním dodáním a to prostřednictvím tzv. pera či pumpy na základě vlastního uvážení a zkušeností z předešlých podobných situací. Příliš vysoká dávka inzulínu může zapříčinit velmi strmý pokles glykémie vedoucí často k závažné a život ohrožující hypoglykémii. Naopak nedostatek inzulínu v těle neumožní glukóze vstoupit do některých buněk a tělo si pak získává energii nefyziologickým způsobem za vzniku toxických látek, které mohou vézt k život ohrožující ketoacidóze, tedy překyselení organizmu. [1][2][3]

Znalost chování glykémie v průběhu různého druhu a intenzity fyzické aktivity jsou pro pacienty klíčovou informací pro správné nastavení inzulínových dávek v takovýchto situacích.

Dnešní možnosti sledování fyzické aktivity prostřednictvím mobilní a nositelné elektroniky spolu s kontinuální monitorací glykémie představují efektivní způsob, jak zpřístupnit pacientům s diabetem zpětnou vazbu o chování glykémie v období zvýšené fyzické aktivity.

Webová aplikace Diani tyto a některé další parametry (dávkování inzulínu a množství konzumovaných sacharidů) umožňuje monitorovat a zobrazovat v reálném čase. Intenzita fyzické aktivity ve formě počtu kroků za minutu spolu s hodnotami glykémie měřené v 5-ti minutových intervalech jsou velmi cenným nástrojem pro pochopení chování glykémie v různých situacích.

Vzhledem k velkému množství dat však ani samotní pacienti nejsou dosud schopni poučit se zcela ze všech situací, které se tímto způsobem podaří zaznamenat.

Cílem této práce bylo pomoci pacientům nalézt situace se zvýšenou fyzickou aktivitou, které zapříčinily pokles glykémie, a tím snadněji nalezli v datech místa, v nichž byla regulace glykémie problematická, a poučili se z takovýchto záznamů pro příští podobnou situaci.

Tato analýza dat může dále sloužit i pro vědecké účely v souvislosti se zkoumáním vlivu různé intenzity fyzické aktivity na velikost poklesu glykémie u pacientů s diabetem.

# Přehled současného stavu

Současná mobilní a nositelná elektronika umožňuje kontinuální sběr dat, jejich ukládání, zpracování, zobrazení a zejména pak komplexní analýzu. Pro pacienty s chronickým onemocněním jako je diabetes mellitus a potažmo pro jejich lékaře je právě analýza velkého množství dat získaných z těchto zařízení klíčovým prostředkem pro zvýšení podpory jejich rozhodovacích procesů a odhalení situací, v nichž má pacient s diabetem problém zvládnout své glykemické výkyvy.

V dnešní době existuje mnoho webových aplikací, které umožňují sběr, analýzu a zobrazení dat sbíraných z mobilních přístrojů jako jsou glukometry, inzulínové pumpy či kontinuální monitory glykémie. Z hlediska monitorace fyzické aktivity má v dnešní době pacient několik možností. V nejlepším případě již existují mobilní, webové či desktopové aplikace, které jsou schopny přijímat data naměřená komerčně dostupným aktivity trackerem, případně umožňují propojení například s jinou mobilní aplikací, která získává takováto data skrze funkce chytrého telefonu [4][5][6][7]. V případě takovýchto aplikací je pak jedinou komplexnější analýzou pouze vizuální porovnání uživatelem grafických či numerických výstupů z jednotlivých zařízení. Například porovnání křivky glykémie s grafem změn intenzity fyzické aktivity v určitém časovém intervalu.

Nalézt však mezi těmito typy dat závislosti, které by umožnily pacientům či lékařům efektivně a včasně reagovat na změny v léčbě, případně se poučit z jich proběhlých situací, vyžaduje adekvátní znalost dané problematiky, nehledě na časovou náročnost při detailním prohlížení velkého množství sbíraných dat.

V rámci vývoje telemonitoringovacího systému pro pacienty s chronickým onemocněním diabetes mellitus na Společném pracovišti FBMI ČVUT a 1. LF UK je vyvíjena i webová aplikace Diani. Tato aplikace umožňuje sběr, ukládání, zobrazení a analýzu dat z glukometru, kontinuálního monitoru glykémie, inzulínové pumpy a diabetického deníku. Získaná data jsou přehledně zobrazena uživateli ve formě grafů a tabulek. Systém zobrazuje mj. i alarmy upozorňující na kritické hodnoty glykémie.

Dalším krokem je tedy vyhledání a identifikace stavů zvýšené fyzické aktivity, která zapříčiňuje pokles glykémie, díky čemuž se pak uživatel bude moci poučit o chování glykémie v různých situacích zvýšené fyzické aktivity a případně pak předejít nežádoucím glykemickým výkyvům v příští podobné situaci. Toto řešení by tedy vyplnilo nedostatky dosavadních podobných systémů, které se omezují pouze na zobrazení či manuální registraci jednotlivých dat.

# Metody

Následující kapitoly popisují proces, kterým bylo dosaženo výše popisovaných cílů. Jedná se o sběr dat, díky němuž bylo možné definovat podmínky pro nalezení požadovaných situací. Návrh a implementace vyhledávacích algoritmů a jejich následná implementace a vizualizace do webové aplikace Diani. Posledním bodem procesu je testování vytvořených algoritmů na nasbíraných datech a následná analýza odborníkem (Ing. Holubovou) pro vyhodnocení správnosti detekčního systému.

# Sběr dat

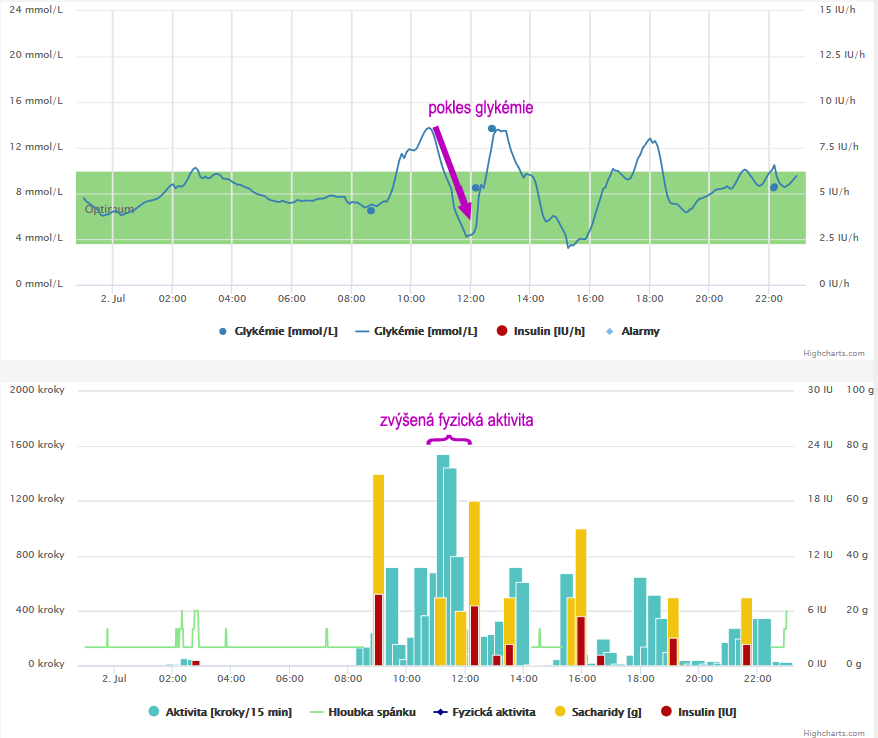
Sdílení dat bylo schváleno v rámci klinické studie ve Fakultní nemocnici Motol lékařskou etickou komisí. Data jsou sbírána od pacientů s diabetem vybavenými zařízeními pro self-monitoring. Set zařízení obsahuje osobní glukometr přenášející automaticky hodnoty glykémie do mobilní aplikace diabetického deníku. Dále smartphone s aplikací diabetického deníku pro manuální registrace glykémií, inzulínových dávek, konzumovaných sacharidů a fyzické aktivity. Do setu dále patří krokoměr pro automatický přenos dat určujících míru fyzické aktivity a kontinuální monitor glykémie, který zaznamenává hodnoty glykémie v 5-ti minutových intervalech.

Anonymizovaná nasbíraná data jsou automaticky přenášena do relační databáze. Tato data je možné zobrazit prostřednictvím webové aplikace Diani ve formě grafů a tabulek.

# Tvorba algoritmů a jejich implementace

Pro nalezení správných a požadovaných segmentů ve sbíraných datech bylo třeba definovat charakter těchto dat a podmínky, které budou daný segment vymezovat.

Pro hledání segmentů obsahujících informace o zvýšené fyzické aktivitě spolu s poklesem glykémie byly vymezeny 2 monitorované veličiny, a to glykémie měřenou v 5-ti minutových intervalech a dále velikost intenzity fyzické aktivity, tj. počet kroků měřených v 1-minutových intervalech.



Obr. 1 Grafické znázornění naměřených dat (aplikace Diani) s vyznačeným příkladem hledaného segmentu

Na základě stanovených kritérií (viz Obr. 1) Ing. Holubovou jsem navrhl specifikaci, kterou jsem po konzultaci formuloval do následujících vstupních podmínek pro hledání:

* rozmezí velikosti poklesu glykémie, tj. rozdíl počáteční a koncové hodnoty glykémie v segmentu
* minimální dobou trvání poklesu glykémie v segmentu
* rozmezím intenzity fyzické aktivity, tj. počtem kroků/5 minut

Z těchto pravidel jsem vytvořil prvotní implementaci vyhledávacího algoritmu v jazyce C#.

Protože se jedná o dvě různé měřené veličiny, bylo třeba určit tzv. řídcí měřenou veličinu, podle níž budou vyhledávána data pro další zpravování. Na základě znalosti self-monitoringu, byla řídící měřenou veličinou určena glykémie z kontinuálního monitoru, protože tato data jsou sbírána jen příležitostně a v krátkých časových úsecích (1-3 týdny) z důvodu omezeného množství dostupných senzorů. Data z kontinuálního monitoru glykémie (dále jen CGM) jsou vybrána z databáze a zpracována na serverové straně aplikace Diani.

Po nalezení dat s poklesem glykémie jsou data rozdělena do segmentů, resp. do seznamu obsahujícího datové struktury typu *DateTimeOffset* s informací o počátku a konci segmentu. Segmenty, které jsou vzájemně vzdáleny na časové ose v intervalu do 11 minut, jsou sloučeny. Tato konstanta byla určena na základě konzultace s Ing. Holubovou. Konstanta ošetřuje možné výpadky měření senzoru CGM nebo krátkodobého nárůstu glykémie, který přerušuje jinak kontinuální pokles glykémie.

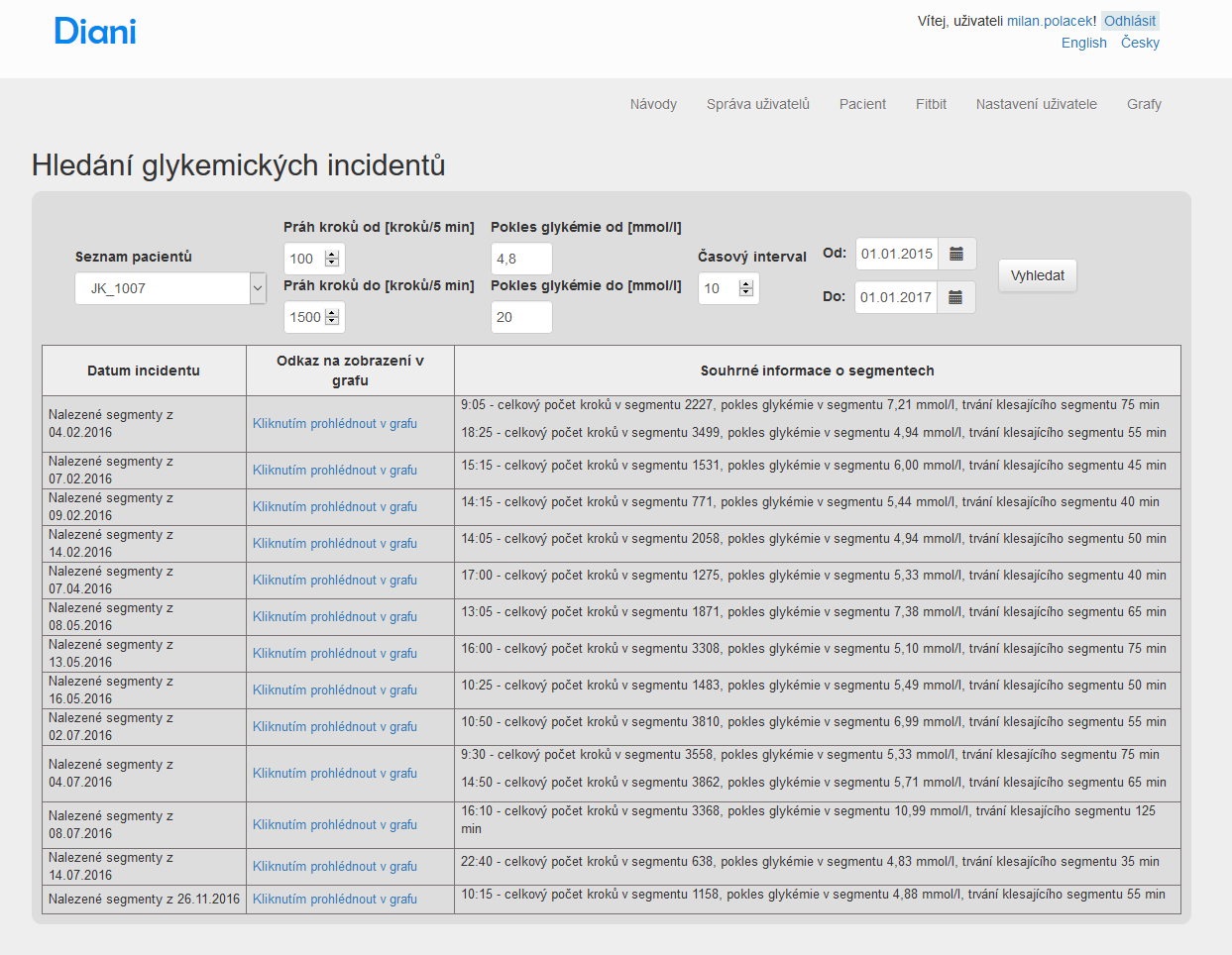
Na základě segmentů s poklesem glykémie (řádově desítky minut) jsou vytvořeny širší segmenty pro výběr dat fyzické aktivity. Tyto širší segmenty (9 denní) byly zvoleny pro zrychlení dotazů do databáze (snížení počtu navazování spojení). Každý širší segment je po výběru z databáze kontrolován dle zadaných kritérií. Ze širšího segmentu jsou vybrána data pomocí dotazovacího jazyka LINQ na základě segmentů poklesu. Specifický segment fyzické aktivity je dále testován, zda vyhovuje zadaným kritériím. V datech se zpočátku zobrazovaly kratší segmenty, než byly odborníkem určeny.

Algoritmus byl tedy doplněn o podmínku ošetřující, kdy v případě aktivity mimo kritéria je vzorek zařazen do hledané oblasti, je-li aktivita dva vzorky nazpět (10 min) v nastavených mezích. Tato konstanta byla určena na základě předešlých výsledků, kdy nalezené výsledky byly rozděleny do více segmentů, které byly od sebe vzdáleny do 10 minut.

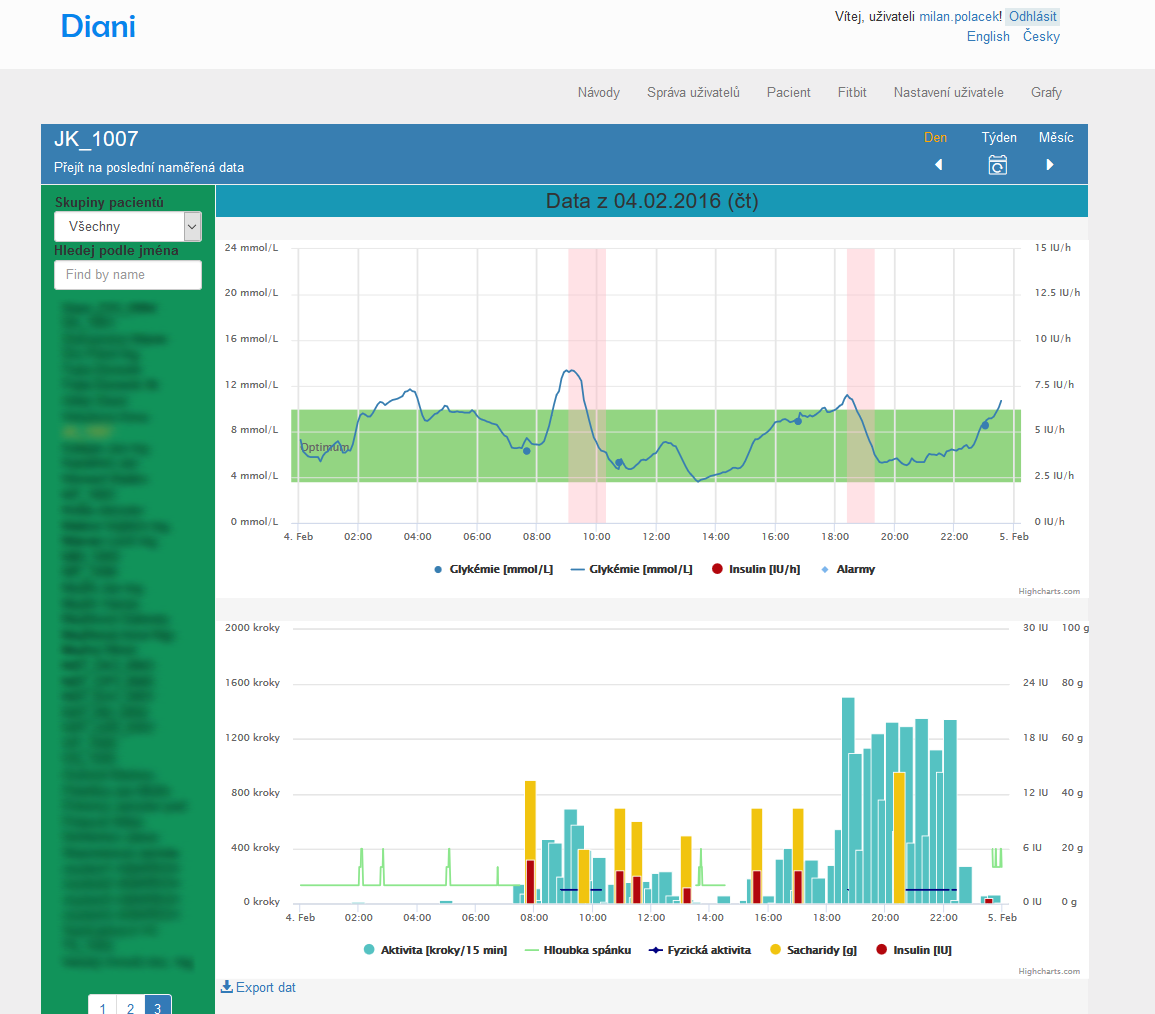
Při zařazení segmentu je uložena informace o počátku, konci a počtu kroků (vykonané aktivity) v daném segmentu.

Na základě výše zmíněných segmentů fyzické aktivity vybraných dle kritérií jsou vybrána naměřená data glykémie a v případě, že pokles glykémie v daném segmentu splňuje podmínky kritérií, je daný segment zachován a je zde uložena navíc informace o hodnotě absolutního poklesu glykémie v daném segmentu (tedy rozdílu počáteční a koncové hodnoty v segmentu).

Výsledná data jsou následně zobrazena (viz Obr. 2) v tabulkovém seznamu s odkazem na přehledový graf v aplikaci Diani (viz Obr. 3).



Obr. 2 GUI Diani s výsledekem vyhledávaných dat pro zadaná kritéria pro celé období měření



Obr. 3 Příklad jednodenního přehledového grafu změřených veličin se výrazněnými segmenty ve webové aplikaci Diani.

# Testování a datová analýza

K testování vytvořených algoritmů bylo využito dat nasbíraných od 3 pacientů s diabetem 1. typu, kteří po čas nošení kontinuálního monitoru glykémie používali současně i krokoměr (tzv. set). Tito pacienti byli vybráni z důvodu jejich dlouhodobého sběru dat. Informace o množství naměřených dat u jednotlivých pacientů je zobrazen v Tab. 1.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Celkový počet monitorovaných dní | Celkový počet naměřených kroků | Celkový počet hodin s CGM | Počet hodnot z CGM |
| Pacient č. 7 | 913 | 8243594 | 4188 | 49321 |
| Pacient č. 70 | 368 | 3595252 | 1356 | 14630 |
| Pacient č. 135 | 319 | 1996371 | 1479 | 17171 |

Tab. 1 Informace o množství naměřených dat z CGM a krokoměru u jednotlivých pacientů

Kontrola správného vyhledávání segmentů algoritmem byla provedena vedoucím práce (expertem) u náhodně nastavených parametrů u 3 pacientů. Kontrola proběhla nad exportovanými daty v csv (comma separated values) formátu.

Pro ověření funkčnosti slučování segmentů byla dále provedena kontrola podílu počtu 5 minutových součtů kroků v segmentu, které spadaly do nastaveného rozmezí intenzity fyzické aktivity, vůči počtu těch, které spadaly mimo toto rozmezí.

# Výsledky

Při nastavení požadovaných kritérií pro nalezení hledaných segmentů bylo možné vyhledat situace, při nichž za předpokladu stejné doby trvání a intenzity fyzické aktivity došlo k obdobnému poklesu glykémie. Příklady takovýchto segmentů spolu s nastavenými parametry pro každého ze tří testovaných pacientů lze vidět v níže uvedených tabulkách (viz Tab. 2, Tab. 3, Tab. 4). Kritéria byla volena na základě doporučení Ing. Holubové.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Datum incidentu | Počátek segmentu | Trvání | Celkový počet kroků v segmentu | Pokles glykémie v segmentu |  | Průměrný pokles glykémie/100kroků |
| 22.3.2014 | 14:20 | 45 | 3511 | 3,22 |  | 0,09 |
| 13.1.2016 | 14:05 | 40 | 3647 | 3,02 |  | 0,08 |
| 6.2.2016 | 0:45 | 40 | 3198 | 3,44 |  | 0,11 |
| průměr |  |  | 3452 | 3,23 | průměr | 0,09 |
|  |  |  |  |  | Směrodatná  odchylka | 0,01 |

Tab. 2 Vybrané nalezené segmenty pro pacienta č. 7 s kritérii 400-600 kroků/5min, pokles glykémie 3-4 mmol/l a minimální délka segmentu 25 min

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Datum incidentu | Počátek segmentu | Trvání | Celkový počet kroků v segmentu | Pokles glykémie v segmentu |  | Průměrný pokles glykémie/100kroků |
| 27.8.2015 | 16:45 | 35 | 1697 | 2,00 |  | 0,12 |
| 31.8.2015 | 17:20 | 35 | 2083 | 2,94 |  | 0,14 |
| 2.12.2016 | 16:50 | 30 | 1769 | 2,55 |  | 0,14 |
| průměr |  |  | 1850 | 2,50 | průměr | 0,13 |
|  |  |  |  |  | Směrodatná  odchylka | 0,01 |

Tab. 3 Vybrané nalezené segmenty pro pacienta č. 70 s kritérii 200-400 kroků/5min, pokles glykémie 2-3 mmol/l a minimální délka segmentu 20 min

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Datum incidentu | Počátek segmentu | Trvání | Celkový počet kroků v segmentu | Pokles glykémie v segmentu |  | Průměrný pokles glykémie/100kroků |
| 4.2.2016 | 18:40 | 35 | 2609 | 3,06 |  | 0,12 |
| 2.7.2016 | 11:15 | 30 | 2515 | 3,61 |  | 0,14 |
| 9.7.2016 | 14:05 | 35 | 2031 | 3,38 |  | 0,17 |
| průměr |  |  | 2385 | 3,35 | průměr | 0,14 |
|  |  |  |  |  | Směrodatná  odchylka | 0,02 |

Tab. Vybrané nalezené segmenty pro pacienta č. 135 s kritérii 400-600 kroků/5min, pokles glykémie 3-4 mmol/l a minimální délka segmentu 25 min

V datech bylo nalezeno více segmentů, než je uvedeno instancí ve výše uvedených tabulkách. Počet nalezených segmentů pro jednotlivá nastavení pro pacienta číslo 7 bylo 7, pro pacienta č. 70 bylo 6 segmentů a pro pacienta č. 135 byly 3 segmenty. Analýza byla provedena nad daty v období od 1.1.2014 (počátek studie) do 1.1.2017. Výběr těchto konkrétních segmentů ve výše uvedených tabulkách byl učiněn proto, že algoritmus vyhledávání má jen pomoci expertovi nalézt segmenty dle zadaných kritérií. Výběr segmentů s podobným charakterem (doba a intenzita zátěže, absolutní pokles glykémie) je právě na expertovi (popř. pacientovi).

Výsledky testování funkčnosti slučování segmentů jsou zobrazeny v příloze B (viz Tab. 5, Tab. 6, Tab. 7, Tab. 8, Tab. 9).

# Diskuse

Nový modul do webové aplikace Diani pro vyhledávání má svá omezení. Jedná se o řadu několika jednodušších filtrů, které se snaží nalézt podobné incidenty.

Protože se jedná o zcela nový přístup k analýze dat u pacientů s diabetem, nelze ověřit věrohodnost výsledků. A je třeba spoléhat na analytickou kontrolu expertem. V budoucnu je budou také testovat další lékaři a inženýři podílející se na této studii.

Na základě mé vizuální analýzy dat v jednotlivých segmentech jsem nalezl nesprávně označené (sloučené) segmenty, které ale expert označil za správně určené. Do přílohy byli doplněny tabulky nalezených segmentů v období od 1.2.2016 od 18.2.2016 u pacienta číslo 135 s minimální poklesem 3 mmol/l v segmentu a minimální délkou segmentu 10 minut. Jak je z hodnot tabulek v příloze patrné, podíl těchto hodnot kolísá ve všech zvolených kritérií aktivity (počtu kroků za 5 minut). Tedy dalším možným vylepšením algoritmu je kritérium pro hledání segmentů s určitým poměrem hodnot aktivity v zadaných kritérií a mimo ně.

Další doplňující kritérium bude omezení délky segmentů shora. V současném algoritmu je možno jen určit minimální délku segmentu. V původním návrhu byl předpoklad, že ostatní kritéria (fyzická aktivita a absolutní pokles glykémie) omezí vyhledávání na dostatečně specifické incidenty a kritérium minimální délky segmentu bude jen doplňující. Z naměřených hodnot se ale později zjistilo, že aktivita kolísá ve značně velkém rozptylu (100 kroků/5min) a bylo by vhodné omezit délku segmentu z obou stran intervalu.

Data jsem se po konzultaci s Ing. Holubovou rozhodl nepředzpracovávat, protože zatím nejsou známy postupy pro vyhledávání jednotlivých incidentů a je vhodné je pro vizuální analýzu zachovat nezměněná. Tento fakt ale nevylučuje, že v budoucnu budou data pro vyhledávácí algoritmus předzpracovávána např. dolní propustí, aby se potlačily výkyvy a kolísání v měřených veličinách.

# Závěr

Vytvořená aplikace umožňuje automaticky vyhledávat situace zvýšené fyzické aktivity související s poklesem glykémie. Díky tomuto řešení se může pacient poučit z předešlých podobných situací, při nichž došlo k nežádoucím výkyvům glykémie, a provést změny léčebného režimu tak, aby bylo takovýmto výkyvům v příští podobné situaci již zamezeno. Pozitivním vlivem může být i změna přístupu pacienta k vykonávání aktivit, kterým se dříve vyhýbal v důsledku špatného zvládání glykemických výkyvů v jejich průběhu.

Tento nástroj je vhodný i pro výzkumné účely z hlediska zkoumání vlivu různé intenzity fyzické aktivity na míru poklesu glykémie u pacientů s diabetem.

Aplikace bude kontinuálně upravována dle požadavků pacientů a lékařů, kteří systém testují. Dalším plánovaným rozšířením je zavedení kritéria pro hledání poklesů s následkem hypoglykémie.

Výsledky této práce budou prezentovány na 52. diabetologických dnech v Luhačovicích.

# Reference

[1] RUŠAVÝ, Z., BROŽ,J., Diabetes a sport. 2012, Praha: Maxdorf. 183. ISBN: 978-80-7345-289-6

[2] NAGI, D.K., *Exercise and sport in diabetes*. 2nd ed. 2006: Wiley. 236. ISBN: 978-

0470022061

[3] REGENSTEINER, Judith G. Diabetes and exercise. New York, NY: Humana Press, c2009. ISBN 9781597452601.

[4] iPro 2 System, Medtronic, https://investor.fitbit.com/press/press-releases/press-release-details/2016/Medtronic-and-Fitbit-Partner-to-Integrate-Health-and-Activity-Data-into-New-CGM-Solution-for-Simplified-Type-2-Diabetes-Management/default.aspx

[5] Diasend, [online] 2016. Dostupné z: <https://www.diasend.com/cs/>

[6] Glooko, [online] 2016. Dostupné z: https://www.glooko.com/

[7] Cellnovo. Cellnovo System. [online] 2016. Dostupné z: http://www2.cellnovo.com/applications

###### Přílohy

1. Zdrojový kód algoritmu

public void FindGlycemiaIncidents()

{

StructsBL \_structsBL = new StructsBL();

TimeStepsFunctions \_timeSteps = new TimeStepsFunctions();

TimeGlucoseFunctions \_timeGlucose = new TimeGlucoseFunctions();

int allowedTimeSpace = 11;//in minutes

int measuringTimeConstant = 6;//in minutes

int stepThresholdFrom = glycemiaIncident.StepThresholdFrom;//in 5 minute

int stepThresholdTo = glycemiaIncident.StepThresholdTo;//in 5 minute

double pokles = glycemiaIncident.DecreaseGlycemiaFrom;

List<MultiChannelSegment> bloodGlucoseContinualSegments = \_structsBL.GetServerDataOfMeasuredSegments(glycemiaIncident.ClientId, PhysicalQuantityEnum.BloodGlucose);

List<TimeValue> continulaBloodGlucoses = \_timeGlucose.BGlucoseContinualPerTimeList(bloodGlucoseContinualSegments).OrderBy(cgm => cgm.Time).ToList();

TimeValue previous1ValueCBG = continulaBloodGlucoses.FirstOrDefault();

List<TimeValue> decreasingCGM = new List<TimeValue>();

foreach (var valueCBG in continulaBloodGlucoses)

{

//rule for skip first value because we are remember

if (previous1ValueCBG.Time == valueCBG.Time)

{

continue;

}

if ((valueCBG.Time - previous1ValueCBG.Time).TotalMinutes < measuringTimeConstant)

{

double? decreaseValueCGM = valueCBG.Value - previous1ValueCBG.Value;

if (decreaseValueCGM < 0 )

{

//adding begin of segment

if (decreasingCGM.Count > 0)

{

if (decreasingCGM.Last().Time != previous1ValueCBG.Time)

decreasingCGM.Add(previous1ValueCBG);

}

else

{

//adding when its begin of measuring

decreasingCGM.Add(previous1ValueCBG);

}

//add present (last) value

decreasingCGM.Add(valueCBG);

}

}

previous1ValueCBG = valueCBG;//memorizing prev value

}

//grouping decreasing data to segments

var groupedDecreasingSegments = SplitToSegments(decreasingCGM, allowedTimeSpace).Where(seg => seg.Item2.Count > 2).ToList();

//select decreasing segment daterange

List<DateTimeRange> dateRangeDecreasingSegments = groupedDecreasingSegments.Select(dgr => dgr.Item1).ToList();

//optimalization fro database request

List<DateTimeOffset> listDayToSelect = dateRangeDecreasingSegments.Select(a => new DateTimeOffset(a.Since.Date.Ticks, a.Since.Offset)).Distinct().ToList();

listDayToSelect.Remove(DateTimeOffset.MinValue);

List<DateTimeRange> optimalizedRangesForDatabase = new List<DateTimeRange>();

DateTimeOffset previousDate = DateTimeOffset.MinValue;

foreach (var currentDayDate in listDayToSelect)

{

//rule for skip first value because we are remember

if (previousDate == DateTimeOffset.MinValue)

{

previousDate = currentDayDate;

optimalizedRangesForDatabase.Add(new DateTimeRange() { Since = currentDayDate.AddDays(-1), Until = currentDayDate.AddDays(1) });

continue;

}

if ((currentDayDate - optimalizedRangesForDatabase.Last().Since ).TotalDays <= 7)

{

optimalizedRangesForDatabase.Last().Until = currentDayDate.AddDays(1);

previousDate = currentDayDate;

}

else

{

optimalizedRangesForDatabase.Last().Until = previousDate.AddDays(1);

optimalizedRangesForDatabase.Add(new DateTimeRange() { Since = currentDayDate.AddDays(-1), Until = currentDayDate.AddDays(1) });

previousDate = currentDayDate;

}

}

List<Tuple<DateTimeRange,int>> activityMarkUpSegments = new List<Tuple<DateTimeRange, int>>();

//finding activity data according decreasing segments and exlude cgmDecreaseGroup where is without activity (sleeping,sitting, lost activity data)

foreach (var actualDateRange in optimalizedRangesForDatabase)

{

List<MultiChannelSegment> activitySegments = \_structsBL.GetServerDataOfMeasuredSegments(glycemiaIncident.ClientId, actualDateRange, PhysicalQuantityEnum.MinuteSteps);

List<TimeValue> stepsInCurrentSegment = \_timeSteps.StepsPerTimeList(activitySegments, false).OrderBy(s => s.Time).ToList();

List<DateTimeRange> daterangeInSelectedSegment = dateRangeDecreasingSegments.Where(dr => dr.Since > actualDateRange.Since && dr.Since < actualDateRange.Until).ToList();

foreach (var currentDateRangeSeg in daterangeInSelectedSegment)

{

TimeSpan interval = new TimeSpan(0, 5, 0); // 5 minutes.

DateTimeRange roundedDateRange = new DateTimeRange()

{

Since = new DateTimeOffset((currentDateRangeSeg.Since.Ticks / interval.Ticks) \* interval.Ticks, currentDateRangeSeg.Since.Offset),

Until = new DateTimeOffset((currentDateRangeSeg.Until.Ticks / interval.Ticks) \* interval.Ticks, currentDateRangeSeg.Until.Offset)

};

List<TimeValue> steps = stepsInCurrentSegment.Where(sv=> roundedDateRange.Since < sv.Time && sv.Time < roundedDateRange.Until).ToList();

List<TimeValue> filteredSteps = new List<TimeValue>();

//fast protection for finded unless segments

if (steps.Count()<= glycemiaIncident.TimeInterval/5)

{

continue;

}

for (int i = 0; i < steps.Count(); i++)

{

if (steps[i].Value >= stepThresholdFrom && steps[i].Value<= stepThresholdTo)

filteredSteps.Add(steps[i]);

else

{

bool added = false;

//rule adding lower then threshold if old value is over threshold

if (i > 0)

{

if (steps[i - 1].Value >= stepThresholdFrom && steps[i-1].Value <= stepThresholdTo)

{

added = true;

filteredSteps.Add(steps[i]);

}

}

if (i > 1 && added == false)

{

if (steps[i - 2].Value >= stepThresholdFrom && steps[i-2].Value <= stepThresholdTo)

filteredSteps.Add(steps[i]);

}

}

}

var markUpSegmentsInSegment = SplitActivityToSegments(filteredSteps, allowedTimeSpace).Where(div => (div.Until - div.Since).TotalMinutes > glycemiaIncident.TimeInterval).ToList();

//filler for segment info

List<Tuple<DateTimeRange, int>> stepsSegmentsWithTotals = new List<Tuple<DateTimeRange, int>>();

foreach (var stepsDaterange in markUpSegmentsInSegment)

{

if (!activityMarkUpSegments.Exists(sst=> sst.Item1.Since== stepsDaterange.Since && sst.Item1.Until == stepsDaterange.Until)) {

stepsSegmentsWithTotals.Add(new Tuple<DateTimeRange, int>(stepsDaterange, (int)steps.Where(sv => stepsDaterange.Since <= sv.Time && sv.Time <= stepsDaterange.Until).Sum(a => a.Value)));

}

}

activityMarkUpSegments.AddRange(stepsSegmentsWithTotals);

}

}

List<Tuple<DateTimeRange, int,double>> markUpSegments = new List<Tuple<DateTimeRange, int,double>>();

foreach (var activityDateRangeWithTotals in activityMarkUpSegments)

{

//inaccurate data search, to conzultate with Anna

var segmentDecCGM = decreasingCGM.Where(cgmd => activityDateRangeWithTotals.Item1.Since < cgmd.Time && cgmd.Time < activityDateRangeWithTotals.Item1.Until)

.OrderBy(val=>val.Time).ToList();

double decreaseGlycemiaInSegment = (double)(segmentDecCGM.FirstOrDefault().Value - segmentDecCGM.LastOrDefault().Value);

if ( glycemiaIncident.DecreaseGlycemiaFrom <= decreaseGlycemiaInSegment && decreaseGlycemiaInSegment <= glycemiaIncident.DecreaseGlycemiaTo)

{

markUpSegments.Add(new Tuple<DateTimeRange, int, double>( activityDateRangeWithTotals.Item1, activityDateRangeWithTotals.Item2, decreaseGlycemiaInSegment));

}

else

{

continue;

}

}

//grouped data per day for easier displaying

List<List<Tuple<DateTimeRange, int, double>>> markUpByDays = markUpSegments.GroupBy(tv => new DateTimeOffset(tv.Item1.Since.Year, tv.Item1.Since.Month, tv.Item1.Since.Day, 0, 0, 0, tv.Item1.Since.Offset))

.Select(g => g.OrderBy(tv => tv.Item1.Since).ToList()).ToList();

foreach (var dayIncidents in markUpByDays)

{

string segments = $"{glycemiaIncident.ClientId}?groupFilter=-666&interval=Day&startDate={dayIncidents.FirstOrDefault().Item1.Since.Date}";

List<string> infoAboutSegments = new List<string>();

foreach (var dateRangePair in dayIncidents)

{

segments += $"&{Constants.FormInputDateAreas}={dateRangePair.Item1.Since.DateTime}&{Constants.FormInputDateAreas}={dateRangePair.Item1.Until.DateTime}";

infoAboutSegments.Add($"{dateRangePair.Item1.Since.ToString("t")} - celkový počet kroků v segmentu {dateRangePair.Item2}, pokles glykémie v segmentu {dateRangePair.Item3:f2} mmol/l, trvání klesajícího segmentu {(dateRangePair.Item1.Until - dateRangePair.Item1.Since).TotalMinutes} min");

}

DateTimeOffset dateOfDay = new DateTimeOffset();

if (dayIncidents.FirstOrDefault() != null)

{

dateOfDay = dayIncidents.FirstOrDefault().Item1.Since.Date;

incidentList.Add(new Tuple<DateTimeOffset, string, List<string>>(dateOfDay, segments, infoAboutSegments));

}

}

}

/// <summary>

/// Spliting data to

/// </summary>

/// <param name="timeValueList"></param>

/// <param name="allowedTimeSpace"></param>

/// <param name="interval"></param>

/// <returns></returns>

private List<Tuple<DateTimeRange, List<TimeValue>>> SplitToSegments(List<TimeValue> timeValueList, int allowedTimeSpace, int interval = 6)

{

timeValueList = timeValueList.OrderBy(t => t.Time).ToList();

List<Tuple<DateTimeRange, List<TimeValue>>> grouped = new List<Tuple<DateTimeRange, List<TimeValue>>>();

DateTimeOffset lastTime = new DateTimeOffset();

foreach (var currentTimeValue in timeValueList)

{

if (grouped.Count == 0)

{

grouped.Add(new Tuple<DateTimeRange, List<TimeValue>>(new DateTimeRange(), new List<TimeValue>() { currentTimeValue }));

lastTime = currentTimeValue.Time;

continue;

}

if ((currentTimeValue.Time - lastTime).TotalMinutes < interval)

{

grouped.Last().Item2.Add(currentTimeValue);

}

else

{ //if segments in time tolerance I join them

if ((currentTimeValue.Time - lastTime).TotalMinutes < allowedTimeSpace)

{

grouped.Last().Item2.Add(currentTimeValue);

}

else

{

grouped.Last().Item1.Since = grouped.Last().Item2.First().Time;

grouped.Last().Item1.Until = lastTime;

grouped.Add(new Tuple<DateTimeRange, List<TimeValue>>(new DateTimeRange(), new List<TimeValue>() { currentTimeValue }));

}

}

lastTime = currentTimeValue.Time;

}

return grouped;

}

private List<DateTimeRange> SplitActivityToSegments(List<TimeValue> timeValueList, int allowedTimeSpace, int interval = 6)

{

timeValueList = timeValueList.OrderBy(t => t.Time).ToList();

List<DateTimeRange> grouped = new List<DateTimeRange>();

DateTimeOffset lastTime = new DateTimeOffset();

foreach (var currentTimeValue in timeValueList)

{

if (grouped.Count == 0)

{

grouped.Add(new DateTimeRange() { Since = currentTimeValue.Time, Until = currentTimeValue.Time });

lastTime = currentTimeValue.Time;

continue;

}

if ((currentTimeValue.Time - lastTime).TotalMinutes >= interval)

{

if ((currentTimeValue.Time - lastTime).TotalMinutes >= allowedTimeSpace)

{

grouped.Last().Until = lastTime;

grouped.Add(new DateTimeRange() { Since = currentTimeValue.Time, Until = currentTimeValue.Time });

}

else

{

grouped.Last().Until = currentTimeValue.Time;

}

}

else

{

grouped.Last().Until = currentTimeValue.Time;

}

lastTime = currentTimeValue.Time;

}

return grouped;

}

1. Tabulky

Všechny nalezené segmenty jsou od pacienta číslo 135 v měřeném období 1.2.2016-18.2.2016 s následujícími kritérii:

* minimální absolutní pokles glykémie 3mml/l
* minimální délka segmentu 10 minut
* aktivita dle titulku tabulky

|  |  |  |  |
| --- | --- | --- | --- |
| 5.2.2016 | 15:40 | celkový počet vzorků | 13 |
|  | 16:40 | mimo hranici | 4 |
| 7.2.2016 | 15:10 | celkový počet vzorků | 7 |
|  | 15:35 | mimo hranici | 4 |
| 7.2.2016 | 19:55 | celkový počet vzorků | 12 |
|  | 20:50 | mimo hranici | 3 |
| 9.2.2016 | 14:10 | celkový počet vzorků | 11 |
|  | 15:00 | mimo hranici | 4 |
| 10.2.2016 | 12:10 | celkový počet vzorků | 11 |
|  | 13:00 | mimo hranici | 0 |

Tab. 5 Počty hodnot aktivity v nalezených segmentech pro kritérium aktivity 0-100 kroků/5 min

|  |  |  |  |
| --- | --- | --- | --- |
| 4.2.2016 | 9:05 | celkový počet vzorků | 13 |
|  | 10:05 | mimo hranici | 6 |
| 5.2.2016 | 13:10 | celkový počet vzorků | 10 |
|  | 13:55 | mimo hranici | 6 |
| 5.2.2016 | 15:40 | celkový počet vzorků | 8 |
|  | 16:15 | mimo hranici | 4 |
| 9.2.2016 | 19:10 | celkový počet vzorků | 9 |
|  | 19:50 | mimo hranici | 4 |

Tab. 6 Počty hodnot aktivity v nalezených segmentech pro kritérium aktivity 100-200 kroků/5 min

|  |  |  |  |
| --- | --- | --- | --- |
| 7.2.2016 | 15:15 | celkový počet vzorků | 10 |
|  | 16:00 | mimo hranici | 5 |

Tab. 7 Počty hodnot aktivity v nalezených segmentech pro kritérium aktivity 200-300 kroků/5 min

|  |  |  |  |
| --- | --- | --- | --- |
| 4.2.2016 | 9:00 | celkový počet vzorků | 17 |
|  | 10:20 | mimo hranici | 4 |
| 5.2.2016 | 13:05 | celkový počet vzorků | 11 |
|  | 13:55 | mimo hranici | 4 |
| 5.2.2016 | 15:40 | celkový počet vzorků | 13 |
|  | 16:40 | mimo hranici | 1 |
| 6.2.2016 | 14:05 | celkový počet vzorků | 12 |
|  | 15:00 | mimo hranici | 1 |
| 6.2.2016 | 18:20 | celkový počet vzorků | 9 |
|  | 19:00 | mimo hranici | 5 |
| 7.2.2016 | 15:10 | celkový počet vzorků | 7 |
|  | 15:40 | mimo hranici | 3 |
| 7.2.2016 | 19:55 | celkový počet vzorků | 12 |
|  | 20:50 | mimo hranici | 1 |
| 8.2.2016 | 17:55 | celkový počet vzorků | 11 |
|  | 18:45 | mimo hranici | 1 |
| 9.2.2016 | 14:10 | celkový počet vzorků | 11 |
|  | 15:00 | mimo hranici | 2 |
| 9.2.2016 | 19:10 | celkový počet vzorků | 9 |
|  | 19:50 | mimo hranici | 3 |
| 10.2.2016 | 12:10 | celkový počet vzorků | 11 |
|  | 13:00 | mimo hranici | 0 |
| 10.2.2016 | 17:05 | celkový počet vzorků | 21 |
|  | 17:45 | mimo hranici | 4 |

Tab. 8 Počty hodnot aktivity v nalezených segmentech pro kritérium aktivity 0-200 kroků/5 min

|  |  |  |  |
| --- | --- | --- | --- |
| 4.2.2016 | 9:05 | celkový počet vzorků | 13 |
|  | 10:05 | mimo hranici | 4 |
| 5.2.2016 | 13:10 | celkový počet vzorků | 10 |
|  | 13:55 | mimo hranici | 4 |
| 5.2.2016 | 15:40 | celkový počet vzorků | 8 |
|  | 16:25 | mimo hranici | 5 |
| 6.2.2016 | 18:20 | celkový počet vzorků | 9 |
|  | 19:00 | mimo hranici | 3 |
| 9.2.2016 | 14:15 | celkový počet vzorků | 9 |
|  | 14:55 | mimo hranici | 5 |
| 9.2.2016 | 19:10 | celkový počet vzorků | 9 |
|  | 19:50 | mimo hranici | 3 |

Tab. 9 Počty hodnot aktivity v nalezených segmentech pro kritérium aktivity 100-300 kroků/5 min