FAKULTA INFORMAČNÍCH TECHNOLOGIÍ ČVUT V PRAZE

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Název:	Portál výsledků analýzy dat a dalších informací o STK
Student:	Aleksandra Parkhomenko
Vedoucí:	Ing. Lucie Svitáková
Studijní program:	Informatika
Studijní obor:	Znalostní inženýrství
Katedra:	Katedra aplikované matematiky
Platnost zadání:	Do konce letního semestru 2020/21

Pokyny pro vypracování

Cílem práce je zanalyzovat data o STK (stanice technické kontroly) a navrhnout přístupy pro nalezení různých anomálií v nich. Dále vytvořit jednoduchý portál, který bude tyto výsledky prezentovat a bude zobrazovat i další užitečné informace o stanicích (otevírací doba apod.).

Detailní pokyny:

1. Proveďte analýzu dat o STK (https://data.irozhlas.cz/opendata/).
2. Navrhněte metody, které použijete pro detekci podezřelého chování na stanicích (např. kontroly po zavírací době).
3. Pomocí scrapingu získejte užitečné informace o jednotlivých stanicích (např. otevírací doba, cena kontroly, atd.).
4. Vytvořte webový portál, který bude tyto výsledky zobrazovat. Tedy jak výstupy ne/nalezených nesrovnalostí, tak informace o jednotlivých stanicích či další statistiky.

Seznam odborné literatury

Dodá vedoucí práce.

Ing. Karel Klouda, Ph.D.
vedoucí katedry

doc. RNDr. Ing. Marcel Jiřina, Ph.D.
děkan

V Praze dne 13. ledna 2020

FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE

Bakalářská práce

Portál výsledků analýzy dat a dalších informací o STK

Aleksandra Parkhomenko

Katedra aplikované matematiky
Vedoucí práce: Ing. Lucie Svitáková

4. června 2020

Poděkování

Chtěla bych poděkovat Ing. Lucii Svitákové a Ing. Marku Sušickému za pomoc, cenné rady, trpělivost a čas, které mi v průběhu zpracování bakalářské práce věnovali.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracovala samostatně a že jsem uvedla veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 2373 odst. 2 zákona č. 89/2012 Sb., občanský zákoník, ve znění pozdějších předpisů, tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo"), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu) licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 4. června 2020 . . . . . .

České vysoké učení technické v Praze
Fakulta informačních technologií

Odkaz na tuto práci

Parkhomenko, Aleksandra. *Portál výsledků analýzy dat a dalších informací o STK*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2020.

Abstrakt

Práce se zabývá analýzou otevřeného datasetu, který obsahuje záznamy jednotlivých prohlídek na stanicích technické kontroly v roce 2018, návrhem metod pro odhalení podezřelého chování na stanicích s využitím získaných znalostí a vývojem webového portálu pro reprezentaci výsledků a poskytování dalších užitečných informací pro uživatele. V rámci práce byly navrženy tři metody pro detekci podezřelého chování na stanicích: hledání kontrol po zavírací době, kontrola hustoty prohlídek na základě počtu kontrolních linek a odhalení podezřele častých souběhu značek automobilů. Taktéž byl navržen webový portál, který zobrazuje výsledky aplikování těchto metod.

Klíčová slova STK, otevřená data, webový portál, detekce anomálií

Abstract

The purpose of this bachelor thesis is an analysis of an open dataset which contains records of inspections at the vehicle inspection stations in 2018, designing methods for detecting suspicious behavior at the stations based on the acquired knowledge and development of a web portal for the representation of the results and providing other useful information for users. Three methods for the detection of suspicious behavior at stations were proposed: searching for inspections after the closing time, checking the density of inspections based on the number of inspection lines and detecting suspiciously frequent concurrences of car brands. A web portal has also been designed to display the results of applying these methods.

Keywords opendata, webportal, vehicle inspection, anomaly detection

Obsah

Seznam obrázků

Seznam tabulek

Seznam výpisů

Úvod

V současné době je lidem dostupné nesrovnatelně větší množství informací než kdykoliv v minulosti. Data se sbírají, uchovávají a zpracovávají ve všech sférách života člověka a mají významný vliv na dnešní společnost. Existuje mnoho různých způsobů rozdělení dat na typy: podle zdroje, formátu, účelu využití i podle dalších kritérií. Tato práce se zaměřuje na zvláštní typ – otevřená data. To jsou data, která jsou vzdáleně dostupná a účel jejich využití není omezen.

Jedním z poskytovatelů otevřených dat jsou orgány státní správy. Ty mají k dispozici enormní objem informací, které jsou již nějakou dobu aktivně zveřejňovány. Některé z těchto informací však nejsou dobře strukturované, analyzované a vizualizované a nejsou tedy plnou měrou využity.

Příkladem takového datasetu je STK 2018, což je úplný seznam jednotlivých kontrol na STK v České republice v roce 2018 obsahující skoro čtyři miliony záznamů. Po vhodné úpravě a zpracování těchto dat z nich bude možné vyvodit užitečné závěry. K čemu by mohl být tento dataset použit? Například k detekci podvodů na některých stanicích. Jedním z aktuálních problémů v dopravním sektoru je, že kontrolou procházejí automobily, které by projít neměly. O přesném počtu automobilů nezpůsobilých k provozu lze jen spekulovat. Například v roce 2019 bylo v Čechách pouze 5 % automobilů shledáno „nezpůsobilých“, kdežto v Německu to ve stejném roce bylo 20 %.

Na základě výše uvedených důvodů jsem se rozhodla zpracovat tento dataset a vytvořit webový portál reprezentující výsledky analýzy. Celá práce je rozdělena do čtyř hlavních kapitol:

První kapitola obsahuje teoretickou část práce, obecně pojednává o otevřených datech a následně konkrétně o datasetu STK 2018. V této kapitole jsou dále prozkoumány různé metody analýzy velkých datových souborů a postupy pro práci s nimi.

Druhá kapitola obsahuje první polovinu praktické části práce. Zahrnuje tedy postup pro zpracovávání datasetu, sestavení statistického přehledu a rov-

něž se věnuje aplikaci metod zmíněných v první kapitole na skutečná data.

Třetí kapitola obsahuje návrh webového portálu a poslední kapitola zahrnuje druhou polovinu praktické části práce a zabývá se popisem implementace RESTového API a webového portálu.

Kapitola 1

Cíl práce

Prvním cílem této práce je zpracovat otevřený dataset STK 2018, což by usnadnilo a urychlilo další práci s informacemi a umožnilo získat užitečné znalosti. Zpracovávání se skládá ze:

- vhodné úpravy dat, tzn. převodu původního datového souboru do stavu, ve kterém se s ním bude jednoduše pracovat;
- vyčištění dat, a tedy odstranění chybných a poškozených záznamů;
- provedení analýzy;
- ukládání;
- vytvoření RESTového API pro přístup.

Pro lepší výsledky analýzy bude vhodné k seznamu prohlídek připojit další data. Například seznam STK, který obsahuje základní informace o stanicích a další údaje, které budou dohledány na jejich webových stránkách, například pracovní doby, ceny, případně kapacity.

Druhým cílem je pomocí zpracovaného datasetu a informací o stanicích vytěžených scrapingem navrhnout metody pro detekci podezřelého chování na STK a případně využitím těchto metod zaznamenat takové chování u jednotlivých stanic. Toto je v současné době velmi aktuální, jelikož statistiky prohlídek ukazují neskutečnou úspěšnost a náhodné kontroly automobilů na silnici odhalují zřejmé podvody.

Pro reprezentaci výsledků analýzy a zobrazení získaných informací o stanicích bude vytvořen webový portál, který umožní uživateli v těchto datech vyhledávat a bude obsahovat různé druhy vizualizací statistik, provozní doby, porovnání cen, odhad zátěží s souvislosti s časem apod. Portál by měl také upozorňovat na příznaky podezřelého chování na stanicích technické kontroly.

Kapitola 2

Rešerše

Tato kapitola se zaměřuje na objasnění pojmů a popis dat, které s touto prací souvisí. Jsou v ní obecně popsána otevřená data a následně jsou konkrétně probrány sloupce datasetu STK 2018 a dalších tabulek vhodných ke zpracování v rámci této práce. Dále jsou prozkoumány obvyklé postupy pro analýzu velkých datových souborů a metody vytěžování znalostí z nich. Následně jsou uvedeny a zobrazeny některé existující portály, které jsou spojeny se stanicemi technické kontroly.

2.1 Otevřená data v České republice

V této části první kapitoly je vysvětlen pojem otevřená data a probrán aktuální vztah veřejné správy v České republice ke zveřejňování informací na webu.

2.1.1 Co jsou otevřená data?

Otevřená data jsou dle § 3 odst. 11 zákona č. 106/1999 Sb., o svobodném přístupu k informacím, informace zveřejňované způsobem umožňujícím dálkový přístup v otevřeném a strojově čitelném formátu, jejichž způsob ani účel následného využití není omezen a které jsou evidovány v národním katalogu otevřených dat [6].

2.1.2 Národní katalog otevřených dat

Národní katalog otevřených dat veřejné správy ČR [7] (dále jen NKOD) je webový portál, který obsahuje přes 130 000 veřejných datových sad od 39 různých poskytovatelů. Mezi nimi jsou jednotlivá ministerstva a města, Český statistický úřad, Český úřad zeměměřický a katastrální a další.

V současné době se projekt Otevřená data v České republice rozvíjí rychleji, než kdykoliv v minulosti. Pořád se zvyšuje transparentnost státních or-

gánů a následně počet otevřených pro analýzu a využití datových sad, evidovaných v NKOD.

Datové sady se zveřejňují jak na žádost podle zákona č. 106/1999 Sb., o svobodném přístupu k informacím [6], tak i z vlastní iniciativy institucí.

V NKOD je ke stanicím technické kontroly a státním technickým prohlídkám dostupný pouze jeden dataset, který je středem pozornosti této práce.

2.2 Stanice technické kontroly

V této části je definován pojem stanice technické kontroly, objasněny typy prohlídek a vozidel, sepsána s tím spojená legislativa a jsou také uvedeny některé statistiky důležité pro problém podvodů na stanicích.

Stanice technické kontroly (STK) jsou speciální zařízení, která mají oprávnění k provedení technických prohlídek různých druhů vozidel. Proto, aby toto oprávnění dostaly, musejí splnit všechny požadavky, které jsou uvedené v § 16 vyhlášky č. 211/2018 Sb., o technických prohlídkách vozidel.

2.2.1 Druhy prohlídek

Existuje několik základních typů státních technických prohlídek vozidel:

- pravidelná technická prohlídka;
- technická prohlídka před registrací vozidla;
- opakovaná technická prohlídka;
- technická prohlídka před schválením technické způsobilosti vozidla;
- technická prohlídka ADR;
- evidenční kontrola;
- technická prohlídka na žádost zákazníka [8].

Pravidelnou technickou prohlídkou se rozumí kontrola prováděná v zákonem stanovené lhůtě, která činí 6, 4 nebo 1 rok ode dne zápisu vozidla do registru silničních vozidel a následně pravidelně 1, 2 nebo 4 roky, a to v závislosti na kategorii a hmotnosti vozidla [9]. Provádí se v plném rozsahu, tj. obsahuje všechny kontrolní úkony.

Kontrolními úkony jsou:

- identifikace vozidla, tzn. kontrola registrační značky, identifikačního čísla a povinného štítku výrobce;
- brzdové zařízení: kontrola stavu celého brzdového systému;

- řízení: kontrola mechanického stavu řízení;
- výhledy: kontrola pole výhledu, stavu zasklení a stěračů skla;
- kontrola svítilen, světlometů, odrazek a elektrického zařízení;
- kontrola náprav, kol, pneumatik a zavěšení náprav;
- kontrola podvozku a části připevněné k podvozku;
- kontrola jiného vybavení včetně stavu rychloměru, bezpečnostních pásů, lékárničky a dalších;
- obtěžování okolí: kontrola emisi, hlučnosti;
- a další prohlídky vozidel k dopravě osob kategorie M2 a M3: únikové východy, systém odmrazováni a větrání, sedadel, schodů apod.

Opakovanou technickou prohlídkou se rozumí kontrola prováděná při zjištění vážné nebo nebezpečné závady a uskutečňuje se do 30 dnů od prohlídky, na které byla závada zjištěna. Po uplynutí lhůty 30 dnů se provádí v plném rozsahu, jinak jsou kontrolovány systémy a konstrukční části, na kterých byla závada zjištěna.

Technickou prohlídkou před schválením technické způsobilosti vozidla se rozumí kontrola prováděná tehdy, pokud technická způsobilost vozidla zatím nebyla schválena a vozidlo nemělo registraci v České republice. Provádí se v plném rozsahu.

Technickou prohlídkou ADR (z francouzského Accord européen relatif au transport international des marchandises Dangereuses par Route) se rozumí kontrola vozidla určeného k přepravě nebezpečných věcí, která je doplněna kontrolou shody některých parametrů s platnými požadavky dohody ADR.

Evidenční kontrolou se rozumí kontrola registrační značky, identifikačního čísla a povinného štítku výrobce. Je součástí každé technické prohlídky libovolného vozidla.

Technická prohlídka na žádost zákazníka je technickou prohlídkou, která je prováděná v rozsahu odpovídajícímu požadavkům zákazníka. Po kontrole se vozidlu nepřiděluje kontrolní nálepka a neprovádí se zápis o výsledcích technické prohlídky do technického průkazu vozidla.

Součástí analyzovaného datasetu je také **technická silniční kontrola**, která se liší tím, že se kontrola provádí pomocí mobilní kontrolní jednotky a na základě nařízení policisty.

2.2.2 Důležité informace o prohlídkách

Dalšími důležitými podklady pro smysluplnou analýzu seznamů technických prohlídek je popis typů závad, kapacity kontrolních linek a definice emise.

Závady zjištěné při technických kontrolách se dělí na:

- lehké (**A**): jsou závady, které zásadně neovlivňují způsobilost vozidla. Například poškození některého písmenného nebo číselného znaku registrační značky, nesnadné odjištění parkovací brzdy, vnější poškození brzdového ventilu apod;
- vážné (**B**): jsou závady, které nepřímo ovlivňují bezpečnost provozu vozidla a nepříznivě působí na životní prostředí. Například vytékání brzdové kapaliny z nízkotlaké části kapalinových brzd, slyšitelný únik vzduchu ze vzduchojemu, vážné poškození nájezdové brzdy apod. Také je nezbytné uvést, že při zjištění závady typu B se vozidlo musí dostavit na opakovanou prohlídku ve lhůtě 30 dnů;
- nebezpečné (**C**): jsou závady, které přímo ohrožují bezpečnost provozu a vozidlo je z toho důvodu pro další provoz nezpůsobilé. To je například absence brzdné síly na jednom nebo více kolech, nepohyblivost ovládacího prvku provozní brzdy, nedostatečný účinek nouzového brždění apod [8].

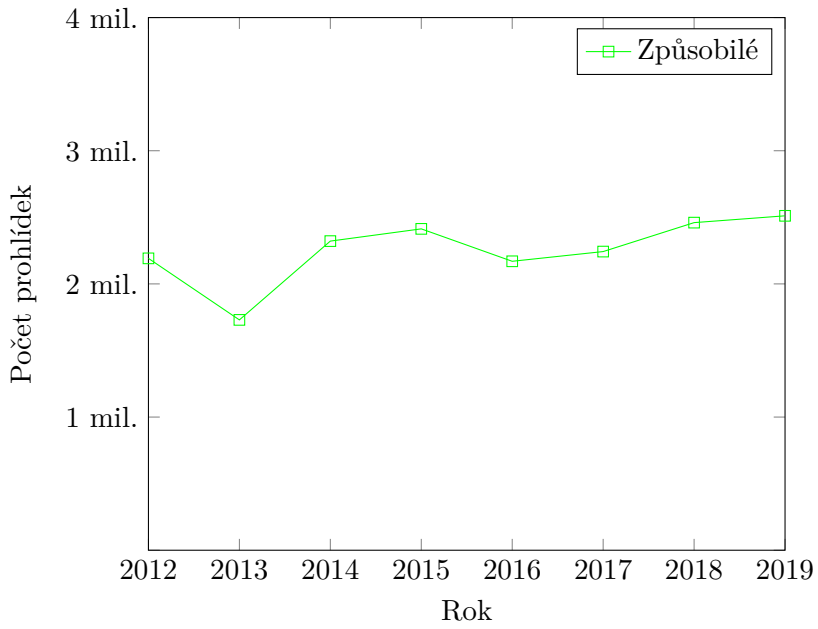
Kapacita kontrolních linek stanic (počet prohlídek za rok) se odhaduje pomocí dosazení hodnot koeficientů aktuálních pro každý rok do speciálních vzorců [8]. Je individuální pro každou stanici a záleží na počtu pracovníků, počtu kontrolních stání na lince a provozní době. Koeficienty jsou zveřejněny Ministerstvem dopravy ČR.

Ministerstvo dopravy taktéž publikuje průměrný počet minut trvání pravidelné prohlídky vozidla (bez měření emisí) a koeficienty pracnosti pro ostatní druhy prohlídek vůči pravidelné.

Například pro vozidla kategorie M1 v roce 2018 platilo:

- 27,91 minut – časová pracnost pravidelné prohlídky (bez měření emisí);
- $27{,}91 \times 0{,}336 = 9{,}38$ minut – časová pracnost evidenční kontroly;
- $27{,}91 \times 0{,}512 = 14{,}29$ minut – časová pracnost opakované prohlídky;
- $27{,}91 \times 1{,}140 = 31{,}82$ minut – časová pracnost technických prohlídek před registrací;
- $27{,}91 \times 1{,}158 = 32{,}32$ minut – časová pracnost prohlídky před schválením technické způsobilosti vozidla.

Měření emise je povinným kontrolním úkonem pro získání osvědčení o technické způsobilosti vozidla. Čím starší je motor, tím více je v něm karbonu, což způsobí snížení jeho výkonu a naopak zvýšení produkce emisí. Vysoké hodnoty emisí představují znečištění okolí, proto jsou považované za vážný problém. Emise se měří v emisních stanicích, často na stejném místě, kde se provádějí ostatní body. Časový rozsah měření je minimálně 30 minut.



Obrázek 2.1: Počet prohlídek po rocích: způsobilé

2.2.3 Statistiky minulých let

Na stránkách ministerstva dopravy jsou ke stažení dostupné celkové souhrnné počty prohlídek provedených na STK a hodnocení způsobilosti a průměrného počtu závad vozidel při pravidelných technických kontrolách z let 2012–2019.

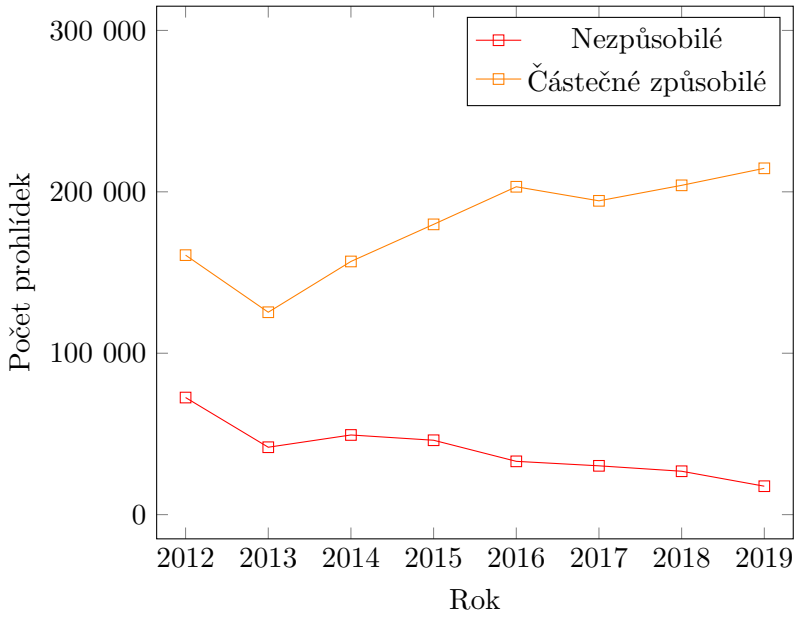
Z těch statistik je vidět, že počet prohlídek s výsledkem „Nezpůsobilé“ má zřemou klesající tendenci (viz obrázek 2.2).

V roce 2016 zpřísnilo Ministerstvo dopravy podmínky kontrol na STK. Od té doby je nutné dodávat fotodokumentaci o prohlídce, nejsou již tolerovány drobné technické úpravy nezapsané do technického průkazu vozidla a zpřísnily se podmínky pro měření emisi. Stále ovšem státní technickou prohlídkou prochází víc, než 90 % automobilů.

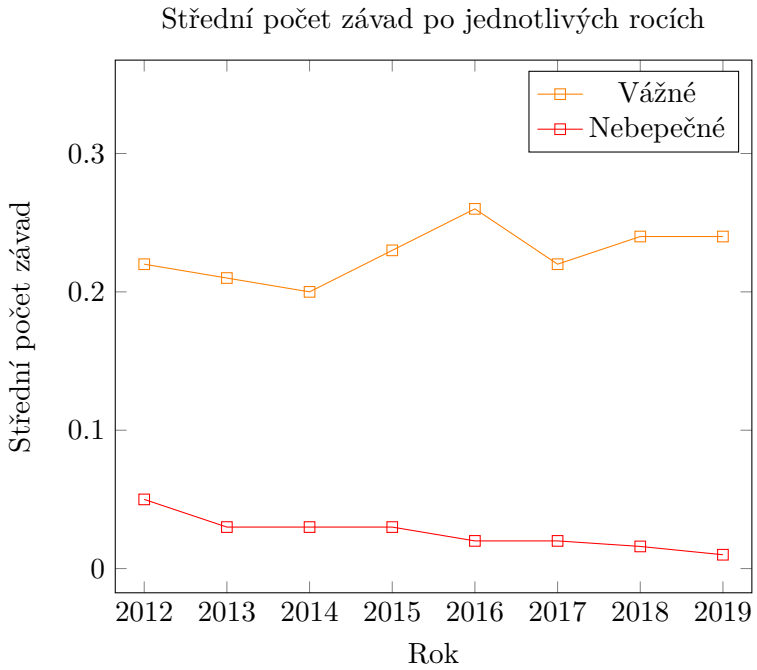
Na obrázcích 2.1, 2.2 a 2.3 je vidět vývoj počtu automobilů způsobilých, nezpůsobilých a částečně způsobilých k provozu a takže střední počty závad zjištěných na prohlídkách mezi roky 2012 a 2019.

2.3 Popis dat

V této části jsou popsána data, která budou zpracována v analytické kapitole této práce. Především jsou definovány sloupce datasetu STK 2018 a pojmy, které s tím souvisí. Potom jsou probrány další pomocné datasety a informace.



Obrázek 2.2: Počet prohlídek po rocích: nezpůsobilé a částečné způsobilé



Obrázek 2.3: Střední počet závad po jednotlivých rocích

2.3.1 STK 2018

Dataset, který je středem pozornosti této práce, je úplný seznam jednotlivých kontrol na stanicích technické kontroly v roce 2018. Je evidován v národním katalogu otevřených dat a byl získan od Ministerstva dopravy podle zákona č. 106/1999 Sb., o svobodném přístupu k informacím.

Tato distribuce datové sady:

- není způsobem výběru nebo uspořádáním obsahu vlastním duševním výtvorem autora;
- nenaplňuje znaky pro vznik práva pořizovatele databáze;
- v sobě neobsahuje autorským právem chráněná díla [10].

Z těchto důvodů není tato datová sada z hlediska autorského práva a sui generis databázových práv chráněná jako celek, stejně jako nejsou chráněny její dílčí části. Každý tak může databázi volně vytěžovat a zužitkovat, tzn. data v ní obsažená volně užívat k jakýmkoli účelům, včetně komerčních.

```
<record
    STK="3114"
    DrTP="Evidencni kontrola"
    VIN="VF3MJAHXHGS280168"
    DatKont="2018-01-02T11:15:08.083"
    TZn="PEUGEOT"
    TypMot="AH01"
    DrVoz="OSOBNI AUTOMOBIL"
    ObchOznTyp="3008"
    Ct="M1"
    DatPrvReg="2017-01-09T00:00:00"
    Km="39227"
    ZavA="0"
    ZavB="0"
    ZavC="0"
    VyslSTK="zpusobile"
    VyslEmise="---"
/>
```

Výpis 2.1: Ukázka řadků z STK2018

Dataset je ke stážení ve formátu XML a obsahuje 3 728 369 záznamů o jednotlivých prohlídkách na stanicích technické kontroly v celé České republice a zahrnuje jak údaje o výsledcích kontroly, tak i údaje o vozidle samotném (viz výpis 2.1).

Seznam a popis sloupců:

- **STK** – unikátní čtyřciferný identifikátor stanice, ve které se prohlídka uskutečnila;
- **DrTP** – druh technické prohlídky: evidenční, pravidelná, opakovaná a jiné. Jsou detailně probrány v sekci 2.2.1;
- **DatKont** – datum a čas provedení kontroly ve formátu yyyy-MM-ddTHH:mm:ss.SSS;
- **ZavA/ZavB/ZavC** – počet objevených závad v průběhu prohlídky, a to typu A – lehká závada, B – vážná závada a C – nebezpečná závada;
- **VyslSTK** – finální výsledek kontroly: způsobilé, nezpůsobilé, částečně způsobilé;
- **VyslEmise** – výsledek měření emise: vyhovuje, nevyhovuje, částečně vyhovuje;
- **TypMot** – konkréntní typ motoru, instalovaného do vozidla. Například AEF, C9DC, 4D56 a jiné. Celkem cca 60 000 různých typů;
- **TZn** – značka (výrobce) vozidla.
- **ObchOznTyp** – model vozidla;
- **DrVoz** – druh vozidla: osobní automobil, traktor, motocykl a jiné;
- **Ct** – kategorie vozidla. Existují:
  - **L** – motorová vozidla zpravidla s méně než čtyřmi koly;
  - **M** – motorová vozidla, která mají nejméně čtyři kola a používají se pro dopravu osob;
  - **N** – motorová vozidla, která mají nejméně čtyři kola a používají se pro dopravu nákladů;
  - **O** – přípojná vozidla;
  - **T** – traktory zemědělské nebo lesnické;
  - **S** – pracovní stroje;
  - **R** – ostatní vozidla, která nelze zařadit do výše uvedených kategorií [9].
- **DatPrvReg** – datum první registrace vozidla (zápisu do registru vozidel) ve formátu yyyy-MM-ddT00:00:00.000;
- **Km** – počet najetých kilometrů;
- **VIN** – vehicle identification number (identifikační číslo vozidla).

2.3.2 Pomocné datasety

Další zdroje informací, které budou použity:

- Seznam STK podle krajů – tabulka, která byla stažená z webových stránek Ministerstva dopravy [11] a obsahuje:
  - číslo STK – unikátní čtyřciferný identifikátor stanice;
  - rozsah oprávnění stanice:
    - OA – Osobní automobil,
    - NA – Nákladní automobil,
    - TRA – Traktor,
    - ZS – Zkušební stanice,
    - ADR – Přeprava nebezpečného nákladu;
  - adresu včetně ulice, města a PSČ;
  - provozovatele STK;
  - kontaktní údaje stanice;
  - ORP – obec s rozšířenou působností;
  - okres;
  - kraj.
- Potřebné podklady pro výpočet kapacitní potřeby technických prohlídek, kapacity kontrolních linek stanic technické kontroly a počtu skutečně provedených technických prohlídek k 1. 1. 2018, které jsou také dostupné z webových stránek ministerstva dopravy [12].

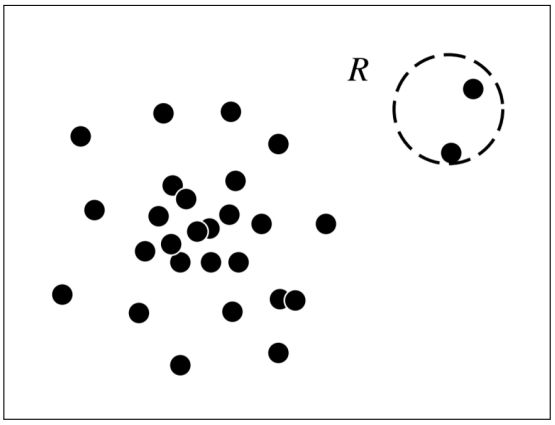
2.4 Postupy pro analýzu

V této části jsou probrány postupy a nástroje pro zpracování a analýzu velkých datových souboru, což zahrnuje postupy pro odhalení anomálií v datech, shlukování a výpočet korelace mezi daty.

2.4.1 Detekce anomálií

Detekce anomálií řeší problém hledání vzorců v datech, která neodpovídají očekávanému chování. Existují následující postupy pro odhalení odlehlých bodů:

- statistické metody – metody, které jsou postavené na myšlence, že anomálie je pozorování, které je považované za částečně nebo zcela irelevantní, protože není generované předpokládaným stochastickým modelem [13] (překlad vlastní);



Obrázek 2.4: Ukázka odlehlých bodů (region R) [1]

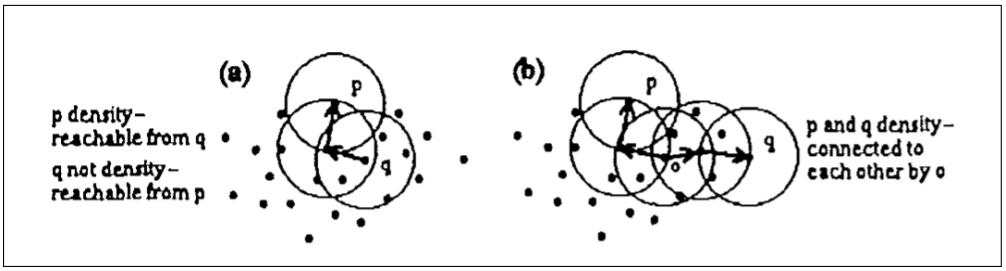
- metody založené na blízkosti datových bodů, považují datový bod za anomálii, pokud má v určitém rádiu příliš málo sousedů, tj. vzdálenost od nejbližších sousedů se výrazně liší od střední vzdáleností v datasetu;

- metody založené na shlukování, předpokládají, že normální datové body patří do velkých a hustých shluků, když anomálie patří do malých nebo dokonce jednoprvkových shluků (viz obrázek 2.4);

- metody založené na klasifikaci, používají unární klasifikátor, natrénovaný na normálních bodech a predikující příslušnost k normální třídě, a považují za anomálie ty body, které do této třídy nepatří [1, s. 543-581].

Popis normální třídy pro mnohodimenzionální a zešikmený dataset, který je středem pozornosti této práce, je velmi rozsáhlý a nepoměrně těžký úkol. Stejně jako sestavení statistického modelu. Proto v rámci analýzy bude použita shlukovací metoda pro odhalení odlehlých bodů založená na hustotě, která se nazývá **DBSCAN** (Density-Based Spatial Clustering of Applications with Noise). Tento algoritmus je shlukování na základě hustoty s detekcí šumu – regionů s nízkou hustotou, které jsou v případě detekce anomálií samotné anomálie [1, s. 471-473].

Při běhu algoritmu DBSCAN se datové body s velkou hustotou postupně přidávají do shluků. Algoritmus vyžaduje dva vstupní parametry: ε – maximální vzdálenost mezi datovými body, aby se považovaly za sousední, a *MinPts* - minimální počet sousedů bodu, aby se považoval za jádro. Jádro je bod, kolem kterého se jeho sousedy a sousedy sousedů tvořen shluk. Hraničním se nazývá bod, který je dosažitelný na základě hustoty z jádra posloupností bodů (viz obrázek 2.5 (b)), ale sám nemá dostatečný počet sousedů, aby byl považován za jádro (viz bod p na obrázku 2.5 (a)). Ostatní body, které nejsou ani jádro, ani hraniční, jsou šum – anomálie [1, s. 471-473].



Obrázek 2.5: DBSCAN: Ukázka spojení bodů na základě hustoty [2]

Zkoumaný dataset obsahuje 16 sloupců. S takovým počtem dimenzí je těžké vizualizovat anomálie a provádět složité výpočty. Nejjednodušším způsobem, jak to udělat, je zbavit se několika sloupců. V tomto případě se ale ztrácejí důležité informace. Proto je vhodné se zamyslet nad metodami redukcí dimenzionality. Redukce dimenzionality je transformace z prostoru vyšší dimenze do prostoru nižší dimenze s co nejmenší ztrátou informace. Nejvýznamnější metodou je Principal Component Analysis (PCA), tj. analýza hlavních komponent. Metoda spočívá v tom, že najde nové proměnné, které jsou lineárními funkcemi sloupců v původním datovém souboru. Výsledné nové proměnné (komponenty) maximalizují rozptyl v datech a nekorelují spolu [14].

2.4.2 Shlukování

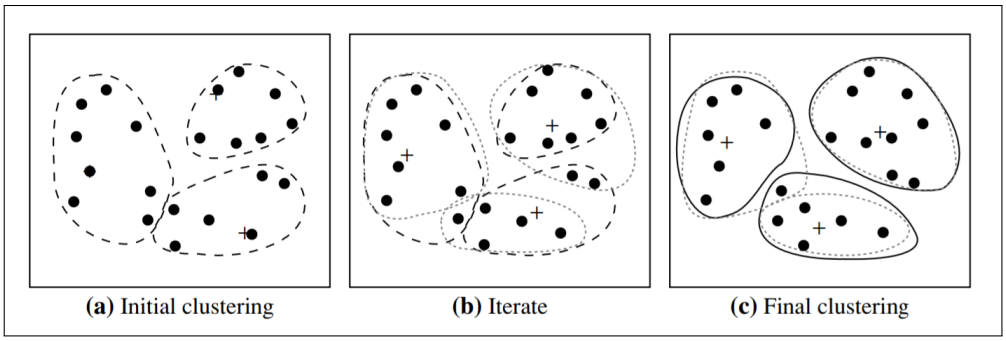
Shluková analýza je další metoda, která se používá pro zkoumání datových sad. Cílem shlukové analýzy je najít v datech jejich podmnožiny, shluky, tvořené podobnými datovými body a v rámci následného rozboru výsledných shluků odhalit skryté vzorce.

Nejvýznamnějším algoritmem pro shlukovou analýzu je **K-means**. Povinným vstupním parametrem algoritmu je k - počet výsledných shluků – do kolika podmnožin je potřeba rozdělit data. Tento vstupní parametr lze odhadnout například pomocí „Elbow“ metody. Metoda spočívá v tom, že se spočítá suma čtverců vzdáleností datových bodů od středů shluků, kterým patří, pro různé počty shluků. Potom se na grafu hledá místo „zlomu“, tzv. loket, ve kterém suma čtverců vzdáleností začíná klesat pomaleji [15].

Následně jsou buď náhodně, nebo cíleně, vybrány k bodů a vypočítá se euklidovská vzdálenost ostatních datových bodů do nich. Datové body, které jsou bližší k jednomu z vybraných k bodů, než k ostatním, tvoří shluk (viz obrázek 2.6 (a)). Potom jsou přepočítány nové středy shluků a opět se spočte vzdáleností ostatních datových bodů do nich (viz obrázek 2.6 (b)). Tento cyklus skončí, když se mezi iteracemi nezmění příslušnost žádného z bodů do shluku nebo se nepřekročí maximální počet iterací (viz obrázek 2.6 (c)).

Výstupem algoritmu K-means jsou středy výsledných shluků a labely, přiřazené každému záznamu ve vstupních datech a označující příslušnost k jed-

nomu z clusterů. Algoritmus velmi záleží na výběru počátečních středů a může skončit v lokálním optimu, proto je doporučené ho spouštět několikrát s různými vstupními středy [1, s. 451-454].

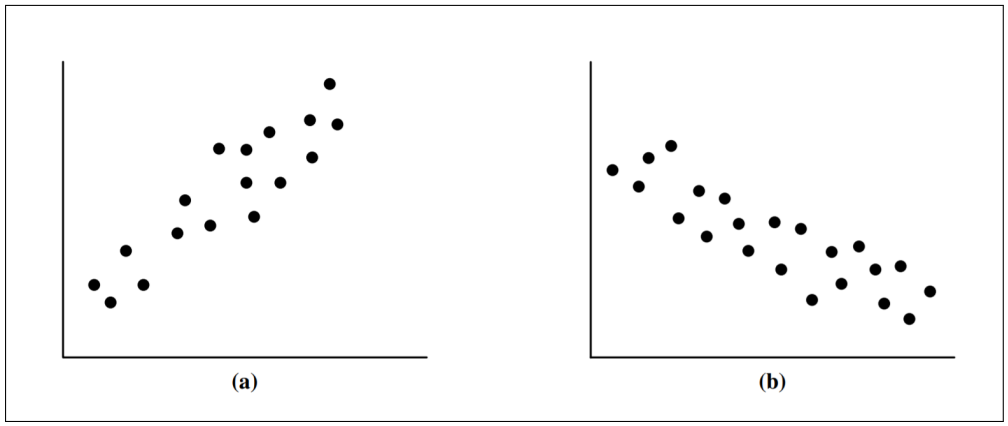


Obrázek 2.6: Ukázka rozdělení dat na shluky při běhu algoritmu K-means [1]

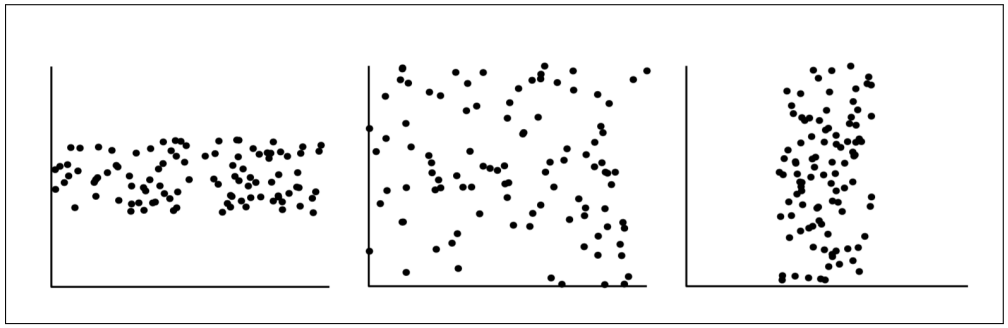
2.4.3 Korelace

„Dva atributy, X a Y, jsou korelovány, pokud jeden atribut implikuje druhý“ (překlad vlastní) [1]. Korelace může být:

- **pozitivní** – pokud se X zvětšuje se zvyšováním hodnot Y (viz obrázek 2.7 (a));
- **nulová** – pokud není možné vztah mezi X a Y popsat lineární funkcí (viz obrázek 2.8);
- **negativní** – pokud se X zvětšuje se snižováním hodnot Y (viz obrázek 2.7 (b)) [1].



Obrázek 2.7: Pozitivní a negativní korelace [1]



Obrázek 2.8: Nulová korelace [1]

Existuje několik vzorců pro výpočet korelačního koeficientu. Tří z nejvýznamnějších jsou:

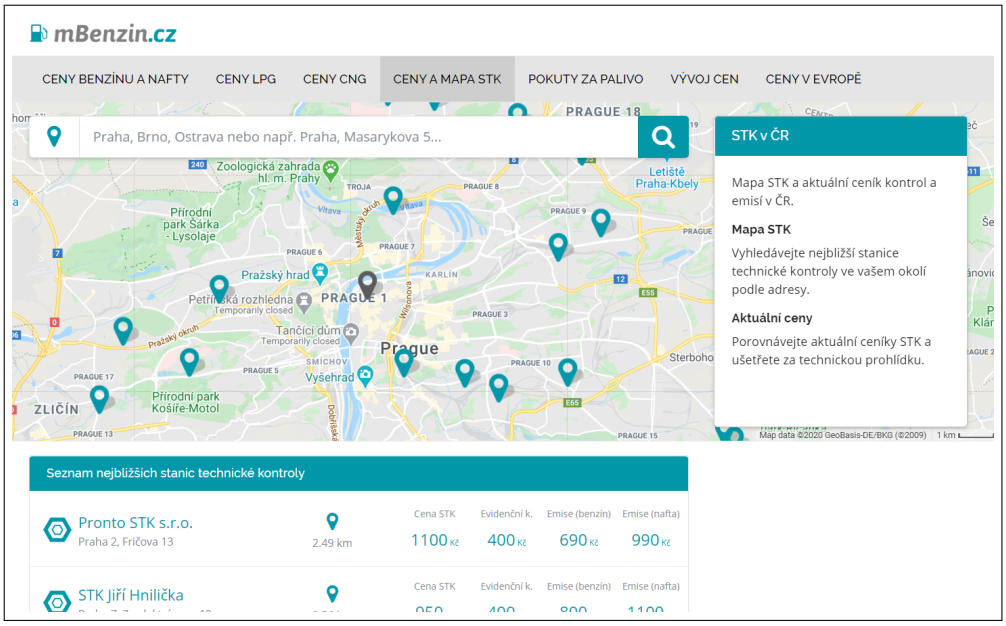
- **Pearsonův korelační koeficient** - potřebuje normálně distribuované spojité proměnné. Jde o nejčastěji používaný korelační koeficient;
- **Spearmanův korelační koeficient** - používá se pro non-normální distribuce (pro data s extrémními a odlehlými hodnotami). Pracuje pouze s pořadími pozorovaných hodnot, nikoli s jejich skutečnými hodnotami;
- **Kendallův korelační koeficient** - vyplývá ze stejných předpokladů, jako Spearmanův korelační koeficient. Je upřednostňován, pokud je ve zkoumaných datech málo záznamů nebo málo odlehlých hodnot [16].

V rámci analýzy bude vypočten Spearmanův korelační koeficient pro jednotlivé dvojice atributů, protože se v tomto případě nejedná o normální distribuci, jelikož data jsou zřejmě zešikmená, a to mimo jiné kvůli nedostatku kontrol s výsledkem „částečně způsobile“ a „nezpůsobile“.

2.5 Existující portály

V této části jsou uvedené a zobrazené tři nalezené existující portály spojené se stanicemi technické kontroly, jejichž koncepce je nejpodobnější té, která je navržená v rámci této práce. Následující podsekce obsahují popis vybraných portálů a aplikací, příklady poskytovaných informací o stanicích a ukázky grafického rozhraní.

2.5.1 mbenzin.cz

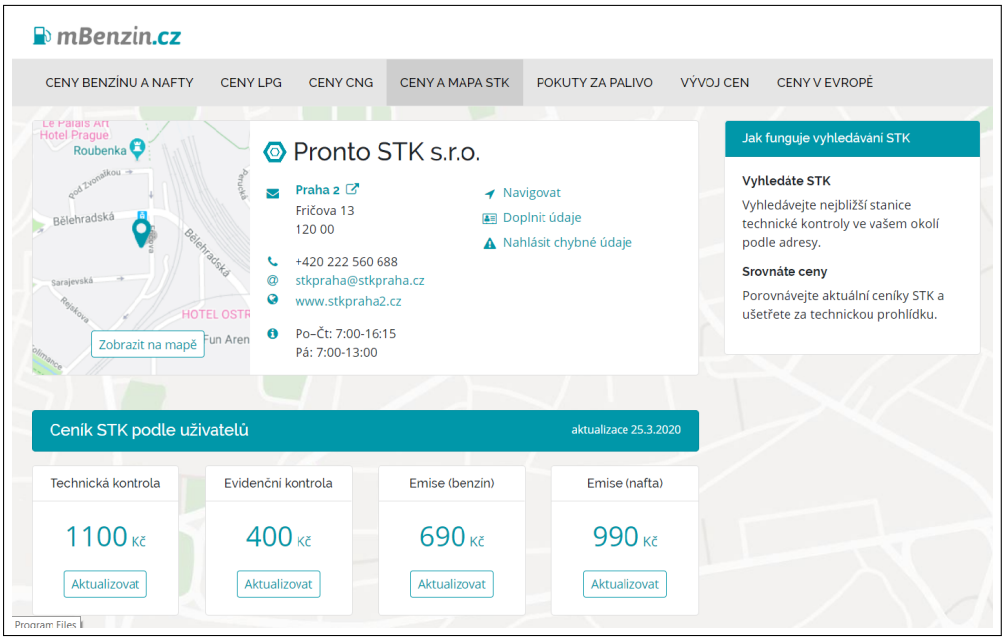


Obrázek 2.9: Portál `www.mbenzin.cz/STK` [3]

Webový portál mbenzin [3] je rozsáhlá a mnohoúčelová aplikace, která je zaměřena na zobrazení a porovnání cen různých druhů paliv v České republice i dalších státech Evropské unie a také informací o stanicích technické kontroly. Obsahuje taktéž aktuální novinové články týkající se automobilů, kontrol, cen benzinu a nafty apod.

Nejdůležitější záložkou portálu pro tuto práci je záložka „CENY A MAPA STK“ (viz obrázek 2.9), která poskytuje informace o nejbližších stanicích technické kontroly a umožňuje uživateli v nich vyhledávat podle adresy. Kliknutím na název každé stanice se otevírá stránka obsahující podrobnější údaje, tj. provozní doba, ceny, telefonní číslo, e- mailová adresa a koordináty na mapě (viz obrázek 2.10).

Portál má minimalistické, pohodlné a intuitivní uživatelské rozhraní, ale konkrétně v záložce STK obsahuje málo funkčností a poskytuje jen základní informace.



Obrázek 2.10: Portál `www.mbenzin.cz/STK` [3]

2.5.2 stanice-technicke-kontroly.cz

Portál stanice-technicke-kontroly [4] je na rozdíl od předchozího převážně věnován STK a kromě informací obsahuje inzerci služeb vyřízení technické prohlídky nebo registraci vozidla místo vlastníka (viz obrázek 2.11).

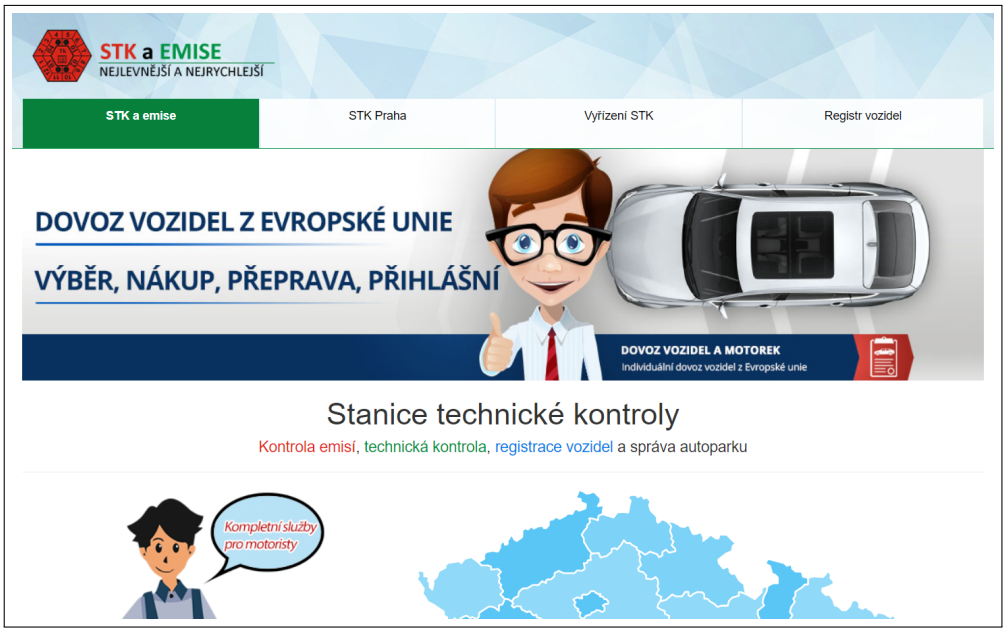
Na úvodní stránce je umístěná mapa, která umožňuje uživatelům vyhledávat stanice podle kraje. žádné další možnosti vyhledávání zde nejsou. Výsledky jsou vypsané ve velmi základní podobě. Kliknutím na název jednotlivé stanice se otevírá stránka obsahující podrobnější údaje, byť také základní. U vybraných stanic jsou však údaje detailnější než u jiných (viz obrázek 2.12).

Obecně vzato má tato webová aplikace příliš jednoduché a zastaralé grafické uživatelské rozhraní, nepříjemné a jednostranné vyhledávání a občas problémy s formátováním výstupů.

2.5.3 seznam-stk.cz

Další portál je seznam-stk [5]. Jak říká název, je primárně určen pro zobrazení seznamů stanic technické kontroly a kromě toho neobsahuje žádné další funkčnosti a informace.

Na úvodní stránce webové aplikace je interaktivní mapa České republiky, která bohužel není funkční, a seznamy krajů a měst, podle kterých lze najít STK (viz obrázek 2.13). Kliknutím na název každé stanice se otevírá stránka obsahující pouze adresu, oprávnění a kontakt stanici. Taktéž není ani uvedena

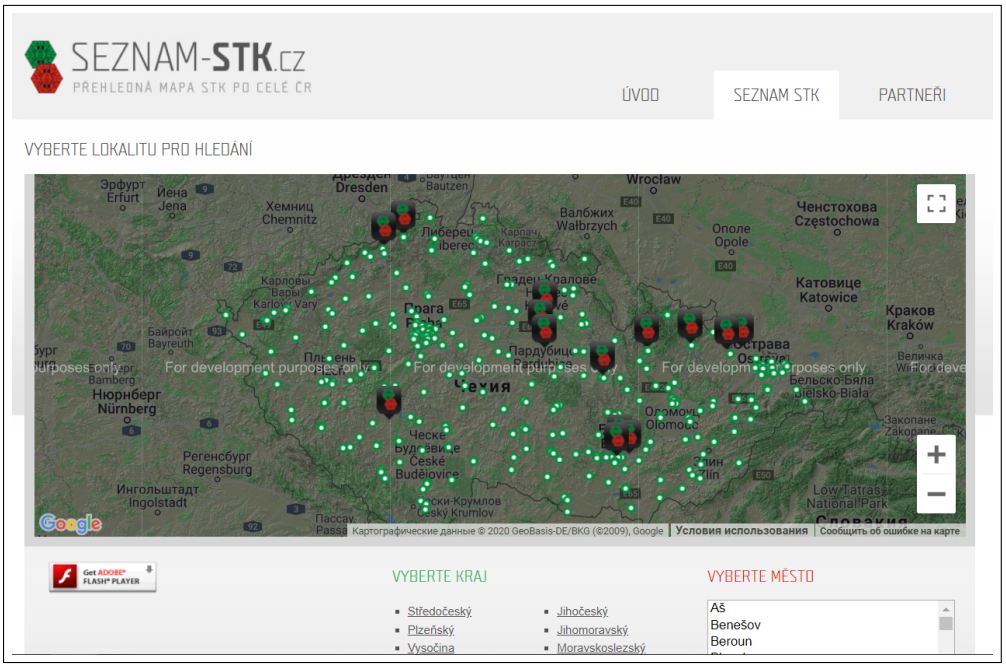


Obrázek 2.11: Portál `www.stanice-technicke-kontroly.cz/` [4]

Obrázek 2.12: Portál `www.stanice-technicke-kontroly.cz/` [4]

provozní doba (viz obrázek 2.14).

Je vidět, že portál není dopracován a nemá v sobě nic, kromě jednoduchého seznamu stanic, který je možné proklikat.



Obrázek 2.13: Portál `www.seznam-stk.cz/` [5]

Obrázek 2.14: Portál `www.seznam-stk.cz/` [5]

2.5.4 Závěr analýzy stávajících řešení

V rámci analýzy stávajících řešení byly nalezeny a probrány tři existující portály (mebenzin.cz, stanice-technicke-kontroly.cz a seznam-stk.cz), poskytující podobné informace, jaké bude poskytovat portál vyvinutý v rámci této práce.

Nejpopulárnější ze zmíněných portálů mbenzin i portál seznam-stk obsahují interaktivní mapu, na které jsou zobrazené jednotlivé stanice. Proto jedním z menších cílů implementace bude import podobné funkce do výsledné aplikace.

Další výhodou webového portálu mbenzin je přítomnost cenového přehledu. Jsou zde ceny technické prohlídky, evidenční kontroly a měření emisí. Možnost seřazení stanic podle ceny technické kontroly je další menší cíl implementace.

Obecně lze říci, že hlavní koncept každého z uvedených portálů je jednoduše řečeno „tabulka“, která obsahuje seznam stanic, ve kterém se lze různými způsoby vyhledávat a kliknutím na každou stanici se zobrazí podrobnější informace.

Je vidět, že žádný z uvedených portálů neobsahuje statistický přehled kontrol. A jistě nikde není reprezentace výsledků metod znalostního inženýrství, což bude největší výhodou a originální funkcí implementovanou do webové aplikace.

Kapitola 3

Analýza dat

Táto kapitola se zabývá zpracováváním tabulek zmíněných v sekci 2.3, aplikací popsaných v sekci 2.4 postupů, návrhem metod pro detekci podezřelého chování a vytěžováním dalších informací z datasetů.

Celá analýza byla provedena za použití nástroje `Jupyter Notebook` [17] pro jazyk Python.

Pro zpracování datasetů, analýzu a vytěžování informací z nich byl zvolen rychlý a jednoduchý nástroj `pandas` [18]. Pandas je knihovna pro jazyk Python speciálně určena pro práci s 2D tabulkami.

3.1 Příprava dat

Pro snadnou a efektivní manipulaci s daty je potřeba:

- převést je do pohodlného formátu, přičemž pro tuto práci byl zvolen formát CSV, což je klasické jednoduché rozšíření, postačující pro účely analýzy (viz sekci 3.1.1);
- odstranit chybné a poškozené záznamy;
- dohledat a vytěžit z webových stránek další informace potřebné pro analýzu (viz sekci 3.1.2).

3.1.1 Oprava souboru

Dataset popsáný v sekci 2.3.1 je dostupný z adresy uvedené v NKOD, a to ve formátu XML. Při otevření souboru libovolným XML parserem nebo validátorem se zobrazí „Fatal Error at Line 1, Char 283“ nebo podobná chyba.

Při bližším pohledu na strukturu souboru je vidět, že se skládá z jednotlivých XML tagů „record“, které jsou umístěny jeden po druhém a samotné položky jsou atributy těchto tagů (viz výpis 3.1).

```
<record STK="3205" ... VyslEmise="---"/>
<record STK="3114" ... VyslEmise="---"/>
                   ...
<record STK="3604" ... VyslEmise="---"/>
```

Výpis 3.1: Struktura souboru STK2018

`Char 283` je místo, kde končí první tag „record“. Chyba na prvním řádku, kterou vrací validátor, je způsobena tím, že soubor neodpovídá běžné stromové struktuře XML dokumentu, která požaduje, aby byla skupina tagů umístěna do jednoho rodičovského tagu a rovněž vyžaduje prolog, obsahující verzi XML a kódování (viz výpis 3.2).

```
<?xml version="1.0" encoding="UTF-8"?>
<records>
    <record STK="3205" ... VyslEmise="---"/>
    <record STK="3114" ... VyslEmise="---"/>
                       ...
    <record STK="3604" ... VyslEmise="---"/>
</records>
```

Výpis 3.2: Požadovaná struktura souboru STK2018

Po přidání chybějících elementů na začátek a na konec dokumentu se musejí také odstranit zbytečné nové řádky (`\n`). Tím končí oprava souboru.

Následně pomocí knihovny `xml.etree` [19] byly jednotlivé atributy každého tagu „record“ načteny do polí. V datasetu byl nalezen jenom jedem testovací záznam, přičemž ostatní jsou platné. Z výsledných datových struktur byl vytvořen pandas dataframe, který byl potom převeden do CSV tabulky a uložen na disk pro budoucí použití.

Ostatní datové sady zmíněné v sekci 2.3.2 není potřeba upravovat ani převádět do CSV, protože jsou již ve formátu xlsx, který je možné načíst do dataframe příkazem `pd.read_excel()`.

Skript a ukázka výsledků jsou součástí notebooku `xml_to_csv.ipynb`.

3.1.2 Vytěžování informací z webových stran

V tabulce se seznamem STK podle krajů (viz sekci 2.3.2) chybějí některé užitečné údaje o stanicích pro analýzu a zobrazení na portálu. Například pro sestavení seznamu kontrol provedených mimo pracovní dobu je potřeba dohledat tuto dobu a pro porovnání cen na STK v různých krajích a jejich zobrazení na portálu je nutné tyto ceny také zjistit.

Potřebné informace jsou již zobrazené na portálu mbenzin.cz (viz obrázek 2.10). Tyto údaje jsou zobrazené v poměrně jednotném formátu a nacházejí se na stejném místě pro každý výsledek hledání, proto je lze vytáhnout jednoduchým iterativním postupem.

Pro získání dat byly použity knihovny `requests` [20] a `beautifulsoup` [21]. Knihovna requests slouží pro zasílání HTTP požadavků a Beautiful Soup je určená pro snadné parsování získaných HTML reprezentací stránek. Beautiful Soup objekt, který je výsledkem parsování, je hnízdová datová struktura, která umožňuje filtrované vyhledávání tagů.

V současné době obsahují jednotlivé části většiny webových aplikací unikátní identifikátory. Tyto identifikátory jsou umístěny do atributů HTML tagů. To jsou například atributy `id`, `class`, `itemprop` apod. Taktéž nejenom pouze `id` tagu a rovněž některé unikátní vlastnosti. Právě podle těchto atributů je možné vytěžovat jenom potřebné informace.

Po analýze HTML reprezentací webových stránek portálu mbenzin byla nalezena umístění hledaných informací, tj. cen technické kontroly, cen evidenční kontroly, cen kontroly emisí benzínových a naftových motorů a provozních dob stanic. Na portálu je implementováno fulltextové vyhledávání, kde text je parametr specifického HTTP požadavku. Tato vlastnost byla využita pro získání informací o všech dostupných stanicích.

Bohužel je po doběhnutí skriptu vidět, že provozní doby stanic nejsou v jednotném formátu a bylo potřeba je rozparsovat, aby bylo možné se v nich jednoduše orientovat. Některé údaje bylo nutné parsovat a dohledávat ručně.

Skript a ukázka výsledků jsou součástí notebooku `STK_info.ipynb`.

3.2 Základní informace a statistiky

Důležitou součástí analýzy libovolného datasetu je sběr základních statistik a závislostí mezi daty. Existuje několik užitečných nástrojů pro automatizaci tohoto procesu. Výsledky z podobných knihoven jsou ale samozřejmě jenom výchozím bodem, protože pro každou datovou sadu je vyžadován individuální postup.

Jednou z takových knihoven je `pandas-profiling` [22], která pro pandas dataframe generuje kompletní statistický popis ve formátu HTML obsahující:

- datové typy sloupců;
- unikátní a chybějící hodnoty;
- minimum, maximum, medián, rozsah jednotlivých číselných sloupců;
- směrodatné odchylky, koeficienty špičatosti a šikmosti;
- nejčastější hodnoty;
- histogramy;
- Spearman, Pearson a Kendall matice.

Pomocí této knihovny byl vygenerován dokument popisující jednotlivé sloupce, které jsou popsáné v dalších sekcích.

3.2.1 STK

Informace o střední hodnotě a směrodatné odchylce jsou pro sloupec **„STK“**, který představuje číselný identifikátor stanice, zbytečné. Jsou ale relevantní tyto statistiky:

- Celkový počet stanic, které jsou součástí datasetu, je **552**. Což je o 180 víc než oficiální počet STK v roce 2018. To je způsobené tím, že v datové sadě jsou také přítomné záznamy o technických silničních kontrolách, které odpovídají identifikátorům mezi 8 000 a 10 000 a tvoří 185 odlišné hodnoty;
- Největší počet kontrol v roce 2018 byl proveden na stanici číslo 3413. Celkem stanice obsloužila 73 558 vozidel. To, že má největší počet kontrol není anomální, protože jde o největší stanici technické kontroly v České republice, která se nachází v Plzni a obsahuje 6 linek pro osobní vozidla a 2 linky pro nákladní vozidla. Z toho počtu je ale 47 537 pravidelných prohlídek, což je ovšem vzhledem k odhadu časové náročnosti pravidelné prohlídky příliš mnoho:
  - provozní doba stanice je 5 dní v týdnu 11 hodin denně, což je maximálně 660 minut každý pracovní den;
  - v roce 2018 bylo 250 pracovních dnů;
  - časová pracnost pravidelné prohlídky osobního vozidla je 27,91 minut, u nákladního vozidla pak 47,19 minut a u motorového vozidla 16,10 minut;
  - pravidelných prohlídek motorových vozidel bylo provedeno 1 936, což jsou 4 % z celkového počtu.

  Hrubý odhad počtu pravidelných kontrol na zkoumané stanici v roce 2018, pokud stanice bude provádět jenom tento typ prohlídek, je 47 294. V datasetu je navíc ještě 26 021 kontrol jiných druhů. Je vidět, že prohlídka se v průměru koná o mnohem rychleji, než je její odhadovaná časová pracnost.

3.2.2 DrTP

Dalším sloupcem je **„DrTP“**, který reprezentuje druh technické prohlídky a nabývá 14 různých hodnot:

- pravidelná: **2 471 966** záznamů (**66,3 %**);
- evidenční: **820 573** záznamů (**22 %**);
- opakovaná: **211 330** záznamů (**5,6 %**);

- před registrací: **186 819** záznamů (**5 %**);
- na žádost zákazníka: **17 993** záznamů (**0,5 %**);
- před schválení technické způsobilosti vozidla: **5 309** záznamů (**0,14 %**);
- ADR: **4 318** záznamů (**0,12 %**);
- pred registrací – opakovaná: **3 886** záznamů (**0,12 %**);
- technická silniční kontrola: **2 581** záznamů (**0,07 %**);
- technická silniční kontrola – opakovaná: **1 666** záznamů (**0,04 %**);
- technická silniční kontrola – opakovaná po dopravní nehodě: **1 271** záznamů (**0,03 %**);
- ADR – opakovaná: **309** záznamů (**0,008 %**);
- před schválení technické způsobilosti vozidla – opakovaná: **203** záznamů (**0,005 %**);
- nařízená technická prohlídka: **145** záznamů (**0,004 %**).

3.2.3 VIN

Třetí sloupec je **„VIN“**, který reprezentuje unikátní identifikátor vozidla. Obsahuje **84,2 %** odlišných hodnot. Nejčastějšími hodnotami jsou některé kombinace tří čísel, například 005, 106 nebo 092. Což není platný VIN, ale většina z těchto záznamů se nevztahuje k vozidlům, ale k přívěsům a návěsům.

3.2.4 TypMot

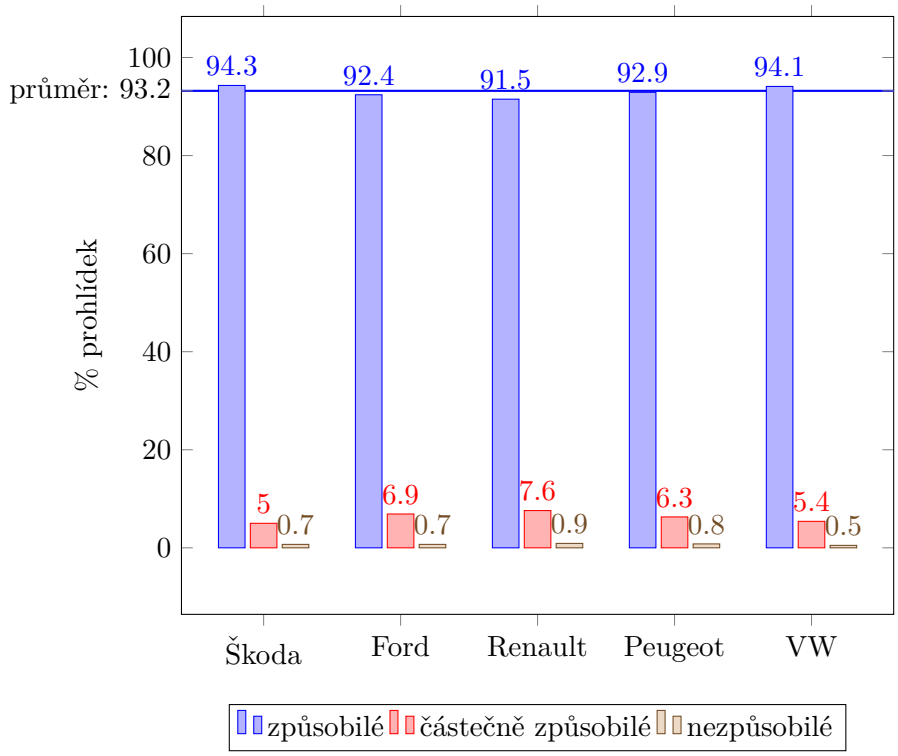
V datasetu je 62 848 různých typů motorů (sloupec **„TypMot“**). Z celého počtu prohlídek 266 522 (7,2 %) nemají uvedený typ motorů, nejčastěji protože jde o přípojné vozidlo, přívěs apod. Nejčastějším typem je motor „BXE“, který je nejčastěji instalován do automobilů Škoda a Volkswagen.

3.2.5 TZn

Dalšími sloupci jsou **„TZn“** a **„ObchOznTyp“**. TZn představuje značku vozidla a obsahuje 6 266 odlišných hodnot. ObchOznTyp je model vozidla a obsahuje 67 142 odlišných hodnot. Nejčastějšími značkami jsou:

- Škoda: **898 984** záznamů (**24,1 %**), nejčástější model:

  Octavia – **342 735** záznamů (**38,1 %**);



Obrázek 3.1: Výsledky kontrol podle značky vozidla

- Ford: **259 757** záznamů (**7 %**), nejčástější model:

  Focus – **72 399** záznamů (**29 %**);

- Renault: **192 249** záznamů (**5,2 %**), nejčástější model:

  Megane – **61 041** záznamů (**31,8 %**);

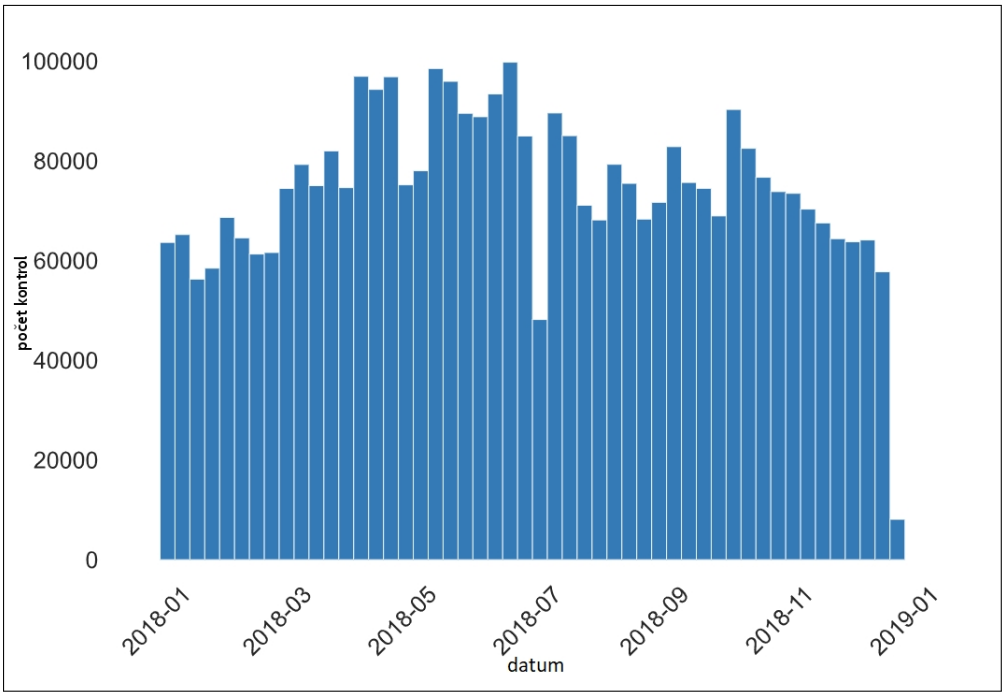
- Peugeot: **180 757** záznamů (**4,8 %**), nejčástější model:

  206 – **36 376** záznamů (**20,1 %**);

- Volkswagen: **166 904** záznamů (**4,5 %**), nejčástější model:

  Golf – **40 364** záznamů (**24,2 %**).

Na obrázku 3.1 je vidět, kolik procent automobilů příslušné značky prošlo nebo neprošlo státní technickou kontrolou v roce 2018.

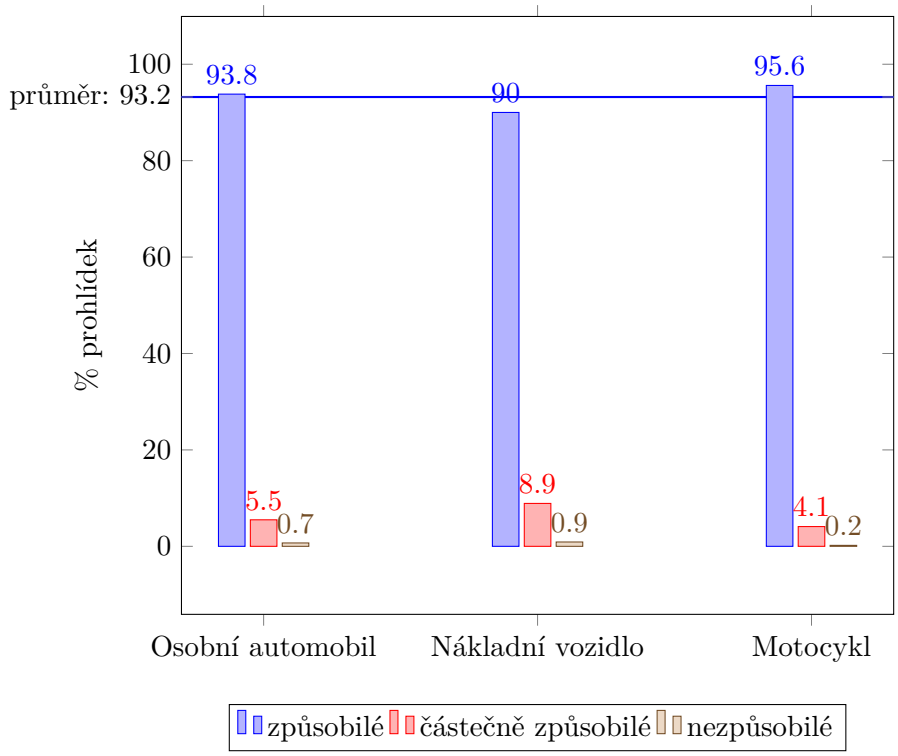


Obrázek 3.2: Počet kontrol podle data

3.2.6 DatKont a DatPrvReg

V datasetu jsou dva datové sloupce: **„DatKont“** a **„DatPrvReg“**, které reprezentují datum a čas prohlídky a datum první registrace vozidla. Základní informace o sloupcích:

- První prohlídka v roce 2018 byla provedena 2. ledna v 05:38:33 a poslední 31. prosince v 16:07:02;
- Z histogramu na obrázku 3.2 je vidět, že rozložení kontrol během roku není rovnoměrné a nejpracnějšími měsíci jsou duben, květen a červen;
- Nejstarší datum registrace vozidla v datasetu je 1. ledna 1753, což je jednoduše minimální možné datum v sloupci datetime na SQL serveru [23]. Dalšími jsou automobily registrované v roce 1900, mezi kterými jsou například vozidla modelu Škoda Favorit, vyráběných v letech 1988 až 1995, což opět vypovídá o nesprávně zadaných záznamech.
- Mimo tato chybná data prohlídek je nejstarší datum registrace osobního vozidla 31. prosince 1917. Jde o klasický automobil modelu Oakland 34 Touring, který se ve světě prodával od roku 1917. Mělo jenom 2 závady typu A a úspěšně prošlo státní technickou prohlídkou.



Obrázek 3.3: Výsledky kontrol podle druhu vozidla

3.2.7 DrVoz a Ct

Sloupce **„DrVoz“** a **„Ct“** jsou druh a kategorie vozidla. Dataset obsahuje 45 různých druhů a 135 různých kategorií vozidel. Nejčastějšími druhy jsou:

- Osobní automobil: **2 705 196** záznamů (**72,6 %**);
- Nákladní vozidlo: **439 599** záznamů (**11,8 %**);
- Motocykl: **200 821** záznamů (**5,4 %**).

Na obrázku 3.3 je vidět, kolik procent automobilů příslušného druhu prošlo nebo neprošlo státní technickou kontrolou v roce 2018.

3.2.8 Km

První číselný sloupec datasetu je **„Km“** a reprezentuje počet najetých kilometrů. Střední počet kilometrů je **158 320**, medián je pak **96 955** a maximum je **9 944 330**.

3.2.9 ZavA, ZavB a ZavC

Dalšími sloupci jsou počet závad typu A (**„ZavA“**), typu B (**„ZavB“**) a typu C (**„ZavC“**). Počet prohlídek, ve kterých nebyly zjištěny žádné závady: **1 618 466** (**43,4 %**).

Základní statistiky:

- typ A:
  - střední počet: **2,001**;
  - maximální počet: **44**;
  - směrodatná odchylka: **2,57**;
- typ B:
  - střední počet: **0,166**;
  - maximální počet: **37**;
  - směrodatná odchylka: **0,8**;
- typ C:
  - střední počet: **0,009**;
  - maximální počet: **23**;
  - směrodatná odchylka: **0,135**;

3.2.10 VyslEmise a VyslSTK

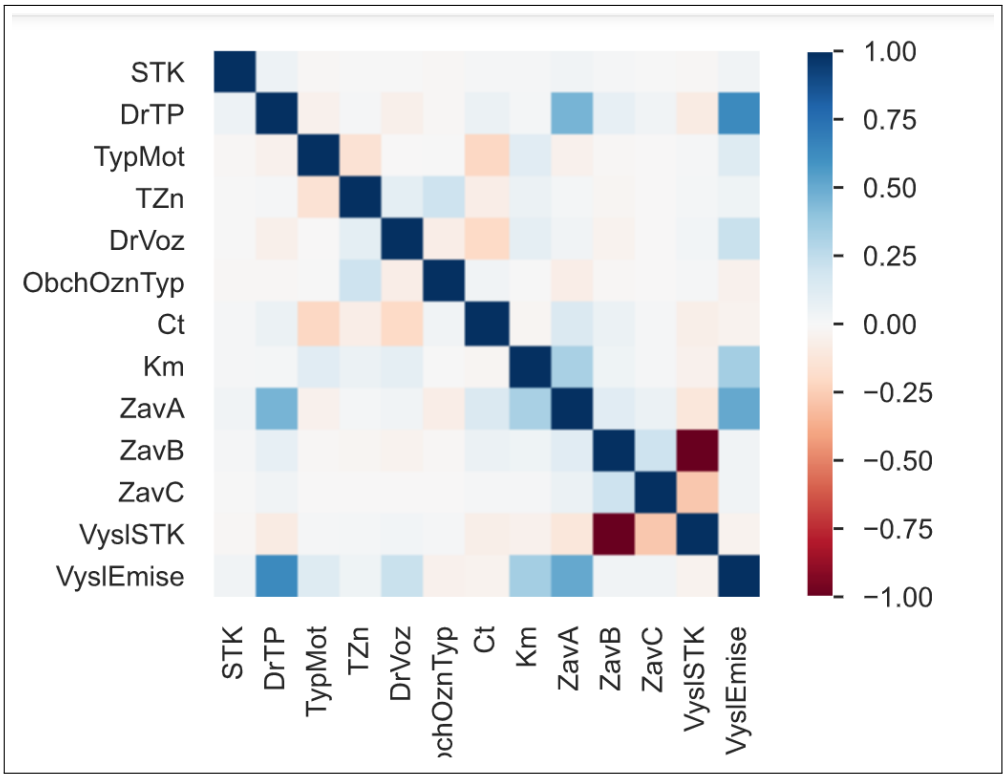
Poslední dva sloupce (**„VyslSTK“**) a (**„VyslEmise“**) jsou výsledky technické prohlídky a měření emisí. U velké části prohlídek (35,5 %) není uveden výsledek měření emisí, což je způsobené druhem technické kontroly, například evidenční kontrola nezahrnuje měření emisí. Z celého datasetu je 6 484 prohlídek, u kterých je nevyhovující výsledek emisí a 918, u kterých je výsledek částečně vyhovující, což je jenom 0,27 % a 0,04 %.

3.2.11 Korelace

Celkem je v datově sadě 176 773 chybějících hodnot, což tvoří jenom 0,3 %. Rozložení sloupců podle typu je následující: 9 kategorických, 2 datové, 5 numerických.

Pro snadný výpočet korelačního koeficientu je potřeba kategorické sloupce převést do numerických, což jde jednoduše pomocí nativní funkce pandasu `cat.codes`, která byla aplikována na každý sloupec typu `object`.

Obrázek 3.4 znázorňuje výsledky výpočtu Spearmanova korelačního koeficientu pro jednotlivé dvojice sloupců.



Obrázek 3.4: Korelační matice: Spearmanův korelační koeficient

Je vidět, že nejvíce korelovanými hodnotami jsou:

- VyslSTK a ZavB/ZavC: výsledek kontroly a počet závad typu B/C. Vysvětlení je jednoduché – výsledek libovolné kontroly, na které byla zjištěna alespoň jedna závada typu B/C se hned mění na „částečně způsobilé“ nebo „nezpůsobilé“;

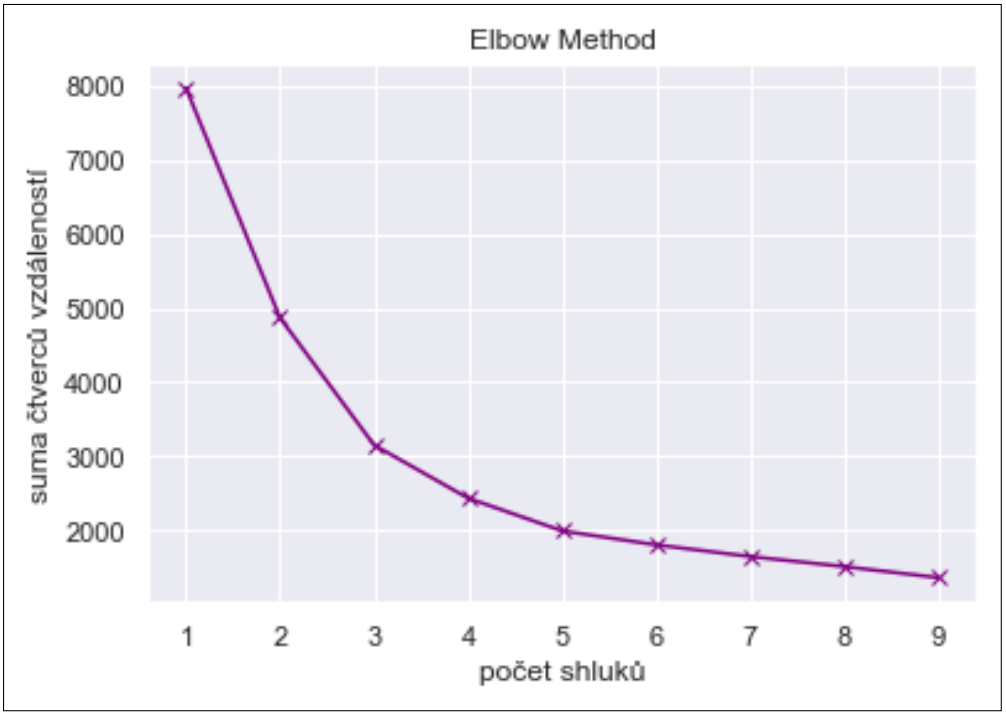
- VyslEmise a DrTP: výsledek měření emise a druh technické prohlídky. Důvod tohoto je triviální. Výsledek měření emise se neuvádí u některých druhů prohlídek;

- ZavA a Km: počet závad typu A a počet najetých kilometrů. Vysvětlení je také očividné. Čím více je automobil v provozu, tím víc se opotřebuje;

- ZavA a DrTP: počet závad typu A a druh technické prohlídky. Největší počet závad A se zjišťuje během opakované prohlídky a během prohlídky ADR se nezjišťují žádné, skoro stejně jako během evidenční kontroly, tudíž počet závad typu A je závislý na typu prohlídky;

- ZavA a VyslEmise: počet závad typu A a výsledek měření emisí. Samotný výsledek emisí a počet závad typu A nejsou silně korelované, ale korelované jsou počet závad typu A a přítomnost výsledku emise.

3.3 Shlukování

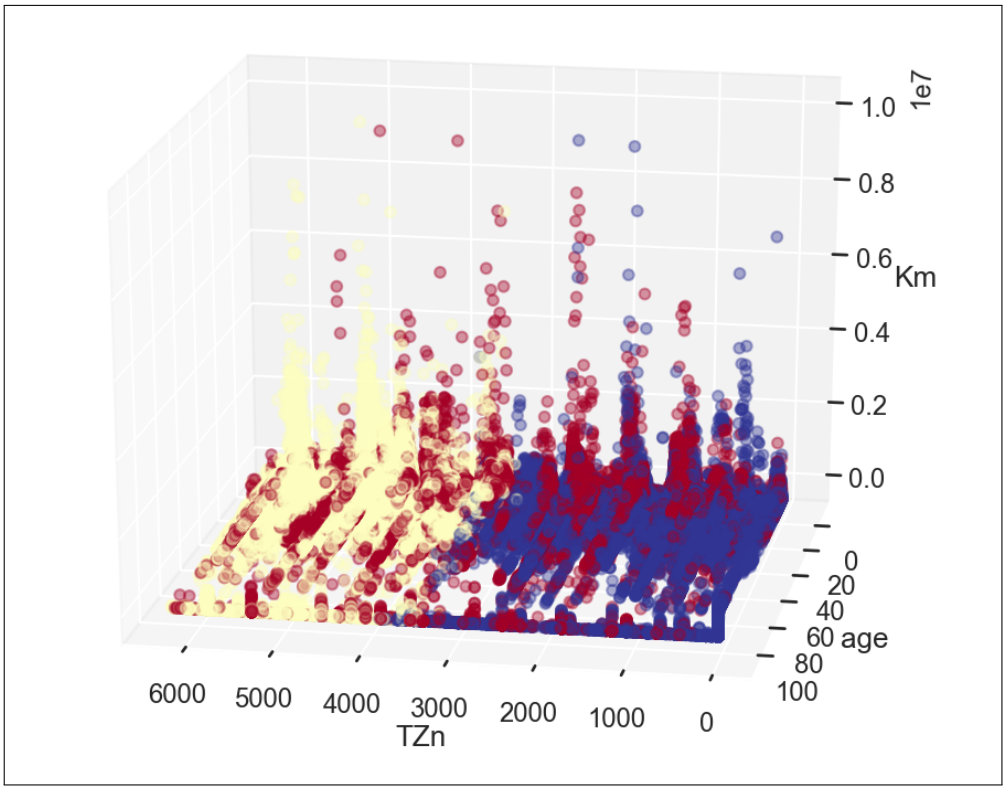


Obrázek 3.5: K-means: Odhad nejlepšího počtu shluků pomocí Elbow metody

Pro odhalení skrytých vzorů v datech se mimo jiné používá shluková analýza. V této sekci je popsán postup hledání podmnožin dat, shluků, v datasetu STK2018.

Shlukování bylo provedeno pomocí algoritmu K-means, popsaného v sekci 2.4.2. Byla využita opensourcová knihovna pro datovou analýzu `scikit-learn` [24].

Kategorické sloupce je pro jednodušší výpočet vzdáleností potřeba převést do numerických pomocí nativní funkce pandasu `cat.codes`, která byla aplikována na každý sloupec typu `object`. Nekategorické sloupce „VIN“ a „DatKont“ jsou pro shlukování zbytečné, stejně jako identifikátor stanice („STK“). Sloupec „DatPrvReg“ v sobě ale nese informace o stáří automobilů, což je důležitý údaj, který může ovlivnit rozložení dat do shluků. K-means v `scikit-learn` se bohužel neumí vypořádat s typy `timestamp` nebo `datetime`, proto byl „DatPrvReg“ omezen na počet roků uplynulých od roku registrace.

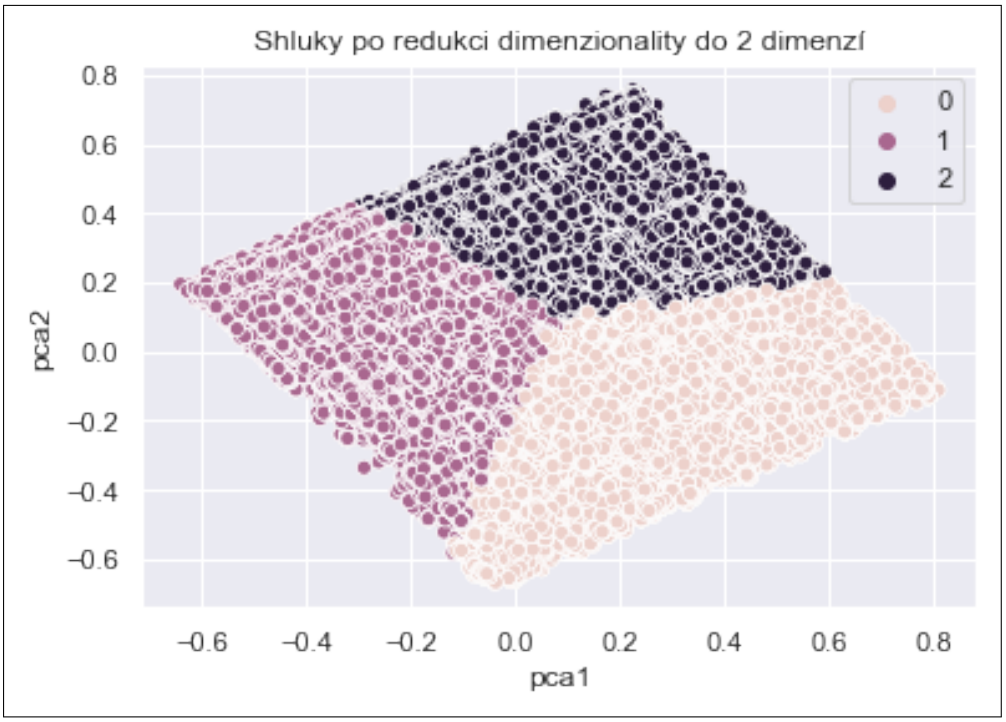


Obrázek 3.6: K-means: Shluky před redukcí dimenzionality

Po odstranění zbytečných sloupců a konvertování kategorických sloupců do numerických zůstalo v dataframe 9 sloupců typu `int`, které v sobě nesou informace o vozidle, které se zúčastnilo prohlídky a nikoliv o samotné prohlídce. Rozsahy hodnot jednotlivých položek jsou ale velmi rozdílné a můžou zešikmit výsledné shluky, proto je nutné je normalizovat. V balíčku `preprocessing` knihovny `scikit-learn` existuje třída `MinMaxScaler`, pomocí které lze škálovat hodnoty a vyhnout se ztrátě vlivu malých čísel na výslednou vzdálenost.

Po normalizaci dat je potřeba odhadnout nejlepší počet shluků. Existuje několik metod pro takový odhad. V rámci této práce byla využita metoda, která se nazývá „Elbow Method“. Metoda spočívá v tom, že se spočítá suma čtverců vzdáleností datových bodů od středů shluků, kterým patří, pro různé počty clusterů. Potom se vykreslí graf závislostí sumy a počtu shluků, na kterém se vyhledá místo „zlomu“. Na výsledném grafu (viz obrázek 3.5) je vidět, že suma čtverců vzdáleností výrazně klesla po změně počtu shluků z 2 na 3 a potom začala klesat rovnoměrně. Proto za výsledné místo zlomu bylo vzato číslo 3.

Po doběhnutí algoritmu byly vytvořeny tři shluky (viz obrázek 3.6), které obsahují 1 224 645, 1 919 929 a 583 795 prvků. Data obsahují 9 dimenzí a rozdělení na shluky není zřejmé, proto bylo pro lepší vizualizaci rozhodnuto



Obrázek 3.7: K-means: Shluky po redukci dimenzionality do 2 dimenzí

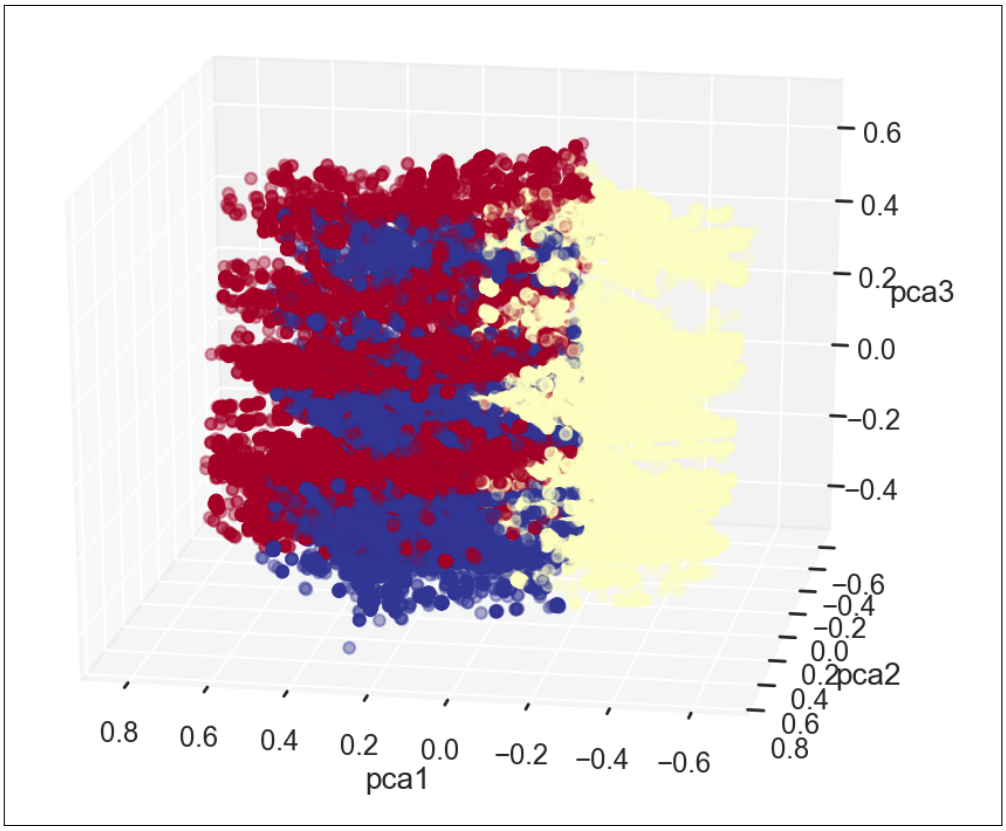
použít metodu, která se nazývá Principal Component Analysis (PCA) – analýza hlavních komponent. Metoda spočívá v tom, že redukuje počet dimenzí pomocí projekcí na nižší dimenzionální prostor.

Byla použita třída `PCA` z balíčku `decomposition` knihovny `scikit-learn`. Analýza rozptylu v datech v závislosti na počtu komponent ukázala, že optimální počet komponent je mezi 2 a 3. Po redukci dimensionality do 2 dimenzí (viz obrázek 3.7) a do 3 dimenzí (viz obrázek 3.8) jsou výsledné tři shluky zjevně rozdělené. Postup byl převzat z článku [25].

Pro každý cluster byl vygenerován stejný `profile report`, jako v sekci 3.2. Tyto reporty ukazují, že shluky jsou rozdělené primárně podle značky vozidla. První cluster obsahuje většinou automobily značky Škoda a další, druhý – Ford, Renault a další, třetí – Fiat, Audi, BMW a další. Střední stáří vozidla, počty najetých kilometrů a závad různých typů jsou u všech třech shluků skoro stejné. Shluky mají dokonce i stejný průměr úspěšností u technické prohlídky.

Celkové výsledky shlukové analýzy vypadají následovně. Data jsou rozdělitelná především podle značky vozidla, mimo kterou nejsou mezi clustery žádné rozdíly, což znamená, že data nejsou pro shlukovou analýzu vhodná, nebo v nich nejsou žádné zjevné segmenty.

Skript a ukázka výsledků jsou součástí notebooku `clustering.ipynb`.



Obrázek 3.8: K-means: Shluky po redukci dimenzionality do 3 dimenzí

3.4 Odhalení podezřelého chování

Jedním z cílů této práce je odhalení podezřelého chování na stanicích technické kontroly. Součástí zmíněných v sekci 2.3 dat nejsou kamerové záznamy, proto nelze s jistotou tvrdit, že stanice podvádějí, ale je možné objevit některé podezřelé příznaky. Například kontroly mimo pracovní dobu stanice, anomálně vysoká hustota prohlídek v určité časy, časové souběhy kontrol a další.

3.4.1 Kontroly mimo pracovní dobu

Jednou ze základních věcí, kterou je možné zjistit z dat, která jsou již k dispozici, je množství kontrol, provedených mimo pracovní dobu stanic, tj. po zavírací době nebo mimo pracovní den. Pro provedení této analýzy bude potřeba celý seznam jednotlivých kontrol a seznam stanic s informacemi o pracovní době vytěženými z webu, které jsou popsané v sekci 3.1.2.

Kvůli tomu, že u některých ze stanic není známá jejich pracovní doba, budou kontroly, které na nich provedly, přeskočeny. Tímto se ovšem ztratí jen 712 360 (19 %) záznamů, což není pro celkovou analýzu kritický objem.

Po načítání CSV souborů do `pandas dataframe` je důležité převést sloupec DatKont v tabulce se seznamem prohlídek do typu `datetime` z nativního modulu Pythonu, který umožňuje snadný přístup k jednotlivým položkám timestampu, které budou využity: počtu hodin, minut a dnů. Funkce `weekday()` vrací den v týdnu pro daný datetime objekt. Ale pozor: 0 odpovídá pondělí a 6 je neděle.

Díky tomu, že je ve výsledném seznamu stanic pracovní doba uvedená v jednotném formátu, lze zjistit, zdali byla kontrola provedena v daném časovém rozmezí, a to parsováním pracovní doby (ve formátu string) pomocí regulárního výrazu, zachycením konkrétních čísel pomocí `match.group()` a porovnáním s jednotlivými položkami timestampu.

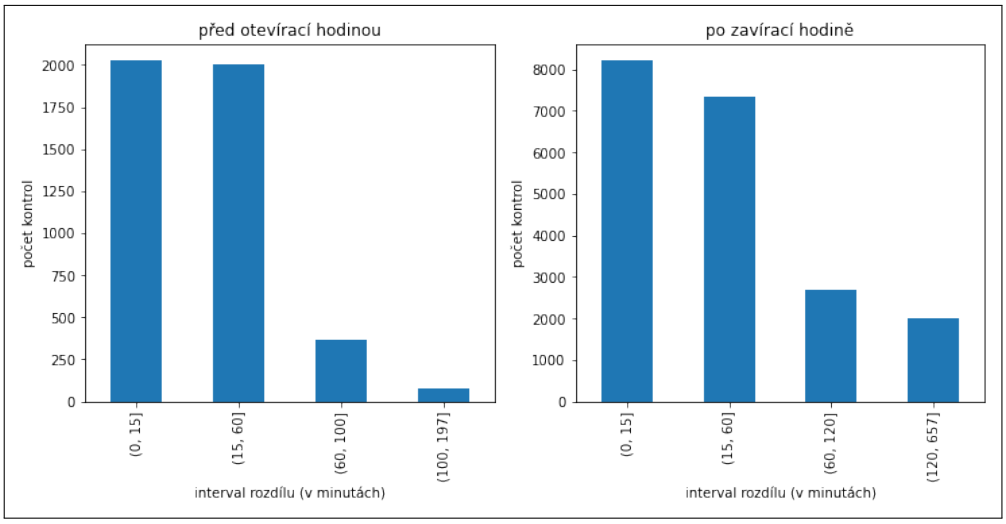
Po doběhnutí scriptu, který zároveň ukládal kontroly mimo pracovní dobu do samostatného CSV souboru, se ukázalo, že 235 z 247 zkoumaných stanic provedly alespoň jednu takovou prohlídku v roce 2018. Samozřejmě se také musí vzít v úvahu to, že některé pracovní doby uvedené na webových stránkách nejsou platné. Bohužel ale neexistuje žádný zdroj informací, například kromě obvolávání jednotlivých stanic, ze kterého je možné vytěžit skutečná data.

Celkový počet prohlídek mimo uvedenou pracovní dobu je 101 278, což jsou 3 % z 3 016 009 záznamů. Z té množiny bylo 25 117 (25 %) provedeno mimo pracovní den:

- Pondělí: 380;
- Úterý: 368;
- Středa: 405;
- Čtvrtek: 381;
- Pátek: 6 236;
- Sobota: 17 063;
- Neděle: 284.

Počty kontrol v sobotu a v pátek vzbuzují podezření, že některé provozní doby nejsou správně zadané. Stejná situace je pak u kontrol po zavírací době stanic, kdy u některých stanic tyto prohlídky tvořily 20–30 % celoročního počtu. Proto byly pro ověření ručně dohledány pracovní doby těch stanic, které mají počet kontrol mimo uvedenou pracovní dobu větší než 200.

Po opravě a odstranění některých záznamů zůstalo jenom 26 718 prohlídek mimo pracovní dobu, z nichž 1 994 mimo pracovní den. Z těchto pak 127 v neděli a 1 867 v sobotu. Data v databázi byla upravena a ve finální reprezentaci výsledků na portálu se chybné informace nezobrazí.



Obrázek 3.9: Počty prohlídek mimo pracovní dobu podle odchylky od uvedené pracovní doby

Na obrázku 3.9 je vidět, že většina kontrol provedených mimo provozní doby stanic, byla provedena v průběhu 60 minut před otevírací nebo po zavírací hodině, ze kterých víc než polovina během 15 minut. Toto chování podezřelé není. Podezřelejší jsou ale kontroly, které patří do posledního intervalu na obou grafech na obrázku 3.9. Většina z těchto kontrol byla provedená v pátek večer. Z nich 32 bylo provedeno do 6 hodiny ráno a 195 po 8 hodině večer.

Celkové rozložení podezřelých prohlídek z hlediska provozní doby stanic je vidět na obrázku 3.10, který zobrazuje počty prohlídek mimo pracovní dobu podle času a dnu kontroly.

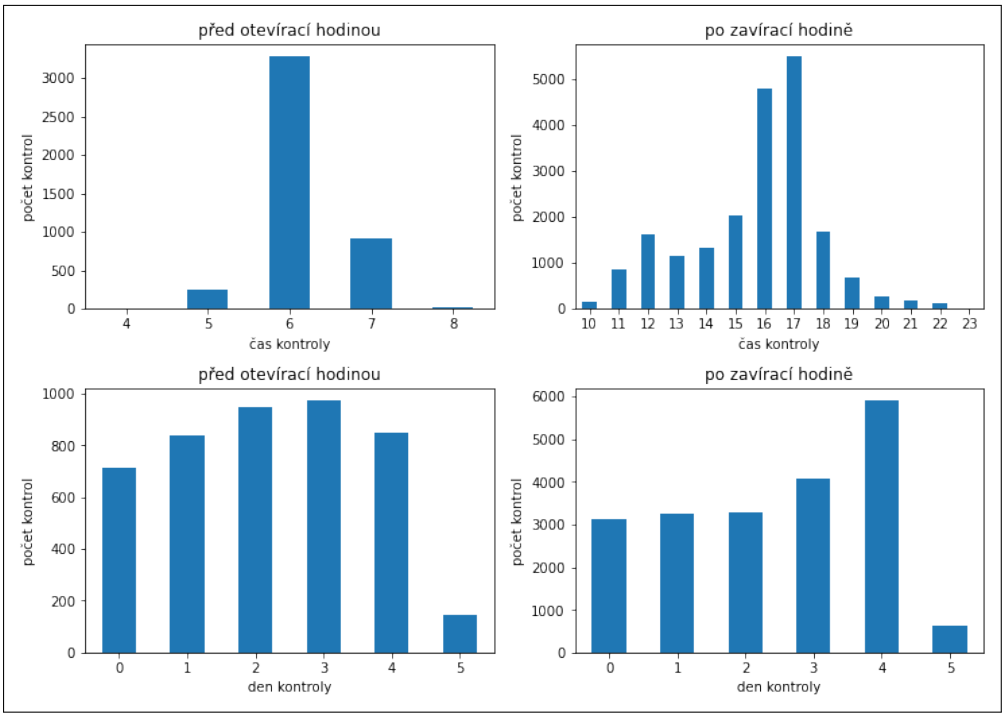
Prohlídky mimo pracovní dobu byly zaznamenány u **229** z 247 zkoumaných stanic a **43** stanic provádělo kontroly mimo pracovní den.

Skript a ukázka výsledků jsou součástí notebooku `out_of_working_hours.ipynb`.

3.4.2 Kapacita

Součástí datasetu STK2018 bohužel není doba trvání prohlídky, ale jenom datum a čas zápisu do systému. Stejně tak není součástí seznamu stanic dostupného na stránkách Ministerstva dopravy ani počet kontrolních linek. Ale na se každý rok zveřejňuje časové pracnosti kontrol a některé stanice na svých webových stránkách uvádějí počet kontrolních linek. Z toho lze odhadnout průměrný počet prohlídek, které stanice provede během určitého časového intervalu, a následně lze tento počet porovnat se skutečným počtem.

Byly zjištěny počty kontrolních linek a provozní doby některých vybraných



Obrázek 3.10: Počty prohlídek mimo pracovní dobu podle času a dnu kontroly

Tabulka 3.1: Kapacita vybraných stanic technické kontroly

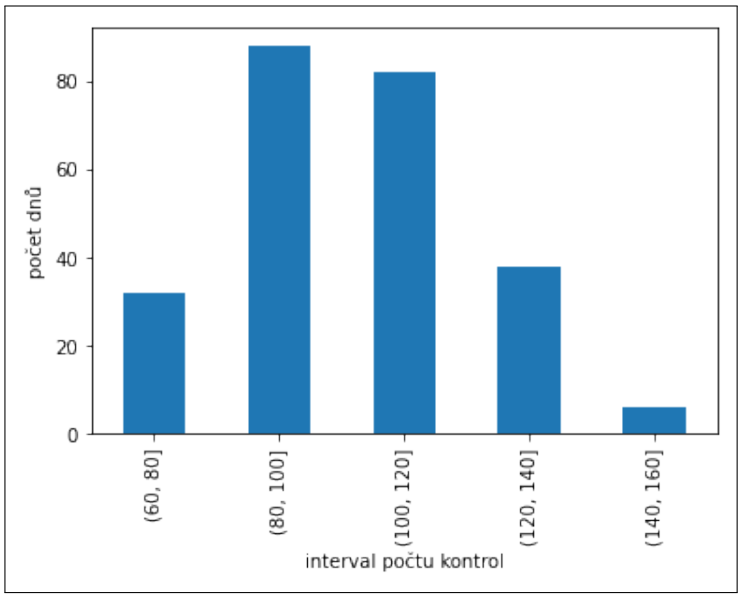
STK	kapacita	evidenční	pravidelná	pracnost	pracovní minuty
3851	2	6 665	18 342	574 443	330 000
3413	8	11 818	30 139	952 032	1 320 000
3814	3	3 316	11 944	364 461	612 000
3114	4	12 378	11 122	426 521	624 000
3234	2	5 696	10 747	353 377	215 100

stanic. Potom byly u těchto stanic vypočítány pravidelné prohlídky a evidenční kontroly kategorie M1 a odhadnuta celková časová pracnost v minutách za celý rok na základě hodnot uvedených na stránkách Ministerstva dopravy a popsaných v sekci 2.2.2. Následně byly vypočítány počty pracovních minut, které měla každá stanice v roce 2018, a to na základě počtů kontrolních linek, pracovní doby a počtu pracovních dnů v roce 2018. Výsledky jsou zobrazené v tabulce 3.1.

Spočítané pracovní minuty samozřejmě úplně neodpovídají skutečnému času strávenému na provedení prohlídek. Například jsou v počtu pracovních minut také prohlídky jiných kategorií a zároveň není ošetřena doba mezi jednotlivými prohlídkami a taktéž doba, ve které nejsou klienti apod. Také musí být vzato v úvahu to, že odhad pracnosti je jenom odhad a některé prohlídky

mohou trvat kratší dobu. Proto za podezřelé může být považován jenom skutečně velký rozdíl mezi odhadem pracnosti a celkovým počtem pracovních minut.

Tento rozdíl je vidět u stanice číslo 3851 (54 %). Pro tuto stanici bylo zjištěno, že v průměru provádí 101 kontrol kategorie M1 denně, ze kterých je 74 pravidelných a 27 evidenčních. Což je pro stanici, která obsahuje jenom dvě kontrolní linky poměrně vysoký počet. Na obrázku 3.11 je vidět, že v některých dnech stanice provedla více než 120 kontrol a v dalších dokonce více než 140.



Obrázek 3.11: Hustota prohlídek kategorie M1 stanice číslo 3851

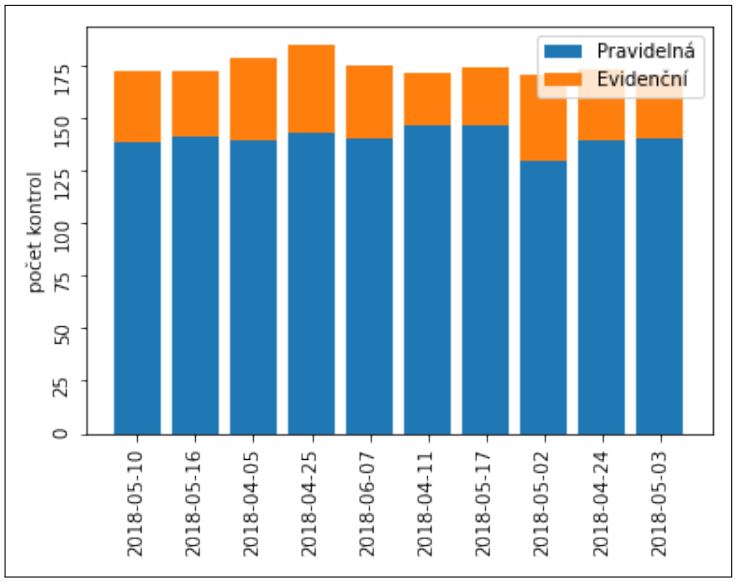
Lze předpokládat, že v dny s největší hustotou prohlídek stanice provedla anomální počet evidenčních kontrol, proto je počet záznamů tak vysoký. Ale na obrázku 3.12 je vidět, že počet pravidelných prohlídek v těch dnech sahá nad 120.

Pro uvedený počet pracovních linek a pracovní doby, která činí 11 hodin denně, musí platit, že každá pravidelná prohlídka trvala cca 11 minut a evidenční kontrola méně než 4 minuty, což je skutečně málo a s ohledem na odhad pracnosti toto vypadá podezřele.

Skript a ukázka výsledků jsou součástí notebooku `capacity.ipynb`.

3.4.3 Časové souběhy kontrol

Další metodou navrženou v rámci této práce pro odhalení anomálního chování na stanicích technické kontroly je hledání časově souběžných kontrol automobilů jednotlivých značek.



Obrázek 3.12: Počet prohlídek kategorie M1 stanice číslo 3851 v dnech největší hustoty

Pro každou značku vozidla, kterých je 6 266, bylo spočítáno kolikrát se na stejné stanici potkala s každou další značkou. Setkáním se rozumí časový souběh, což s ohledem na odhad trvání prohlídky bylo rozhodnuto stanovit 27 minut před a 27 minut po. Také z datasetu musejí být vyloučeny záznamy typu návěs, přívěs, přípojné vozidlo apod. Tyhle zvláštní typy jsou souběžné s určitými značkami automobilu a nic podezřelého v tom není. Nejčastější dvojice značek jsou samozřejmě kombinace Škody a dalších nejpopulárnějších značek v České republice, konkrétně pak Ford a Škoda, Renault a Škoda, Škoda a Peugeot, Volkswagen a Škoda apod. Toto vůbec není překvapující nebo anomální.

Pro zachycení anomálního chování pro každou dvojici značek byla proto vypočtena velmi naivní pravděpodobnost, že se potkají, na základě počtu automobilů příslušné značky v datasetu. Následně byl vypočten odhad počtu setkání na základě pravděpodobnosti a procentuální rozdíl mezi odhadem a skutečným počtem.

Dvojice značek automobilů s největším procentuálním rozdílem jsou:

- Gloria a Mercedes: Gloria je vojenské vozidlo, které se v datasetu vyskytuje jenom 4krát a vždy spolu s dalším vojenským vozidlem značky Mercedes (identifikátory vozidel jsou různé, takže jde jenom o souběh značek, nikoliv automobilů);

- Irisbus a SOR, Irisbus a Karosa: jsou to značky autobusů a je normální, že automobily patřící do jednoho autoparku jezdí na technickou prohlídku spolu;
- Porsche a Italmoto: Italmoto je velmi vzácná značka motocyklu (16 záznamů), která byla prohlédnuta na jedné stanici v jeden den a Porsche je jednoduše nejvzácnější značka, která v ten čas náhodou přijela na prohlídku.

Dvojice značek automobilů s největším procentuálním rozdílem a počtem setkání větším než 1000, jsou:

- Tatra a DAF;
- Scania a DAF;
- MAN a DAF;
- a další modely nákladních automobilů.

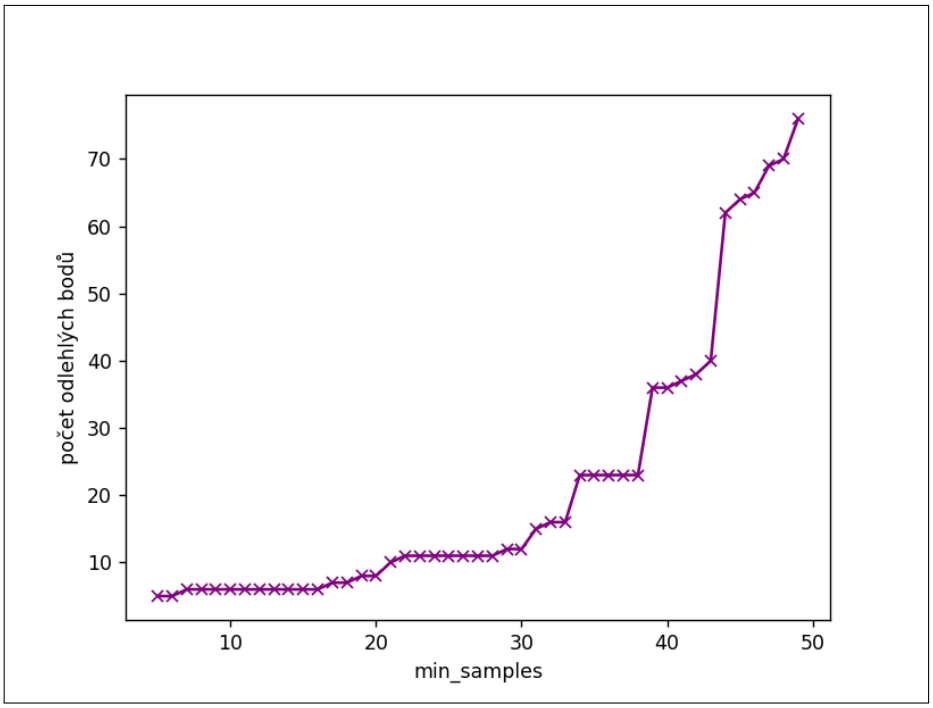
Z toho plyne, že nákladní automobily, traktory a tahače mají sklony jezdit na prohlídku spolu, proto byly z datasetu vyloučeny všechny typy vozidel, kromě osobních automobilů a skript byl spuštěn znovu.

Dvojice značek osobních automobilů s největším procentuálním rozdílem a počtem setkání větším než 100, jsou:

- Ferrari a BMW;
- Lexus a Mercedes-Benz;
- Volvo a Lexus;
- Mercedes-Benz a Porsche;
- BMW a Porsche.

Dvojice značek osobních automobilů s největším procentuálním rozdílem a počtem setkání větším než 1000, jsou:

- Volvo a Mercedes-Benz;
- Volvo a BMW;
- BMW a Mercedes-Benz;
- Toyota a Volvo;
- Audi a Mercedes-Benz.



Obrázek 3.13: DBSCAN: Počet odlehlých bodů podle parametru min_samples

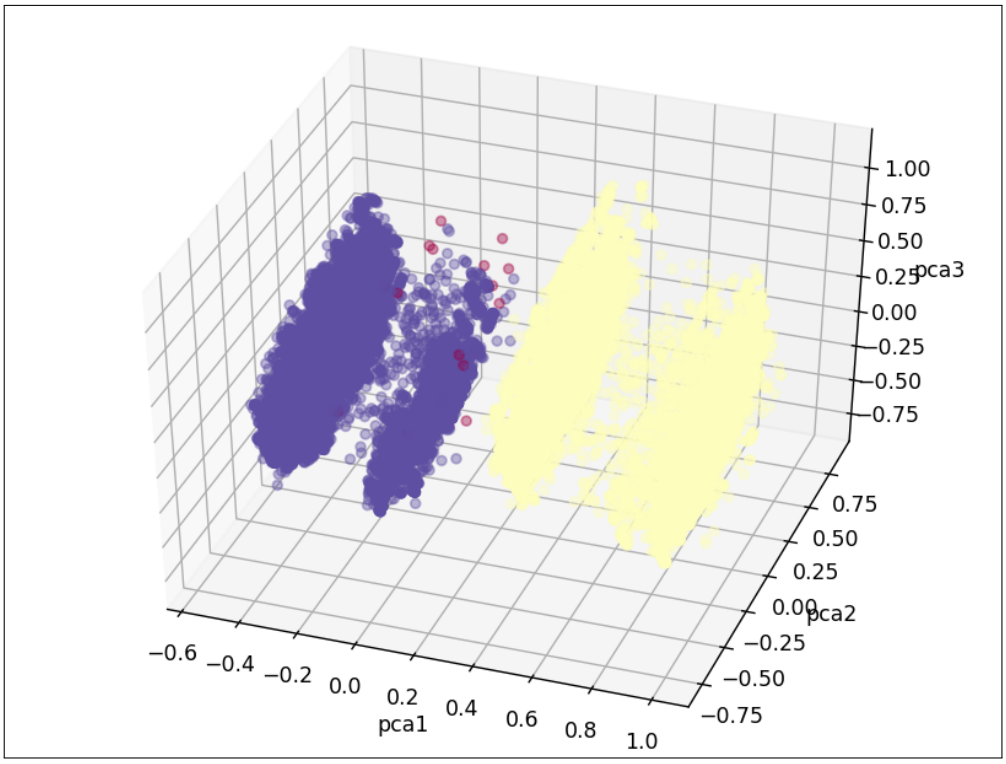
Většina uvedených souběžných značek patří do skupiny drahých automobilů a lze předpokládat, že vozidla příslušných značek mají sklon jezdit na kontroly do určité skupiny STK, což podezřelé není.

Anomálními ale jsou například časté souběhy dražších automobilů a levnějších. Častými jsou myšleny souběhy, jejichž počet daleko přesahuje odhad. Například:

- Alfa Romeo a Fiat se na jedné ze stanic setkaly dvakrát tolik, než bylo odhadnuto;
- Saab a Audi nebo Mercedes se také na jedné ze stanic setkaly dvakrát tolik, než bylo odhadnuto;

Výsledky této analýzy poukázaly na přítomnost příznaků podezřelého chování, tj. anomálně častých souběhů značek automobilů, na 46 z 419 zkoumaných stanic.

Skript a ukázka výsledků jsou součástí notebooku `vehicle_conjunction.ipynb`.



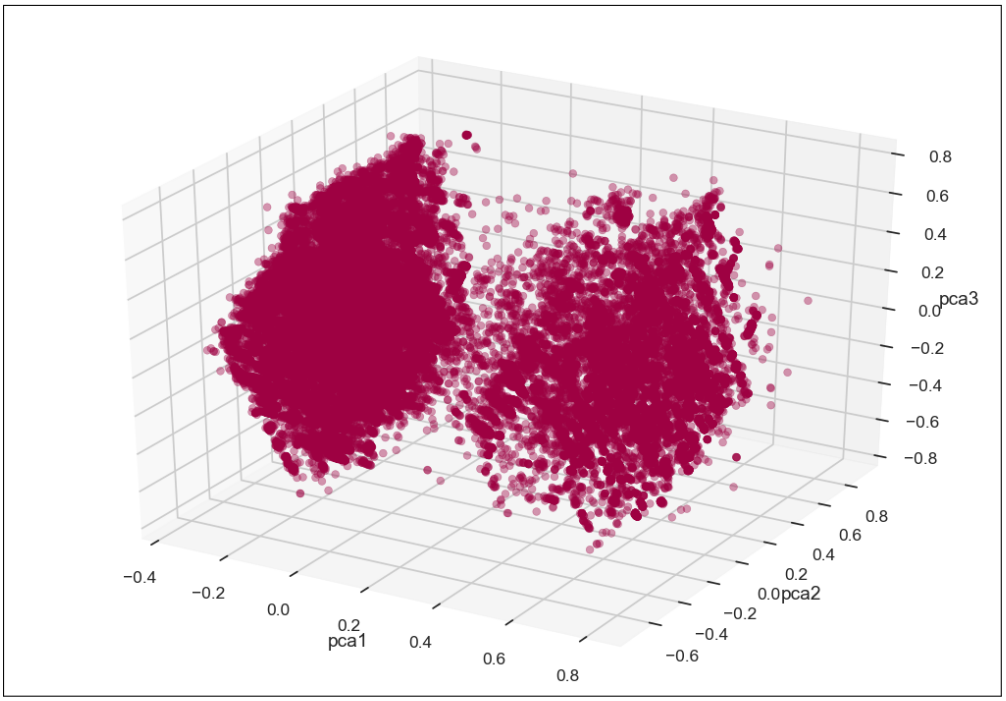
Obrázek 3.14: DBSCAN: Výsledek běhu algoritmu

3.5 Detekce anomálií

V rámci této práce byla provedena detekce anomálií pomocí shlukovací metody DBSCAN, popsané v sekci 2.4.

Nejprve byly všechny hodnoty převedeny do numerických a normalizovány stejně jako v sekci 3.3. Potom byly opět vypočítány hlavní komponenty a výsledný trojdimenzionální dataset byl použit jako vstup do algoritmu DBSCAN. Byla využita třída DBSCAN balíčku `cluster` knihovny `scikit-learn`. Bohužel implementace tohoto algoritmu je paměťově náročná a rychle vyčerpávala celou dostupnou paměť (cca 9 GB), pokud vstupní data obsahovala víc než 100 000 řádků. Proto byl objem vstupních dat omezen na tento počet záznamů a algoritmus byl spuštěn opakovaně.

Důležitými vstupními parametry algoritmu jsou maximální vzdálenost dvou sousedních bodů a minimální počet sousedů proto, aby se bod považoval za jádro (`min_samples`). Při vzdáleností menší než 0,2 se za odlehlé považovalo 20—25 % datových bodů a při vzdáleností větší než 0,2 naopak skoro žádné procento. Proto bylo rozhodnuto, že je 0,2 optimální vzdálenost. Na obrázku 3.13 je vidět vývoj počtu nalezených odlehlých bodů se zvýšením `min_samples`. Počet anomálií mírně roste do cca 38, potom začíná růst rychle. Z toho bylo



Obrázek 3.15: DBSCAN: Výsledek běhu algoritmu po vyloučení sloupců VyslSTK a VyslEmise

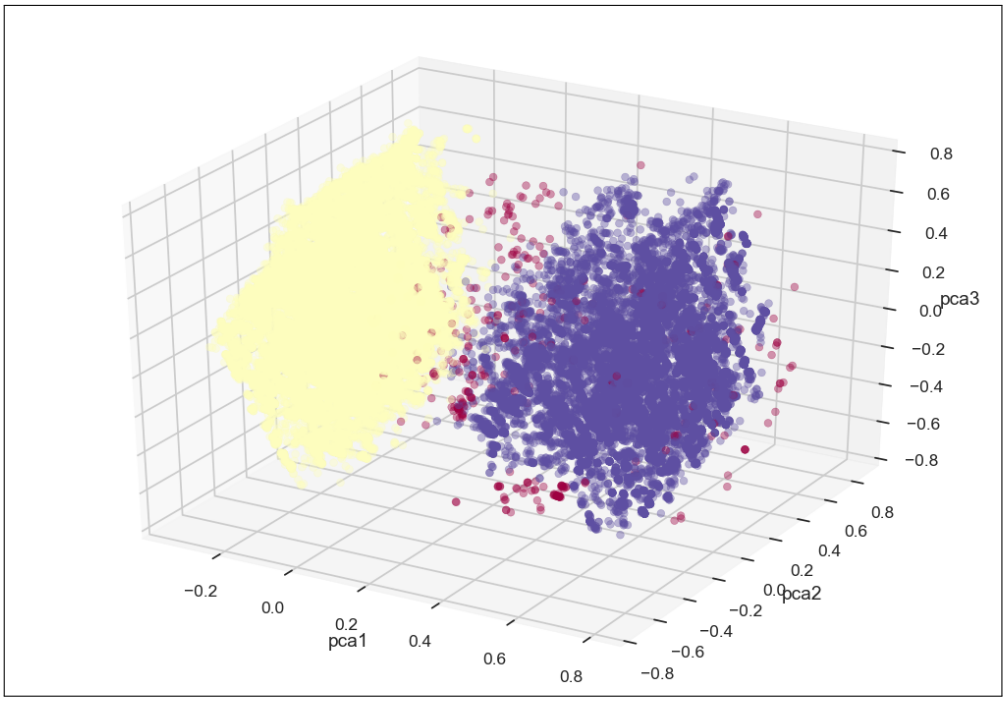
rozhodnuto, že optimální počet je 38 (místo zlomu grafu).

Zajímavou vlastností algoritmu DBSCAN je to, že jako vstupní parametr není vyžadován počet shluků. Při každém spuštění vypadal výsledek algoritmu skoro stejně (viz obrázek 3.14) a obsahoval 2 hlavní shluky a cca 20 odlehlých bodů (na grafu jsou označeny červenou barvou).

Algoritmus byl spuštěn opakovaně a všechny nalezené body, byly seskupeny do jednoho datasetu.

Většinu (95.7 %) odlehlých bodů tvoří prohlídky, které dostaly jako výsledek kontroly „částečně způsobilé“a jako výsledek měření emisí „částečně vyhovuje“ (52 %) nebo „nevyhovuje“ (45.6 %). Samozřejmě tehdy mají na rozdíl od celého datasetu mnohem vyšší průměrný počet vad typu B (nevyhovující emise většinou spadají do této kategorie) a nepatrně vyšší počet vad ostatních typů. Průměrný počet najetých kilometrů a stáří automobilů, patřících do anomálií, se skoro neliší od normálních bodů. Překvapivé je, že nejčastější značkou mezi anomáliemi je Fiat, druhou nejčastější je Škoda.

Obecně vzato byly tyto body algoritmem považovány za anomálie nejspíše kvůli tomu, že rozdělení prohlídek podle výsledku kontroly a měření emisí je zřejmě zešikmené. Po vyloučení dvou sloupců, VyslSTK a VyslEmise, vypadalo rozdělení na shluky úplně jinak (viz obrázek 3.15) a se stejnými vstupními parametry algoritmu nebyl nalezen žádný odlehlý bod.



Obrázek 3.16: DBSCAN: Výsledek běhu algoritmu po vyloučení sloupců VyslSTK a VyslEmise a změně parametru ε

Po změně parametru ε, který označuje maximální vzdálenost mezi body, aby se považovaly za sousední, na 0,15 bylo nalezeno mnohem víc odlehlých bodů (viz obrázek 3.16, anomálie jsou označené červenou barvou). Většina z těchto bodů reprezentovala opakované technické silniční prohlídky a další druhy, kterých je v datasetu málo. Po vyloučení sloupce „DrTP“ (druh prohlídky) nebyla nalezena žádná anomálie a výsledek vypadal podobně jako na obrázku 3.15.

Skript a ukázka výsledků jsou součástí notebooku `anomaly_detection.ipynb`.

Kapitola 4

Návrh

V této kapitole jsou definovány hlavní případy užití webové aplikace, která bude vytvořena v rámci této práce, popsán databázový model aplikace, navrhnuta architektura portálu a popsány technologie a nástroje, které budou použity při implementaci.

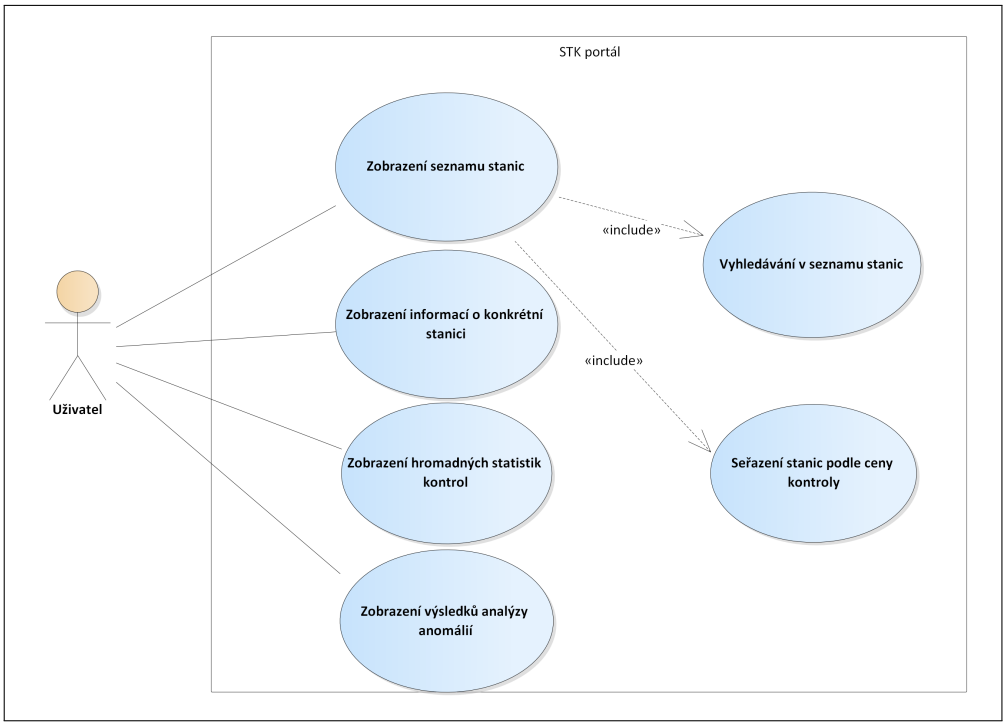
4.1 Případy užití

Případ užití je seznam akcí, které popisují interakcí mezi systémem, v tomto případě webovým portálem, a uživatelem. Alistair Cockburn udává tuto definici případů užití: „Případem užití je popis možných sekvencí interakcí mezi systémem a externími aktéry souvisejících s konkrétním cílem.“ (překlad vlastní) [26, s. 15].

Definování případů užití je důležitou součástí návrhu libovolného systému. Během tohoto procesu se stanovují cíle projektu, které pomáhají odhalit případné slabé stránky. Také se usnadňuje vymezení primárních bodů pro implementaci.

Pro zobrazení případů užití existuje velké množství různých nástrojů. V rámci této práce budou využity UML diagramy nakreslené pomocí programu Enterprise Architect [27] od společností Sparx Systems.

Na obrázku 4.1 jsou uvedeny základní případy užití portálu, tj. funkcionality, které bude navrhovaný systém poskytovat svým uživatelům. Model obsahuje pouze jednoho aktéra, jelikož účel a způsob použití webové aplikace nezáleží na typu uživatele.



Obrázek 4.1: Model případů užití

4.1.1 Zobrazení seznamu stanic

Umožňuje uživateli zobrazit úplný seznam stanic technické kontroly v České republice.

Základní scénář:

1. Uživatel si přeje zobrazit celý seznam stanic technické kontroly v České republice.
2. Uživatel stiskne tlačítko *Stanice* v hlavičce portálu.
3. Uživateli se načte stránka *Stanice* obsahující tabulku ze všemi stanicemi v České republice se sloupci STK ID, provozovatel, město, adresa a cena prohlídky.

Alternativní scénář:

1. Scénář začíná ve 3. kroku základního scénáře, jestliže se nenačte seznam stanic.
2. Systém zobrazí hlášku v podobě: „Je nám líto, ale při načítání seznamu stanic došlo k chybě.“

4.1.2 Vyhledávání v seznamu stanic

Umožňuje uživateli vyhledávat v úplném seznamu stanic technické kontroly v České republice.

Základní scénář:

1. Po načtení seznamu stanic si uživatel přeje v tom seznamu vyhledávat.
2. Uživatel zadá číslo, adresu, město nebo provozovatele stanice do textového pole umístěného nad seznamem stanic.
3. Systém přefiltruje seznam stanic a vyloučí záznamy, u kterých žádný z atributů neodpovídá požadavku.
4. Uživateli se zobrazí přefiltrovaný seznam stanic, které odpovídají požadavku.

Alternativní scénář:

1. Scénář začíná v 4. kroku základního scénáře, jestliže po filtrování nezbude ani jedna stanice.
2. Systém zobrazí hlášku v podobě: „Je nám líto, ale požadovaná stanice nebyla nalezena.“

4.1.3 Seřazení stanic podle ceny kontroly

Umožňuje uživateli seřadit stanice buď již přefiltrované po vyhledávání, nebo všechny stanice technické kontroly podle ceny technické prohlídky.

Základní scénář:

1. Po načtení seznamu stanic nebo po filtraci seznamu si uživatel přeje seřadit stanice podle ceny technické kontroly.
2. Uživatel klikne na sloupec *Cena*.
3. Uživateli se zobrazí seznam stanic seřazený vzestupně podle ceny technické prohlídky.

Alternativní scénář:

1. Scénář začíná ve 3. kroku základního scénáře, jestliže u některých stanic nebude uvedená cena kontroly.
2. Stanice, u kterých nebude uvedená cena, budou zobrazené na konci seřazeného seznamu.

4.1.4 Zobrazení informací o konkrétní stanici

Umožňuje uživateli zobrazit kompletní informace o každé stanici technické kontroly včetně statistik kontrol v roce 2018.

Základní scénář:

1. Po načtení seznamu stanic si uživatel přeje zobrazit podrobnější informace o konkrétní stanici.

2. Uživatel klikne na číslo požadované stanice zobrazené ve formě tlačítka.

3. Uživateli se načte stránka obsahující údaje o vybrané stanici: rozsah oprávnění, provozní doba, e-mail, telefonní číslo apod. A taktéž statistiky kontrol v roce 2018 ve formě grafů.

Alternativní scénář:

1. Scénář začíná ve 3. kroku základního scénáře, jestliže pro vybranou stanici není zaznamenán některý z údajů.

2. Uživateli se zobrazí jenom ty sekce údajů, pro které jsou k dispozici data.

4.1.5 Zobrazení hromadných statistik kontrol

Umožňuje uživateli zobrazit hromadné statistiky kontrol na stanicích technické kontroly v roce 2018.

Základní scénář:

1. Uživatel si přeje zobrazit hromadné statistiky kontrol na stanicích technické kontroly v České republice.

2. Uživatel stiskne tlačítko *Statistiky* v hlavičce portálu.

3. Uživateli se načte stránka *Statistiky*, obsahující různé grafy a informace, které popisují statistiky, získané během analýzy datasetu STK 2018 a týkající se prohlídek obecně.

Alternativní scénář:

1. Scénář začíná ve 3. kroku základního scénáře, jestliže se nenačtou grafy.

2. Systém zobrazí hlášku v podobě: „Je nám líto, ale při načítání statistik došlo k chybě.“

4.1.6 Zobrazení výsledků analýzy anomálií

Umožňuje uživateli zobrazit výsledky analýzy anomálií – odhalení podezřelého chování na stanicích technické kontroly.

Základní scénář:

1. Uživatel si přeje zobrazit výsledky analýzy anomálií a podezřelého chování v záznamech kontrol.
2. Uživatel stiskne tlačítko *Anomálie* v hlavičce portálu.
3. Uživateli se načte stránka *Anomálie* obsahující různé grafy a informace získané během analýzy datasetu STK 2018. Ty popisují anomálie a příznaky podezřelého chování.

Alternativní scénář:

1. Scénář začíná ve 3. kroku základního scénáře, jestliže se nenačtou grafy.
2. Systém zobrazí hlášku v podobě: „Je nám líto, ale při načítání anomálií došlo k chybě.“

4.2 Databázový model

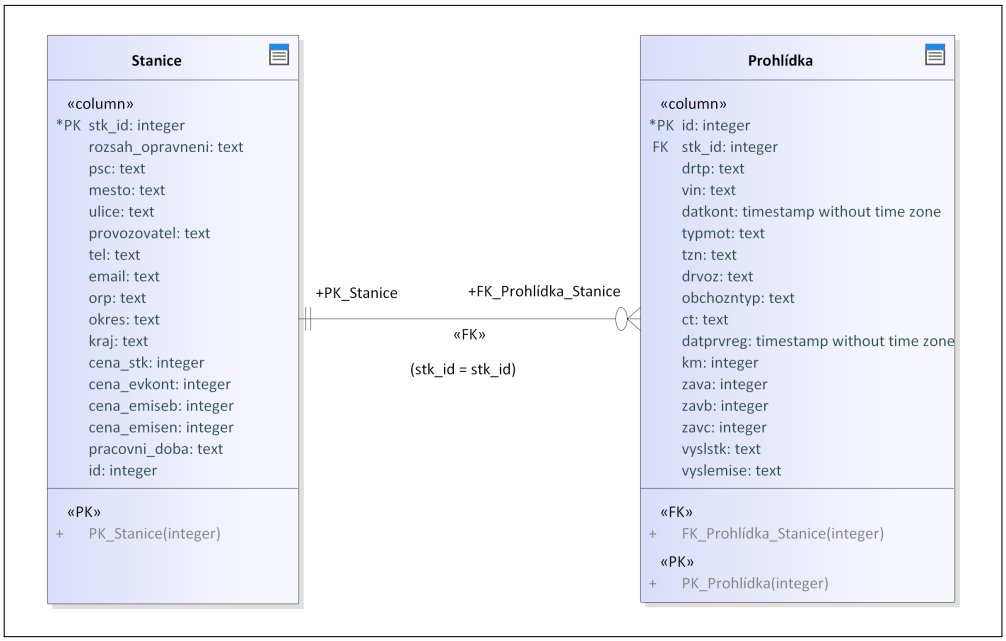
Tato sekce popisuje navržený způsob ukládání dat do relační databáze. Pro tyto účely byla zvolena databáze PostgreSQL. Databázový model aplikace je velmi jednoduchý, jelikož se data na portálu pouze zobrazují a nemění se a zároveň jsou všechny položky tabulek předem definované zdroji, odkud byly stažené.

Databázový model zahrnuje pouze dvě entity: stanice a prohlídka. Prohlídka obsahuje přesně ty sloupce, jak jsou definované v datasetu STK2018 a popsané v sekci 2.3. Stanice obsahuje všechny sloupce ze seznamu stanic také popsaného v sekci 2.3 a zároveň informace, které byly vytěžené z webu v rámci 3.1.2, tj. cena technické prohlídky, cena evidenční kontroly, cena emise benzínového a naftového vozidla a pracovní doba stanice.

Tabulky jsou navzájem propojené cizím klíčem „stk_id“ označujícím unikátní identifikátor stanice. Jelikož se kontrola provádí v určitém místě, vztah mezi entitami má kardinalitu 1:N, kdy je každému záznamu o kontrole jednoznačně přiřazena stanice.

4.3 Architektura a technologie

Implementace portálu bude rozdělena na dvě části, a to na backendový server a samotnou webovou aplikaci (dále jen *klient*). Backendový server se bude připojovat k databázi a obsluhovat RESTové požadavky ze strany klienta.



Obrázek 4.2: Databázový model

Klient bude zpracovávat odpovědi backendu a obsahovat frontendové části portálu.

4.3.1 Backend

Backendový server bude vyvinut v jazyce `Java` s použitím frameworku `Spring Boot`. Hlavním případem užití serveru bude poskytování dat pro webovou aplikaci, klienta, která pomocí nich bude zobrazovat statistiky kontrol a informace o stanicích.

Server bude napojen na databázi a bude do ní posílat požadavky podle volání jednotlivých endpointů klientem. Jelikož bude databáze obsahovat velký počet záznamů kontrol, volání, která se budou tykat celé sady prohlídek (například počet kontrol po jednotlivých měsících), budou hned vracet požadovaná čísla, nikoli samotné prohlídky. Naopak volání, týkající se jednotlivých stanic, budou vracet seznamy prohlídek, aby se jednodušeji zobrazovaly grafy a statistiky.

4.3.2 Klient

Druhá část portálu bude vyvinuta na platformě `Angular` [28] v jazyce `TypeScript` [29]. Webová aplikace bude obsahovat následující obrazovky:

- **Úvod**: hlavní uvítací stránka portálu. V průběhu analýzy stávajících řešení bylo také zjištěno, že podobné portály obsahují mapu České re-

publiky, na které jsou označeny stanice, proto by v tomto portálu také měla být.

- **O portálu**: bude obsahovat popis jednotlivých stránek, definovat k čemu portál slouží a obsahovat stručný popis dat.
- **Stanice**: bude obsahovat seznam stanic, pole pro vyhledávání a možnost seřazení stanic podle ceny technické prohlídky.
- **Jednotlivá stanice**: na rozdíl od ostatních stránek není součástí hlavičky portálu, protože je generována zvlášť pro každou stanici, na kterou klikne uživatel. Bude obsahovat informace o vybrané stanici, zobrazovat statistiky kontrol v roce 2018 a anomálie. Mezi grafy budou patřit minimálně:
  - rozdělení kontrol podle výsledku,
  - počet kontrol podle dnů v tydnu,
  - počet kontrol podle časového intervalu (pro odhad zátěže),
  - nejčastější značky automobilů.
- **Statistiky**: bude zobrazovat hromadné statistiky kontrol v roce 2018. Mezi grafy budou patřit minimálně:
  - rozdělení kontrol podle výsledku,
  - počet kontrol podle měsíce,
  - průměrné stáří automobilů, průměrný počet najetých kilometrů apod.,
  - nejčastější značky automobilů,
  - ceny prohlídek podle kraje.
- **Anomálie**: bude zobrazovat vybrané výsledky analýzy podezřelého chování na stanicích technické kontroly.

Kapitola 5

Implementace

Tato kapitola se věnuje postupu tvorby webového portálu, tj. databázovým tabulkám, struktuře backendového serveru a frontendové aplikace a testování. Implementace je řízena návrhem představeným v kapitole 4 a je zaměřená na realizaci případů užití, které jsou popsány v sekci 4.1. Při vývoji byla využita prostředí IntelliJ IDEA [30] a WebStorm [31] společnosti JetBrains.

5.1 Databáze

Pro ukládání tabulek byl zvolen opensourcový objektově-relační databázový systém PostgreSQL [32]. Celá práce byla vypracována pod vedením OpenDataLabu, otevřené laboratoře založené ve spolupráci s ČVUT FIT. Databáze byla vytvořena na serveru, který náleží společnosti Profinit EU, s.r.o., který je partnerem OpenDataLabu. Server vyžaduje přístup z interní VPN. Pro přístup z vlastního počítače bez připojení na VPN musí být na server přidán veřejný SSH-klíč stanice.

Pomocí nástroje pgAdmin [33], což je administrátorský program pro správu databázového serveru PostgreSQL, byly v databázi založeny tabulky se sloupci přesně odpovídajícími sloupcům výsledných CSV souborů vytvořených po zpracovávání datasetů a stahování informací z webu. Následně bylo pomocí knihovny psycopg2 [34], což je PostgreSQL adaptér pro Python, navázáno připojení na server a nahrána data. Z toho důvodu, že dataset STK2018 obsahuje cca 3 700 000 záznamů, byla data nahrána v cyklu po částech.

Po nahrání dat se v databázi nacházely dvě tabulky, a to `kontroly` (3 724 641 řádků) a `seznam_stk` (372 řádků), které obsahovaly všechna potřebná data pro následnou tvorbu portálu.

5.2 Backend

V této sekci je popsána struktura té části aplikace, která poskytuje přístup k datům a je následně volaná klientem. Pro implementaci backendové části projektu byl použit framework `Spring Boot` [35] verze 2.2.6 a nástroj pro automatizaci buildu aplikace `Maven` [36]. Šablona projektu byla vygenerována pomocí nástroje `Spring Initializr` [37]. Provolávání endpointů běžícího backendového serveru je jediný způsob, jak druhá část aplikace (*klient*) přistupuje k datům.

5.2.1 Struktura

Jelikož se backendová část aplikace v tomto projektu zabývá pouze zpracováním `GET` požadavků do databáze, je její struktura velmi jednoduchá (viz obrázek 5.1, na kterém jsou vidět všechny třídy implementované v rámci této části aplikace).

Složka **entity** obsahuje dvě třídy, `Inspection` a `Station`, jejichž instance reprezentují záznamy v databázových tabulkách. Rozhraní tříd jsou triviální: privátní proměnné, namapované na jednotlivé atributy v tabulkách pomocí anotace `@Column`, a standardní sada metod pro objekt v Javě: gettery, settery, `equals()`, `hashCode()` a `toString()`. Celá třída je anotovaná `@Entity` a `@Table` s názvem reprezentované tabulky.

Složka **repo** obsahuje interface, které zjednodušují proces přístupu k databázovým tabulkám a získávání potřebných záznamů. Funkčnost je podrobněji popsána v sekci 5.2.2.

Složka **controller** obsahuje třídy, které definují endpointy, a data, které budou požadována z databáze při jejich volání. Funkčnost je podrobněji popsána v sekci 5.2.3.

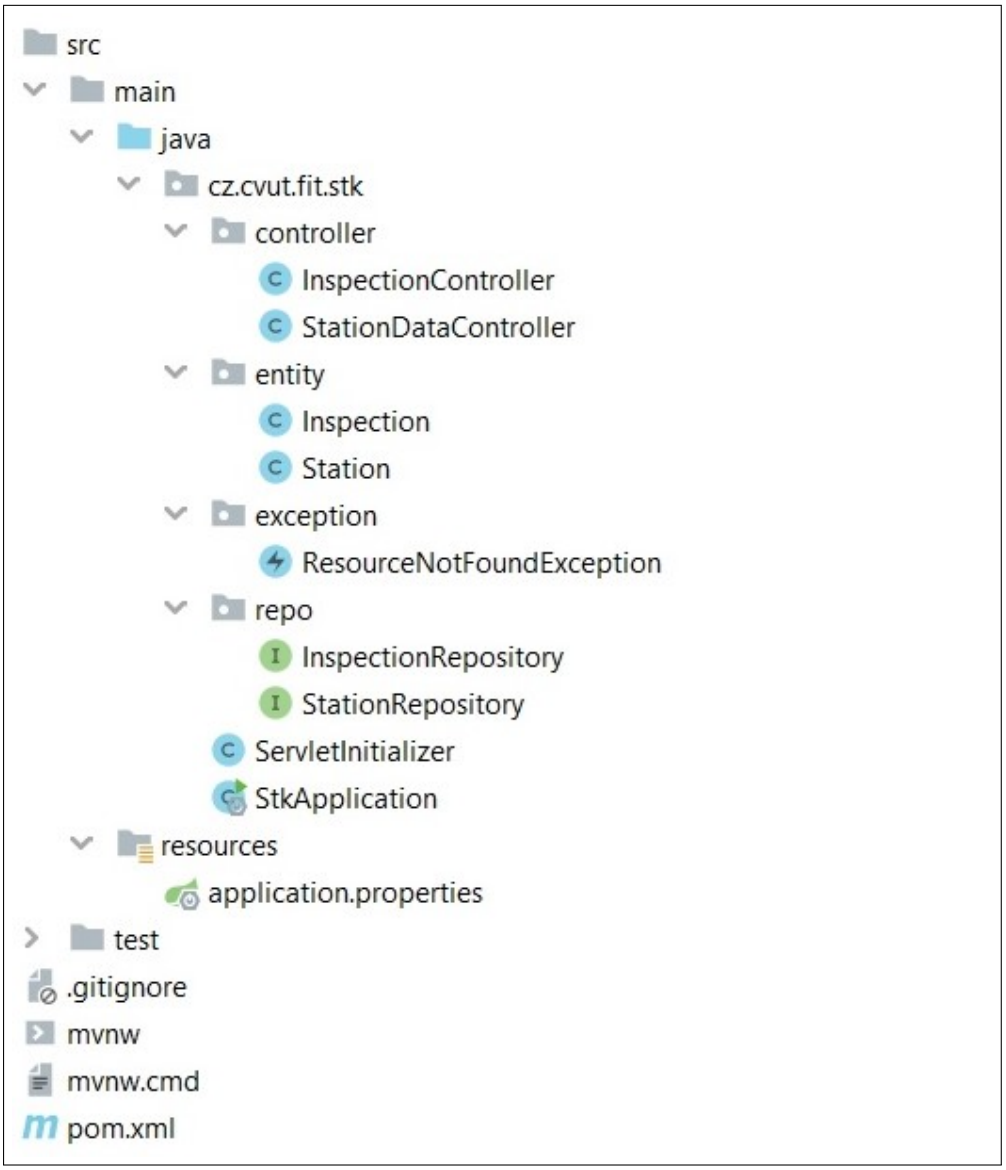
Složka **exception** obsahuje jednu výjimku, `ResourceNotFoundException`, která se vyvolává v případě, kdy při volání endpointu nebudou nalezena žádná data.

Složka **test** obsahuje testy databázových volání a je podrobněji popsána v sekci 5.2.4.

Třída **StkApplication** je hlavní třída aplikace obsahující metodu *main*. Je anotovaná `@SpringBootApplication` a indikuje hlavní třídu Spring Boot aplikace.

Soubor **application.properties** je soubor sloužící k definici důležitých konfigurací pro projekt. V tomto případě definuje primárně způsob připojení k databázi, tzn. link na server, přihlašovací údaje a JDBC (tady *postgresql*).

Soubor **pom.xml** je základní XML reprezentace celého projektu pro nástroj Maven. Obsahuje popis projektu (název, verzi apod.) a taktéž poskytuje snadný způsob vkládání a správy závislostí. Kromě závislostí Spring Bootu, vygenerovaných automaticky při založení projektu, byl do souboru taktéž přidán `org.postgresql`.



Obrázek 5.1: Struktura backendové aplikace

5.2.2 JPA

Modul `JPA` frameworku `Spring` je velmi užitečný nástroj pro práci s databázovými tabulkami. Stačí jenom definice třídy typu `interface`, která je anotovaná `@Repository` a dědí generickou třídu `JpaRepository` s parametry `ObjectClass` a `id`, kde `ObjectClass` je v tomto případě Inspection nebo Station a `id` je typu Long. Podobný prázdný interface umožňuje získat buď všechny záznamy z tabulky, na který je namapován příslušný objekt, nebo jeden záznam podle `id`.

Pro účely tohoto projektu jsou ale potřebné jiné typy dotazů. Modul JPA dovoluje vytvářet jiné dotazy pomocí definice metod přímo ve třídě, která dědí `JpaRepository`. Dotaz, který bude potom poslán do databáze, je generován na základě názvu metody a není ani potřeba definovat její tělo. Například v rámci této práce byly vytvořeny následující metody:

- `findByStationId(Long stationId)` – vrací prohlídky, které byly provedeny na vybrané stanice. Je používána pro zobrazení statistik kontrol na jednotlivé stanici.
- `countByInspectionDateBetween(Date before, Date after)` – vrací počet prohlídek, které byly provedeny mezi určitými daty. Je používána pro zobrazení hromadných statistik kontrol v roce 2018.
- `findByInspectionDateBetweenAndStationId(Date before, Date after, Long stationId)` – vrací prohlídky, které byly provedeny na vybrané stanice mezi určitými daty. Je používána pro zobrazení statistik kontrol na jednotlivé stanici.

Je vidět, že podobným způsobem lze libovolně filtrovat prohlídky a stanice, vracet počet řádku s určitými vlastnostmi apod. Modul JPA tedy poskytuje možnosti jak vytvářet, modifikovat a mazat záznamy v databázi. V rámci tohoto projektu jsou však zbytečné, protože backendový server slouží jenom pro získávání dat.

5.2.3 REST

Aplikace obsahuje dva controllery: `InspectionController` a `StationController`, každý z nich je věnován své databázové tabulce. Controllery jsou anotovány `@RestController` a `@RequestMapping`. Do vstupního parametru `path` anotace `@RequestMapping` je dána kořenová cesta controlleru. Jednotlivé metody jsou anotovány `@GetMapping` se vstupním parametrem, označujícím cestu, kterou je volána konkrétně tato metoda.

Do controllerů jsou pomocí anotace `@Autowired` importovány JPA repositáře (StationRepository do StationControlleru a InspectionRepository do InspectionControlleru) a volané v jednotlivých metodách. Metody zavolají vybranou metodu repositáře, zalogují výsledek pomocí jednoduchého loggeru

a vrátí `ResponseEntity` se statusem *OK* a nalezenými stanicemi nebo kontrolami v těle, nebo vyvolají `ResourceNotFoundException`, pokud s požadovánými filtry nebude nic nalezeno.

Výsledek volání endpointu `/api/inspections/6014`, který vrací kontrolu s konkrétním id je následující:


```
{
    "id":6014,
    "stationId":3234,
    "inspectionType":"pravidelna",
    "inspectionDate":"2018-01-12T07
       :14:01.840+0000",
    "engineType":"780",
    "vehicleBrand":"KTM",
    "vehicleType":"MOTOCYKL",
    "vehicleModel":"450 EXC",
    "vehicleCategory":"LC",
    "firstRegistrationDate":"2009-05-28T22
       :00:00.000+0000",
    "mileage":1000,
    "defectCountA":0,
    "defectCountB":0,
    "defectCountC":0,
    "inspectionResult":"zpusobile",
    "emissionControlResult":"---",
    "vin":"VBKEXA40X9M372337"
}
```


Výpis 5.1: Příklad výsledku volání backendu

5.2.4 Testování

Testování backendového serveru je realizováno pomocí Unit testů. Unit testing je testování funkčností oddělených jednotek kódu. V průběhu testování se provolávají všechny endpointy a je kontrolováno, jestli výsledek odpovídá očekávané struktuře. Případná změna struktury vrácených odpovědi může způsobit nedefinované chování aplikace klient, proto se testy, nacházející ve složce `test`, automaticky spouštějí při každém sestavení projektu pomoci nástroje `Maven`.

Pro implementaci testu byla použita opensourcová knihovna `JUnit` verze 5 [38] a její metody `assertTrue`, `assertEquals` a `assertFalse`. Byly kontrolovány struktury odpovědi, statusy požadavků a správné vyvolávání chyby.

5.3 Klient

Druhá část portálu je vytvořena na platformě `Angular` [28] v jazyce `TypeScript` [29]. Webová aplikace volá backendový server podle toho, jaké informace chce uživatel na stránce zobrazit. Na základě odpovědi pak generuje reprezentaci požadavků, tj. vypisuje tabulky a kreslí grafy. Aplikace má minimalistický grafický design, jednoduchou hlavičku pro navigaci po portálu a šest obrazovek.

5.3.1 Struktura

Struktura projektu *klient* je vidět na obrázku 5.2.

Soubor **main.ts** je hlavní vstupní bod aplikace, který definuje primární komponentu, používanou při spouštění aplikace, tedy `AppComponent`.

Vyvinutý webový portál je SPA (Single Page Application), což je webová aplikace, která je umístěna na jedné stránce. Celý kód je obnoven pomocí jediného zásobníku stránek a přechod mezi stránkami je uskutečněn bez obnovení celé stránky. Soubor **index.html** reprezentuje hlavní stránku a je skoro prázdný bez započtení importu hlavní komponenty (`AppComponent`) do těla.

Složka **assets** by měla obsahovat obrázky a další média importovaná do projektu, což je v tomto případě jenom pár obrázků a symbolů.

Složka **services** obsahuje dva interface a jejich implementace, které volají backend: `InpectionsAPIService`, vracející seznam kontrol, a `StationsAPIService`, vracející seznam stanic.

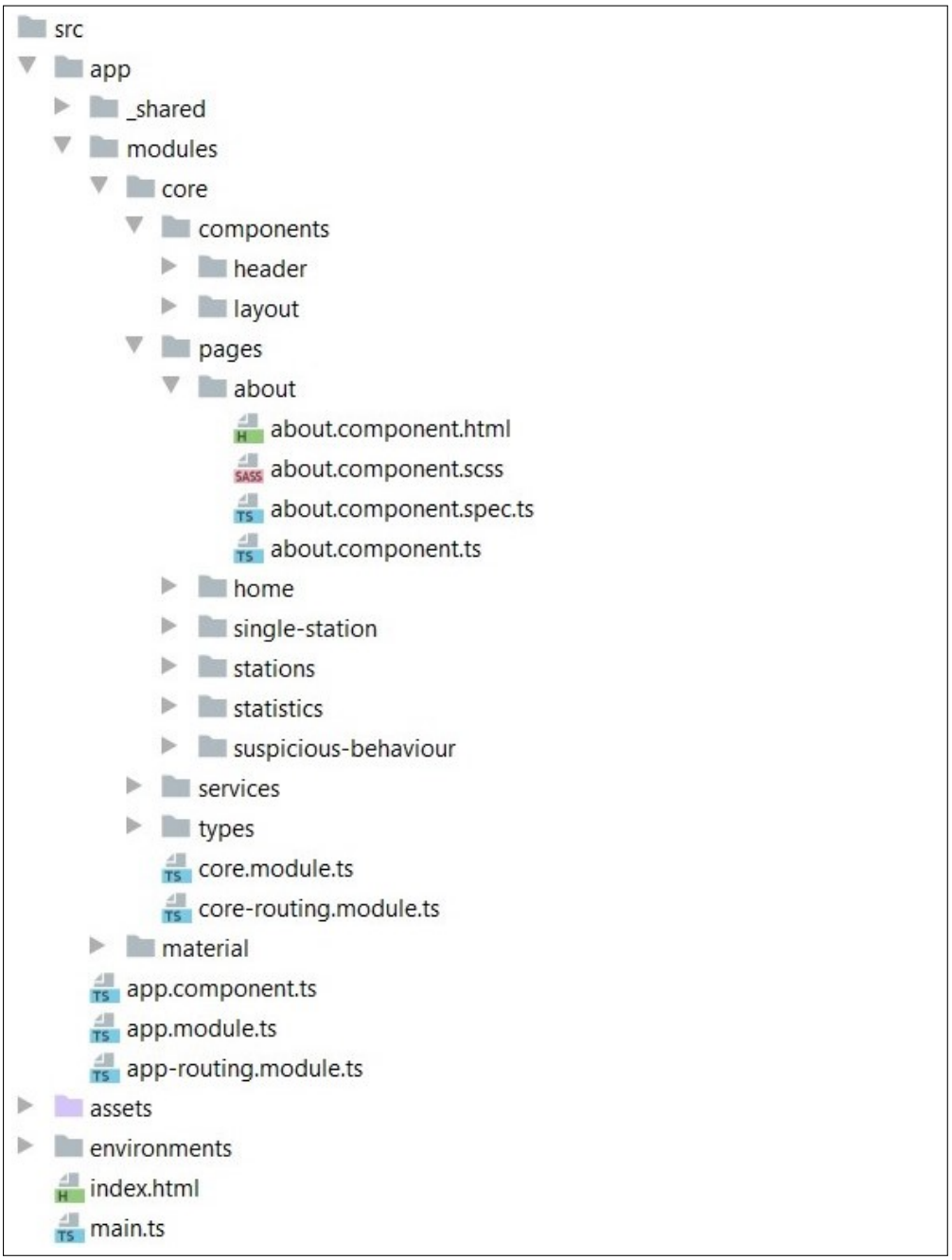
Složka **types** obsahuje definice typů, do kterých se automaticky převádějí odpovědi backendu: `InpectionsInterface` a `StationsInterface`, ve kterých jsou definovány atributy jednotlivých stanic a prohlídek.

Složka **pages** obsahuje šest dalších složek – jednu pro každou obrazovku. Složka každé obrazovky se skládá z:

- její HTML reprezentace (například about.component.html);
- SCSS souboru pro definici stylu (například about.component.scss);
- TypeScript souboru, který reprezentuje komponentu a zabývá se získáváním a zpracováním dat, která budou zobrazena na stránce (například about.component.ts);
- dalšího TypeScript souboru, který obsahuje testy funkčností komponent (například about.component.spec.ts).

Soubor **core-routing.module.ts** je modul, který definuje routing – přechody mezi stránkami a také cesty, podle kterých jsou dostupné jednotlivé komponenty.

Složka **components** obsahuje dvě složky: **header** (hlavička) a **layout** (rozložení stránky). Header obsahuje navigační panel a jsou v něm definovány



Obrázek 5.2: Struktura webové aplikace

vztahy mezi tlačítky v hlavičce portálu a komponentami. Layout definuje, kde budou na stránce umístěny jednotlivé komponenty: okraje, výšku a šířku obsahu apod.

5.3.2 Volání backendu

Primárním účelem portálu je poskytování uživatelům informací o stanicích a technických kontrolách v roce 2018. Proto je jednou z nejdůležitějších částí implementace propojení klientu a backendu – vybudování způsobu pro jednoduché získávání dat a jejich zobrazování na webové stránce.

Za volání backendové části portálu z klientu jsou zodpovědné třídy `InpectionsAPIService` a `StationsAPIService`. Třídy jsou anotovány `@Injectable()`. Označení třídy touto anotací zajistí, že kompilátor vygeneruje metadata nezbytná k vytvoření závislostí třídy, pokud bude třída importována do komponenty. V konstruktoru těchto služeb je inicializována třída `HttpClient` z balíčku `@angular/common/http`, která potom uvnitř metod volá jednotlivé endpointy pomocí vlastní metody `get()`. Metoda vrací generický objekt `Observable` který je zodpovědný za zpracování asynchronních požadavků, a to s parametrem označujícím typ výsledku volání. Observable objekty jsou součástí knihovny `RXJS` [39].

V jednotlivých komponentách jsou následně vytvořeny instance služeb pomocí anotace `@Inject`. Při načítání stránky se potom spouští odebírání (příkaz `subscribe`) a data se uloží do lokální proměnné, která je zpřístupněna z HTML reprezentace komponenty.

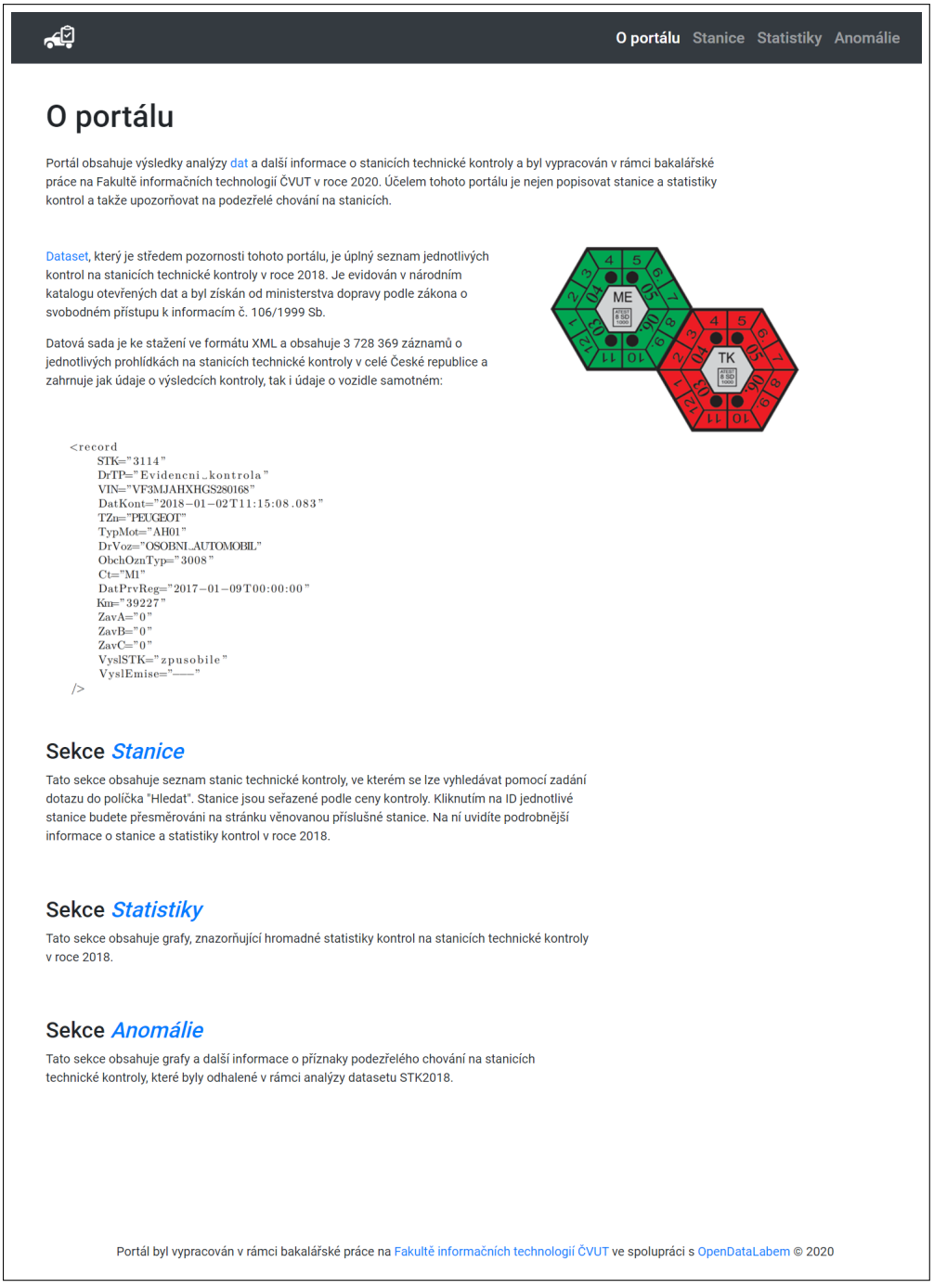
5.3.3 Komponenty

Podle návrhu, představeného v kapitole 4, bylo vytvořeno 6 komponent, které odpovídají různým obrazovkám portálu.

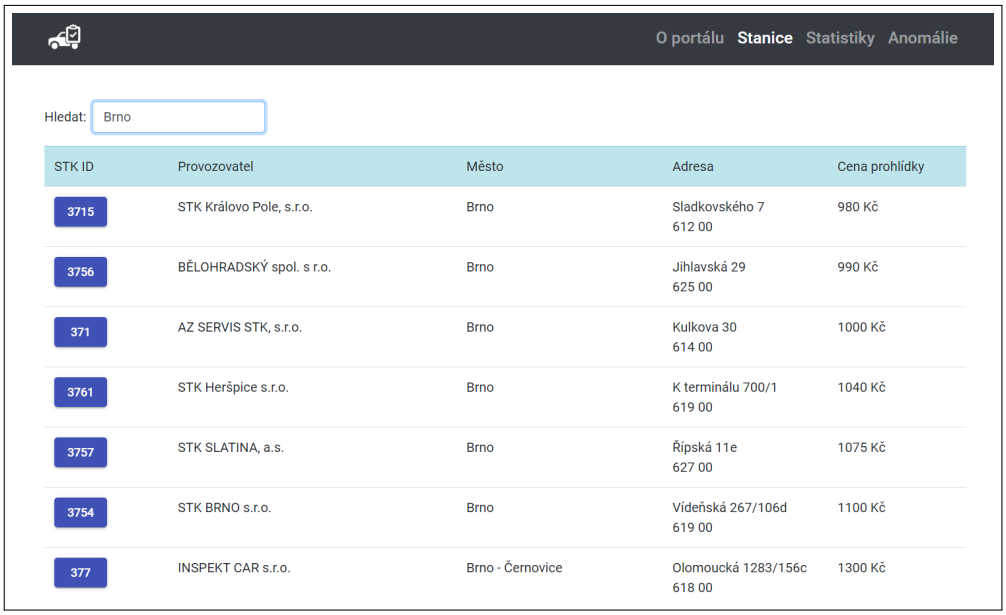
HomeComponent je komponenta reprezentující úvodní stránku portálu. V ideálním případě by měla obsahovat uvítací sekci a mapu České republiky s na ní označenými stanicemi technické kontroly. V průběhu implementace se ale bohužel nepodařilo doladit mapu tak, aby zobrazovala všechny stanice tak, jak bylo zamýšleno v návrhu. Proto bylo rozhodnuto odložit přidání této funkce do vydání další verze aplikace.

AboutComponent je komponenta reprezentující stránku obsahující popis portálu a jednotlivých sekcí. Tato obrazovka podobná domovské stránce obsahuje stručný popis portálu v celku a následně popisuje každou sekci, které jsou uvedeny v hlavičce, zvlášť a taktéž zahrnuje linky na tyto sekce. Je v podstatě návodem na používání portálu a doplňkovou navigací. Na obrázku 5.3 je vidět, jak vypadá výsledná obrazovka.

StationsComponent je komponenta reprezentující stránku obsahující seznam stanic technické kontroly v České republice. V podstatě jde o minimalistickou tabulku s 5 sloupci: číselný identifikátor stanice, při stisknutí kterého se



Obrázek 5.3: Výsledná obrazovka: AboutComponent



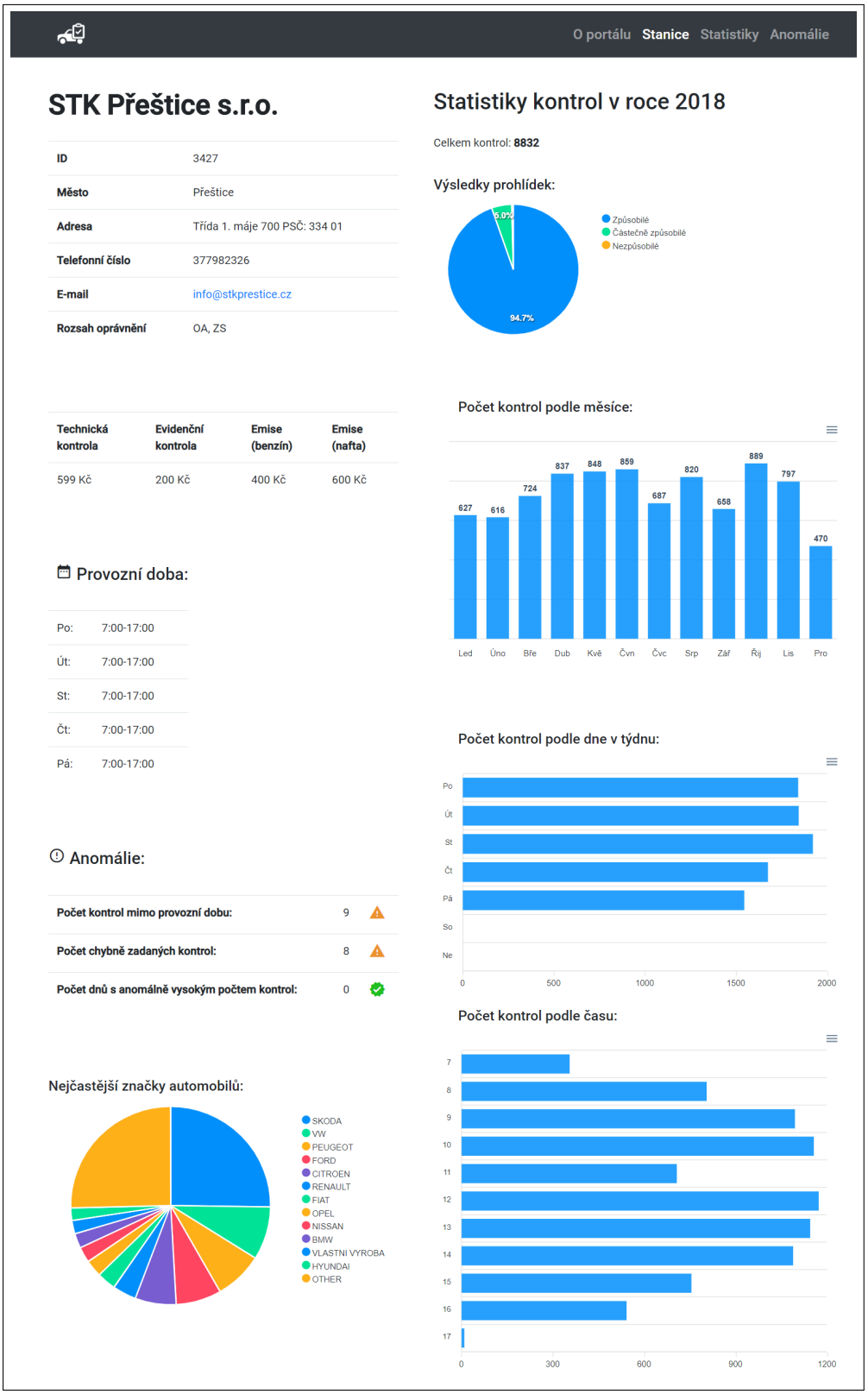
Obrázek 5.4: Výsledná obrazovka: StationsComponent

otevře stránka věnovaná příslušné stanici; provozovatel stanice; město; adresa a cena pravidelné prohlídky, podle které je seřazen seznam. Na obrázku 5.4 je vidět, jak vypadá výsledná obrazovka.

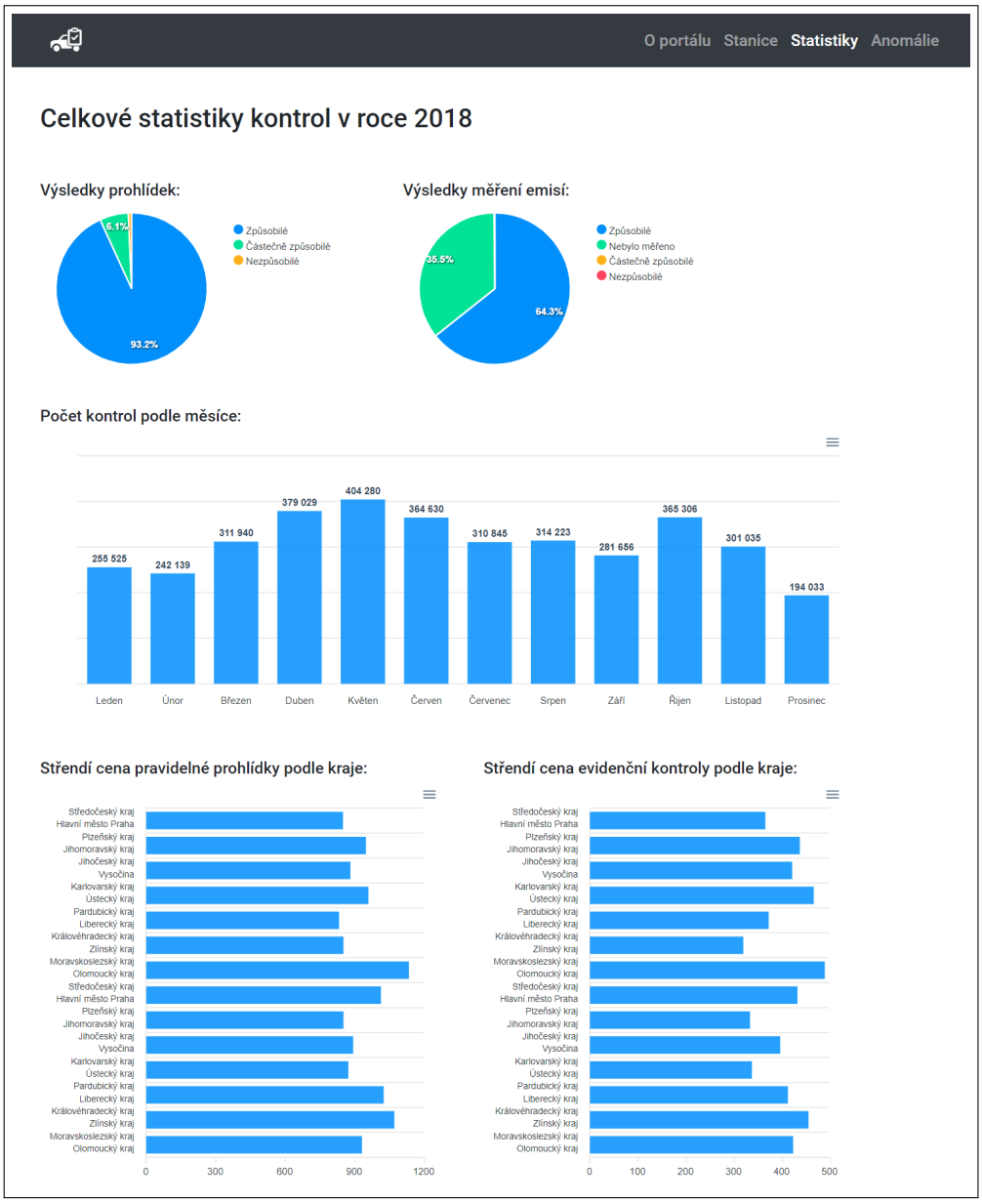
SingleStationComponent je komponenta reprezentující stránku obsahující popis jednotlivé stanice, která byla vybrána ze seznamu. Stránka je rozdělena na tři části. První část je popis stanice, který zahrnuje ID, město, adresu, telefonní číslo, e-mail, rozsah oprávnění, ceny prohlídek a měření emisí, pokud jsou k dispozici, a provozní doby, pokud je také k dispozici. Druhá část obsahuje statistiky kontrol v roce 2018, tzn. celkový počet, rozložení výsledků, počet kontrol podle měsíce, dne v týdnu a času a nejčastější značky kontrolovaných automobilů. Třetí část obsahuje zjištěná z dat anomálie. Na obrázku 5.5 je vidět, jak vypadá výsledná obrazovka.

StatisticsComponent je komponenta reprezentující stránku obsahující hromadné statistiky kontrol na STK v roce 2018. Na ní jsou zobrazeny podobné statistiky, jaké jsou umístěny na stránce věnováné jednotlivé stanici. Potom jsou zobrazeny ceny kontrol a úspěšnost na technických prohlídkách podle kraje. Na obrázku 5.6 je vidět, jak vypadá výsledná obrazovka.

SuspiciousBehaviourComponent je komponenta reprezentující stránku obsahující výsledky analýzy datasetu STK2018 na předmět podezřelého chování na stanicích. Obsahuje 4 sekce: kontroly mimo provozní dobu, anomální hustota prohlídek, chybně zadané záznamy a časové souběhy kontrol. Na obrázku 5.7 je vidět, jak vypadá výsledná obrazovka.

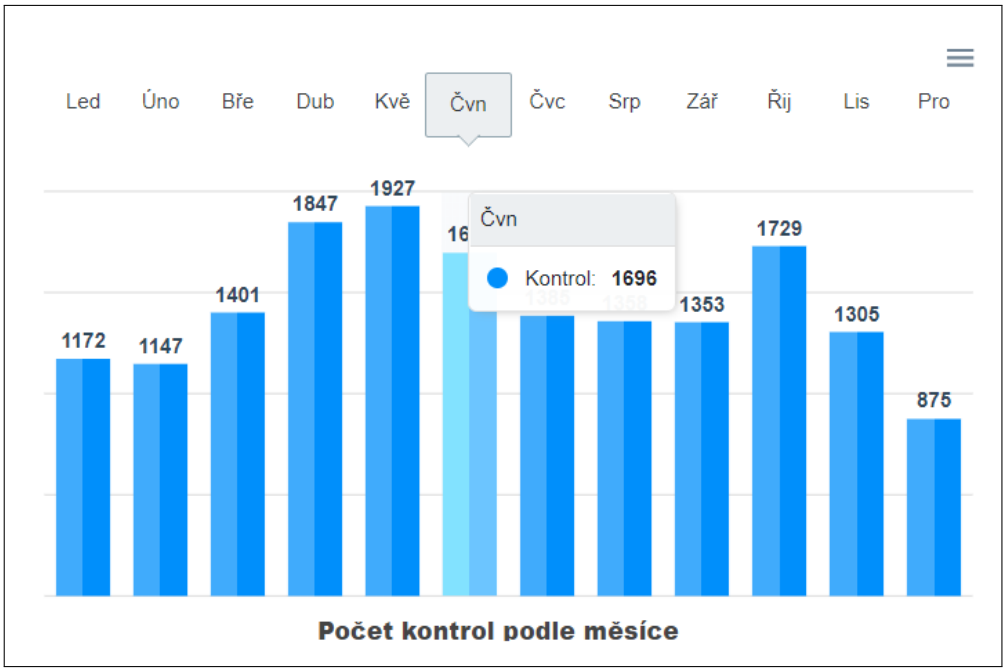


Obrázek 5.5: Výsledná obrazovka: SingleStationComponent



Obrázek 5.6: Výsledná obrazovka: StatisticsComponent

Obrázek 5.7: Výsledná obrazovka: SuspiciousBehaviourComponent



Obrázek 5.8: Příklad grafu, obsahujícího počet kontrol podle měsíce

5.3.4 Grafy

Důležitou části zobrazení statistik na portálu jsou grafy. Pro jejich vykreslení byla zvolena opensourcová knihovna `ApexCharts` [40]. Knihovna obsahuje rozsáhlé množství typů interaktivních grafů.

Grafy se inicializují v souborech TypeScript (rozšíření `.ts`), které obsahují logiku komponenty. Grafy jsou objekty třídy `ChartComponent` z knihovny `ApexCharts`. V konstruktoru komponenty, reprezentující stránku, jsou definovány parametry jednotlivých grafů: tvar vstupních dat, popisy os, barvy apod. Následně po obdržení odpovědi od backendu se uvnitř metody `subscribe()` aktualizuje parametr `data`. Na tuto událost reaguje HTML reprezentace stránky a kreslí graf.

Například pro vykreslení grafu, který se zobrazuje na stránce věnované jednotlivé stanici (viz obrázek 5.8)a která obsahuje přehled počtů kontrol podle měsíce, jsou prohlídky tříděny podle měsíce a počty jsou předány do parametru data. Potom jsou výsledné parametry předány do atributů tagu `apx-chart` a na stránce se zobrazí graf.

5.3.5 Testování

Součástí každé složky obsahující komponentu je soubor s rozšířením `spec.ts`, který obsahuje testy. Jde o soubory napsané v jazyce TypeScript a zahrnující

testy vytvoření komponenty, testování jejího chování a volání služeb, které jsou volané při načítání zkoumané stránky.

Před každým testem se pomocí utility `TestBed` uvnitř metody `beforeEach()` vytváří konfigurace testovacího modulu. Potom se v jednotlivých testech inicializuje komponenta a metodou `expect()` se testují jednotlivé části stránky a funkčnosti.

Testy se spouští příkazem `ng test`. Při běhu příkazu se hledají soubory s rozšířením `spec.ts` a postupně se spouštějí.

Závěr

Hlavními cíli této práce byla analýza otevřené datové sady STK2018, návrh metod detekce podezřelého chování na stanicích technické kontroly a vývoj webového portálu reprezentujícího výsledky analýzy a užitečné informace o stanicích.

V rámci této práce byly navrženy tři metody detekce podezřelého chování na stanicích technické kontroly na základě záznamů o prohlídkách.

První metodou je hledání kontrol mimo pracovní dobu stanice. Výsledky této analýzy poukázaly na přítomnost příznaků podezřelého chování na větší části (235 ze 247) zkoumaných stanic.

Další metodou je odhalení podezřelé krátkosti prohlídky na základě dohledaného počtu kontrolních linek. Výsledky této analýzy poukázaly na přítomnost příznaků podezřelého chování na 2 z 5 zkoumaných stanicích.

Poslední metodou je detekce podezřele častých souběhů značek automobilů na jednotlivých stanicích. Výsledky této analýzy poukázaly na přítomnost příznaků podezřelého chování na 46 ze 419 zkoumaných stanicích.

Výsledný webový portál se podle návrhu zaměřuje na reprezentaci statistik vytěžených z analyzovaného datasetu. Webová aplikace obsahuje všechny požadované funkce a má minimalistické a intuitivní uživatelské rozhraní. Pomocí scrapingu byly vytěženy informace o pracovních dobách stanic a cenách prohlídek, které jsou také zobrazené na portálu.

V budoucnosti by bylo možné analýzu rozšířit o porovnání výsledků kontrol mezi různými lety, bude ale potřeba zažádat o další data. S takovými daty lze navrhnout další metody detekce podezřelého chování, například anomální přírůstek počtu provedených kontrol beze změny počtu kontrolních linek. Výsledky porovnání by byly užitečným doplněním do portálu. Taktéž by bylo možné přidat do portálu další funkce, například mapu s vyznačenými stanicemi.

Literatura

[1] Han, J.; Kamber, M.; Pei, J.: *Data mining: concepts and techniques. 3rd ed.* Waltham: Elsevier Inc., 2011, ISBN 978-0-12-381479-1.

[2] Ester, M.; Kriegel, H.-P.; Sander, J.; aj.: A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining [online]. AAAI Press, 1996, str. 226–231. Dostupné z: https://doi.org/10.1023/A:1009745219419

[3] CENY A MAPA STK [online]. Mobildrive, 2020. [cit. 2020-05-09]. Dostupné z: https://www.mbenzin.cz/STK

[4] STK a emise [online]. Bartos015, 2018. [cit. 2020-05-09]. Dostupné z: https://www.stanice-technicke-kontroly.cz/

[5] SEZNAM-STK [online]. Agentura Kryštof s.r.o., 2012. [cit. 2020-05-09]. Dostupné z: http://www.seznam-stk.cz/

[6] ČESKO: Zákon č. 106 ze dne 11. května 1999 o svobodném cit.u k informacím. 1999. Dostupné z: https://aplikace.mvcr.cz/sbirka-zakonu/ViewFile.aspx?type=c&id=3256

[7] Portál otevřených dat [online]. Ministerstvo vnitra České republiky [cit. 2020-04-09]. Dostupné z: https://data.gov.cz/

[8] ČESKO: Vyhláška č. 211 ze dne 20. září 2018 o technických prohlídkách vozidel. 2018. Dostupné z: https://aplikace.mvcr.cz/sbirka-zakonu/ViewFile.aspx?type=c&id=38506

[9] ČESKO: Zákon č. 56 ze dne 10. ledna 2001 o podmínkách provozu vozidel na pozemních komunikacích a o změně zákona č. 168/1999 Sb., o pojištění odpovědnosti za škodu způsobenou provozem vozidla a o změně některých

souvisejících zákonů (zákon o pojištění odpovědnosti z provozu vozidla), ve znění zákona č. 307/1999 Sb. 2001. Dostupné z: https://aplikace.mvcr.cz/sbirka-zakonu/ViewFile.aspx?type=c&id=3599

[10] ČESKO: Zákon č. 121 ze dne 7. dubna 2000 o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon). 2000. Dostupné z: https://aplikace.mvcr.cz/sbirka-zakonu/ViewFile.aspx?type=c&id=3424

[11] STK - Seznam STK podle krajů [online]. Ministerstvo dopravy České republiky [cit. 2020-04-09]. Dostupné z: https://www.mdcr.cz/Dokumenty/Silnicni-doprava/STK/STK-Seznam-STK-dle-kraju?returl=/Dokumenty/Silnicni-doprava/STK

[12] Statistiky pro výpočty kapacit [online]. Ministerstvo dopravy České republiky [cit. 2020-04-09]. Dostupné z: https://www.mdcr.cz/Statistiky/Silnicni-doprava/STK/Statistiky-pro-vypocty-kapacit

[13] Anscombe, F. J.: Rejection of Outliers. In: Technometrics [online]. ročník 2, č. 2, 1960: s. 123–146. Dostupné z: https://doi.org/10.1080/00401706.1960.10489888

[14] Maldonado, P.; Vašata, D.: Redukce dimenzionality [přednáška]. Praha: FIT ČVUT v Praze, 20. listopadu 2019.

[15] Gove, R.: Using the elbow method to determine the optimal number of clusters for k-means clustering. In: Blocks [online]. 2017, [cit. 2020-05-01]. Dostupné z: https://bl.ocks.org/rpgove/0060ff3b656618e9136b

[16] Akoglu, H.: User's guide to correlation coefficients. In: Turkish journal of emergency medicine [online]. ročník 18, č. 3, 2018: s. 91–93. Dostupné z: https://doi.org/10.1016/j.tjem.2018.08.001

[17] Project Jupyter: The Jupyter Notebook 6.0.1 [software]. 2020, [cit. 2020-05-20]. Dostupné z: https://jupyter.org/

[18] The Pandas Development Team: Pandas 1.0.0 [software]. 2020, [cit. 2020-04-12]. Dostupné z: https://pandas.pydata.org/

[19] Python Software Foundation: The ElementTree XML API 3.3 [software]. 2020, [cit. 2020-04-12]. Dostupné z: https://docs.python.org/3/library/xml.etree.elementtree.html

[20] A Kenneth Reitz Project: Requests 2.23.0 [software]. 2020, [cit. 2020-04-12]. Dostupné z: https://requests.readthedocs.io/en/latest/

[21] Leonard Richardson: Beautiful Soup 4.9.1 [software]. 2020, [cit. 2020-04-12]. Dostupné z: https://www.crummy.com/software/BeautifulSoup/bs4/doc/

[22] Simon Brugman: Pandas Profiling 2.8.0 [software]. 2020, [cit. 2020-04-12]. Dostupné z: https://pandas-profiling.github.io/pandas-profiling/docs/master/rtd/index.html

[23] Oracle Corporation: Oracle Documentation [online]. 2020, [cit. 2020-05-09]. Dostupné z: https://docs.oracle.com/

[24] Pedregosa, F.; Varoquaux, G.; Gramfort, A.; aj.: Scikit-learn: Machine Learning in Python. In: Journal of Machine Learning Research [online]. ročník 12, 2011: s. 2825–2830, [cit. 2020-05-18]. Dostupné z: http://jmlr.org/papers/v12/pedregosa11a.html

[25] Kavyazin, D.: Principal Component Analysis and k-means Clustering to Visualize a High Dimensional Dataset. In: Medium [online]. A Medium Corporation, 2019. [cit. 2020-05-18]. Dostupné z: https://medium.com/@dmitriy.kavyazin/principal-component-analysis-and-k-means-clustering-to-visualize-a-high-dimensional-dataset-577b2a7a5fe2

[26] Cockburn, A.: *Writing effective use cases*. Boston: Addison-Wesley, 2000, ISBN 978-0201702255.

[27] Sparx Systems: Enterprise Architect 14 [software]. 2020, [cit. 2020-05-12]. Dostupné z: https://sparxsystems.com/

[28] Google: Angular 2.0 [software]. [cit. 2020-04-12]. Dostupné z: https://angular.io/

[29] Microsoft Corporation: TypeScript 3.5.3 [software]. [cit. 2020-04-12]. Dostupné z: https://www.typescriptlang.org/

[30] JetBrains: IntelliJ IDEA Ultimate 2019.3.3 [software]. [cit. 2020-04-12]. Dostupné z: https://www.jetbrains.com/idea/

[31] JetBrains: WebStorm 2020.1.1 [software]. [cit. 2020-04-12]. Dostupné z: https://www.jetbrains.com/webstorm/

[32] The PostgreSQL Global Development Group: PostgreSQL 12.3 [software]. 2020, [cit. 2020-04-12]. Dostupné z: https://www.postgresql.org/

[33] The pgAdmin Development Team: pgAdmin 4.20 [software]. 2020, [cit. 2020-04-12]. Dostupné z: https://www.pgadmin.org/

[34] The Psycopg Team: Psycopg – PostgreSQL database adapter for Python 2.8 [software]. 2019, [cit. 2020-04-12]. Dostupné z: https://www.psycopg.org/docs/

[35] Webb, P.; Syer, D.; Long, J.; aj.: Spring Boot Reference Documentation [online]. 2020, [cit. 2020-04-12]. Dostupné z: https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/

[36] The Apache Software Foundation: Apache Maven Project Guide [online]. 2020, [cit. 2020-04-12]. Dostupné z: https://maven.apache.org/guides/

[37] Nicoll, S.; Syer, D.; Bhave, M.: Spring Initializr 0.8.0 [software]. 2019, [cit. 2020-04-12]. Dostupné z: https://start.spring.io/

[38] Bechtold, S.; Brannen, S.; Link, J.; aj.: JUnit 5 User Guide [online]. [cit. 2020-04-12]. Dostupné z: https://junit.org/junit5/docs/current/user-guide/

[39] Lesh, B.; Driscoll, D.; Wortmann, J.-N.; aj.: RXJS 6.4.0 [software]. 2020, [cit. 2020-05-28]. Dostupné z: https://rxjs-dev.firebaseapp.com/

[40] ApexCharts: ApexCharts 3.19.0 [software]. 2020, [cit. 2020-05-28]. Dostupné z: https://apexcharts.com/

Příloha A

Seznam použitých zkratek

API Application Programming Interface

CSV Comma-Separated Values

DBSCAN Density-Based Spatial Clustering of Applications with Noise

HTTP HyperText Transfer Protocol

JDBC Java Database Connectivity

JPA Java Persistence API

NKOD Národní katalog otevřených dat

PCA Principal Component Analysis

REST Representational State Transfer

SCSS Sassy Cascading Style Sheets

SPA Single Page Application

SQL Structured Query Language

SSH Secure Shell

STK Stanice technické kontroly

UML Unified Modeling Language

URL Uniform Resource Locator

VIN Vehicle Identification Number

VPN Virtual Private Network

XML eXtensible Markup Language

Příloha B

Obsah přiloženého CD

- `readme.txt` ..... stručný popis obsahu CD
- `exe` ..... adresář se spustitelnou formou implementace
- `src`
  - `analysis` ..... zdrojové kódy analýzy
  - `impl` ..... zdrojové kódy implementace
  - `thesis` ..... zdrojová forma práce ve formátu LaTeX
- `text` ..... text práce
  - `thesis.pdf` ..... text práce ve formátu PDF
  - `thesis.ps` ..... text práce ve formátu PS