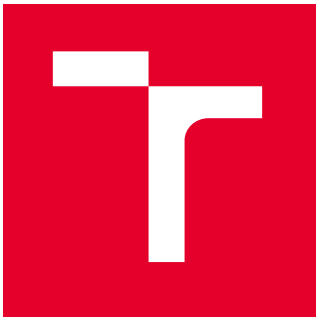


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

ANONYMIZACE DAT V UŽIVATELSKÉ APLIKACI

DATA ANONYMIZATION IN USER APPLICATION

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE — **Bc. Tadeáš Zachoval**

AUTHOR

VEDOUCÍ PRÁCE — **Ing. Jan Benedikt**

SUPERVISOR

BRNO 2024

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Diplomová práce

magisterský navazující studijní program **Informační bezpečnost**

Ústav telekomunikací

Student: Bc. Tadeáš Zachoval ***ID:*** 211820

Ročník: 2 ***Akademický rok:*** 2023/24

NÁZEV TÉMATU:

Anonymizace dat v uživatelské aplikaci

POKYNY PRO VYPRACOVÁNÍ:

Vytvořte teoretický model aplikace, který bude pracovat s uživatelskými daty (osobní údaje a geolokace osob). Proveďte analýzu v oblasti anonymizace a pseudoanonymizace. V rámci analýzy se zaměřte i na zasazení práce s osobními daty v právním rámci České republiky. Model následně rozšiřte o metody anonymizace a pseudoanonymizace uživatelských dat, tak aby bylo zaručeno, že v případě, kdy dojde k úniku dat z jakékoliv části modelu, nebude možné na základě těchto získaných dat identifikovat místo nebo osobu, které s daty souvisí. Cílem práce je uživatelská aplikace umožňující anonymizaci a pseudoanonymizaci uživatelských dat. Vstupem aplikace bude uživatelský dataset ve strojově čitelném formátu dat a výstupem budou anonymizovaná data. V případě pseudoanonymizace vytvořte v aplikaci funkcionalitu pro možnost zpětné konverze pseudoanonymizovaných dat do původní podoby při zachování původní úplnosti a integrity dat.

DOPORUČENÁ LITERATURA:

[1] RAGHUNATHAN, Balaji. The complete book of data anonymization: from planning to implementation. Boca Raton: CRC Press, c2013. ISBN 978-143-9877-302.

[2] ARBUCKLE, L. - EMAM, E.K. Building an anonymization pipeline: Creating Safe Data. . [s.l.]: O'Reilly, 2020. ISBN 978-149-2053-439.

Termín zadání: 5.2.2024 ***Termín odevzdání:*** 21.5.2024

Vedoucí práce: Ing. Jan Benedikt

doc. Ing. Jan Hajný, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Diplomová práce je zaměřena na de-identifikaci osobních údajů. Práce se skládá z teoretické a praktické části. Teoretická část popisuje právní pohled na osobní údaje a je především zaměřena na pojmy a možné de-identifikační metody a rozdíl mezi nimi. Stěžejní částí teoretické části je popis základních a některých pokročilých anonymizačních a pseudonymizačních technik. V praktické část se pak práce věnuje naprogramování aplikace v programovacích jazyce Python s grafickým uživatelským prostředím. Aplikace implementuje algoritmus na generování zkušebních falešných množin dat, které lze využít pro de-identifikaci. Hlavní funkcionalitou aplikace je však anonymizace a pseudonymizace osobních údajů. K tomuto účelu jsou v aplikaci uplatněny vybrané de-identifikační techniky.

KLÍČOVÁ SLOVA

Osobní údaje, lokalizační údaje, de-identifikace, anonymita, anonymizace, pseudonymizace, k-anonymita, diferenciální soukromí, Python

ABSTRACT

The master thesis is focused on de-identification of personal data. The thesis consists of theoretical and practical parts. The theoretical part describes the legal view of personal data and is mainly focused on the concepts and possible de-identification methods and the difference between them. The core of the theoretical part is a description of basic and some advanced anonymization and pseudonymization techniques. The practical part of the thesis then focuses on programming an application in the Python programming language with a graphical user interface. The application implements an algorithm to generate a testing fake dataset that can be used for de-identification. However, the main functionality of the application is the anonymization and pseudonymization of personal data. For this purpose, selected de-identification techniques are applied in the application.

KEYWORDS

Personal data, location data, de-identification, anonymity, anonymization, pseudonymization, k-anonymity, differencial privacy, Python

ZACHOVAL, Tadeáš. *Anonymizace dat v uživatelské aplikaci*. Diplomová práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2023. Vedoucí práce: Ing. Jan Benedikt

Prohlášení autora o původnosti díla

Jméno a příjmení autora:	Bc. Tadeáš Zachoval
VUT ID autora:	211820
Typ práce:	Diplomová práce
Akademický rok:	2023/24
Téma závěrečné práce:	Anonymizace dat v uživatelské aplikaci

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno ....

podpis autora*

*Autor podepisuje pouze v tištěné verzi.

PODĚKOVÁNÍ

Rád bych zde poděkoval vedoucímu diplomové práce panu Ing. Janu Benediktovi za odborné vedení, konzultace, trpělivost a podnětné návrhy při zpracování mé diplomové práce.

Obsah

Seznam obrázků

Seznam tabulek

Seznam výpisů

Úvod

Diplomová práce se věnuje oblasti, která úzce souvisí s osobními údaji, přesněji pak de-identifikací těchto údajů. De-identifikace je pojmem nadřazeným pro anonymizaci a pseudonymizaci osobních údajů. Důvod proč tato data de-identifikovat je jednoduchý, v dnešní době sbírá osobní údaje na internetu skoro každý, kdo poskytuje služby a pokud by tato sesbíraná data měla být jakýmkoliv způsobem publikována, musí být zajištěno aby nebylo narušeno soukromí osob, od kterých byla data shromážděna. V tu chvíli přichází na řadu de-identifikace (anonymizace nebo pseudonymizace) a díky těmto mechanizmům je následné zacházení s osobními údaji značně jednodušší. Právě z těchto důvodů se mnohdy veškerá sesbíraná data před uložením nebo dalším zacházením s nimi anonymizují, či pseudonymizují.

Práce si klade za cíl definovat pojmy, které se v oblasti de-identifikace a osobních údajů vyskytují a dále také provést rešerši základních technik, které se při de-identifikaci osobních údajů využívají. Dalším cílem práce je vytvořit návrh a samotnou aplikaci, ve které budou aplikovány vybrané techniky de-identifikace. Diplomovou práci tvoří tak dvě části, část teoretická a část praktická.

V teoretické části je práce zaměřena na definici pojmů, které jsou jak v praxi, tak i v této práci používány a jsou nutné k pochopení některých úseků práce. Část se věnuje samotné definici de-identifikace a hlavně vysvětlení rozdílu mezi anonymizací a pseudonymizací. Dále v této části práce lze nalézt popis několika různých technik anonymizace a pseudonymizace a to ať základních, tak i těch více sofistikovanějších. V neposlední řadě jsou zde věnovány řádky i samotné de-identifikaci lokalizačních údajů a strojově čitelným souborům, se kterými vyvinutá aplikace pracuje.

Praktická část věnuje svůj obsah tvorbě aplikace v programovacím jazyce Python s grafickým uživatelským prostředím. Vytvořená aplikace umožňuje tvorbu testovací množiny osobních údajů nebo načtení dříve takto vytvořeného datasetu. Aplikace se zaměřuje na některé vybrané techniky de-identifikace a implementuje je. Vybranými anonymizačními metodami jsou základní anonymizační techniky, k-anonymita a diferenciální soukromí. V aplikaci je také uplatněna pseudonymizace včetně zpětné de-pseudonymizace dat.

1 Teoretická část

V úvodu teoretické části se práce zabývá osobními údaji a nakládání s nimi jakožto s citlivými údaji a de-identifikaci důležitých dat. Práce nahlíží na toto téma jak z pohledu ukotvení v českých právních předpisech, tak i z pohledu technického. Mimo jiné jsou zde popsány pojmy, se kterými se v práci lze setkat a jsou důležité pro správné pochopení textu práce. V neposlední řadě jsou zde popsány různé techniky, jak lze aplikovat de-identifikační metody mezi které patří anonymizace, či pseudonymizace na soubor dat.

1.1 Právní rámec ochrany osobních údajů v České republice

V českém právním systému se ochrana osobních údajů upravuje jednak v Listině základních práv a svobod, a také samostatným zákonem o zpracování osobních údajů. Konkrétně článek 13 Listiny základních práv a svobod stanoví základní principy a zásady týkající se ochrany soukromí a osobních údajů. Článek 10 odstavec 3 této Listiny říká, že každý má právo na ochranu údajů o své osobě před neoprávněným shromažďováním, zveřejňováním, či jiným zneužíváním těchto údajů.[1]

Od dubna roku 2019 v České republice vyšel v platnost nový zákon (zákon č. 110/2019 Sb. o zpracování osobních údajů) zpracovávající příslušné předpisy Evropské unie. Tento právní předpis harmonizuje směrnici Evropské unie (směrnici 2016/680) a nařízení Evropské unie 2016/679, známé také jako GDPR (z angl. Genaral Data Protection Regulation, obecné nařízení o ochraně osobních údajů). Nově vzniklý zákon nahradil předchozí zákon č. 101/2000 Sb., o ochraně osobních údajů a o změně některých zákonů. Nový zákon tak přinesl přísnější podmínky pro práci s osobními údaji. Poskytuje subjektům údajů více práv a zároveň ukládá správcům a zpracovatelům, kteří s osobními údaji manipulují, přísnější povinnosti a vyšší sankce za jejich porušení.

1.1.1 Terminologie

Sekce terminologie obsahuje výběr vybraných pojmů z oblasti osobních údajů a zákona o zpracování osobních údajů, neboť nařízení GDPR platí pro každého, kdo zpracovává a shromažďuje osobní údaje subjektů žijící na území celé Evropské unie. Subjektem údajů je myšlena jakákoliv fyzická osoba, ke které jsou vztahovány osobní údaje.

Zpracování osobních údajů zahrnuje různé operace nebo soubory operací, které se provádějí s osobními údaji, buď pomocí automatizovaných postupů, nebo ručně.

Mezi tyto operace patří shromáždění, zaznamenání, uspořádání, strukturování, uložení, přizpůsobení, modifikace, vyhledávání, nahlížení, použití, sdílení, přenos, šíření, seřazení, kombinace, omezení, výmaz nebo zničení osobních údajů.

Správce určuje účel a způsoby zpracování osobních údajů. Správcem může být například právnická nebo fyzická osoba, orgán veřejné moci, agentura či jiný subjekt. Sám správce může být i zpracovatelem osobních údajů. V případě, kdy správce pověří zpracovatele správou osobních údajů, automaticky se tak nezbavuje odpovědnosti, nýbrž naopak tato odpovědnost i nadále zůstává na straně správce a nově i na straně zpracovatele. Z toho vyplývá, že v daný moment jsou za zpracovávání a dodržování právních předpisů zodpovědné obě strany.

Zpracovatelem osobních údajů, jak bylo uvedeno výše, může být stejná osoba, či organizace jako je správce osobních údajů, která má za úkol provádět zpracování osobních údajů pro správce. Pakliže se nejedná o tentýž subjekt, musí být mezi správcem a zpracovatelem uzavřena smlouva.

Článek 4 nařízení GDPR definuje osobní údaje následovně: "*veškeré informace o identifikované nebo identifikovatelné fyzické osobě (dále jen „subjekt údajů“); identifikovatelnou fyzickou osobou je fyzická osoba, kterou lze přímo či nepřímo identifikovat, zejména odkazem na určitý identifikátor, například jméno, identifikační číslo, lokační údaje, síťový identifikátor nebo na jeden či více zvláštních prvků fyzické, fyziologické, genetické, psychické, ekonomické, kulturní nebo společenské identity této fyzické osoby*". [2] Příkladem takového osobního údaje může být jméno a příjmení, adresa bydliště, IP adresa (unikátní číslo, jenž je přiděleno libovolnému zařízení komunikujícímu prostřednictvím internetového protokolu) apod.

České právní předpisy pracují i s pojmem citlivé údaje, těmi jsou myšleny veškeré osobní údaje vypovídající např. o rasovém nebo etnickém původu, politických názorech, náboženství, genetický údaj, biometrický údaj zpracovávaný za účelem jedinečné identifikace fyzické osoby, údaj o zdravotním stavu, o sexuálním chování, o sexuální orientaci, či údaj týkající se rozsudků v trestních věcech a trestných činů. Celou definici citlivých údajů lze nalézt v §66 odstavec 6 zákona č. 110/2019 Sb., zákon o zpracování osobních údajů. [3]

Pseudonymizací se rozumí proces, kdy zpracovávané osobní údaje již nelze připisovat konkrétní osobě, či subjektu bez použití dalších informací. [4] Stejně jako s osobními údaji, i s pseudonymizací pracuje článek 4 nařízení EU 2016/679, který pseudonymizaci definuje jako: "*zpracování osobních údajů tak, že již nemohou být přiřazeny konkrétnímu subjektu údajů bez použití dodatečných informací, pokud jsou tyto dodatečné informace uchovávány odděleně a vztahují se na ně technická a organizační opatření, aby bylo zajištěno, že nebudou přiřazeny identifikované či identifikovatelné fyzické osobě*". [2]

1.2 De-identifikace osobních údajů

V následující části bude detailněji rozebráno téma de-identifikace dat. Zjednodušeně řečeno de-identifikace dat slouží k odstranění nebo jiné úpravě sesbíraných dat tak, aby byla znemožněna re-identifikace (možnost zpětně identifikovat subjekt údajů) a zároveň pro upravený dataset by mělo platit zachování užitečnosti dat pro další práce (např. analýzu, výzkum apod.). Pod de-identifikaci dat lze zahrnout pojmy jako anonymizace, pseudonymizace, identifikátor, či kvazi-identifikátor.

Pod pojmem anonymizace si lze představit proces, kdy jsou osobní údaje, s nimiž se nakládá, zpracovány a po tomto způsobu zpracování je znemožněna zpětná identifikace nebo odvození totožnosti konkrétní osoby. [4] Poté, co jsou osobní údaje anonymizovány, nepodléhají nadále obecnému nařízení o ochraně osobních údajů.

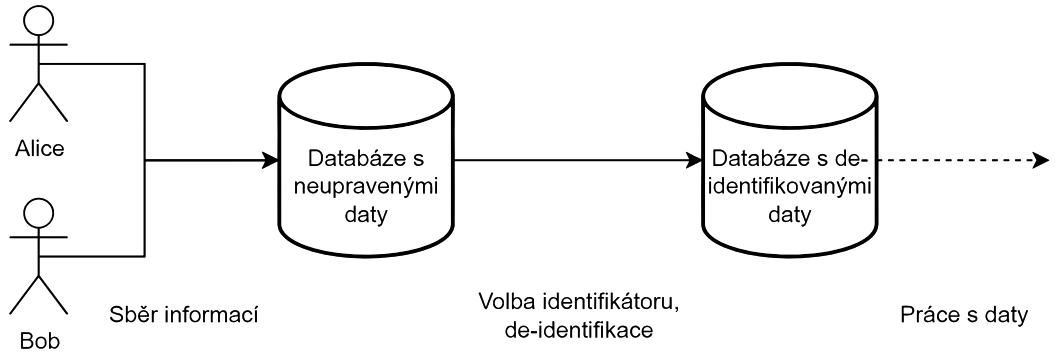
Při práci s osobními údaji se také často vyskytuje pojem identifikátor, či kvazi-identifikátor. Jedná se o atributy sesbíraného datasetu, které mohou přímo, či nepřímo identifikovat subjekt údajů. Rozdíl mezi jednotlivými typy identifikátory bude podrobněji rozebrán v práci.

Mnohdy jsou některé pojmy v různých publikací navzájem zaměňovány. Práce tak definuje de-identikaci jako obecný proces úpravy dat, díky kterému je zamezeno asociaci mezi sesbíranými daty a subjekty údajů, anonymizaci a pseudonymizaci jako metody de-identifikace.

Za poslední dobu se oblast zpracování osobních údajů a věcí spojené s tímto zpracováním rozvinula natolik, že v dnešní době snad neexistuje organizace, která by tyto údaje neshromažďovala. A proto je na každého zpracovatele kladen důraz, aby při zpracovávání osobních údajů a případném využívání těchto dat bylo co nejvíce zajištěno ochrany soukromí subjektů poskytujících tyto osobní údaje. De-identifikace má za cíl datasety obsahující osobní údaje upravit tak, že umožňuje odstranit údaje citlivé na ochranu soukromí, které identifikují jednotlivce, zatímco jiné užitečné informace zůstávají zachovány. Příkladem, kdy se taková de-identifikace využívá v praxi je například při zveřejnění smluv, které je podmíněno zákonem, nebo například při zveřejnění soudních rozhodnutí. V obou případech je nutné dokument upravit tak, aby splňoval podmínky jednak zachování věcného jádra dokumentu, tak i ochranu soukromí vystupujících stran. De-identifikace je tedy důležitým nástrojem, který mohou organizace použít k minimalizaci rizika ohrožení soukromí spojené s vytvářením, používáním, archivací, sdílením a zveřejňováním dat obsahujících osobní údaje. Odstranění identifikátorů v době jejich shromažďování může snížit náklady spojené s používáním a archivací dat, protože snižuje riziko ztráty soukromí spojené s neúmyslným zveřejněním. De-identifikace tak zpracovatelům osobních údajů umožňuje využívat sesbíraná data ve větší míře, než by jinak bylo možné. [5]

Postup de-identifikace dat zobrazuje obrázek 1.1. Kde na začátku procesu je

nutné sesbírat data od subjektů, která budou následně de-identifikována. Poté je nutné ze sesbíraných dat určit, u kterých atributů se jedná o jaký identifikátor. Po přidělení identifikátorů k atributům je nutné zvolit odpovídající de-identifikační metodu a provést ji. Po úspěšné de-identifikaci jsou data připravena k následnému zpracování, ke kterému byla tyto data sesbírána.



Obr. 1.1: Proces de-identifikace

Sběr dat

Na začátku celého procesu je zprvu nutné nejdříve sesbírat data. Během sběru dat se získávají a interpretují informace nebo údaje z více různých zdrojů (v tomto případě osob) s cílem řešit výzkumné otázky, analyzovat výsledky a provádět predikce ohledně budoucích trendů a pravděpodobných událostí. Tento krok představuje základní pilíř ve všech oblastech výzkumu, analýzy a procesů rozhodování, ať už jde o výzkum v oblasti sociálních věd, podnikání nebo zdravotnictví. [6] Typickými situacemi, při kterých lze sbírat data mohou být například ankety, dotazníky nebo i sběry dat od zákazníků na internetových obchodech.

Volba identifikátorů

Jak již bylo zmíněno výše, na základě identifikátorů lze určit, o který subjekt ze sesbíraných dat se jedná. Tudíž je nutné ze shromážděných dat určit, o jaký typ identifikátoru se u každého atributu jedná.

Identifikátory lze dělit na přímé a nepřímé identifikátory. Často je nepřímý identifikátor zaměňován právě za kvazi-identifikátor. Kvazi-identifikátor je obvykle složen z několika atributů, které samy o sobě nemusí jednotlivce jednoznačně identifikovat, ale ve spojení s jinými atributy nebo vnějšími zdroji informací mohou vést k odkrytí identity jedince. U nepřímého identifikátoru se jedná o jeden daný atribut, který

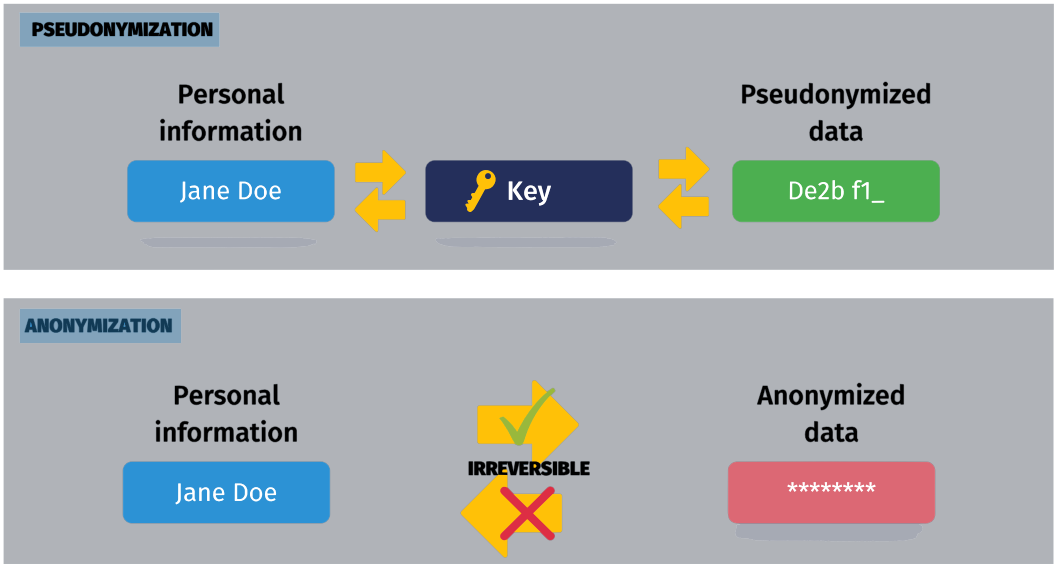
nedokáže jednoznačně určit jednotlivce. Za nepřímý identifikátor lze považovat například PSČ, místo narození, váhu subjektu apod., neboť u této hodnoty je pravděpodobné, že ji bude sdílet větší množství osob. Zatímco u kvazi-identifikátoru, jenž je kombinací těchto atributů (PSČ, datum narození, váha), je už velice pravděpodobné, že odkrytí totožnosti jednice nebude natolik obtížné. Všechny hodnoty, které nejsou přímými identifikátory, jsou nepřímé identifikátory, ale ne všechny nepřímé identifikátory jsou kvazi-identifikátory. U přímého identifikátoru se jedná o atribut, který jednoznačně ztotožňuje osobu (např. rodné číslo).

Přímé identifikátory by měly být v datasetu odstraněny nebo nějakým způsobem upraveny, kvazi-identifikátory by měly být v ideálním případě de-identifikovány způsobem, který zachová alespoň část jejich výpovědní hodnoty pro analýzu dat. [8]

Volba de-identifikační metody

Jakmile jsou data sesbírána a u jednotlivých atributů určeny identifikátory je nutné zvolit metodu de-identifikace. Volba metody závisí mimo jiné také na budoucím využití de-identifikovaných dat. Každá z metod se chová k datasetu různě, proto je nutné volit vhodné metody. Rozlišujeme anonymizaci a pseudonymizaci. V jádru věci jsou obě metody stejné, zakládají si za cíl upravit data, tak aby byla znemožněna re-identifikace.

Zásadním rozdílem mezi metodami je jejich reverzibilita, jak lze vidět na obrázku 1.2.



Obr. 1.2: Rozdíl mezi anonymizací a pseudonymizací [7]

Anonymizace je jednosměrným procesem (jako např. funkce hash, která generuje z vstupních dat řetězec symbolů o konstantní délce). Oproti tomu pseudonymizace

je metoda, kde se vytvoří klíč, který následně pozmění atributy datasetu na nic neříkající data. Zásadní věcí u pseudonymizace je zachovávat zmíněný klíč mimo dosah upraveného datasetu, a zvyšit tak bezpečnost provedené de-identifikace.

De-identifikovaná data

Data, která byla anonymizována, přestávají být osobními údaji a nejsou tak regulována. Z tohoto důvodu je mohou organizace používat k jiným účelům, než pro které byly původně získány. [9] Data, která byla anonymizována nebo pseudonymizována mohou být, s dostatečnou ostražitostí, publikována. Publikovanými daty tak mohou být například veřejné statistiky, výsledky výzkumů, vládní informace apod.

Dle [10] musí de-identifikovaná data splňovat následující tři podmínky, aby bylo možné považovat tyto data za neosobní:

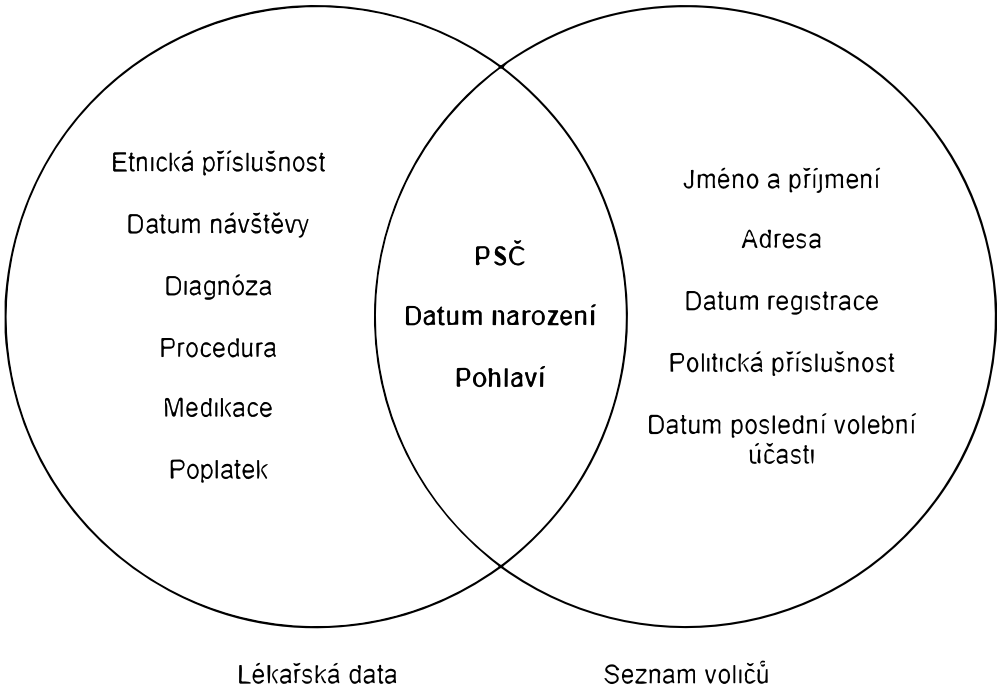
- vyčlenění (singling-out): Jedná se o schopnost identifikovat a oddělit některé nebo všechny záznamy v datovém souboru, které jednoznačně identifikují konkrétního jednotlivce.
- propojitelnost (linkability): Pojem odkazuje na schopnost propojit alespoň dva záznamy v databázi, které se týkají stejné osoby nebo skupiny osob. Propojení může nastat buď v rámci jedné databáze nebo mezi různými databázemi.
- odvození (inference): Znamená schopnost odvodit hodnotu určitého atributu z hodnoty jiných atributů s vysokou pravděpodobností.

Re-identifikace

Re-identifikace je proces, při kterém je snaha rozpoznat subjekty z de-identifikovaných údajů. Protože hlavním cílem de-identifikace je zabránit neoprávněné opětovné identifikaci, nazývají se někdy takové pokusy útoky na opětovnou identifikaci. "Faktická anonymita", někdy označovaná jako "praktická anonymita", říká, že pokud by opětovná identifikace jednotlivců souboru dat vyžadovala nadměrné množství času, odborných znalostí, lidské práce a nákladů, pak lze soubor považovat za "fakticky anonymní". [11] Z praktické anonymity je zřejmé, že při studiu útoků na re-identifikaci by se měly brát v úvahu odborné znalosti, které útočník potřebuje, a také informace, které má k dispozici nebo náklady na tento typ dat, která by mohla být potřebná k propojení s něčí identitou. [12] Je nutné brát v úvahu, že de-identifikační proces není vždy stoprocentní. Pokaždé existuje nějaké riziko re-identifikace. Riziko opětovné identifikace se obvykle uvádí jako procento záznamů v souboru dat, které lze opětovně identifikovat. [13]

Existuje mnoho vědeckých prací na toto téma, často zmiňovanou z nich je práce profesorky Latanya Sweeney, Ph.D, kde v práci *Simple Demographics Often Identify People Uniquely* dospěla k závěru, že 87 % (v té době 216 milionů z 248 milionů)

americké populace lze identifikovat na základě poštovního směrovacího čísla, pohlaví a data narození. A k identifikaci 53 % (tzn. 132 milionů z 248 milionů) jedinců postačí pohlaví, datum narození a místo jejich bydliště. [14] K tomu výsledku jí postačili dvě množiny dat (seznam voličů a lékařská data), jak lze vidět na obrázku 1.3.



Obr. 1.3: Dvě množiny dat, jenž vedly k re-identifikaci subjektů(vytvořen na základě [14])

1.3 Základní de-identifikační techniky

1.3.1 Anonymizační techniky

Níže je blíže přiblížena metoda anonymizace s výčtem anonymizačních technik. Jak již bylo uvedeno dříve, anonymizace je jednosměrným procesem úpravy dat (především pak odstranění identifikátorů), tak aby bylo zamezeno zpětnému identifikování subjektů v datasetu. Existuje několik technik, kterými lze anonymizace dosáhnout. Pro správné zvolení anonymizační techniky je nejdříve nutné pochopit, k čemu mají být výsledná anonymizovaná data použita. Neboť každá anonymizační technika má svoje vlastnosti a tudíž platí, že pro daný účel se nemusí hodit každá technika. [4] V neposlední řadě je také dobré myslet na základní myšlenku de-identifikace, a to aby data po procesu byla použitelná pro další použití a především aby nebylo ohroženo soukromí daných subjektů údajů.

Suprese dat

Pod pojmem suprese se skrývá jednoduchá operace odstranění. K odstranění může dojít buď v rámci jednoho atributu nebo lze odstranit celý záznam daného subjektu (v příkladu uvedeném níže by se jednalo o odstranění jednoho řádku). Tato technika se používá v případě, kdy daný atribut není relevantní pro další zpracování. [15]

Příkladem mohou být záznamy s jmény studentů, učitelů a výsledky testů (viz tab. 1.1 a tab. 1.2). Pro další zpracování je důležité pouze to, jakou úspěšnost v procentech měli studenti daných učitelů (není důležité, kdo ze studentů měl jaký výsledek). Tudíž jména studentů jsou v konečném zpracování irelevantní a lze je odstranit. Jedná se o nejspolehlivější techniku anonymizace, neboť bez odstraněných identifikátorů je nemožné přiřadit studenta k danému výsledku. V případě, kdy by učitel měl pouze jednoho žáka, je nejlepším řešením odstranění celého záznamu.

Tab. 1.1: Data před anonymizací technikou suprese

Jméno studenta	**Jméno učitele**	**Výsledek testu** [%]
František M.	Zelený A.	97
Martina S.	Marková D.	85
Ivan K.	Marková D.	39
Věra O.	Zelený A.	66
Lukáš F.	Zelený A.	53

Tab. 1.2: Data po anonymizaci technikou suprese

Jméno učitele	**Výsledek testu** [%]
Zelený A.	97
Marková D.	85
Marková D.	39
Zelený A.	66
Zelený A.	53

Maskování dat

Maskování dat je proces, kde je snahou skrýt důležité, citlivé nebo osobní informace obsažené v souboru dat tak, že jsou nahrazeny náhodnými znaky, nepravdivými údaji nebo fiktivními informacemi. Tím je vytvořena verze dat, která není pravdivá, ale zároveň zachovává strukturu původního souboru. [16]

Jako příklad lze uvést případ, kdy by si nadnárodní společnost ukládala jména zákaznických firem a jejich telefonní čísla. A pro budoucí zpracování řeší pouze to,

odkud daná firma pochází, tudíž pro zpracování postačí jméno společnosti a telefonní předvolba telefonního čísla. Zbytek tel. čísla se zamaskuje například znakem "X". Výsledek lze vidět na tabulce 1.3.

Tab. 1.3: Data před a po anonymizaci technikou maskování dat

Název společnosti	**Tel. číslo společnosti před maskováním**	**Tel. číslo společnosti po maskováním**
GreenTech Inc.	+1 (555) 658-6544	+1 (XXX) XXX-XXXX
Eco-Wave Ltd.	+44 9514 658 4233	+44 XXXX XXX XXXX
Zelená řešení a.s.	+420 966 345 333	+420 XXX XXX XXX

Záměna dat

Neboli data swapping, neznamená nic jiného než prohození jednotlivých atributů jednotlivých záznamů. Dalšími jmény pod kterými jej lze nacházet jsou shuffling nebo permutace dat. [17]

Jednoduchý příklad záměny dat lze vidět v tabulce 1.4. Výsledná data vypadají stejně jako data před záměnou, avšak podoba po provedení záměny dat neodhaluje žádnou reálnou osobu.

Tab. 1.4: Data před (vlevo) a po (vpravo) anonymizaci technikou data swapping

Osoba	**Pozice**	**Plat [v tis. Kč]**
1	Programátor	81
2	Kadeřnice	41
3	Prodavač	33
4	Učitelka	50
5	Údržbář	32

Osoba	**Pozice**	**Plat [v tis. Kč]**
1	Údržbář	41
2	Učitelka	33
3	Kadeřnice	50
4	Programátor	32
5	Prodavač	81

Generalizace dat

V případě generalizace se jedná o zobecnění záznamů. Míra zobecnění je nastavena u každé skupiny identifikátorů zvlášť. Se zvyšující úrovní generalizace (jak je hodně daný záznam zobecněn) úměrně klesá užitečnost výsledných dat. Zároveň však se zvyšující generalizací se snižuje riziko zpětné identifikace. Technika zobecnění se často používá v souvislosti s věkem nebo ji lze použít pro generalizaci poštovního směrovacího čísla.

Tab. 1.5: Data před (vlevo) a po (vpravo) anonymizaci technikou generalizace

Osoba	Věk	Váha [kg]	Výška [cm]
1	38	81	177
2	22	51	162
3	67	63	156
4	55	90	188
5	29	76	176

Osoba	Věk	Váha [kg]	Výška [cm]
1	30-40	80	175
2	20-30	50	160
3	60-70	60	155
4	50-60	90	190
5	20-30	80	175

Narušení dat

Jedná se o anonymizační techniku, která je založena na přidání náhodného šumu. Tuto metodu lze použít pro číselné záznamy tak, že mění jednotlivé identifikátory s určitou hodnotou a operací. [18] Je nutné zvolit přiměřenou velikost šumu, aby byl splněn cíl anonymizace a zároveň zůstala zachována vypovídající hodnota upravených dat.

Například k platům zaměstnanců byla přidána malá náhodná hodnota v rozmezí +/- 5000 Kč, což činí obtížnější re-identifikaci na základě jejich přesné mzdy. Avšak výsledný průměrný plat zaměstnanců touto modifikací dat nebyl pozměněn.

Tab. 1.6: Data před (vlevo) a po (vpravo) anonymizaci technikou data perturbation

Zaměstnanec	Plat [v Kč]
1	83 352
2	43 243
3	31 542

Zaměstnanec	Plat [v Kč]
1	78 769
2	45 148
3	34 220

1.3.2 Pseudonymizační techniky

Dle obecného nařízení o ochraně osobních údajů by měly být osobní údaje pseudonymizovány takovým způsobem, aby je bylo možné spojit, nebo jinak přiřadit, k určitému subjektu a to jen a pouze za pomoci použití dodatečných dat (např. šifrovacího klíče), jenž by měli být ukládány separátně od pseudonymizovaných údajů a měli by být řádně uloženy a chráněny. [19]

Následující techniky pseudonymizace jsou přebrány z dokumentu popisující pseudonymizaci od Agentury Evropské unie pro kybernetickou bezpečnost (ENISA). [20]

Čítač

Přestavuje jednoduchou pseudonymizační techniku, jenž nahrazuje identifikátor monotónním čítačem. Na začátku se nastaví čítač na počáteční hodnotu (nemusí být striktně 0), ta se s každým inkrementovaným identifikátorem zvyšuje. Technika čítače je vhodná především pro menší datasety. [21] Pseudonymizační klíč není nutné uchovávat, ale v tom případě je nutné aby byla původní tabulka před provedením pseudonymizace zachována v nezměněném stavu.

Generátor náhodných čísel

Jedná se o prakticky stejnou techniku jako u předchozího čítače s tím rozdílem, že místo hodnoty, která se vždy inkrementuje o předem nastavenou hodnotu se hodnota volí pokaždé nová pomocí generátoru náhodných čísel. Za nevýhodu této techniky lze považovat uchovávání klíče, neboť se musí uchovávat celá tabulka (viz tabulka 1.7) a pokud není ošetřeno, je možné že k jednomu pseudonymu bude přiřazen více než jeden záznam.

Tab. 1.7: Data před (vlevo) a po (uprostřed) pseudonymizací pomocí generátoru náhodných čísel, pseudonymizační klíč (vpravo)

Jméno	**Prac. pozice**
Martin K.	Programátor
Bob P.	Servisní technik
Alice D.	Profesorka

Jméno	**Prac. pozice**
06532	Programátor
3241665	Servisní technik
965	Profesorka

Čítač	**Jméno**
06532	Martin K.
3241665	Bob P.
965	Alice D.

Funkce hash

Funkce hash je postavená na matematické funkci, která vytváří z vstupního řetezce znaků hodnotu o pevně dané délce. Podobně jako anonymizace, jedná se o jednosměrný úkon (z výstupu tedy nelze zpětně zjistit vstupní řetězec). Hash funkce by také měla splňovat podmínku kolize, tzn. že není možné, aby dva různé vstupní řetězce měli stejný výstupní řetězec.

Autentizační kód zprávy (MAC)

Pomocí MAC (Message Authentication Code, autentizační kód zprávy) lze zaručit autentizaci i integritu zprávy, jedná se o kontrolní součet, který se následně přidá k posílané zprávě. [22] Nejznámějším verzí je HMAC (Hash-based message authentication code, autentizační kód zprávy s využitím hash funkce). Jak již název napovídá, technika využívá také funkci hash, akorát s tím rozdílem, že je k této funkci přidán klíč, bez kterého nelze vytvořit pseudonym.

Šifrování

Šifrování využívá k zabezpečení dat kryptografii. Proces vytváří z vstupních dat šifrovaná data, při kterém jsou do procesu začleněny kryptografické algoritmy. Rozlišujeme symetrické a asymetrické šifrování.

Symetrické šifrování využívá pro šifrování i dešifrování dat proudové nebo blokové šifry. Bloková šifra rozdělí zprávu (data) na celé jednotky nebo bloky otevřeného textu o velikosti, kterou předem definuje délka klíče (např. 128, 192 nebo 256 bitů). [23] U proudových šifer je využíván každý bit (případně bajt) přenášených dat. Pro šifrování i pro dešifrování se využívá totožného klíče. Výhodou tohoto typu šifrování je rychlost provádění kryptografických operací (šifrování a dešifrování). Nevýhoda tohoto algoritmu spočívá ve sdíleném klíči pro šifrování a dešifrování. Zástupci symetrických šifer jsou například AES (Advanced Encryption Standard), 3DES (Triple Data Encryption Standard), DES (Data Encryption Standard).

Zatímco u symetrického šifrování je využíván jeden klíč pro obě operace u asymetrického šifrování je využita dvojice klíčů (veřejný a soukromý), každý jeden pro jiný typ operace. Zašifrování dat probíhá pomocí veřejného klíče adresáta. Poté, co adresát obdrží takto zašifrovaná data, musí pro dešifrování těchto dat použít adresát svůj soukromý klíč (ke kterému má přístup jen a pouze sám adresát). To zajišťuje, že pouze adresát, může dešifrovat a získat tak původní data. Ostatní mohou pouze zašifrovaná data vidět, ale nemají přístup k jejich obsahu bez soukromého klíče adresáta. Díky tomu může kdokoliv poslat adresátovi zašifrovanou zprávu, což je veliká výhoda oproti symetrickému šifrování, které musí řešit bezpečnou distribuci symetrického klíče. Mezi známé asymetrické algoritmy řadíme např. RSA (Rivest Shamir Adleman), ECC (Elliptic-Curve Cryptography, kryptografie nad eliptickými křivkami).

1.4 Pokročilé de-identifikační techniky

1.4.1 K-Anonymity

Koncept anonymizační techniky k-anonymity byl představen v roce 1998, a stojí na myšlence, že spojením sad dat s podobnými atributy lze zastřít identifikační informace o jakémkoli subjektu, který se na těchto datech podílí. Písmeno "k"ve slově k-anonymity označuje proměnnou, která představuje počet výskytů jednotlivých kombinací hodnot v souboru dat. Pokud je k=2, hovoříme o 2-anonymitě. Z toho vyplývá, že data byla zobecněna natolik, že v datovém souboru existují alespoň dvě sady každé kombinace dat.

K-anonymity pracuje tak, že se podobné osoby seskupí a datová pole, která obsahují identifikační informace, se zobecní nebo potlačí. Příkladem může být soubor dat, který obsahuje atributy věku, pohlaví, poštovního směrovacího čísla a nemocí, kterou subjekt trpí. Aby byla data K-anonymní s hodnotou k=3, je nutné zajistit, aby pro každou kombinaci kvazi-identifikátorů existovaly alespoň tři subjekty, které mají stejnou hodnotou. To by vyžadovalo zobecnění nebo potlačení některých informací, například nahrazení přesného věku věkovým rozmezím nebo maskováním poštovního směrovacího čísla (viz tabulka 1.8).

Tab. 1.8: Data před (vlevo) a po provedení (vpravo) 4-anonymity

Věk	**PSČ**	**Nemoc**
65	532 04	Rakovina
33	518 21	Cukrovka
29	277 11	Srdeční nemoc
27	277 13	Srdeční nemoc
56	512 35	Rakovina
39	512 45	Cukrovka
35	538 33	Srdeční nemoc
25	277 05	Žloutenka
21	277 23	Cukrovka
51	538 63	Rakovina
34	538 34	Srdeční nemoc
51	512 33	Rakovina

Věk	**PSČ**	**Nemoc**
<30	27* **	Žloutenku
<30	27* **	Cukrovka
<30	27* **	Srdeční nemoc
<30	27* **	Srdeční nemoc
30-40	5** **	Cukrovka
30-40	5** **	Cukrovka
30-40	5** **	Srdeční nemoc
30-40	5** **	Srdeční nemoc
>50	5** **	Rakovina
>50	5** **	Rakovina
>50	5** **	Rakovina
>50	5** **	Rakovina

V tuto chvíli, při pohledu na tabulku 1.8 lze vidět, že v případě, kdy bude znát útočník údaj (např. jeho cílem je osoba s věkem 50+ let, o které ví že bydlí na adrese s PSČ 512 33) o potencionálním subjektu a ví, že vystupuje v této tabulce, bude vědět, že jeho cíl má rakovinu. Tuto skutečnost, nedostatek, k-anonymity řeší technika l-diversity.

Algoritmy k-anonymity

Část algoritmy k-anonymity se zaměřuje na různé algoritmy navržené k transformaci dat tak, aby byla dosažena požadovaná úroveň k-anonymity. V následujících odstavcích jsou představeny některé existující algoritmy. V závislosti na účinnosti použitého algoritmu a vstupní datové sady se může kvalita vytvořených tříd ekvivalence lišit. Pokud datová sada obsahuje odlehlé hodnoty, je možné že i vytvořené skupiny budou mít relativně velký rozptyl hodnot nebo budou mít vytvořené třídy počet prvků výrazně vyšší než "k".[24] Mezi algoritmy k-anonymity lze zařadit např.

Mondrian, Datafly, Incognito apod. První jmenovaný lze zařadit do algoritmu shoradolů (začíná se s jednou skupinou, obsahující celý dataset a algoritmus dělí tuto skupinu s postupem dále). Zbylé jmenované algoritmy spadají do skupiny algoritmů zdola-nahoru (od menších, elementárních, prvků se utváří skupiny).

Datafly vytvoří seznam četností seskupením záznamů se stejnými hodnotami atributů kvazi-identifikátorů, čímž v souboru dat vzniknou jedinečné kombinace hodnot kvazi-identifikátorů. Algoritmus kontroluje, zda se v seznamu četností vyskytují položky s méně než "k"prvky, čímž je zajištěna ochrana osobních údajů. Pokud ano, najde atribut s nejvíce odlišnými hodnotami a zobecní jej na všechny záznamy. Algoritmus běží, dokud není méně než "k"prvků, které nepatří do žádné třídy ekvivalence s velikostí větší než "k". Tyto prvky jsou potlačeny a algoritmus je ukončen. Aktuální stav seznamu četností představuje vygenerované třídy ekvivalence.[24] Přístup nezaručuje minimální k-anonymní řešení, nalezené řešení je však vždy k-anonymní.[25]

Mondrian metoda začíná s jednou ekvivalenční třídou, která zahrnuje celou datovou sadu, a pokračuje v rozdělení této jedné ekvivalenční třídy na více menších tříd. Jedná se o greedy ("lakomý") a rekurzivní algoritmus. Jeho "lakomost"spočívá v tom, že algoritmus vždy vybere "nejširší"atribut, předpokládajíc, že to povede k nejvyváženějšímu rozdělení aktuální ekvivalenční třídy. Rekurze algoritmu znamená, že prakticky funguje na podobný způsob jako průchod stromem do hloubky. Bere jednu ekvivalenční třídu (skupinu) a rozdělí ji na dvě: nazvěme první nově vytvořenou ekvivalenční třídu "vlevo"a druhou "vpravo". Algoritmus pokračuje v rozdělení "vlevo"ekvivalenční třídy, opět získává novou "vlevo"a "vpravo". Pokračuje ve splnění dalšího rozdělení "vlevo"a tak dále, dokud nenarazí na "vlevo"ekvivalenční třídu, která nemůže být dále rozdělena. Rozdělení je neplatné, pokud je velikost alespoň jedné z dvou výstupních ekvivalenčních tříd menší než "k". Poté se pokouší rozdělit "vpravo", pokračuje dále ve "stromu"prostřednictvím rekurzivních volání, dokud nejsou nalezeny platné rozdělení. Jinak se vrací zpět nahoru ve "stromu", tj. podél zásobníku rekurzivních volání. [24]

Incognito využívá přístup zdola nahoru se strategií prohledávání do šířky k navigaci po mřížce za účelem nalezení všech k-minimálních vektorů vzdálenosti. Po nalezení všech vektorů algoritmus vypočítá jejich informační ztrátu a vybere řešení s nejmenší informační ztrátou jako optimální řešení. Algoritmus může tímto způsobem najít globální optimum.[25]

Nedostatky a útoky na model k-anonymity

Při použití modelu k-anonymity je nutné dbát na vhodně zvolený parametr "k", jenž určuje velikost daných tříd ekvivalence. Při volbě velkého parametru se může užitečnost anonymizovaných záznamů zmenšit a být tak nepoužitelná pro další použití,

oproti tomu při volbě malého parametru je zde riziko nedostatečné anonymizace dat.

Jedním z útoku je útok na homogenitu modelu. Ten poukazuje na problém nedostatečné diverzifikace jednotlivých skupin (jak lze vidět v tabulce 1.8, kde skupina s osobami nad 50 let má shodnou nemoc). Jako dalším typ lze uvést útok se znalostí pozadí, tzn. že útočník zná některé informace o osobě z tabulky, které mohou dopomoci identifikovat jednotlivý záznam. [26] Příkladem tak může být situace, kdy útočník ví, že se v tabulce 1.8 nachází záznam o osobě (jeho blízké, u které zná tyto údaje: věk 27 let, trvalé bydliště 277 13 a kupuje často inzulín). Z těchto záznamů, kterými disponuje útočník, tak dokáže identifikovat jednotlivý záznam v tabulce.

1.4.2 L-diversity

Další z představených pokročilých anonymizačních technik je l-diversity. Jedná se o techniku, která doplňuje předchozí k-anonymity model a zároveň díky tomu odpadají některé nedostatky tohoto modelu. L-diversity tak představuje vložení další entropie/diverzity do anonymizovaného datasetu. Cílem l-diverzity je rozšířit anonymizaci, po provedení techniky k-anonymity (zobecněním a maskováním identifikátorů), také na citlivé záznamy v datasetu. Princip l-diverzity požaduje, aby v každé třídě ekvivalence záznamů mohlo být nejvýše 1/l jejích záznamů s identickou hodnotou důvěrného atributu. [27] Příklad l-diversity lze vidět v tabulce 1.9.

Tab. 1.9: Data po provedení k-anonymity a před provedení metody l-diversity

Věk	**PSČ**	**Nemoc**
<30	27* **	Žloutenku
<30	27* **	Cukrovka
<30	27* **	Srdeční nemoc
<30	27* **	Srdeční nemoc
30-40	5** **	Cukrovka
30-40	5** **	Cukrovka
30-40	5** **	Srdeční nemoc
30-40	5** **	Srdeční nemoc
>50	5** **	Rakovina
>50	5** **	Rakovina
>50	5** **	Rakovina
>50	5** **	Rakovina

Věk	**PSČ**	**Nemoc**
<30	27* **	Žloutenku
<30	27* **	Cukrovka
<30	27* **	Srdeční nemoc
<30	27* **	Srdeční nemoc
>30	53* **	Cukrovka
>30	53* **	Srdeční nemoc
>30	53* **	Srdeční nemoc
>30	53* **	Rakovina
>30	51* **	Rakovina
>30	51* **	Cukrovka
>30	51* **	Cukrovka
>30	51* **	Rakovina

Při převzetí tabulky, na které byla aplikovaná technika k-anonymity (pro k=4),

bylo na tabulku aplikováno l-diversity (kde l=3), to znamená že v tabulce v jednotlivých skupinách záznamů se nebude vyskytovat více než l-1 možností citlivých údajů.

V tuto chvíli, kdy útočník má stejné informace o subjektu jako měl v předchozím případě (tzn. osoba 50+ let, PSČ bydliště je 512 33 a vystupuje v této tabulce), tak je soukromí této osoby zajištěno více než v případě, kdy byla aplikovaná pouze metoda k-anonymity. A to z důvodu, že útočník v danou chvíli nemůže jednoznačně určit, jakou nemocí disponuje oběť.

L-diversity algoritmy

Stejně jako u k-anonymity, tak i u l-diversity existují různé algoritmy zabývající se implementací l-diversity. Mezi známé algoritmy l-diversity lze uvést BGSI [28], Clustering-Based Frequency l-diversity [29], V-MDAV (Variable-size Maximum Distance to Average Vector) [30] apod. Například algoritmus BGSI nabízí silnější mechanismus ochrany soukromí (s menší ztrátou informací) tím, že podporuje jedinečné a různorodé citlivé hodnoty uvnitř skupin, což snižuje možnost útoků na odhalení atributů. [28] Nicméně tyto algoritmy jsou velice komplexní a jejich detailnější rozbor není pro práci podstatný.

Nedostatky a útoky na l-diversity

Modelu l-diversity může být mnohdy složité dosáhnout a v některých případech i zbytečné. Co se týče útoků, model je náchylný na skewness útoky (útoky zkreslením) a útoky na podobnost. [31] Při uvažování datasetu s 50 záznamy, který splňuje 2-diversitu (tzn. ve skupině se nachází shodný počet možností pozitivní/negativní) skewness útok spočívá v tom, že jestliže by byl útočník schopen propojit konkrétního pacienta, lze mít za to, že vybraný pacient má 50% pravděpodobnost, že jeho záznam je pozitivní, namísto podstatně menší pravděpodobnosti pro celý dataset. [32]

U útoku na podobnost lze uvažovat následující příklad. Jestliže bude tabulka splňující l-diverzitu a v jedné skupině se tak vyskytnou různé typy rakoviny, útočník může z této situace prosperovat neboť, už jen informace, že některý ze subjektů bojuje s rakovinou, lze považovat za určité soukromí. Pokud by se nejednalo o nemoc, ale např. o číselný údaj, útočník se může dostat do situace, kdy v jedné skupině l-diversity se vyskytnou velmi podobné údaje a celá skupina bude mít následně velmi úzký rozptyl těchto hodnot. [27]

1.4.3 T-closeness

Anonymizační model t-closeness (neboli t-uzavřenost) je zdokonalením myšlenky předchozích modelů k-anonymity a l-diverzity, to hlavně z pohledu jejich odolnosti.

Článek [32] definuje princip tohoto modelu jako: "*O třídě ekvivalence se říká, že má t-uzavřenost, jestliže vzdálenost mezi rozložením citlivého atributu v této třídě a rozložením atributu v celé tabulce není větší než prahová hodnota t. O tabulce se říká, že má t-uzavřenost, jestliže všechny třídy ekvivalence mají t-uzavřenost.*"Pro určení vzdálenosti této vzdálenosti mezi rozloženími se často využívá ukazatele Earth Mover's Distance (dále jen EMD). Využití EMD má tu výhodu, že zohledňuje sémantickou blízkost hodnot atributů.

Algoritmy t-closeness

Zástupci algoritmů metody t-closeness jsou například SABRE (Sensitive Attribute Bucketization and Redistribution Framework for t-closeness), CODIP (Complete Disjoint Projections) apod. Pro první uvedený algoritmus platí, že se nejprve rozdělí původní data na sadu "kbelíků"(buckets) s podobnými hodnotami citlivých atributů a poté jsou vybrány záznamy z každého "kbelíku"pro vytvoření tříd ekvivalence. To zaručuje různorodost hodnot v každé třídě ekvivalence. Omezení tohoto algoritmu však spočívá v tom, že může vytvářet anonymizovaná data s nízkou kvalitou. Neboť "kbelíky"jsou generovány iterativně, což může vést k vytvoření více "kbelíků"a tak vzniknou ekvivalenční třídy s větším počtem záznamů, a tím dojde k větším ztrátám informací. [33]

Nedostatky a útoky na t-closeness

Použití EMD v metodě t-closeness může způsobit ztížení určení blízkosti mezi získanou znalostí a t-hodnotou. Přístup vyžaduje, aby rozložení citlivých atributů v jednotlivých třídách ekvivalence bylo podobné celkovému rozložení v datové tabulce. [31]

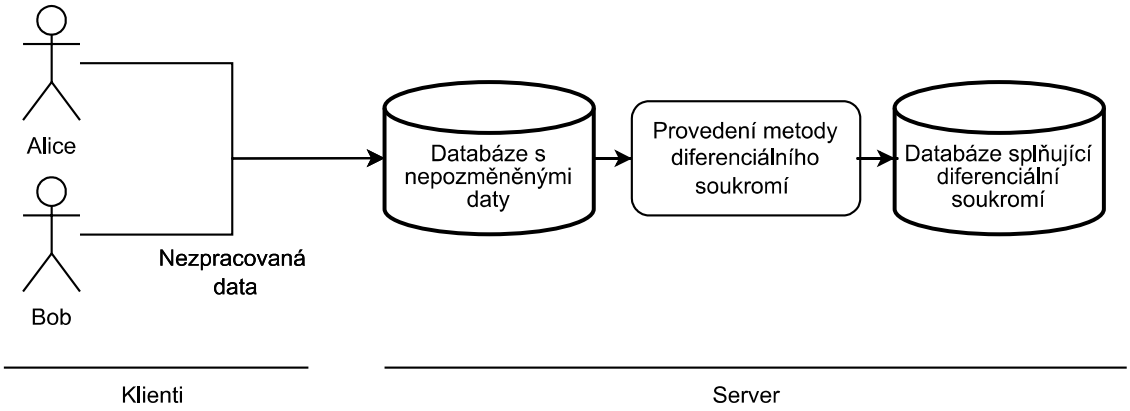
1.4.4 Diferenciální soukromí

Diferenciální soukromí je matematický koncept používaný ve statistické analýze a analýze strojového učení k ochraně soukromí jednotlivců v datech. Umožňuje shromažďovat, analyzovat a sdílet statistické odhady založené na osobních údajích a zároveň zajišťuje ochranu před různými útoky na soukromí. Tyto útoky zahrnují re-identifikaci, propojení záznamů a diferenciaci a mohou zahrnovat i neznámé nebo nepředvídané útoky. [34]

Diferenciální soukromí je realizováno použitím mechanismu na jakoukoli informaci vystavenou ze souboru dat. Diferenciální soukromí funguje tak, že do dat vnáší řízenou náhodnost nebo šum tak, aby chránil soukromí subjektů údajů. Mechanismus využívá řadu technik, jako je náhodná odpověď, zamíchání nebo aditivní šum. Konkrétní volba mechanismu je v podstatě přizpůsobena povaze a kvalitě informací. Koncept je navržen tak, aby zajistil informačně-teoretickou záruku soukromí tak, že výstup konkrétní analýzy zůstane téměř stejný bez ohledu na to, zda jsou zahrnuty data o konkrétním subjektu údajů, či nikoli. Například jestliže je z anonymní databáze vymazán záznam o jednom subjektu, měl by být výstup z této databáze po provedení mechanismu diferenciálního soukromí přibližně stejný jako výstup z původní databáze, ve které vybraný subjekt nechybí. Pakliže je výsledek rozdílný, a lze s dostatečnou jistotou rozlišit případ, kdy byl či nebyl subjekt z databáze odebrán, poté je možné říci, že vybraný mechanismus nesplňuje diferenciální soukromí. Síla soukromí v diferenciálním soukromí se řídí nastavením parametru soukromí ϵ, známého také jako ztráta soukromí. Při snížení ϵ se však zvyšuje množství šumu nebo "náhodnosti", které mechanismus používá. Pokud je rozpočet na ochranu soukromí příliš malý, odhalené informace se stanou nepoužitelnými pro jakýkoli praktický účel. [35]

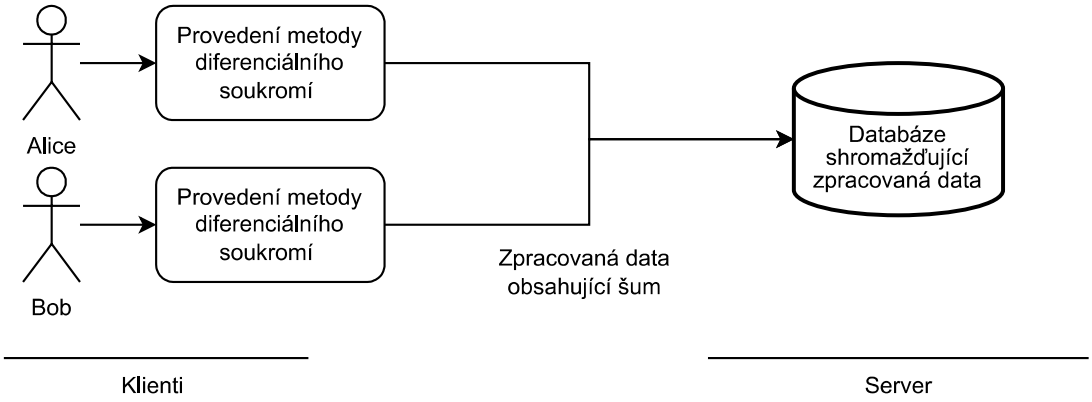
V praxi jsou rozlišovány dva hlavní typy diferenciálního soukromí:

- centrální/globální diferenciální soukromí (Central Differential Privacy, CDP / Global Differential Privacy, GDP): Jak lze vidět na obrázku níže, u tohoto typu diferenciálního soukromí jsou data od jednotlivců shromažďována a ukládána v nešifrovaném formátu důvěryhodným centralizovaným správcem dat, který poté provede mechanismus diferenciálního soukromí na citlivých datech a výstupy uvolní nedůvěryhodné veřejnosti. Toto však může být pro mnoho aplikací nevhodné, neboť to představuje jediný bod selhání pro narušení dat a zatěžuje tak důvěryhodného správce právními a etickými povinnostmi dodržovat soukromí dat.



Obr. 1.4: Centrální/Globální diferenciální soukromí

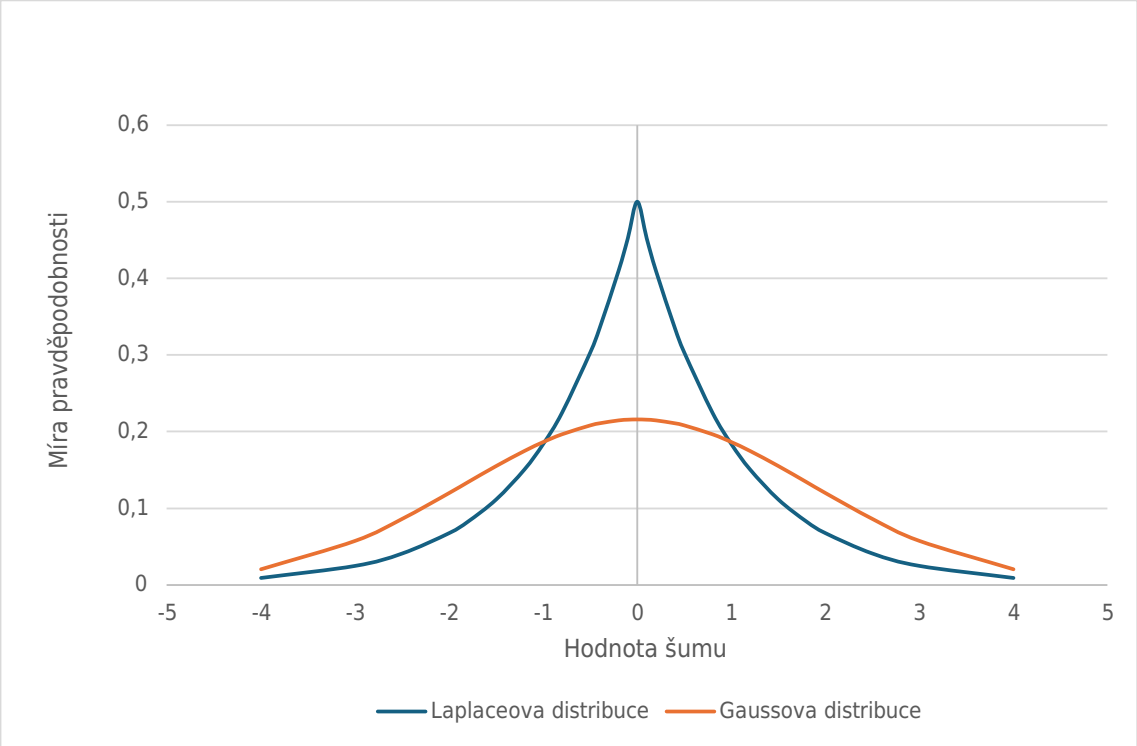
- lokální diferenciální soukromí (Local Differential Privacy, LDP): V případě LDP každý jednotlivec náhodně upravuje svá vlastní data pomocí mechanismu lokálního diferenciálního soukromí ve svém zařízení. Typ lokálního diferenciálního soukromí lze vidět na obrázku 1.5. Datový analytik používá tyto zašuměné nebo náhodné odpovědi k odvození souhrnných statistik kombinovaných dat. Proto se lokální diferenciální soukromí považuje za bezpečnější ve srovnání s předchozím typem. LDP je však mnohem méně efektivní, neboť pro dosažení stejného soukromí je třeba do dat přidat mnohem větší množství náhodného šumu. Lze říci, že přístup lokálního diferenciálního soukromí je obecně účinější pro mnohem větší soubory dat ve srovnání s předchozím typem diferenciálního soukromí.



Obr. 1.5: Lokální diferenciální soukromí

Mechanismy diferenciálního soukromí

K dosažení diferenciálního soukromí lze využít několika mechanismů. Těmi nejznámějšími jsou mechanismy pro přidání aditivního šumu k datům (platí pro číselné záznamy), jmenovitě pak Laplaceův mechanismus a Gaussův mechanismus. Laplaceův mechanismus používá Laplaceovo rozdělení, jenž je symetrickou verzí exponenciálního rozdělení. K pravdivé odpovědi přidá šum ze symetrického spojitého rozdělení. [36] Toto rozdělení je vhodné pro situace, kde je požadováno, aby pravděpodobnost velkých šumů byla malá, ale přesto možná. To umožňuje generovat přesnější odhady, zejména pro hodnoty blízko nuly. Druhý, Gaussovský mechanismus používá Gaussovo (též známé jako normální) rozdělení pravděpodobnosti. Toto rozdělení má vyšší hustotu pravděpodobnosti kolem střední hodnoty, což znamená, že vygenerovaný šum má tendenci být koncentrován kolem skutečných hodnot. Názorné průběhy obou mechanismů lze vidět na obrázku níže (1.6).



Obr. 1.6: Porovnání mechanismů diferenciálního soukromí pro přidání aditivního šumu

1.4.5 Homomorfní šifrování

Homomorfní šifrování je kryptografická technika, která umožňuje třetím stranám provádět operace se zašifrovanými texty, aniž by tyto texty byly před těmito operacemi dešifrovány. Výhodou je tak uchování citlivých informací v zašifrované podobě

během provádění výpočtů nebo analýz třetími stranami. Jedním z hlavních omezení homomorfního šifrování je vyšší výpočetní náročnost a velikost šifrovaných dat.

Homomorfní šifrování lze dělit do čtyř typů [37]:

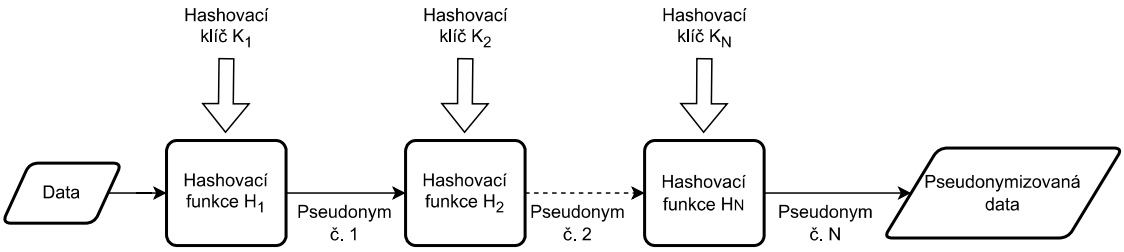
- částečně homomorfní šifrování (Partially Homomorphic Encryption, PHE): Nejjednodušší forma homomorfního šifrování umožňuje provádět nad šifrovanými daty buď sčítání nebo násobení, avšak ne obě operace současně. Tento typ šifrování umožňuje vypočítat součet nebo součin určité sady dat.
- poněkud homomorfní šifrování (Somewhat Homomorphic Encryption, SWHE): SWHE umožňuje provádět nad zašifrovanými daty jak operace sčítání, tak i násobení. Nicméně, má určitá omezení. Počet operací, které lze provést nad šifrovanými daty, je omezen a přesnost výsledků se může snižovat s každou další operací. To znamená, že čím více operací se provádí, tím méně přesné mohou být výsledky.
- stupňované plně homomorfní šifrování (Leveled Fully Homomorphic Encryption, LFHE): Jedná se o pokročilou formu homomorfního šifrování, která umožňuje provádět libovolný počet aritmetických operací nad šifrovanými daty, pokud je předem definována posloupnost těchto operací. To znamená, že lze vytvořit určitou "úroveň"výpočtů, které jsou předem stanovené, a následně tyto operace provádět nad zašifrovanými daty.
- plně homomorfní šifrování (Fully Homomorphic Encryption, FHE): Plně homomorfní šifrování je nejvíce pokročilou technologií homomorfního šifrování, která umožňuje neomezené výpočty na zašifrovaných datech a to v předem nedefinovaném pořadí takovýchto výpočtů. Nepraktičnost schémat FHE se projevuje jejími náročnými výpočtovými vlastnostmi.

Mezi příklady částečného homomorfního šifrování lze uvést algoritmy ElGamal, Benaloh, Paillier. K známým SWHE schématům patří schéma Boneh, Goh, and Nissim (BGN) a Polly Cracker. [38]

1.4.6 Řetězení funkce hash

Bezpečné kryptografické hashovací funkce se často nepovažují za vhodné techniky pseudonymizace, přednost se dává autentizačním kódům a hashovacím funkcím s využitím klíče (tzv. soli, která se přidává ke vstupním řetězcům). Pokročilé techniky však lze také získat řetězením hashovacích funkcí, což je vrstevnatý přístup, kdy jsou dočasně generovány mezilehlé pseudonymy pro získání konečného pseudonymu. Protivník musí kompromitovat veškeré entity, aby pseudonymizaci zvrátil, což vyžaduje znalost všech pseudonymů. Tím je zajištěna další vlastnost, které nelze dosáhnout jedinou hashovací funkcí s klíčem - žádná entita, která obdrží meziprodukt pseudonymu, jej nemůže zvrátit a první entita nemůže porovnat konečné pseudonymy

s původními identifikátory. Názorný zjednodušený obecný příklad řetězení funkce hash lze vidět níže na obrázku 1.7. Z toho vyplývá skutečnost, že s narůstajícím počtem entit provádějící hashování, úměrně narůstá i bezpečnost výsledných pseudonymizovaných dat.



Obr. 1.7: Řetězení funkce hash

1.5 Lokalizační údaje

V následující sekci je pojednáváno o de-identifikaci lokalizačních údajů, neboť metody de-identifikace těchto údajů mohou být poněkud rozdílné oproti údajům jako jsou například jméno a příjmení, rodné číslo apod. Takto de-identifikovaná data lze použít například při analýzách, které vyžadují znát lokalitu nebo trajektorii pohybu. Názorný příklad lze vidět na různých rozborech dat z dob, kdy vypukla pandemie koronavirové choroby covid-19. Dalším možným využitím těchto dat lze považovat použití v oblasti marketingu, kdy lze předpovídat cesty spotřebitelů k nákupům, poskytování doporučení v místě zájmu apod.

Lokalizačním údajem je myšlen údaj, díky kterému je možné ať už přímo nebo nepřímo identifikovat subjekt osobních údajů. Z pohledu GDPR se tak jedná o osobní údaj, avšak přesnější definici lze nalézt v §91 odst. 1 zákona č. 127/2005 Sb. o elektronických komunikacích, jenž definuje lokalizační údaje jako údaje : "*..., které určují zeměpisnou polohu telekomunikačního koncového zařízení uživatele veřejně dostupné služby elektronických komunikací.*"[39]

Nejběžnějším způsobem, jakým se lokalizační údaje zapisují jsou pomocí zeměpisných souřadnic (tzv. koordinátů). Ty se zapisují ve tvaru dvojice souřadnic (zeměpisné šířky a délky) a případně někdy i třetí hodnoty, jenž definuje zeměpisnou výšku. Souřadnice jsou udávány ve stupních a zapisují se třemi možnými způsoby. Jedním způsobem je jako desetinné číslo stupně, druhým je zápis celých stupňů a minuty jsou udány v desetinném čísle, či poslední možným způsobem jako stupně a minuty v celých číslech a sekundy v desetinném čísle. Veškerá desetinná čísla jsou zapisována pomocí desetinné tečky. Takovýto způsob zápisu lokalizačních údajů je

dán systémem WGS84 (World Geodetic System 1984). Existují i alternativy zápisu souřadnic jako například systém S-JTSK (Souřadnicový systém Jednotné trigonometrické sítě katastrální) nebo systém ETRS89 (Evropský terestrický referenční systém 1989).

1.5.1 De-identifikační metody lokalizačních údajů

Možností, jak lokalizační údaje anonymizovat je metoda zhrubnutí polohy. V jádru věci se jedná pouze o jiný název pro generalizaci polohy. Jak již bylo popsáno výše v této práci, jedná se o zobecnění záznamů. V praxi to znamená, že se hodnoty koordinátů zaokrouhlí na určitý počet desetinných míst. Avšak tento způsob nelze považovat za příliš použitelný, neboť hustota osídlení není vždy v každé oblasti stejná, a tak nelze zaručit stejnou anonymitu pro subjekt údajů v hustě osídleném městě jako pro subjekt údajů nacházející se v mírně osídlené části země. Toto dokazují autoři v článku [40], kde uvádějí příklad zaokrouhlení souřadnic na celá čísla, což zapříčiní posun lokality až o 47 kilometrů. Kdežto v případě zaokrouhlení na jedno desetinné číslo je lokalita posunuta pouze o cca 3,7 kilometrů. Autoři v článku také zmiňují metodu mikroagregace. Metoda spočívá ve vytvoření jednoho centroidu (centrálního bodu) z několika dvojic souřadnic. Mikroagregace způsobila, že při použití dvou dvojic souřadnic se lokalita nezměnila a při použití více dvojic souřadnic nebyly posuny lokality tak razantní. Avšak nevýhodou uvádějí autoři významné vymazání dat a nemožnost předpovídat jakékoliv trajektorie pohybu.

1.6 Strojově čitelné soubory

Strojově čitelné soubory představují řešení pro efektivní manipulaci s daty v počítačových systémech. Tyto soubory jsou navrženy tak, aby umožňovaly snadnou čitelnost a zpracování dat počítačovými programy, což je klíčové pro automatizaci procesů, integraci systémů, analýzu dat apod. Různé formáty strojově čitelných souborů, jako jsou JSON (JavaScript Object Notation), CSV (Comma-separated values), XML (Extensible Markup Language) nabízejí flexibilitu a možnosti pro uložení a výměnu dat ve strukturované podobě. Každý z těchto formátů má své výhody a nevýhody a mají různé možnosti využití.

Díky svým schopnostem jako snadno číst, zapisovat a zpracovávat data jsou strojově čitelné soubory důležitým nástrojem v moderních informačních systémech. Používají se pro ukládání konfiguračních informací, přenos dat mezi různými aplikacemi a systémy, analýzu velkých datových sad a mnoho dalších účelů. Standardizované formáty těchto souborů umožňují kompatibilitu mezi různými platformami a systémy, což podporuje interoperabilitu a efektivní výměnu informací.

2 Praktická část

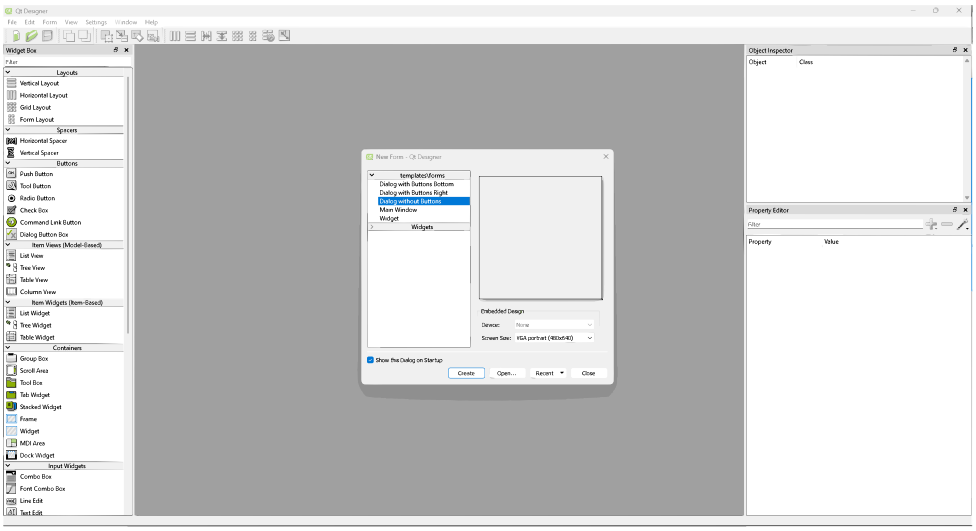
V praktické části se práce zabývá vytvořením aplikace, ve které jsou implementovány anonymizační a pseudonymizační techniky de-indentifikace osobních údajů. Použité techniky byly vybrány na základě rešerže provedené v teoretické části práce. Aplikace je vytvořena v programovacím jazyce Python za využití potřebných knihoven. Aplikace disponuje grafickým uživatelským rozhraním, čímž je zjednodušena její obsluha. Mimo jiné je v aplikaci možné vygenerovat potřebný dataset s falešnými osobními údaji. Z důvodu jednoduchosti syntaxe (umožňující snadné čtení a zápis dat) a podpoře hierarchické struktury dat (umožňující reprezentaci složitějších datových modelů v jednoduché a srozumitelné podobě) aplikace podporuje pro veškeré akce (generování, čtení, zapisování) ve všech svých částech pouze soubory typu JSON (JavaScript Object Notation).

2.1 Vytvoření aplikace

Aplikace je vytvořena v programovacím jazyce Python ve verzi 3.12 na operačním systému Windows 11 (verze 23H2). Vývojovým prostředím pro programování byl zvolen PyCharm (Community Edition) ve verzi 2023.2.5. V celé aplikace je využito, mimo standardně dodávaných knihoven spolu s Pythonem, několik externích knihoven, které bylo nutné doinstalovat. K vytvoření grafického rozhraní byla využita knihovna PyQt5 a s ní distribuovaný program Qt designer. V aplikaci jsou využity knihovny math (zajišťující matematické funkce), random (pro práci s náhodnými čísly), json (práce s JSON soubory), datetime (práce s datumy a časem), GeoPy (knihovna, která dokáže nalézt souřadnice adres, měst, zemí a orientačních bodů po celém světě pomocí geokodérů třetích stran a dalších zdrojů dat [41]), Faker (generuje smyšlené osobní údaje), Shapely (slouží k manipulaci a analýze geometrických objektů v rovině), NumPy (knihovna, která pracuje s vektory, maticemi a vícerozměrnými poli) a knihovna Pandas (ta usnadňuje práci, manipulaci a analýzu dat).

2.1.1 Grafické uživatelské rozhraní aplikace

Jak již bylo řečeno výše, při tvorbě grafického rozhraní byl využit program Qt designer (viz obrázek 2.1), což je nástroj z frameworku Qt pro jazyk Python. Program zjednodušuje vytváření grafického rozhraní pro aplikace psané v jazyce Python. Přední výhodou programu Qt designer je schopnost naprogramovat grafické rozhraní v podobě, které uživatel požaduje. Qt designer je aplikace se schopností WYSIWYG (What You See Is What You Get), což lze přeložit jako schopnost zobrazit produkt v podobě, jakou má a i jakou bude mít v době po exportu z aplikace.



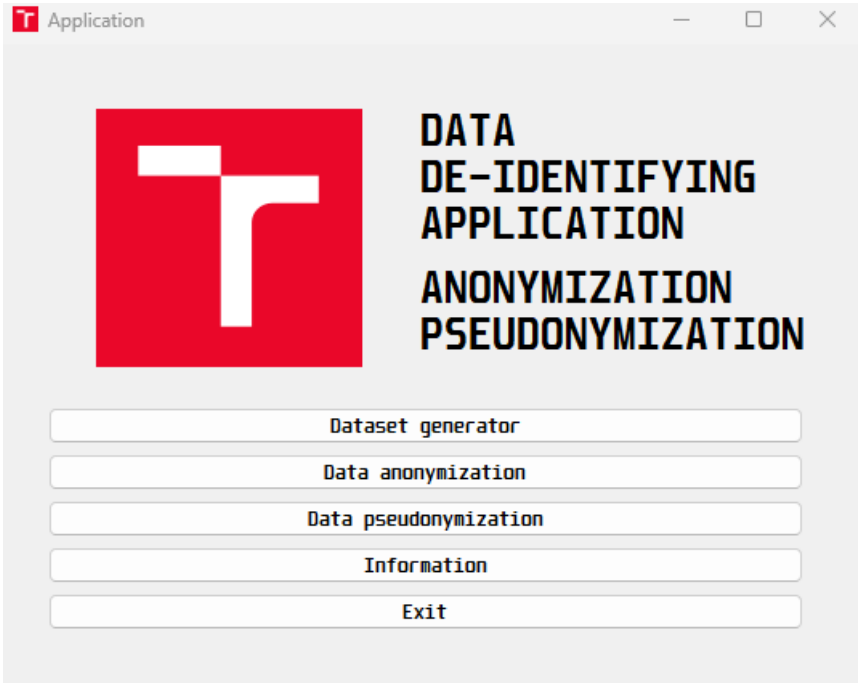
Obr. 2.1: Ukázka aplikace Qt designer

Výstupem takovéhoto programu jsou soubory obsahují popis vizuálních prvků (jako jsou okna, tlačítka, textová pole, seznamy atd.), a definují jejich vzhled, pozici a interakce s uživatelem pomocí XML syntaxe. Výstupními soubory programu Qt designer jsou soubory s příponou .ui (user interface). Tyto soubory lze poté buď importovat pomocí kódu nebo pomocí příkazu z těchto souborů udělat výstupní soubory ve formátu podporující programovací jazyk Python. Výhodou nově vzniklého souboru v Python jazyce je jeho možnost modifikace při práci. Příkaz pro transformaci souboru lze vidět níže. Příkaz je součástí knihovny PyQt5, není nutné jej nijak doinstalovávat do uživatelské stanice. Syntaxe příkazu znamená, že knihovna vygenerovává kód ze souboru zapsaném za "-x"a do nově vzniklého souboru za částí příkazu "-o".

```
D:\Project folder>pyuic5 -x user_interface_file.ui -o python_file.py
```

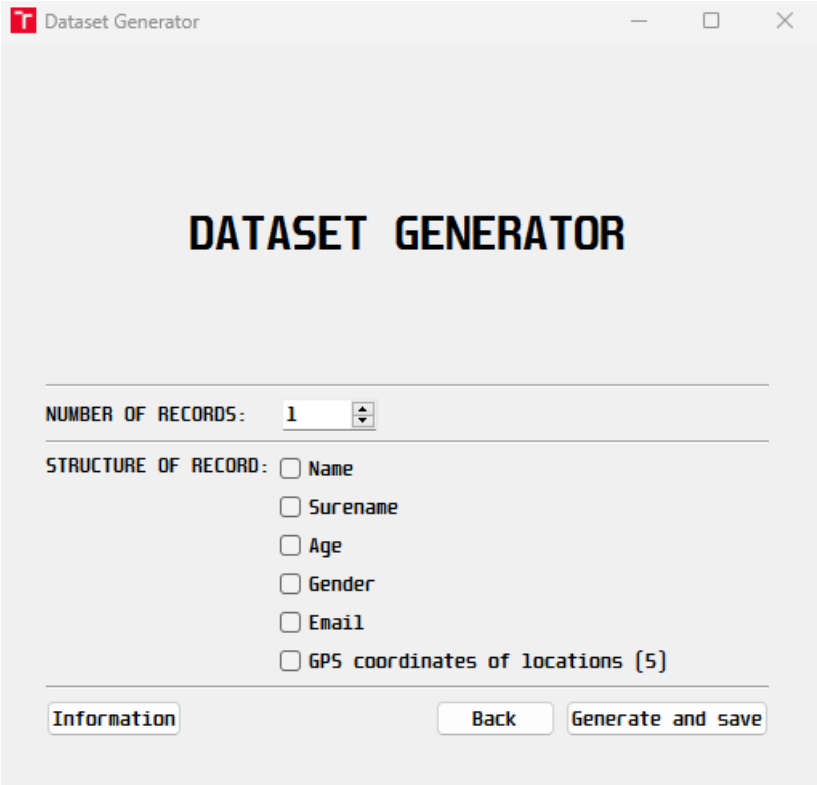
2.2 Popis aplikace

Celá aplikace je vytvořena v podobě obdobné např. instalaci programu. Při každém kroku je vytvořeno a otevřeno nové okno aplikace. Po spuštění lze vidět úvodní obrazovku aplikace, která slouží jako rozcestník celé aplikace (viz obrázek 2.2). Na výběr připadá několika tlačítek, kdy každé přenese uživatele k dalšímu kroku aplikace (obrazovce). V případě tlačítka *Information* se zobrazí informace o aplikaci a tlačítko *Exit* celou aplikaci uzavře.



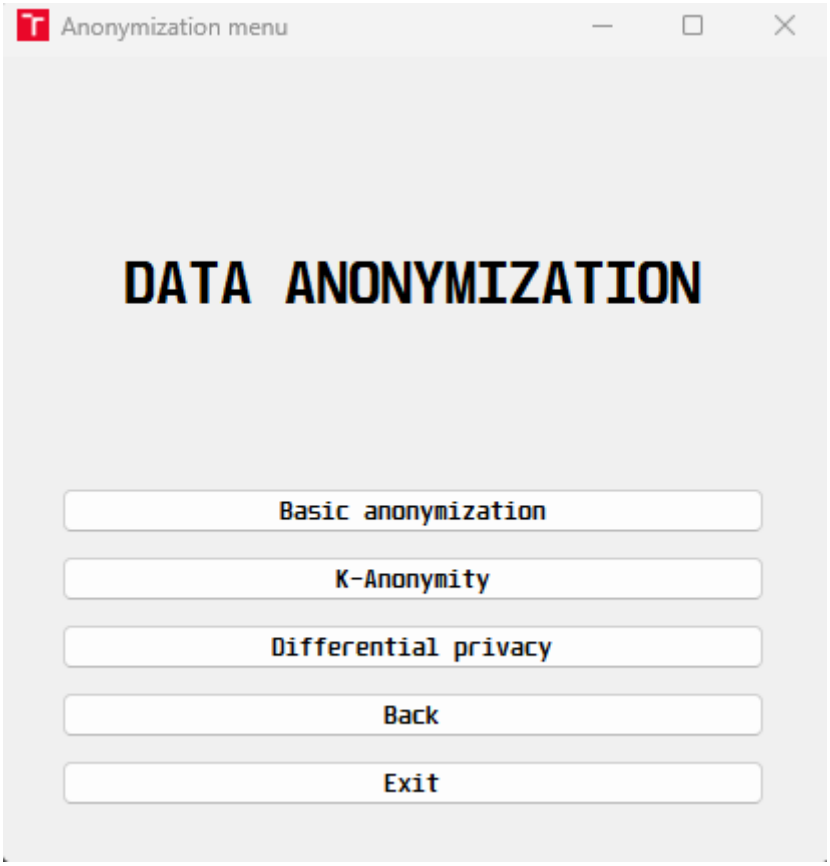
Obr. 2.2: Zobrazení úvodního okna aplikace

První volba *Dataset generator* přenese uživatele do nového okna, ve kterém je možné vytvořit dostatečně velký zkušební dataset pro budoucí testování implementovaných de-identifikačních technik. Jak lze vidět na obrázku 2.3 níže, v tomto okně aplikace si uživatel dokáže vytvořit náhodný dataset s libovolným počtem fiktivních záznamů a o jedinečné struktuře tohoto testovacího datasetu. Na výběr, který definuje strukturu datasetu, je z několika možností atributů záznamů, jmenovitě pak jméno, příjmení, věk, pohlaví, emailová adresa a poslední možností záznamu jsou souřadnice (ve tvaru zeměpisné šířky-latitude a délky-longitude v desitinném čísle). V datasetu je, při výběru těchto souřadnic, zapsáno do záznamu pět takovýchto dvojic souřadnic. Všech pět dvojic souřadnic jsou náhodná místa v okruhu náhodně vygenerovaného trvalého bydliště (cca v okolí 25km od tohoto bydliště). Poslední možností v tomto okně aplikace je možnost vrátit se o krok zpět a nebo celý dataset vytvořit a uložit ho ve formátu JSON v místě adresáře, který uživatel sám zvolí.



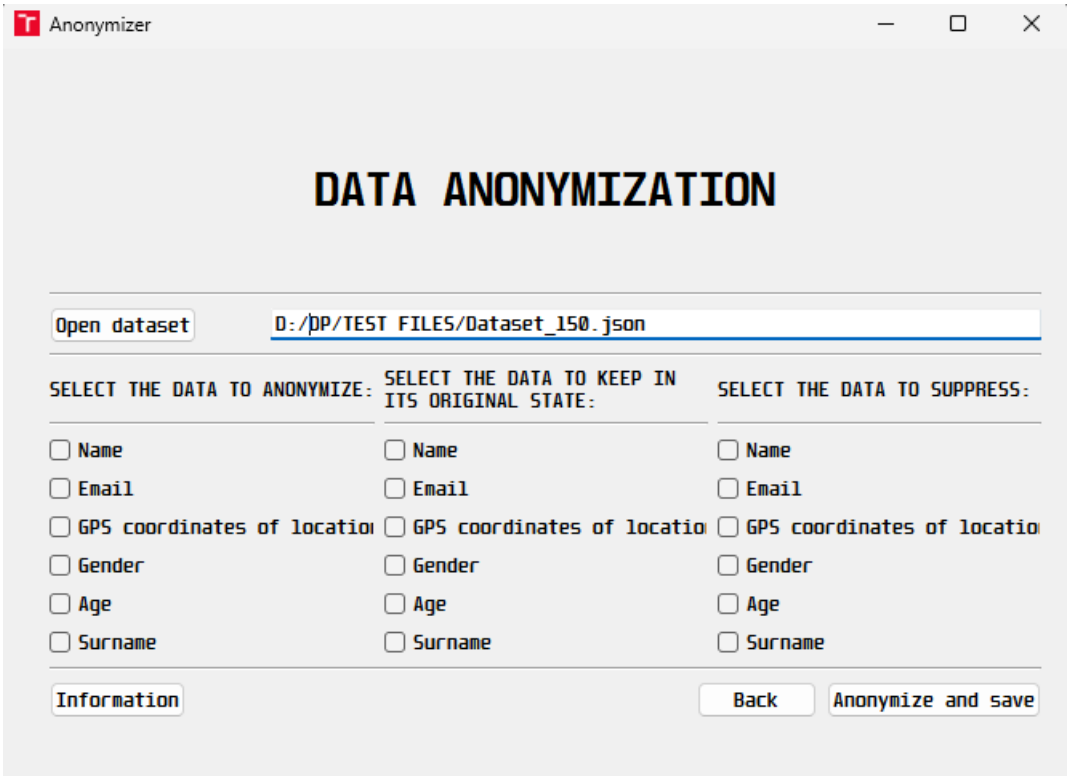
Obr. 2.3: Zobrazení okna aplikace pro vytvoření zkušebního generátoru

Druhou možností výběru z hlavního menu aplikace je *Data anonymization*, jenž jak název napovídá, slouží k výběru anonymizační techniky. Lze vybírat z následujících možností, které jsou vidět na obrázku 2.4. První tlačítko tohoto menu odkáže uživatele na základní možnosti anonymizace dat. Těmito základními metodami anonymizace jsou maskování emailové adresy, zobecnění věku a generalizace souřadnic. Druhá volba je věnována metodě k-anonymity (ve které je použita technika multidimenzionální Mondrian) a poslední implementovanou metodou anonymizace byla zvolena metoda diferenciálního soukromí. Každá metoda anonymizace disponuje tlačítkem informace, která obsahuje základní informace o vybrané metodě.



Obr. 2.4: Podmenu výběru anonymizační techniky

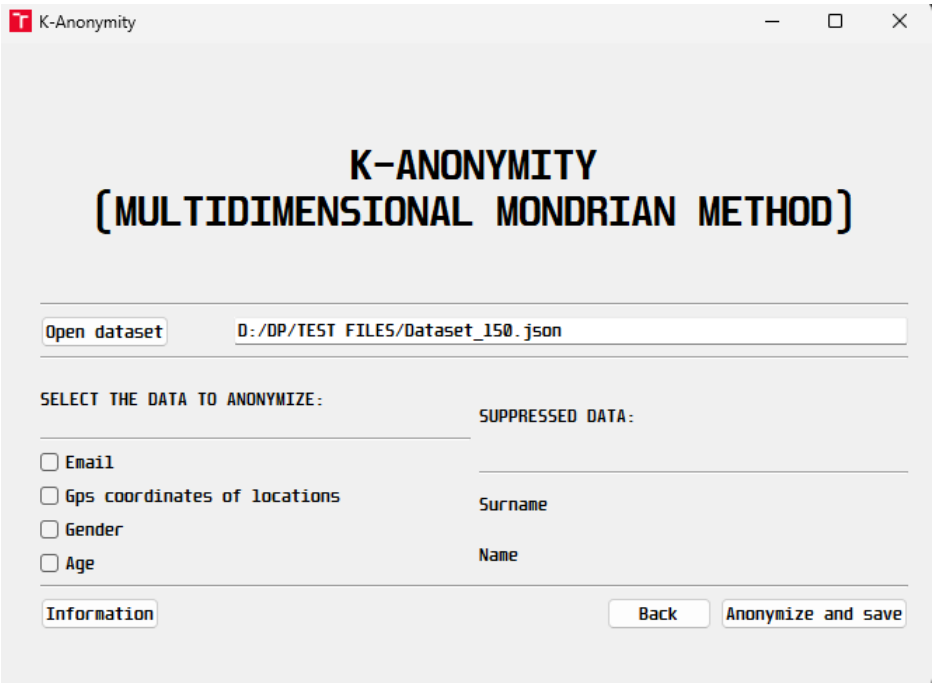
Prvním výběrem z podmenu anonymizace (obrázek 2.4) může uživatel vyzkoušet základní anonymizační techniky. V tomto okně aplikace je uživatel vyzván k vložení požadovaného datasetu. Z tohoto datasetu jsou posléze načteny klíče (z páru klíč-hodnota uložených v JSON souboru) a jsou zobrazeny jako zaškrtávací políčka ve třech sloupcích. V prvním sloupci uživatel zaškrtne data, která požaduje anonymizovat. Zaškrtnutá políčka ve druhém sloupci indikují ta data, která uživatel bude požadovat ponechat v nepozměněném stavu a poslední sloupec slouží pro potlačení dat (tato volba by měla být vybrána pro přímé identifikátory jako je například celé jméno, rodné číslo apod.). Po zaškrtnutí všech políček lze data anonymizovat a uložit ve formátu JSON na požadovaném místě po stisku tlačítka *Anonymize and save*. Okno aplikace pro základní anonymizaci lze vidět na obrázku 2.5 níže.



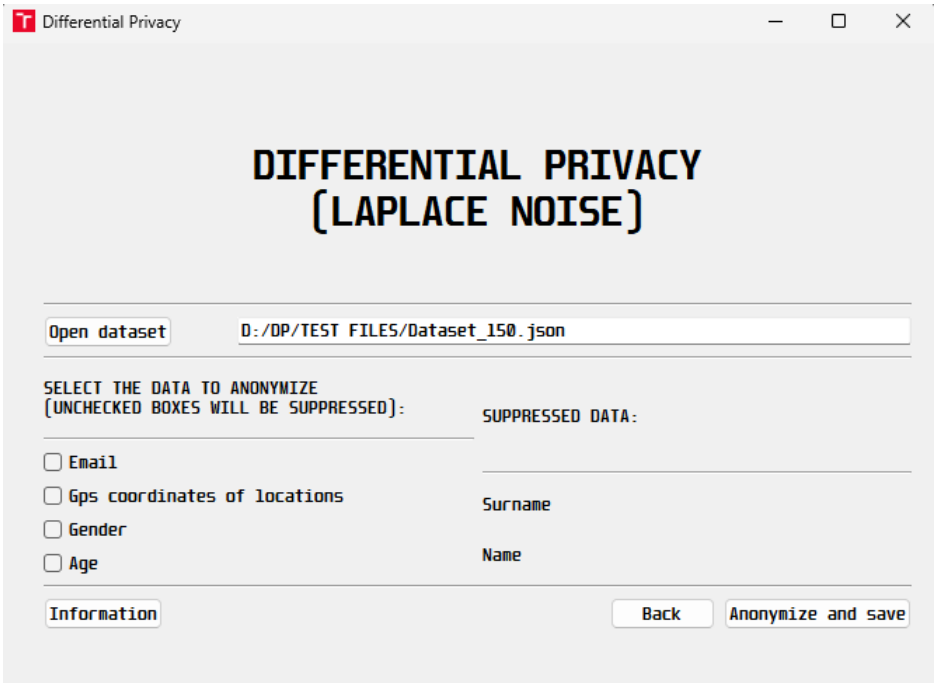
Obr. 2.5: Okno aplikace pro základní anonymizaci dat

Tlačítko *K-anonymity* anonymizačního podmenu přenáší do obdobného okna aplikace (obrázek 2.6) jako předchozí možnost. Zde uživatel opět načte svůj dataset a jestliže se v datasetu objeví klíč obsahující řetězec jména nebo příjmení, budou tyto klíče zařazeny do pravého sloupce indikující data k potlačení (z důvodu, že se jedná o přímý identifikátor). Zbylé načtené klíče jsou vepsány do levého sloupce, kde lze vybrat strukturu anonymizovaného souboru (jinými slovy, které záznamy mají být anonymizovány). Dále pak také uživatel musí nastavit hodnotu "k", jenž slouží k nastavení minimálního počtu záznamů v jednotlivých anonymizovaných skupinách. Po nastavení této hodnoty a vybrání dat k anonymizaci aplikace povolí anonymizaci a uložení takto pozměněných dat.

Poslední anonymizační metodou implementovanou v aplikaci je diferenciální soukromí. Tu lze nalézt pod tlačítkem *Differential privacy* v podmenu anonymizace. I v tomto případě je uživatel vyzván k načtení souboru s osobními údaji a na základě struktury tohoto souboru jsou vypsány klíče do levého či pravého sloupce (na základě zda se jedná o jméno nebo příjmení, či nikoli). V případě zaškrtnutí věku či souřadnic, je k těmto hodnotám vygenerován a přidán náhodný šum, jenž zajistí jejich anonymizaci. V obou případech se využívá Laplaceova mechanismu. Okno *Differential privacy* zobrazuje níže obrázek 2.7.



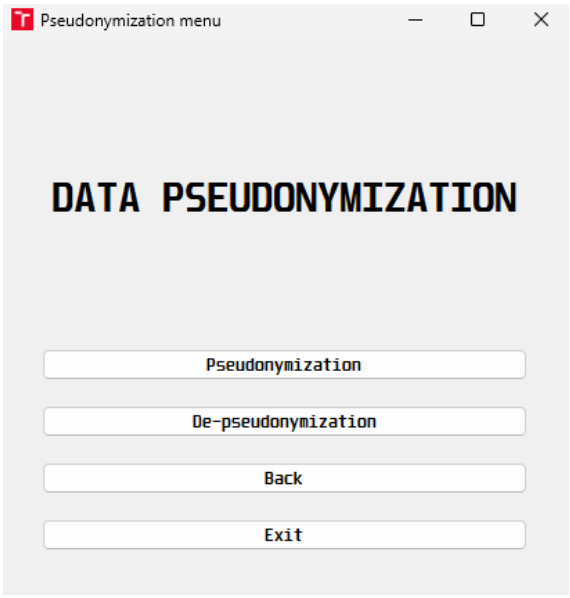
Obr. 2.6: Okno aplikace implementující k-anonymitu



Obr. 2.7: Okno aplikace s metodou diferenciálního soukromí

Tlačítko *Data pseudonymization* v hlavním menu aplikace (obrázek 2.2) otevře, jak lze vidět níže na obrázku 2.8, podmenu pseudonymizace. Toto podmenu disponuje tlačítkem pro pseudonymizaci dat a tlačítkem pro zpětnou de-pseudonymizaci

těchto pseudonymizovaných dat. A nechybí zde ani tlačítko pro vrácení se zpět do hlavního menu aplikace nebo pro ukončení celé aplikace.

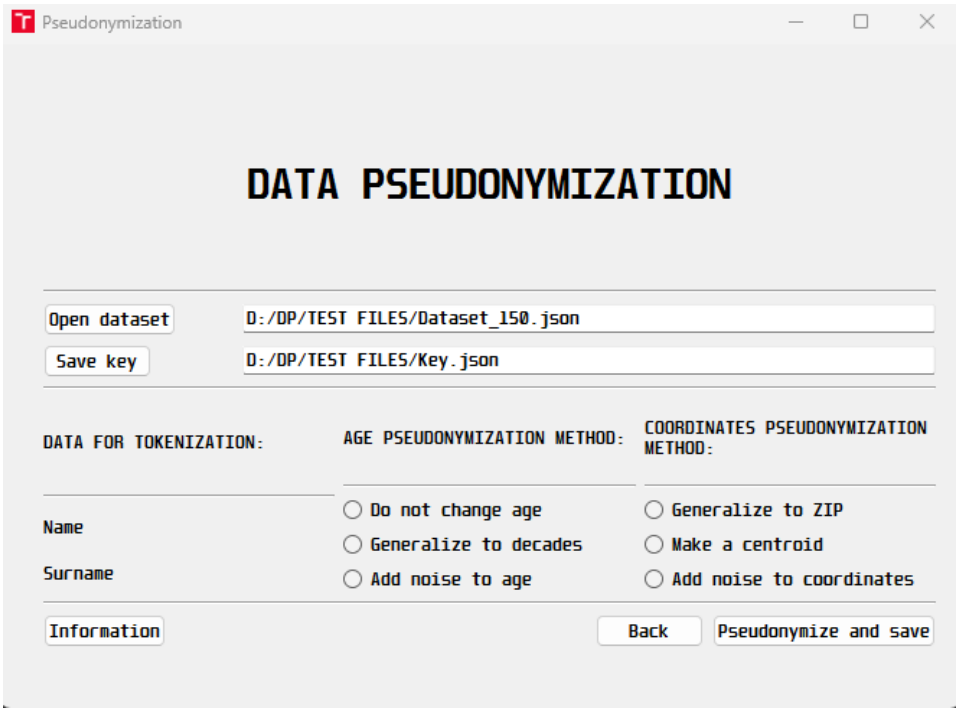


Obr. 2.8: Podmenu aplikace pro výběr pseudonymizace

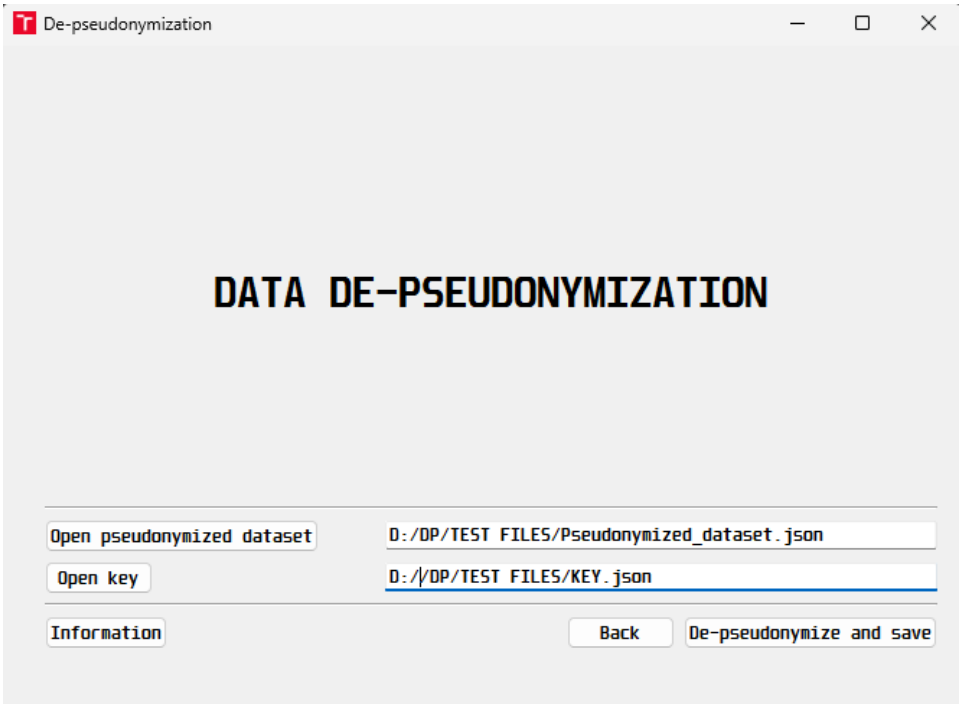
První možnost podmenu pseudonymizace (*Pseudonymization*) otevře okno aplikace pro vytvoření pseudonymizovaného datasetu (obrázek 2.9). Jako první v tomto okně uživatel načte svůj dataset s osobními údaji a následně vybere místo uložení pro svůj pseudonymizační klíč, který je potřebný pro případnou de-pseudonymizaci dat. Po zadání cest k těmto souborům má uživatel možnost vybrat, jak bude naloženo se záznamy obsahující věk a souřadnice. Pakliže dataset obsahuje atribut jména nebo příjmení, jsou tyto hodnoty použity pro vytvoření tokenu za pomoci funkce hash se solí. Pokud se v datasetu vyskytuje emailová adresa, její uživatelské jméno je také tokenizováno a je ponechána pouze doména. Možnosti jak pseudonymizovat věk je buď jej ponechat v nezměněném formátu, přidat k němu šum nebo jej zobecnit po dekádách. V případě pseudonymizace souřadnic je na výběr z vytvoření centroidu, přidání šumu a zobecnění souřadnic na jedno poštovní směrovací číslo. Po pseudonymizaci těchto dat je na konec souboru (pseudonymizačního klíče) vložena hodnota hash vypočtena ze souboru obsahujícího pseudonymizovaná data, jenž slouží pro kontrolu integrity dat (zda pseudonymizovaný soubor nebyl pozměněn).

Druhá možnost podmenu (tzn. tlačítko *De-pseudonymization*) přesune uživatele do okna (viz obrázek 2.10), ve kterém lze provést zpětnou operaci pseudonymizace a na základě klíče, kterým disponuje, zpětně vygenerovat z pseudonymizovaného datasetu zpět originální dataset obsahující veškeré osobní údaje. Za předpokladu, že se hash uložený v klíči shoduje s tím, který je vytvořen ze souboru s pseudonymizovanými daty. Uživatel tak zadá pouze dvě cesty, jednu k pseudonymizačními

klíči a druhou cestu k pseudonymizovanému datasetu a následně lze provést de-pseudonymizaci těchto dat.



Obr. 2.9: Okno aplikace zajišťující pseudonymizaci dat



Obr. 2.10: Znázornění okna aplikace pro de-pseudonymizace

2.3 Implementované de-identifikační metody v aplikaci

V aplikaci je využito několika de-identifikačních metod. Jak je z předchozí kapitoly zřejmé, implementované metody anonymizace jsou základní, k-anonymity a diferenciálního soukromí. Jako pseudonymizační techniky jsou použity metody tokenizace, hash otisků za použití přidané náhodné soli. K pseudonymizačním technikám lze také přidat náhodný šum k záznamům o věku a záznamům obsahující souřadnice. Níže v této kapitole budou detailněji popsány metody, které se k de-identifikaci osobních údajů v aplikaci používají.

2.3.1 Základní anonymizační techniky

Prvními implementovanými anonymizačními technikami, jenž v aplikaci slouží k de-identifikaci dat jsou základní techniky. Použití těchto technik samo o sobě nelze považovat za dostatečnou anonymizaci. V aplikaci jsou pouze demonstrativní a mají pouze informativní charakter, jak mohou být určité záznamy anonymizovány. Aplikace dokáže z těchto základních technik maskovat uživatelské jméno emailové adresy (na pevně stanovenou délku) s tím, že doménová část této adresy zůstane zachována. Další technikou je generalizace (zobecnění) záznamu jedince o věku s tím, že jeho věk bude zobecněn do rozsahu dekád (tzn. jedinec ve věku 33 je zařazen do věkového rozmezí 30-39 let). Poslední zařazenou technikou je zobecnění záznamu obsahující souřadnice. V případě zobecnění souřadnic se jedná o prosté zaokrouhlení na dvě desetinná místa. Pro záznamy obsahující jméno a příjmení není implementována žádná anonymizační technika. Důvodem je skutečnost, že se jedná o přímé identifikátory. A tak pakliže budou tyto atributy vybrány k anonymizaci, ve výsledném datasetu budou zastoupeny hodnotou *None*. Obdobně je tomu tak se záznamy pohlaví subjektů, ty zde nejsou také nijak anonymizovány.

Na obrázku (2.11) níže lze vidět strukturu jednoho vybraného záznamu před a po provedení všech těchto základních anonymizačních technik.

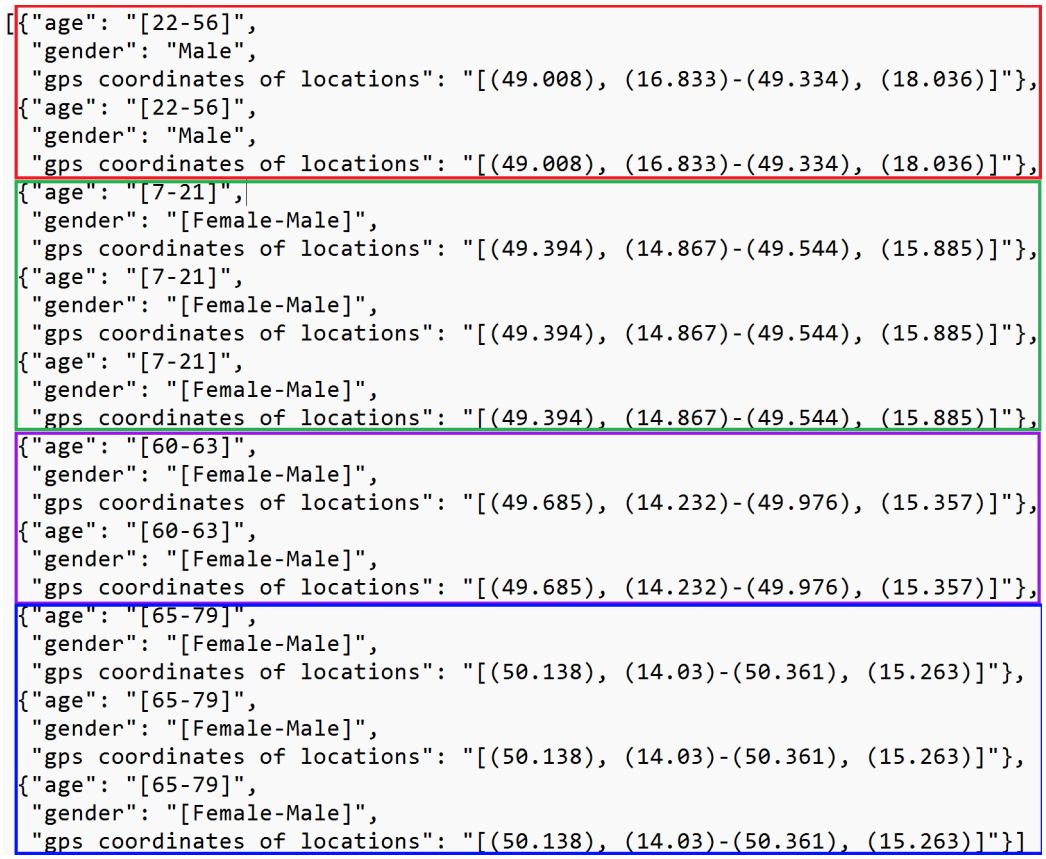
```
[{"Name": "Dalibor",
  "Surname": "Šimek",
  "Email": "daši22@mail.com",
  "Age": 22,
  "Gender": "Male",
  "GPS coordinates of locations": "(48.82460332908224, 16.695888795748868);
                                  (48.990049668232125, 17.080034923926487);
                                  (49.256946344371464, 16.630724418743668);
                                  (49.14631405326232, 17.239521269650595);
                                  (48.822999453145016, 16.519506415542512)"},

{"Email": "XXXXXXXXXX@mail.com",
 "Age": "20-29",
 "Gender": "Male",
 "GPS coordinates of locations": "(48.82, 16.70);
                                  (48.99, 17.08);
                                  (49.26, 16.63);
                                  (49.15, 17.24);
                                  (48.82, 16.52)"}]
```


Obr. 2.11: Záznam před a po provedení základních anonymizačních technik

2.3.2 K-anonymita

V druhé implementované metodě, v metodě k-anonymity, je využíváno multidimenzionálního Mondrian mechanismu. Výhoda tohoto mechanismu spočívá v jeho schopnosti anonymizovat data ve více než jedné dimenzi (jednom atributu) a je schopen lépe zachovat strukturu dat. Mechanismus rozděluje data do skupin s minimálním počtem záznamů "k". Může se tedy stát, že budou skupiny i o několik málo záznamů větší, než uživatel v aplikaci zadá. Pro příklad uvedený níže (obrázek 2.12) platí, že testovací dataset obsahoval deset záznamů o subjektech osobních údajů se strukturou celého jména, věku, pohlaví a souřadnic (pro zjednodušení byl vypočítán jeden bod z pěti souřadnic a ten byl ještě zaokrouhlen) a na tomto datasetu byla provedena anonymizace pomocí metody k-anonymity s parametrem k=2. Jak lze vidět na anonymizovaném datasetu, ačkoli byl parametr k=2, tak se zde nachází i skupiny o třech záznamech. Metoda mondrian spočívá v principu rozdělení dat do skupin a následně sumarizuje nebo generalizuje hodnoty atributů v těchto skupinách tak, aby byla zachována k-anonymita. Příklad byl anonymizován ve třech dimenzích, jedna s věkem, druhá s body souřadnic a třetí s pohlavím subjektu. Veškerá data byla po provedení k-anonymity rozdělena do čtyř skupin. Jediná červená skupina obsahuje záznamy pouze subjektů se stejným pohlavím, jinak jak lze vidět, skupiny jsou rozděleny na základě věku a souřadnic. Aplikace automaticky detekuje výskyt přímých identifikátorů (jména a příjmení) a z výsledného datasetu je odstraňuje.

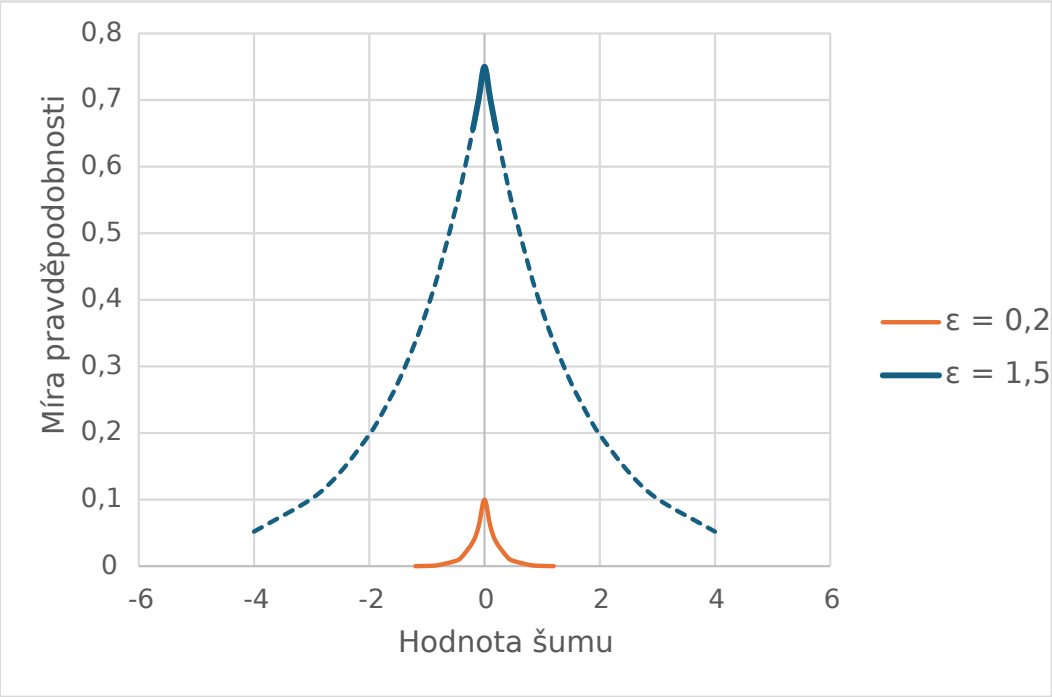


Obr. 2.12: Data po aplikaci k-anonymity (pro k=2)

2.3.3 Diferenciální soukromí

Poslední anonymizační metodou je diferenciální soukromí, ve které aplikace přidává aditivní náhodný šum k souřadnicím a k věku. V praxi lze využít několika mechanismů, nicméně v aplikaci je implementována funkce Laplaceova šumu. Tento šum má symetrické rozložení, které odpovídá Laplaceově distribuci s vrcholem na střední hodnotě a rychlým poklesem směrem k nule na obou stranách. Tento tvar je charakterizován hustotou pravděpodobnosti, která se postupně snižuje se vzdáleností od střední hodnoty. Na obrázku níže (2.13) jsou vyobrazeny funkce použité v aplikaci. Jak bylo uvedeno dříve v práci, parametr ϵ ve schématu diferenciálního soukromí určuje, jak moc může se měnit výstup v reakci na změnu jednoho vstupu. Vyšší hodnota znamená vyšší míru flexibility vůči změnám v datech, což vede k menšímu množství šumu potřebnému k dosažení požadované úrovně soukromí. V první funkci, kde je $\epsilon = 1{,}5$, je výsledný šum menší, protože je vyšší tolerance vůči změnám v datech, což umožňuje použití menšího množství šumu pro dosažení stejné úrovně soukromí. Naopak, ve druhé funkci, kde je $\epsilon = 0{,}2$, je tolerance k změnám v datech nižší, což znamená, že je vyžadováno více šumu pro dosažení stejné úrovně soukromí. To vede k většímu množství přidaného šumu a zdánlivě většímu šumu

výsledku, ačkoli parametr ϵ má menší hodnotu. Znamená to tedy, že $\epsilon = 1{,}5$ byl použit pro šum přidávaný k hodnotám souřadnic, neboť i malá změna na pozici desetinného čísla (např. v setinách) vede ke znatelné změně pozice souřadnic (použité hodnoty z grafu jsou z intervalu [-0,2; 0,2], což značí plná čára a přerušovaná značí pouze další potencionální průběh funkce). Interval takto malých čísel byl vybrán na základě skutečnosti, že změna souřadnic o 0,2 připadá na změnu bodu o 20 až 30 kilometrů (tento údaj je platný pro oblast ČR, kilometry se liší v závislosti na poloze bodu na zeměkouli). Druhý případ, $\epsilon = 0{,}2$ byl použit pro věk a výsledné přidané hodnoty šumu nabývají hodnot celých čísel s podmínkou, že přidaný vygenerovaný šum nesmí být větší jak 10. Změna o 10 let lze považovat už za značné zkreslení výsledného datasetu.



Obr. 2.13: Použité funkce Laplaceova šumu

Výstupem z této metody je anonymizovaný dataset, kde jsou přičteny vygenerované hodnoty šumu. Pakliže uživatel v aplikaci označí k anonymizaci data obsahující emailové adresy nebo pohlaví, tak tyto hodnoty budou v případě emailové adresy zamaskovány a hodnoty pohlaví zůstanou nepozměněny. V případě, že uživatel nějaké data neoznačí, budou automatický vymazány z datasetu, obdobně jako jméno a příjmení. Na příkladu níže (obrázek 2.14) lze vidět příklad anonymizovaného záznamu metodou diferenciálního soukromí. Z příkladu vyplývá, že přidaný šum k věku byl 1 a ke každé souřadnici byl přidán šum v rozmezí -0,2 až 0,2.

```
[{"Name": "Dalibor",
  "Surname": "Šimek",
  "Email": "daši22@mail.com",
  "Age": 22,
  "Gender": "Male",
  "GPS coordinates of locations": "(48.82460332908224, 16.695888795748868);
                                  (48.990049668232125, 17.080034923926487);
                                  (49.256946344371464, 16.630724418743668);
                                  (49.14631405326232, 17.239521269650595);
                                  (48.822999453145016, 16.519506415542512)"},

{"Email": "XXXXXXXXXX@mail.com",
 "Age": 23,
 "Gender": "Male",
 "GPS coordinates of locations": "(48.904969795009755, 16.776255261676386);
                                  (48.79004966823212, 16.880034923926488);
                                  (49.05694634437146, 16.43072441874367);
                                  (49.20300428058533, 17.296211496973605);
                                  (48.94110191210027, 16.63760887449777)"}]
```


Obr. 2.14: Záznam před a po provedení diferenciálního soukromí

2.3.4 Pseudonymizace a de-pseudonymizace

Jako poslední implementovanou de-identifikační metodou je pseudonymizace, ke které je doplněna i část s de-pseudonymizací. Ta dokáže pomocí klíče zpětně získat originální vstupní data z pseudonymizovaných dat. V aplikaci, jak je vidno na obrázku 2.9, slouží k tokenizaci jméno a příjmení (záleží na skutečnosti, zda a jaké záznamy obsahuje zdrojový dataset). V případě, že se ve vstupním souboru dat nachází oba záznamy (myšleno jméno a příjmení), jsou tyto dva záznamy sloučeny do jednoho a následně je řetězec doplněn o náhodně vygenerovanou sůl (jedná se o 16 bytovou sůl v hexadecimálním tvaru). Řetězec je následně zahashován funkcí SHA (Secure Hash Algorithm) o délce 384 bitů. Výsledkem je 384 bitový token. Tento token je uložen do výsledného pseudonymizovaného souboru. Záznamy jména, příjmení a vygenerované soli jsou uloženy do souboru s pseudonymizačním klíčem. Údaje o pohlaví jsou ve výchozím nastavení aplikace neměněny a jsou tak přepsány do výstupního souboru. Data obsahující celou emailovou adresu jsou tokenizovány takovým způsobem, že uživatelské jméno se zamění za jméno ve formátu "userXXXXX", kde znaky "XXXXX"značí náhodně vygenerované číslo z rozsahu 0 až 99 999 a doménová část adresy zůstane nepozměněná. Původní uživatelské jméno je uloženo do klíče.

U pseudonymizace věku lze vybírat z, jak již bylo zmíněno, ponechání věku beze změny, generalizace věku a přidání šumu k věku. Za podmínky, že je vybrána generalizace věku, věk je upraven, stejně jako v případě základních anonymizačních technik, do skupin po dekádách. V posledním případě, kdy je do věku přidáván náhodný šum je využito Laplaceova šumu. Ať vybere uživatel jakoukoliv metodu,

vždy je do pseudonymizačního klíče uložen identifikátor, který značí jaká metoda byla vybrána a záznam o hodnotě věku (pouze u metody přidání šumu ukládá aplikace velikost šumu).

Metodami implementovanými pro pseudonymizaci souřadnic byly vybrány přepočet souřadnic na jeden centroid čili zobecnění souřadnic do jednoho společného bodu, jeden bod definovaný poštovním směrovacím číslem a přidáním Laplaceova šumu do každé jednotlivé souřadnice. I zde platí, že daná vybraná metoda je zapsána do pseudonymizačního klíče spolu s přepsanými hodnotami souřadnic z původního souboru. Jen v případě přidání šumu k souřadnicím ukládá aplikace v klíči hodnoty šumu.

Poslední funkcí uplatněná v pseudonymizaci slouží k ověření integrity dat, zda nebyly pseudonymizované záznamy pozměněny. Po samotné pseudonymizaci dat se na konec pseudonymizačního klíče uloží hodnota hash, která je vypočítána právě z pseudonymizovaných dat. Hodnota hash je vytvořena stejným algoritmem jako hash jména a příjmení, akorát s rozdílem že se k vytvoření nepřidává sůl. Použitým algoritmem je tedy SHA-384.

Na obrázcích níže (2.15, 2.16) lze vidět původní a pseudonymizovaná data, ve kterých byl věk zobecněn na dekády, emailová adresa byla tokenizována, celé jméno bylo zahashováno a souřadnice byly zgeneralizovány na jeden záznam poštovního směrovacího čísla. Druhý obrázek zobrazuje pseudonymizační klíč obsahující veškerá potřebná data, která jsou nezbytná pro správnou de-pseudonimizaci do budoucna.


```
[{"Name": "Dalibor",
  "Surname": "Šimek",
  "Email": "daši22@mail.com",
  "Age": 22,
  "Gender": "Male",
  "GPS coordinates of locations": "(48.82460332908224, 16.695888795748868);
                                  (48.990049668232125, 17.080034923926487);
                                  (49.256946344371464, 16.630724418743668);
                                  (49.14631405326232, 17.239521269650595);
                                  (48.822999453145016, 16.519506415542512)"}]


[{"Email": "user43222@mail.com",
 "Age": "20-29",
 "Gender": "Male",
 "GPS coordinates of locations": "691 72",
 "Full name":
"9e60cb6ad1c5defcdf381e5ae6f422f743006cc0ba4e2dfd8bcc6402c3605a6d21bccf7fec7
ca00f2eb0f1936c61ecaa"}]
```


Obr. 2.15: Původní a pseudonymizovaná data

```
[{"Name": "Dalibor",
  "Surname": "Šimek",
  "Email name": "daši22",
  "Age method": "Generalize age",
  "age": 22,
  "GPS method": "GPS to ZIP",
  "gps coordinates of locations": "(48.82460332908224, 16.695888795748868);
                        (48.990049668232125, 17.080034923926487);
                        (49.256946344371464, 16.630724418743668);
                        (49.14631405326232, 17.239521269650595);
                        (48.822999453145016, 16.519506415542512)",
  "Hashing salt": "d88665a8c6613dadf6493c14ef1b96eb"},
 {"hash":
"0fa80f9330b5135204ba26262c681ecd072558631bd8b92f28e70bf7afdf1d524f9ad747055
1be488bfe03819fa4a804"}]
```


Obr. 2.16: Pseudonymizační klíč k předchozím pseudonymizovaným datům s hodnotou hash k ověření integrity

De-pseudonymizace dat využívá pseudonymizovaných dat a pseudonymizačního klíče k vytvoření původního datasetu. Jako první po načtení dat z těchto souborů se vypočítává hash z pseudonymizovaných dat. Za podmínky, že se hodnota tohoto vypočítaného hashe shoduje s hodnotou hash uložené v pseudonymizačním klíči postupuje aplikace v hledání shodného záznamu mezi soubory a uložení tohoto záznamu do nového souboru. Namapování mezi záznamy probíhá za pomoci vypočítání hodnoty hash ze jména a příjmení, ke kterému se přidá uložená hodnota hashovací soli, a hledá se odpovídající hodnota hash celého jména v pseudonymizovaném souboru. Za předpokladu že se tyto hodnoty shodují, tak se převezmou hodnoty z klíče a přepíší se do výsledného souboru. Výsledek de-pseudonymizace z příkladu z obrázků 2.15 a 2.16 zobrazuje obrázek 2.17.


```
[{"Name":"Dalibor",
  "Surname":"Šimek",
  "Gender":"Male",
  "Age":22,
  "Email":"daši22@mail.com",
  "GPS coordinates of locations":"(48.82460332908224, 16.695888795748868);
                                  (48.990049668232125, 17.080034923926487);
                                  (49.256946344371464, 16.630724418743668);
                                  (49.14631405326232, 17.239521269650595);
                                  (48.822999453145016, 16.519506415542512)"}]
```


Obr. 2.17: De-pseudonymizovaná data

2.4 Popis funkcí aplikace

Obsahem této kapitoly je popis vybraných částí kódu, tak aby uživatel měl lepší povědomí o fungování aplikace. Jsou zde rozebrány stěžejní části a vysvětleny tak, aby bylo jednoduché se v případě potřeby v kódu orientovat a lehce vyznat.

2.4.1 Generování testovacího datasetu

První kód aplikace popisuje generování souřadnic. Prvně bylo nutné definovat hranice generování lokalizačních bodů. Tyto hranice určují souřadnice krajních bodů České republiky. Po vygenerování náhodné dvojice souřadnic (pomocí knihovny random) mezi těmito krajními body je pomocí knihovny GeoPy určeno, zda souřadnice náleží bodu na území České republiky. Pokud ano, získá se z těchto souřadnic adresa a vygeneruje se polynom (vygenerování polynomu je zajištěno knihovnou Shapely) o velikosti 50 kilometrů v okolí adresy (viz řádek 25 až 29 výpisu 2.1). K hodnotám souřadnic je tak přičteno 25 kilometrů na každou světovou stranu a celkově tyto kroky vytvářejí oblast kolem aktuálního bodu s rozměry přibližně 50 x 50 kilometrů. V opačném případě, že se vygenerované souřadnice nenachází na území České republiky, se vygeneruje nová dvojice souřadnic. A probíhá celý cyklus znovu. V kódu se nachází i ošetření případu, kdy knihovna GeoPy nedokáže obsloužit požadavek uživatele na dotaz lokace souřadnic. Knihovna využívá OpenStreetMap ke hledání adres a požadavky knihovny GeoPy jsou omezeny a jeden požadavek za vteřinu. Z tohoto důvodu může generování souřadnic chvilku zabrat.

Výstupem této funkce je vrácení hodnot souřadnic definující polynom. Kód generující souřadnice je uveden níže na výpisu 2.1.

Výpis 2.1: Generování GPS souřadnic

```
def generateGPSCoordinates():
    min_latitude = 48.5525
    max_latitude = 51.055702
    min_longitude = 12.091389
    max_longitude = 18.858889
    while True:
        global lat, lon
        lat = random.uniform(min_latitude, max_latitude)
        lon = random.uniform(min_longitude, max_longitude)
        try:
            geolocator = Nominatim(user_agent="211820")
            location = geolocator.reverse(f"{lat},{lon}")
        except GeocoderTimedOut:
            print("Geocoding service timed out. Retrying...")
            time.sleep(0.5)
            continue
        location_address = location.address
        address_split = location_address.split(", ")
        zip_code = address_split[-2]
        zip_code_numeric = zip_code.replace(" ", "")
        if address_split[-1] == "Česko" and
            address_split[0].isnumeric() and
            zip_code_numeric.isdigit():
            # 0.225 is approx. 25km
                coordinates = Polygon([
    (lat + 0.225, lon + (100 / (111.320 * math.cos(lat))) / 4),
    (lat + 0.225, lon - (100 / (111.320 * math.cos(lat))) / 4),
    (lat - 0.225, lon + (100 / (111.320 * math.cos(lat))) / 4),
    (lat - 0.225, lon - (100 / (111.320 * math.cos(lat))) / 4)])
            return coordinates
```

Souřadnice předávané v návratové hodnotě v předchozí funkcí jsou využity při generování pěti náhodných bodů v okolí adresy bydliště, které mají simulovat například nějaký pohyb nebo oblíbené lokace generovaného subjektu osobních údajů.

Výpis 2.2: Generování náhodných souřadnic

```
def polygonRandomPoints(poly, num_points):
    min_x, min_y, max_x, max_y = poly.bounds

    points = []
    while len(points) < num_points:
        random_point = Point([random.uniform(min_x, max_x),
                              random.uniform(min_y, max_y)])
        if random_point.within(poly):
            points.append(random_point)
    return points
```

Poslední funkcí je samotná funkce, která zapisuje množinu dat do souboru, na základě předaných parametrů, které uživatel zvolí v aplikaci. Pro generování náhodných dat je využita knihovna Faker. Jako první program generuje jméno a příjmení (pro liché záznamy platí, že generovaná osoba je mužského rodu a pro sudé záznamy se jedná o osoby ženského rodu). Při generování jmen se ukládá i pohlaví subjektu údajů. Poté program vygeneruje rodné číslo, ze kterého je následně dopočítán věk k aktuálnímu dni a tento věk se uloží. Následuje vytvoření emailové adresy, jenž se skládá z prvních dvou písmen jména, dvou písmen příjmení a věku. Zbylé záznamy (pěti dvojic souřadnic v okolí adresy) jsou brány z výše zmíněných funkcí.

Po vytvoření všech záznamů se na základě parametrů (políček, které uživatel aplikace vyplnil) vypíší záznamy do slovníku a vytvoří se databáze na místě a s požadovaným jménem, které uživatel sám určí. Po uložení celé databáze se aplikace přenese zpět do hlavního menu.

2.4.2 Základní anonymizace

Základní anonymizační techniky pracují na jednoduché bázi, kdy se hodnoty ze souboru pouze upraví nepodmíněně nehledě na jakékoliv duplicity či kolize ve výsledném datasetu. V této metodě se využívá knihoven re (regular expressions) a json. Knihovna re slouží pro práci s textovými řetězci a knihovna json pro prácí s JSON soubory. Kód pro tyto operace lze vidět níže na výpisu 2.3. V případě věku se využívá celočíselného dělení pro správné přiřazení dekády, do které věk subjektu spadá. Emailová adresa se dělí na dvě části rozdělené znakem "@"(zavináče). Uživatelská část je pak zamaskována symboly "X"a doménová část je zachována. Záznamy obsahující souřadnice využívá jednoduchého zaokrouhlení na dvě desetinná místa.

Výpis 2.3: Základní anonymizační techniky

```
def generalizeAge(age):
    decade = (age // 10) * 10
    return f"{decade}-{decade + 9}"
def maskEmail(email):
    name, domain = email.split("@")
    name = "XXXXXXXXXX"
    return f"{name}@{domain}"
def roundGPS(coordinates):
    def roundNumber(number):
        return "{:.2f}".format(float(number.group()))
    rounded_string = re.sub(r"\d+\.\d+",
                    lambda x: roundNumber(x), coordinates)
    return rounded_string
```

2.4.3 K-anonymity

Následující funkce implementuje algoritmus Mondrian pro aplikaci k-anonymity na dataset. Z aplikace funkce přebírá několik parametrů. Jsou to cesta k souboru se záznamy osobních údajů, parametry značící které záznamy jsou požadovány k anonymizaci a které k potlačení, a posledním parametrem je parametr "k"vypovídající o minimálním počtu záznamů v jednotlivé skupině. Nejprve je nutné načíst jednotlivé parametry do proměnných a v případě souřadnic bude nutné rozdělit je na dvojice (v případě anonymizační techniky k-anonymity jsou souřadnice ještě zaokrouhleny na tři desetinná místa). Funkce pro extrakci souřadnic a vypočítání jejich centroidu se zaokrouhlením na tři desetinná místa jsou uvedeny ve výpisu 2.4. Tyto funkce jsou použity i v některých dalších implementovaných de-identifikačních metodách v aplikaci.

Výpis 2.4: Funkce pro rozdělení souřadnic a vypočítání centroidu se zaokrouhlením

```
def extractCoordinates(coords_str):
    coordinates = coords_str.split(';')
    xy_coords = [tuple(map(float,
            coord.strip('()').split(','))) for coord in coordinates]
    return xy_coords

def calculateCentroid(coordinates):
    x_coords = [coord[0] for coord in coordinates]
    y_coords = [coord[1] for coord in coordinates]
    centroid_x = np.mean(x_coords)
    centroid_x = round(centroid_x, 3)
    centroid_y = np.mean(y_coords)
    centroid_y = round(centroid_y, 3)
    return f"({centroid_x}), ({centroid_y})"
```

V k-anonymitě je implementována také funkce pro maskování emailové adresy, která se shoduje s funkcí uvedenou v základní anonymizaci. Pro implementaci techniky k-anonymity (multidimenzionální Mondrian metody) bylo nutné vytvořit několik funkcí a využít knihoven NumPy a Pandas. První funkcí je *summarized* (viz řádky 1 až 8 níže uvedeného výpisu 2.5), která slouží k sumarizaci dat v každém oddíle pomocí intervalové reprezentace. Pro každý záznam, jenž má být anonymizován provádí kroky řazení data ve sloupci odpovídajícím záznamům, pakliže hodnota prvního a posledního záznamu v řazených datech není stejná, vytvoří intervalovou reprezentaci hodnot v tomto sloupci. Na výstupu vrací dataset s intervalovými hodnotami. Druhou funkcí je *anonymize* (řádky 9 až 18 výpisu 2.5). Tato funkce bere jako vstup oddíl a seznam hodnocení (ranks) pro jednotlivé záznamy. Z oddílu je vybrána dimenze (atribut) s nejvyšším počtem unikátních hodnot. Na základě dimenze funkce řadí dataset. Poté rozdělí dataset na dvě části tak, aby byl každý poddataset (skupina) aspoň k-anonymní ("k"parametr udává v aplikaci uživatel). Třetí funkce, na řádcích 19 až 25 výpisu 2.5, *mondrian* nejprve vytváří slovník ranks, který pro každý záznam (kvazi-identifikátor) obsahuje počet unikátních hodnot v daném sloupci. Slovník ranks je seřazen podle počtu hodnot od nejvyššího po nejnižší. Poté volá funkci *anonymize*. Výsledek je vrácen jako výstup.

Výpis 2.5: Funkce aplikující k-anonymitu

```
def summarized(partition, dim):
    for qi in qis:
        partition = partition.sort_values(qi)
        if partition[qi].iloc[0] != partition[qi].iloc[-1]:
            s = f"[{str(partition[qi].iloc[0])}-
                        {str(partition[qi].iloc[-1])}]"
            partition[qi] = [s] * partition[qi].size
    return partition
def anonymize(partition, ranks):
    dim = ranks[0][0]
    partition = partition.sort_values(dim)
    median = partition[dim].count() // 2
    left_side = partition[:median]
    right_side = partition[median:]
    if len(left_side) >= k and len(right_side) >= k:
        return pd.concat([anonymize(left_side, ranks),
                          anonymize(right_side, ranks)])
    return summarized(partition, dim)
def mondrian(partition):
    ranks = {}
    for qi in qis:
        ranks[qi] = len(set(partition[qi]))
        ranks = sorted(ranks.items(), key=lambda t: t[1],
                       reverse=True)
        return anonymize(partition, ranks)
```

2.4.4 Diferenciální soukromí

Poslední implementovanou anonymizační metodou je diferenciální soukromí. I v této metodě aplikace pracuje se souřadnicemi, takže bylo nutné využít stejné funkce jako v případě k-anonymity (funkce *extractCoordinates*). Při tvoření funkcí byly použity knihovny NumPy a Pandas. Jak již bylo řečeno výše v práci, diferenciální soukromí implementované v aplikaci využívá Laplaceova šumu pro přidání náhodné hodnoty k souřadnicím a i k věku. Funkce přidávající šum jsou znázorněny na výpisu níže (2.6). V prvním případě, jak název napovídá, funkce přidává šum k souřadnicím. Využívá se zde knihovny NumPy pro generování hodnot Laplaceova šumu (řádek 7). Generování šumu je omezeno na interval od -0,2 do 0,2 a následně se šum připočítá

k souřadnicím a zapíše do seznamu. Kód pro generování šumu k věku je obdobný, pouze s rozdíly jiného epsilon a podmínkou, aby případný šum nebyl větší než 10 a splňoval podmínku, aby součet šumu a věku nebyl záporný (řádek 18 výpisu 2.6).

Výpis 2.6: Funkce přidávající Laplaceův šum k souřadnicím a věku

```
1  def addingLaplaceNoiseGPS(coords):
2      sens = 1.0
3      eps = 1.5
4      b = sens / eps
5      noisy_coords = []
6      for coord in coords:
7          noise = np.random.laplace(0, b, 1)[0]
8          noise = np.clip(noise, -0.2, 0.2)
9          noisy_coord = (coord[0] + noise, coord[1] + noise)
10         noisy_coords.append(noisy_coord)
11     return noisy_coords
12
13 def addingLaplaceNoiseAge(age):
14     while True:
15         sens = 1.0
16         eps = 0.2
17         noise = np.random.laplace(0, sens / eps, 1)
18         if abs(noise) < 10 and noise + age > 0:
19             return int(noise.item()) + age
```

2.4.5 Pseudonymizace

Kód provádí pseudonymizaci dat a zahrnuje několik funkcí pro různé operace jako přidání šumu k GPS souřadnicím, generalizaci věku, hashování plného jména a tokenizace uživatelských částí emailových adres. Využívá se knihoven NumPy, Pandas, GeoPy, Secrets, re, random, json a haslib. První funkcí (*HashFullName* kód vygeneruje náhodou sůl, přidá jí k celému jménu a provede hash s výstupem v hexadecimálním tvaru dlouhém 384 bitů). V kódu se také používají dříve zmíněné funkce pro získání souřadnic (*extractCoordinates*, převod věku na dekády (*generalizeAge*), přidání šumu do souřadnic(*addingLaplaceNoiseGPS* s rozdílem v návratové hodnotě, nyní funkce vrací hodnotu zašumělých dat a hodnoty jednotlivých šumů), přidání šumu k věku (*addingLaplaceNoiseAge*, zde návratová hodnota nabývá také zašumělých hodnot a hodnot šumu). Vypočítávání centroidu souřadnic se v pseudonymizaci

také vyskytuje, pouze s rozdílem délky zaokrouhlení, nyní je centroid zaokrouhlen na pět desetinných míst. Nově implementovanými funkcemi jsou níže uvedené (výpis 2.7). Pseudonymizace emailu nyní probíhá, na rozdíl od předchozího maskování jednoduchým nahrazením za řetězec symbolů "X", nahrazením uživatelského jména za řetězec složený ze slova "user"a k němu náhodně vygenerovaného čísla z rozmezí 0 až 99 999. Druhá nově vzniklá funkce vytváří ze souřadnic jeden záznam s poštovním směrovacím číslem. V tomto případě se nejdříve vytvoří jeden centroid a z něho se pomocí knihovny GeoPy vyhledá odpovídající poštovní směrovací číslo, pakliže se toto číslo nenajde, přičte se k souřadnicím jedna setina a vyhledává se znovu. Poslední funkcí je vypočtení hodnoty hash z celého souboru pseudonymizovaných dat, kód využívá obdobného algoritmu jako vypočtení hash ze jména subjektu (SHA-384 bez přidané soli) a předává hodnotu hash v hexadecimálním tvaru.

Výpis 2.7: Vybrané funkce pseudonymizace

```
def hashFullName(full_name):
    hashing_salt = secrets.token_hex(16)
    full_name_with_salt = f"{full_name}{hashing_salt}"
    hashed_data = hashlib.sha384(
                            full_name_with_salt.encode()).hexdigest()
    return hashed_data, hashing_salt
def maskEmail(email):
    def generateRandomNumber():
        while True:
            number = random.randint(0, 99999)
            if number not in rng_email:
                rng_email.add(number)
                return number
    random_number = generateRandomNumber()
    email_name, domain = email.split("@")
    masked_name = "user" + str(random_number)
    return f"{masked_name}@{domain}", email_name
def calculateFileHash(file_path):
    hasher = hashlib.new('sha384')
    with open(file_path, 'rb') as f:
        while chunk := f.read(4096):
            hasher.update(chunk)
    return hasher.hexdigest()
```

2.4.6 De-pseudonymizace

Před provedením zpětného procesu pseudonymizace je nejprve zkontrolována integrita dat pomocí porovnání dvou hodnot hash. Jedna hodnota je uložena v pseudonymizačním klíči a druhá se pokaždé vypočítává z nahrávaných pseudonymizovaných dat. Za podmínky, že se tyto dvě hodnoty shodují, proces pokračuje. V opačném případě bude uživateli znázorněna výstraha, že s daty bylo manipulováno a nelze tak provést de-pseudonymizaci. Při shodě hodnot jsou z pseudonymizačního klíče vypočítávány hodnoty hash uložených záznamů (jméno, příjmení, hashovací sůl) a na základě těchto hodnot se v pseudonymizovaných datech hledají shodné záznamy. Nalezená shoda znamená přepsání (případné přepočítání věku a souřadnic na základě použitých metod) hodnot z pseudonymizačního klíče do nově vznikajícího souboru s de-pseudonymizovanými daty.

2.5 Spuštění aplikace

Pro spuštění aplikace je nutné mít na koncovém zařízení nainstalovaný Python, či v případě jeho absence je nutné jej stáhnout. Z důvodu stažení programovacího jazyka Python, nutných knihoven a práce s GeoPy knihovnou je nutné disponovat připojením k internetu. Bez připojení internetu nelze zaručit bezchybný chod aplikace. Při nainstalovaném Pythonu stačí v příkazové řádce v adresáři aplikace zadat následující příkaz:

```
D:\Application>pip install -r requirements.txt
```

Tímto příkazem se nainstalují veškeré potřebné balíčky využité v aplikaci. Následně lze celá aplikace spustit v libovolném vývojovém prostředí podporující programovací jazyk Python nebo jednoduchým příkazem z příkazové řádky:

```
D:\Application>py application.py
```

2.6 Porovnání implementovaných metod

Každá z vybraných implementovaných metod má své klady i zápory. Jak již bylo řečeno dříve, některé metody, jako je například metoda diferenciálního soukromí, se hodí na více obsáhle databáze záznamů. Poměrně značné zdržení aplikace je viditelné u generování obsáhlých datasetů (části generování souřadnic bydliště, jenž slouží k vytvoření pětice souřadnic) a v pseudonymizační technice zobecnění souřadnic na

poštovní směrovací číslo. Toto zpomalení je dané využitím volně dostupné knihovny GeoPy a jejím omezením počtu dotazů na server za vteřinu.

Základní anonymizační techniky mají v aplikaci pouze demonstrativní účel, tudíž je nelze považovat za účinné a odolné proti re-identifikaci. Avšak z pohledu základních metod anonymizace je lze považovat za použitelné.

Oproti tomu, druhou v aplikaci implementovanou anonymizační metodu využívající k-anonymitu (s multidimenzionálním Mondrian algoritmem) lze označit jako účinný nástroj pro ochranu soukromí v datasetech s vysokou dimenzionalitou (velkým množství identifikátorů). Využívá se dělení datového prostoru do skupin tak, aby bylo zaručeno, že každá skupina obsahuje alespoň "k"záznamů, což minimalizuje identifikovatelnost jednotlivců. Za nevýhody této metody lze označit v některých případech větší obecnost záznamů v jednotlivých skupinách a tím i případnou ztrátu vypovídající hodnoty výstupního datasetu.

Další aplikovanou metodou byla metoda diferenciálního soukromí s využitím aditivního Laplaceova šumu. Tato metoda přidávala náhodnou hodnotu šumu k vybraným záznamům (věku a souřadnicím). Velikost zkreslení, resp. velikost šumu závisí na vhodně zvoleném parametru ϵ. Tento přístup by byl aplikovatelný na různé typy dat (především na číselné) a není omezen strukturou datasetu, což znamená, že může být použit ve velkém množství situací, kde by bylo nutné využít ochrany osobních údajů. Negativum metody tkví v klíčovém výběru hodnoty parametru ϵ, tak aby aditivní šum byl adekvátní svojí velikostí k výsledným anonymizovaným datům (aby nebyla příliš zkreslená, ale zároveň aby stále splňovala anonymitu subjektů).

Poslední implementovanou technikou de-identifikace byla pseudonymizace osobních údajů. K dosažení pseudonymizace dat je možné využít několika kombinací metod pseudonymizace. Metoda kombinuje některé dříve použité metody s rozdílem zachování informací o pseudonymizaci ve výsledném pseudonymizačním klíči. Nedostatky této metody spočívají ve stejných negativech jako v diferenciálním soukromí (v případě využití aditivního šumu), či možným velikým zobecněním záznamů. K těmto nevýhodám se přidává případná ztráta soukromí, pakliže bude pseudonymizační klíč prozrazen.

Závěr

Diplomová práce s názvem Anonymizace dat v uživatelské aplikaci pojednává především o ochraně osobních údajů jako je jméno, příjmení, emailová adresa, věk, geolokační údaje apod. A to způsobem ochrany pomocí anonymizace nebo pseudonymizace těchto záznamů.

Zprvu v práci byly rozebrány pojmy týkající se osobních údajů nejen z pohledu právního rámce a celkové de-identifikace dat. Po uvedení pojmů se v teoretické části definuje de-identifikace a jsou rozebrány různé základní techniky anonymizace a pseudonymizace. Na základní de-identifikační techniky navazují pokročilé anonymizační techniky, které se používají i v praxi.

V druhé, praktické, části diplomové práce byla vytvořena aplikace s grafickým uživatelským prostředím, ve které je implementován program pro generování testovacích datasetů s osobními údaji. Výhodou tohoto generátoru je, že uživatel aplikace může definovat jakou strukturu má mít dataset (jaké osobní údaje budou vygenerovány) a počet těchto záznamů v datasetu. Stěžejní částí aplikace je naprogramování vybraných de-identifikačních metod jako jsou základní anonymizační techniky, k-anonymita, diferenciální soukromí, pseudonymizace a zpětná de-pseudonymizace dat.

Při zpětné analýze vytvořeného programu bylo dosaženo závěru, že v případě de-identifikace osobních údajů záleží na spustě faktorů (jako je např. množství a struktura dat), které je nutné při implementování de-identifikačních metod zohlednit. Je také nutné počítat s určitou informační ztrátou de-identifikovaných dat.

Na základě této práce lze říci, že nelze sestavit jednu komplexní univerzální aplikaci, která zaručí de-identifikaci veškerých osobních údajů. Je nutné si vždy definovat cíl de-identifikace a dle toho volit použité metody na základě jejich silných a slabých stránek. A je vždy na zpracovateli osobních údajů, aby rozhodl zda jsou výsledná data opravdu de-identifikovaná, nespadají nadále pod pojem osobní údaj, a proto se na ně nevztahuje ochranný rámec právní úpravy ochrany osobních údajů.

Literatura

[1] ČESKO. Usnesení č. 2/1993 Sb., Usnesení předsednictva České národní rady o vyhlášení LISTINY ZÁKLADNÍCH PRÁV A SVOBOD jako součástí ústavního pořádku České republiky In: *Zákon pro lidi* [online]. © AION CS 2010-2023 [cit. 2023-12-01]. Dostupné z URL:https://www.zakonyprolidi.cz/cs/1993-2

[2] Nařízení Evropského parlamentu a Rady (EU) 2016/679 ze dne 27. dubna 2016 o ochraně fyzických osob v souvislosti se zpracováním osobních údajů a o volném pohybu těchto údajů a o zrušení směrnice 95/46/ES (Obecné nařízení o ochraně osobních údajů). In: *Úřední věstník Evropské unie* [online]. Kapitola I, článek 4 [cit. 2023-10-10]. Dostupné z URL:https://eur-lex.europa.eu/legal-content/CS/TXT/HTML/?uri=CELEX:32016R0679#d1e1481-1-1

[3] ČESKO. Zákon č. 110/2019 Sb., Zákon o zpracování osobních údajů In: *Zákon pro lidi* [online]. © AION CS 2010-2023 [cit. 2023-10-10]. Dostupné z URL:https://www.zakonyprolidi.cz/cs/2019-110

[4] MARQUES, J. a BERNARDINO J. Analysis of Data Anonymization Techniques. In: *Proceedings of the 12th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2020) - Volume 2: KEOD*. s 235-241, ISBN: 978-989-758-474-9. [online]. [cit. 2023-10-10]. Dostupné z URL:https://www.scitepress.org/Papers/2020/101423/101423.pdf

[5] GARFINKEL, S. De-Identification of Personal Information In: *NISTIR 8053*. [online]. [cit. 2023-10-12]. Dostupné z URL:https://nvlpubs.nist.gov/nistpubs/ir/2015/nist.ir.8053.pdf

[6] SIMPLILEARN. What Is Data Collection: Methods, Types, Tools. [online]. 1 Sep 2023. [cit. 2023-10-12]. Dostupné z URL:https://www.simplilearn.com/what-is-data-collection-article

[7] Pseudonymization according to the GDPR [definitions and examples]. dataprivacymanager.net. In: *Blog, Data Breach, Data Privacy, DPO*. [online]. 02/11/2021 [cit. 2023-10-12]. Dostupné z URL:https://dataprivacymanager.net/pseudonymization-according-to-the-gdpr/

[8] GARDHOUSE, K. Direct and Indirect Personal Identifiers: What are they? [online]. April 8, 2023. [cit. 2023-10-12]. Dostupné z URL:https://www.private-ai.com/2023/04/08/personal-identifiers/

[9] Guidance on Anonymisation and Pseudonymisation An Coimisiún um Chosaint Sonraí. [online]. June 2016. [cit. 2023-10-12]. Dostupné z URL:https://www.dataprotection.ie/sites/default/files/uploads/2022-04/Anonymisation%20and%20Pseudonymisation%20-%20latest%20April%202022.pdf

[10] Article 29 Data Protection Working Party (2014) Opinion 05/2014 on Anonymisation Techniques [online]. 0829/14/EN WP216 10 April 2014. [cit. 2024-03-10]. Dostupné z URL:https://ec.europa.eu/justice/article-29/documentation/opinion-recommendation/files/2014/wp216_en.pdf

[11] KNOCHE, P. Factual anonymity of microdata from household and person related surveys - the release of microdata for scientific purposes. In: *Proceedings of the International Symposium on Statistical Confidentiality*. pages 407–413. 1993.

[12] KREHLING, L. De-Identification Guideline. in: *Western Information Security and Privacy Research Laboratory Technical Report WL-2020-01*. [online]. Western University, Canada, 2020. [cit. 2023-10-13]. Dostupné z URL:https://whisperlab.org/technical-reports/de-identification-guideline-WL-2020-01.pdf

[13] Re-identification risk. nist.gov. In: *Glossary*. [online]. [cit. 2023-10-13]. Dostupné z URL:https://csrc.nist.gov/glossary/term/re_identification_risk

[14] SWEENEY, L. Simple demographics often identify people uniquely. In: *Data Privacy Working Paper 3*. [online]. Carnegie Mellon University Pittsburgh, 2000 [cit. 2023-10-13]. Dostupné z URL:https://dataprivacylab.org/projects/identifiability/paper1.pdf

[15] Data Anonymisation: Attribute Suppression ntu.edu.sg [online]. [cit. 2023-10-13]. Dostupné z URL:https://libguides.ntu.edu.sg/c.php?g=927336&p=6698844

[16] RICHMAN, A. Data Anonymization vs Data Masking: Definitions/Use Cases in: *K2VIEW BLOG*. [online]. May 14, 2023. [cit. 2023-10-13]. Dostupné z URL:https://www.k2view.com/blog/data-anonymization-vs-data-masking/

[17] CFI Team. Data Anonymization. [online]. [cit. 2023-10-13]. Dostupné z URL:https://corporatefinanceinstitute.com/resources/business-intelligence/data-anonymization/

[18] SANTONJA, C. D. 6 Personal Data Anonymization Techniques You Should Know About. [online]. 11/03/2022. [cit. 2023-10-13]. Dostupné z URL:https://blog.pangeanic.com/6-personal-data-anonymization-techniques

[19] EVERLÖF, A. GDPR pseudonymization techniques. [online]. May 31, 2018. [cit. 2023-10-13]. Dostupné z URL:https://alexewerlof.medium.com/gdpr-pseudonymization-techniques-62f7b3b46a56

[20] ENISA. Pseudonymisation techniques and best practices. [online]. November 2019. [cit. 2023-10-13]- Dostupné z URL:https://www.enisa.europa.eu/publications/pseudonymisation-techniques-and-best-practices

[21] Best practices and techniques for pseudonymization. isc2.org In: *Blog*. [online]. 14 June 2021. [cit. 2023-10-13]. Dostupné z URL:https://blog.isc2.org/isc2_blog/2021/06/best-practices-and-techniques-for-pseudonymization.html

[22] SHELDON, R. message authentication code (MAC). [online]. [cit. 2023-10-13]. Dostupné z URL:https://www.techtarget.com/searchsecurity/definition/message-authentication-code-MAC

[23] BRETT, Daniel. Symmetric vs. Asymmetric Encryption: What's the Difference? In: *Trentonsystems*. [online]. May 4, 2021 9:30:00 AM. [cit. 2023-10-28]. Dostupné z URL:https://www.trentonsystems.com/blog/symmetric-vs-asymmetric-encryption

[24] MAJKUT, Kristof. k-Anonymization Module Prepared for Change. Final Dissertation. University of Trento: Department of Information Engineering and Computer Science. University of Twente: Faculty of Electrical Engineering, Mathematics and Computer Science. [online]. 2022/2023. [cit. 2023-12-28]. Dostupné z URL:https://essay.utwente.nl/96238/1/Majkut_MA_EEMCS.pdf

[25] DE PASCALE, Daniel a CASCAVILLA, Guiseppe a TAMBURRI, A. Damian a VAN DEN HEUVEL, Willem-Jan. Real-world K-Anonymity applications: The KGen approach and its evaluation in fraudulent transactions. In: *Information Systems*, Volume 15. [online]. May 2023. [cit. 2023-11-25]. Dostupné z URL:https://www.sciencedirect.com/science/article/pii/S0306437923000297

[26] MACHANAVAJJHALA, Ashwin a GEHRKE, Johannes a KIFER, Daniel a VENKITASUBRAMANIAM, Muthramakrishnan L-Diversity: Privacy Beyond k-Anonymity. Cornell University: Department of Computer Science. [online]. [cit. 2024-01-15]. Dostupné z URL:https://personal.utdallas.edu/~muratk/courses/privacy08f_files/ldiversity.pdf

[27] Data Anonymisation and L-Diversity. informationwithinsight.com. [online]. March 12, 2019. [cit. 2023-11-23]. Dostupné z URL:https://informationwithinsight.com/2019/03/12/data-anonymisation-and-l-diversity/

[28] YE, Yang a DENG, Qiao a WANG, Chi a LV, Dapeng a Liu, Yu a FENG, Jianhua BSGI: An Effective Algorithm towards Stronger l-Diversity 2008. DOI: 10.1007/978-3-540-85654-2_3.

[29] Mohammad-Reza Zare-Mirakabad, ZARE-MIRAKABAD, Mohammad-Reza a JANTAN, Aman a BRESSAN, Stéphane. Clustering-Based Frequency l-Diversity Anonymization. In: *Lecture Notes in Computer Science*, vol 5576. 2009. [cit. 2024-01-23]. Dostupné z URL:https://doi.org/10.1007/978-3-642-02617-1_17

[30] JIAN-MIN, Han a TING-TING, Cen a HUI-GUN, Yu An Improved V-MDAV Algorithm for l-Diversity. In: *2008 International Symposiums on Information Processing*, pp. 733-739. [online]. 2008. [cit. 2023-12-23]. DOI: 10.1109/ISIP.2008.110

[31] KAJOL, P. A study on T-Closeness over K-anonymization Technique for Privacy Preserving in Big Data. [online] In: *International Journal of Engineering Research & Technology (IJERT) Volume 08, Issue 05* 21-05-2019 [cit. 2023-11-23]. Dostupné z URL:https://www.ijert.org/a-study-on-t-closeness-over-k-anonymization-technique-for-privacy

[32] Data Anonymisation and L-Diversity. informationwithinsight.com. [online]. March 12, 2019. [cit. 2023-11-23]. Dostupné z URL:https://www.cs.purdue.edu/homes/ninghui/papers/t_closeness_icde07.pdf

[33] Wang R, Zhu Y, Chen TS et al. WANG,Rong a ZHU, Yan a CHEN, Tung-Shou a CHANG, Chin-Chen. Privacy-preserving algorithms for multiple sensitive attributes satisfying t-closeness. In: *Journal of Computer Science and Technology 33*, pp. 1231–1242. November 2018. [cit. 2023-11-23]. DOI: 10.1007/s11390-018-1884-6

[34] WOOD, A. a ALTMAN, M. a BEMBENEK, A. a BUN, M. a GABOARDI, M. a HONAKER, J. a NISSIM, K. a O'BRIEN, D. a STEINKE, T. a VADHAN, S. Differential Privacy: A Primer for a Non-Technical Audience. [online] In: *Vanderbilt Journal of Entertainment & Technology Law, Vol. 21, No. 17, 2018* February 2019 [cit. 2024-03-15] Dostupné z URL:https://ssrn.com/abstract=3338027

[35] HARNESS, Y. What is Differential Privacy? In: *Blog*. [online]. September 09, 2022. [cit. 2024-03-23]. Dostupné z URL:https://dualitytech.com/blog/what-is-differential-privacy/

[36] FATHIMA, Shaistha. Differential Privacy — Noise adding Mechanisms. In: *Becoming Human: Artificial Intelligence Magazine*. [online]. October 1, 2020. [cit. 2024-02-10]. Dostupné z URL:https://becominghuman.ai/differential-privacy-noise-adding-mechanisms-ede242dcbb2e

[37] Homomorphic Encryption. chain.link [online]. December 21, 2023. [cit. 2024-02-11]. Dostupné z URL:https://chain.link/education-hub/homomorphic-encryption

[38] ASWANI, Nimish Jain a CHERUKURI, Kumar. Revisiting Fully Homomorphic Encryption Schemes. Vellore Institute of Technology: School of Information Technology and Engineering (SITE). [online]. [cit. 2024-03-15]. Dostupné z URL:https://arxiv.org/pdf/2305.05904

[39] ČESKO. § 91 odst. 1 zákona č. 127/2005 Sb., o elektronických komunikacích a o změně některých souvisejících zákonů (zákon o elektronických komunikacích) - znění od 1. 1. 2024 In: *Zákon pro lidi* [online]. © AION CS 2010-2024 [cit. 2024-03-10]. Dostupné z URL:https://www.zakonyprolidi.cz/cs/2005-127#p91-1

[40] BALE, C. D. a FISHER, J. L. a SCHNEIDER, M. J. a WEBER, S. a CHANG, S. Legally Anonymizing Location Data Under the GDPR. [online]. July 25, 2023 [cit. 2024-03-14]. DOI: 10.13140/RG.2.2.17076.73609/1

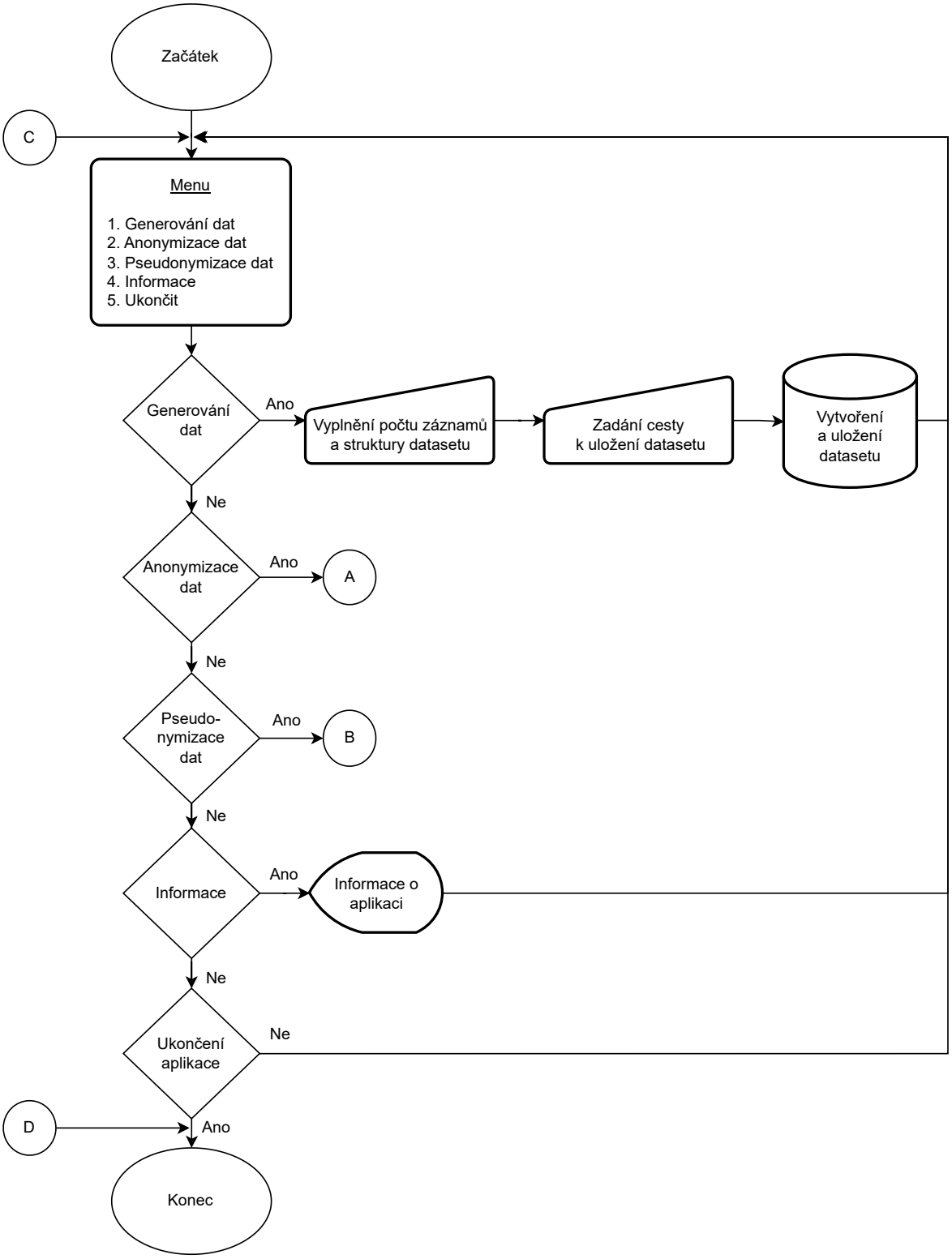
[41] Welcome to GeoPy's documentation! geopy.readthedocs.io [online]. [cit. 2023-11-28]. Dostupné z URL:https://geopy.readthedocs.io/en/stable/

Seznam symbolů a zkratek

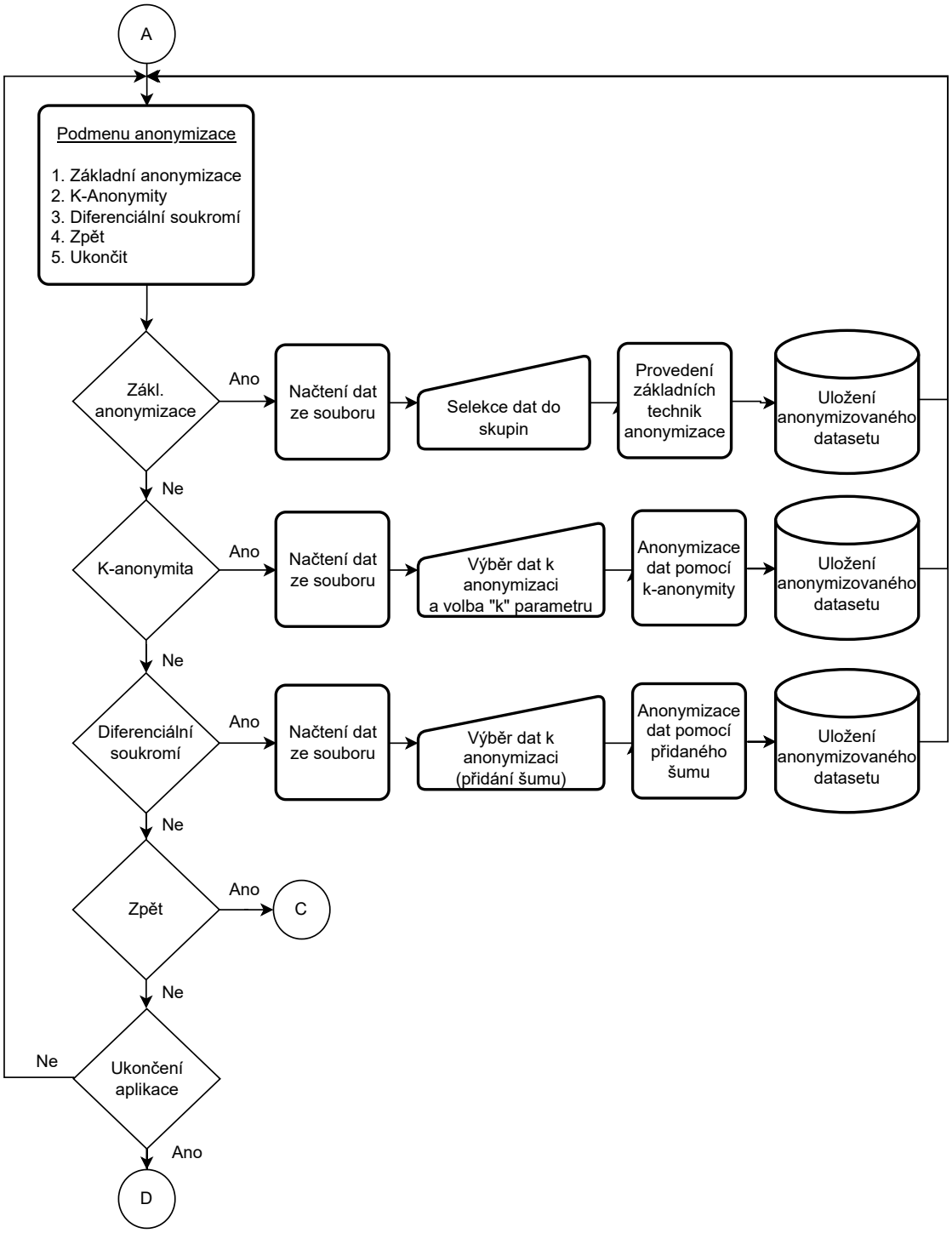
AES	Advanced Encryption Standard
CDP	Central Differential Privacy
CODIP	Complete Disjoint Projections
CSV	Comma-separated values
DES	Data Encryption Standard
ECC	Elliptic-Curve Cryptography
EMD	Earth Mover's Distance
ENISA	The European Union Agency for Cybersecurity
FHE	Fully Homomorphic Encryption
GDPR	Genaral Data Protection Regulation
GDP	Global Differential Privacy
HMAC	Hash-based message authentication code
IP	Internet Protocol
JSON	JavaScript Object Notation
LFHE	Leveled Fully Homomorphic Encryption
LDP	Local Differential Privacy
MAC	Message Authentication Code
PHE	Partially Homomorphic Encryption
RSA	Rivest Shamir Adleman
SHA	Secure Hash Algorithm
SWHE	Somewhat Homomorphic Encryption
V-MDAV	Variable-size Maximum Distance to Average Vector
WYSIWYG	What You See Is What You Get
XML	Extensible Markup Language
3DES	Triple Data Encryption Standard

Seznam příloh

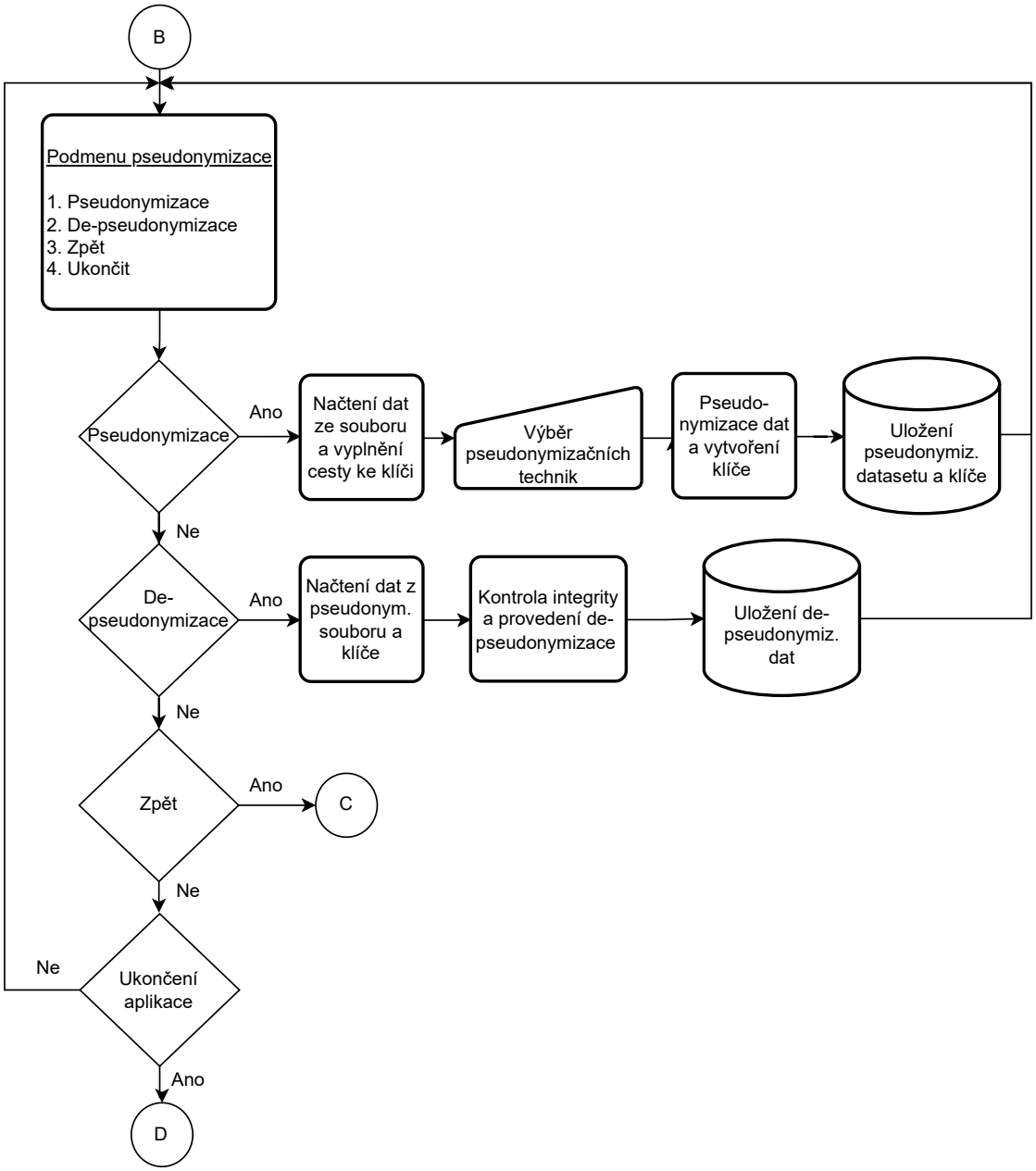
A Vývojový diagram aplikace



Obr. A.1: Vývojový diagram hlavního menu aplikace



Obr. A.2: Vývojový diagram podmenu anonymizace



Obr. A.3: Vývojový diagram podmenu pseudonymizace

B Obsah elektronické přílohy

V elektronické příloze je obsažen jeden adresář, dva textové soubory a několik souborů s kódy aplikace. V kořenovém adresáři se nachází složka "Testing datasets", která obsahuje použitelné testovací datasety s různými počty záznamů. Dále se zde nachází také textový dokument s názvem "README.txt"v němž je napsán návod na spuštění aplikace a druhý textový soubor ("requirements.txt") s definicí seznamu balíčků (packages), které jsou potřebné pro běh aplikace.

```
/ ..... kořenový adresář přiloženého archivu
├── Testing datasets ..... testovací soubory s různým množství fiktivních záznamů
│   ├── Dataset_5.json
│   ├── Dataset_150.json
│   └── Dataset_1000.json
├── anonymizationMenuWindow.py
├── application.py ..... hlavní soubor pro spuštění aplikace
├── basicAnonymInfoWindow.py
├── basicAnonymization.py
├── basicAnonymWindow.py
├── depseudonymization.py
├── depseudonymizationInfoWindow.py
├── depseudonymizationWindow.py
├── differentialPrivacy.py
├── differentialPrivacyInfoWindow.py
├── differentialPrivacyWindow.py
├── generatorDataset.py
├── generatorInfoWindow.py
├── generatorWindow.py
├── informationWindow.py
├── kAnonymity.py
├── kAnonymityInfoWindow.py
├── kAnonymityWindow.py
├── mainWindow.py
├── pseudonymization.py
├── pseudonymizationInfoWindow.py
├── pseudonymizationMenuWindow.py
├── pseudonymizationWindow.py
├── README.txt ..... návod pro spuštění aplikace
├── requirements.txt ..... soubor se seznamem balíčků a jejich verzí
└── VUT.svg ..... logo aplikace
```