DataSentics

Tvorba podkladové databáze identifikátorů osob k analýze dat o dočasné pracovní neschopnosti a invaliditě

Project VS/2018/0380 "Development of microsimulation tools for social insurance projection (DEMTOP)" has been funded with support from the European Commission. This study reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Project VS/2018/0380 Development of microsimulation tools for social insurance projection (DEMTOP) and this document has received financial support from the European Union Programme for Employment and Social Innovation "EaSI" (2014-2020). For further information please consult: http://ec.europa.eu/social/easi The information contained in this study does not necessarily reflect the official position of the European Commission.

Contents

1. Účel a struktura dokumentu 3
2. Shrnutí 4
3. Zdrojová data a jejich příprava 6
4. Spojení databází VZ, SEE a ISPV 13
5. Výpočet podobnostního skóre 18
6. Výpočet unikátnostního skóre 32
7. Finální přiřazení 35
8. Výsledky propojení 37
9. Ovládání nástroje 40

Disclaimer

V případě zásahu do kódu softwaru hrozí ztráta funkčnosti nebo nesprávné fungování a na potíže vzniklé v důsledku tohoto jednání se nevztahuje uživatelská podpora. Doporučujeme tedy do kódu nezasahovat. Výjimkou jsou pouze explicitně popsané úpravy v tomto manuálu.

1. Účel a struktura dokumentu

Tato dokumentace je součástí projektu „Rozvoj dynamického mikrosimulačního modelu důchodového systému MPSV – II.“, části „Tvorba podkladové databáze identifikátorů osob k analýze dat o dočasné pracovní neschopnosti a invaliditě“.

Předmětem plnění této veřejné zakázky bylo vytvoření databáze (a nástroje generujícího tuto databázi), která propojuje zaměstnance uvedené v databázích SEE20, SEE21 a SEE22 s osobami uvedenými v databázi STATMIN VZ. Databáze nemají stejný identifikátor a ztotožnění tedy není jednoduchá databázová operace. Výsledné propojení je pak na straně Ministerstva práce a sociálních věcí (dále MPSV) podkladem k analýze dat dočasné pracovní neschopnosti a invaliditě.

Dokumentace popisuje

- zvolený přístup k propojení jednotlivců v jednotlivých databázích
- výsledky provedeného propojení a
- uživatelskou příručku k ovládání nástroje generujícího propojení jednotlivců v databázích.

Výsledky provedeného propojení jsou uvedeny v kapitole 8. Uživatelská příručka je obsažena v kapitole 9.

Popis zvoleného přístupu je rozdělen do těchto částí

- vstupní databáze a příprava dat (kapitola 3)
- základní spojení databází (kapitola 4)
- výpočet podobnostního (základního) skóre (kapitola 5)
- výpočet unikátnosti (kapitola 6)
- výběr párů na základě podobnostního skóre a unikátnosti (kapitola 7)

V následující kapitole uvádíme shrnutí zvoleného přístupu.

2. Shrnutí

Nástroj DELNEZ[1] slouží pro propojení zaměstnanců ve dvou zdrojích – propojují se jedinci v databázích SEE 20-22 na jedince v databázi STATMIN VZ. Vzhledem k tomu, že tyto dva zdroje nesdílejí stejné jednoznačné identifikátory, je nutné určit propojení na základě algoritmu vyjadřující jednak podobnost jedinců a jednak unikátnost propojení.

Výstupem nástroje DELNEZ je databáze propojující identifikátory z SEE 20-22 a identifikátory ze STATMIN VZ včetně dodatečných informací o propojení – podobnostní skóre, unikátnostní skóre a informace o vybraných propojených dvojicích.

Vstupem do nástroje jsou dva zdroje, které propojujeme, tedy databáze SEE 20, 21 a 22 a databáze STATMIN VZ a dále pomocné databáze INEP, upravená databáze ISPV a další pomocné tabulky.

Databáze STATMIN ANOD není uvedena mezi primárními zdroji, protože oproti STATMIN VZ neobsahuje další informace pro identifikaci osob v databázích SEE20-22. Zároveň předpokládáme konzistenci údajů mezi STATMIN VZ a STATMIN ANOD, protože do tohoto algoritmu vstupují zkontrolované a vyčištěné databáze (součást nástroje pro tvorbu modelových bodů pro model NEMO).

Databáze STATMIN ANOD je využita pro generování statistik, viz kapitola Export výsledků.

Algoritmus propojení sestává z následujících kroků:

1. **Tvrdé databázové propojení** dle roku, pohlaví, roku a měsíce narození a okresu. Tímto vybereme pouze relevantní množinu kandidátů, kteří vstupují do dalšího zpracování.
2. **Výpočet podobnostního skóre** na základně dostupných socio-demografických údajů, a především pak i na základě historické struktury nemocnosti jedince (resp. jeho neschopenek a vyloučené doby). Samotné socio-demografické údaje, které máme v databázích k dispozici, omezí relevantní kandidáty na několik desítek, ale ve většině případů nevedou k jednoznačné identifikaci. Proto je nutné využít dostupnou informaci o obdobích, kdy byl zaměstnanec na neschopence. Tato informace je velmi silný identifikátor pro jedince s bohatou historií nemocí. Z tohoto důvodu je na tuto informaci kladen větší důraz právě pro jedince s víceletou historií v databázích SEE 20-22 oproti základním socio-demografickým údajům. Naopak pro jedince s krátkou historií je větší důraz kladen na socio-demografické údaje. Výpočet podobnostního skóre je založen na více než deseti specifických skóre a probíhá na několika úrovních granularity, a to na úrovni neschopenka–ELDP, na úrovni IDxROK–IDxROK a na úrovni ID–ID (kde je výsledné podobnostní skóre spočteno).
3. **Výpočet unikátnostního skóre**. Podobnostní skóre dává informaci o tom, jak je jedinec z SEE podobný jedinci ve STATMIN VZ, ale samo o sobě již nedává informaci o tom, jak je toto propojení unikátní, tedy jaká je pravděpodobnost, že jedinec z SEE je totožný s jedincem ze STAMIN VZ a s nikým jiným a naopak (jedinec ze STATMIN VZ je totožný s jedincem z SEE a nikým jiným). Může se tedy stát, že k jednomu jedinci v SEE nalezneme kandidáta ze STATMIN VZ s maximálním podobnostním skóre a zároveň existují další kandidáti s maximálním podobnostním skóre (nebo velmi vysokým). V takovémto případě nemůžeme jedince jednoznačně ztotožnit. Na druhou stranu mohou nastat případy, kdy pro jedince z SEE máme kandidáta ze STATMIN VZ, který sice nemá maximální možné podobnostní skóre, ale je to jediný možný kandidát. Pro tyto účely počítáme unikátnostní skóre, a to na základě jednak distribuce podobnostního skóre pro všechny kandidáty vybraného jednotlivce z SEE a

[1] Databáze ELdp a NEschopenek – Ztotožnění

jednak distribuce podobnostního skóre pro všechny kandidáty vybraného jednotlivce ze STATMIN VZ. Toto skóre vyjadřuje, jaký je rozdíl v podobnostním skóre ve vybrané dvojici a všech dalších kandidátech. Pokud je rozdíl malý (uvažované propojení není unikátní), pak je toto skóre nízké, pokud je rozdíl velký, je toto skóre vysoké.

4. **Identifikace finálních přiřazení** pomocí algoritmu, který prochází všechny možné kombinace a vybírá takovou dvojici, která má maximální kombinaci podobnostního a unikátnostního skóre.

Z pohledu technické architektury je nástroj založený na Sparku, frameworku pro distribuované zpracování velkých dat. Řešení je tak připraveno na v čase se zvětšující vstupní databáze. Spark běží na odděleném linuxovém serveru MPSV a ovládaný je pomocí Excelového rozhraní na windowsovém serveru MPSV. Uživatelské rozhraní umožňuje nastavovat několik parametrů (jako rok běhu, tolerance ve výpočtu skóre apod.) tak, aby bylo možné propojovat databáze pravidelně bez zásahu do kódu a aby bylo možné provádět jednoduché úpravy fungovaní algoritmu. Zároveň je možné spouštět jednotlivé kroky algoritmu samostatně.

3. Zdrojová data a jejich příprava

Hlavní zdroje

Hlavními vstupy pro tvorbu podkladové databáze jsou uvedeny v následujících sekcích.

STATMIN VZ

Databáze STATMIN VZ je souhrnem všech osob, za které byl podán Evidenční list důchodového pojištění.

NÁZEV SLOUPCE	TYP SLOUPCE
ID_VZZAM_AN	dlouhé celé číslo
ID_ELDP	dlouhé celé číslo
ID_OSOBA_AN	dlouhé celé číslo
POHLAVI	binární celé číslo
ROKNAR	celé číslo
PSC	text
KVC	text
OD	datum
DO	datum
DNY	celé číslo
VDOBA	celé číslo
ODOBA	celé číslo
VZ	desetinné číslo
ID_ORG_AN	dlouhé celé číslo
IXYEAR	celé číslo
TYPDOKLADU	celé číslo
DUPLICITY	binární celé číslo
SK	dlouhé celé číslo

SEE20

Databáze SEE20 je souhrnem všech záznamů o ukončené dočasné pracovní neschopnosti na základě "Rozhodnutí o dočasné pracovní neschopnosti".

NÁZEV SLOUPCE	TYP SLOUPCE
DATUM_NAROZENI	text
POHLAVI	text
ID_OSOBY	dlouhé celé číslo
TYP_ZUCTOVATELE	text
NUTS_NESCHOPNEHO	text
LAU_ZAMESTNAVATELE	text
NUTS_LEKARE	text
NESCHOPEN_OD	datum
NESCHOPEN_DO	datum
DIAGNOZA	text
DRUH_ZAMESTNANI	celé číslo
DRUH_NESCHOPNOSTI	text
DUVOD_UKONCENI_NEMOCI	celé číslo
DUPLICITY	binární celé číslo
SK	dlouhé celé číslo

SEE21

Databáze SEE21 je souhrnem všech záznamů s žádostí o dávku otcovské poporodní péče.

NÁZEV SLOUPCE	TYP SLOUPCE
DATUM_NAROZENI	text
POHLAVI	text
ID_OSOBY	dlouhé celé číslo
TYP_ZADATELE	celé číslo
ID_DITETE	dlouhé celé číslo
TYP_ZUCTOVATELE	text
NUTS_OSSZ	text
OD	datum
DO	datum
DRUH_ZAMESTNANI	text
ROK	celé číslo

SEE22

Databáze SEE22 je souhrnem všech záznamů jak za dávkové případy čerpání dlouhodobého ošetřovného, tak případy typu "ošetřování" (za ošetřovanou osobu).

NÁZEV SLOUPCE	TYP SLOUPCE
DATUM_NAROZENI	text
POHLAVI	text
ID_OSOBY	dlouhé celé číslo
TYP_PRIPADU	text
TYP_ZUCTOVATELE	text
NUTS_OSSZ	text
OD	datum
DO	datum
POCET_ZAPLACENYCH_DNU	celé číslo
DENNI_VZ	desetinné číslo
DIAGNOZA	text
TYP_UKONCENI	text
TYP_VZTAHU	celé číslo
ID_PRIPADU	dlouhé celé číslo
ID_PRIPADU_OSETROVANI	dlouhé celé číslo
ROK	celé číslo

Pomocné zdroje

Následující datové zdroje byly využity pro obohacení hlavních zdrojů.

INEP

Databáze INEP představuje unikátní evidenci individuálních údajů, které byly sesbírány výlučně pro vytvoření kvalitních vstupních dat pro mikrosimulační model. Tato databáze představuje zúplnění pravidelně dostupných dat ze STATMIN_VZ a STATMIN_ANOD, a to z obou hledisek.

NÁZEV SLOUPCE	TYP SLOUPCE
ID_OSOBY	dlouhé celé číslo
ROK_NAROZENI	celé číslo
CIS_POHLAVI_ID	text

JE_NA_ZIVU	binární celé číslo
DOBA_POJISTENA	celé číslo
ROK	celé číslo
NDP_PECE_O_DITE	celé číslo
NDP_NEZAMESTNANOST	celé číslo
NDP_STUDIUM	celé číslo
NDP_PECE	celé číslo
NDP_OSTATNI	celé číslo
NDP_VOJENSKA_SLUZBA	celé číslo
DOBA_NEPOJISTENA	celé číslo
VYMEROVACI_ZAKLAD	desetinné číslo
VYMEROVACI_ZAKLAD_OSVC	desetinné číslo
DOBA_VYLOUCENA	celé číslo
SKR_ZAMESTNANEC	binární celé číslo
SKR_OSVC	binární celé číslo
SKR_PECE_O_DITE	binární celé číslo
SKR_NEZAMESTNANOST	binární celé číslo
SKR_STUDIUM	binární celé číslo
SKR_PECE	binární celé číslo
SKR_OSTATNI	binární celé číslo
SKR_NEPOJISTEN	binární celé číslo
DUPLICITY	binární celé číslo
SK	dlouhé celé číslo

Upravené ISPV

Databáze ISPV obsahuje informaci o trvání pracovní neschopnosti (v hodinách) na úrovni ID_VZZAM_AN x ID_OSOBA_AN x IXYEAR v databázi STATMIN VZ.

NÁZEV SLOUPCE	TYP SLOUPCE
ID_VZZAM_AN	dlouhé celé číslo
ID_OSOBA_AN	dlouhé celé číslo
ID_ORG_AN	dlouhé celé číslo
ABSNEMOC	dlouhé celé číslo
MISTOVP	text
ROK	celé číslo

Psc_okres

Databáze psc_okres je převodníkem mezi označením LAU, PSČ a názvem (označením) okresu.

NÁZEV SLOUPCE	TYP SLOUPCE
NAZCOBCE	text
PSC	celé číslo
NAZPOST	text
KODOKRESU	celé číslo
NAZOKRESU	text
NAZOBCE	text
NUTS4	text

Maternity_and_care

Databáze Maternity_and_care obsahuje pravděpodobnosti mateřské pro ženy a ošetřovného pro muže a ženy dle věku.

NÁZEV SLOUPCE	TYP SLOUPCE
AGE	celé číslo
PROB_MATERNITY	desetinné číslo
PROB_CARE_WOMEN	desetinné číslo
PROB_CARE_MEN	desetinné číslo

Změna struktury zdrojových databází

Pro účely pozdějšího spojování databází je třeba udělat nejprve několik úprav. První části uvádíme pouze významnější úpravy. V druhé části je uvedena struktura výsledných databází, kde jsou stručně popsány i drobnější úpravy.

Provedené operace

1) **Přidání měsíce narození do databáze STATMIN VZ**

   Pro přidání měsíce narození do STATMIN VZ je třeba tuto databázi napojit na databázi INEP, kde se informace o měsíci narození osob vyskytuje. Vhledem k tomu, že obě databáze mají stejný identifikátor, je propojení relativně jednoduché.

   Upozorňujeme, že pokud nebude docházet k aktualizaci INEP, bude se úplnost této informace v čase snižovat, protože databáze STATMIN VZ je aktualizována každý rok, a tedy bude obsahovat jedince vstupující na trh práce, kteří v neaktualizované databázi INEP nebudou.

   Pokud informaci nejsem schopni získat vyplňujeme hodnotu 0.

2) **Přiřazení okresu k databázím STATMIN VZ, SEE20, SEE21, SEE22**

   Abychom mohli napojit databáze STATMIN VZ a SEE20/21/22, je třeba sjednotit informace v nich obsažené. V databázi STATMIN VZ máme pole PSČ zaměstnance, kdežto v databázi SEE20 máme LAU_neschopneho v databázi SEE21 a SEE22 máme LAU_OSSZ.

   Nejvyšší společné granularity, které jsme schopni dosáhnout je úroveň okresu.

   K získání označení okresu pro databázi STATMIN VZ, SEE20/21/22 použijeme pomocnou tabulku psc_okres.

   Pokud informaci nejsem schopni získat vyplňujeme hodnotu 0.

   Výsledkem je pole SEE_OKRES, resp. VZ_OKRES, ve kterém je číslo, které označuje okres. V případě, že jedno PSČ spadá do více okresů, je v poli SEE_OKRES, resp. VZ_OKRES uvedeno toto PSČ.

3) **Redukce databází z pohledu roků**

   Jako relevantní pro další kroky přichází v úvahu data od roku 2009 (včetně) do posledního společného roku.

Začátek pracovní neschopnosti v SEE20 je tedy stanoven jako max(1.1.2009; Neschopen_od), čímž posuneme začátek pracovní neschopnosti u těch záznamů, kde byl začátek před 1.1.2009.

Z databáze STATMIN VZ bereme jen ty záznamy, které mají hodnotu IXYEAR >= 2009.

4) Spočtení vyloučené doby

V rámci databáze STATMIN VZ spočteme celkovou vyloučenou dobu sečtením pole VDOBA a ODOBA.

V databázi SEE20 spočteme vyloučenou dobu jako rozdíl ve dnech pole Neschopen_do a Neschopen_od. (Před 30.11.2009 ještě odečteme 1 den, protože do 30.11. 2009 pole Neschopen_do obsahovalo informaci "práce schopen od".)

V databázi SEE21 a SEE22 spočteme vyloučenou dobu jako rozdíl ve dnech pole DO a OD.

Dále pro všechny databáze spočteme sumu vyloučené doby v každém roce, protože v databázi STATMIN VZ může být víc ELDP v jednom roce a stejně tak v databázi SEE20 může být více pracovních neschopností v jednom roce.

5) Změna struktury SEE20/22 na strukturu ID x ROK

Informace v databázi SEE20, resp. SEE22 se objeví až po skončení Dočasné pracovní neschopnosti, resp. dlouhodobého ošetřovného. Je tedy možné, že pole Neschopen_od, resp. OD spadá do jiného roku než pole Neschopen_do, resp. DO. Abychom mohli provést spojení s databází STATMIN VZ, je třeba takovéto záznamy rozdělit dle let. Viz následující obrázek

ID_OSOBY	Neschopen_od	Neschopen_do	Vyloucena_doba
123456	10.10.2017	01.02.2018	113

ID_OSOBY	Neschopen_od	Neschopen_do	Vyloucena_doba
123456	10.10.2017	31.12.2017	82
123456	01.01.2018	01.02.2018	31

6) Spojení databází SEE20, SEE21 a SEE22

Vzhledem k tomu, že struktura databází SEE20, SEE21 a SEE22 je stejná, lze vytvořit jednu souhrnnou databázi, kterou dále budeme nazývat SEE.

7) Spočtení pole ELDPCNT

V databázi STATMIN VZ spočteme pro každé ID osoby počet odevzdaných ELDP v každém roce.

V databázi SEE20/21/22 spočteme počet duplicitních/souběžných záznamů pro každé ID osoby v každém roce. Jako duplicitní/souběžný je pro tyto účely označen takový záznam, kde jsou stejné hodnoty v poli ID osoby, Pohlaví, Rok narození, Měsíc narození, Okres, Neschopen Od, Neschopen Do a Rok.

8) Filtrování TYP_ZUCTOVATELE

Protože v databázi STATMIN VZ nejsou uvedeni OSVČ, musíme je odstranit i z databáze SEE20 a SEE22, kde uvažujeme pouze TYP_ZUCTOVATELE se nerovná „SV“.

9) **Výpočet dnů pracovní neschopnosti v databázi ISPV**

Databáze ISPV obsahuje informaci o délce pracovní neschopnosti v granularitě ID osoby x Rok x ID zaměstnavatele.

Délka pracovní neschopnosti je v této databázi uvedena v hodinách, zatímco Vyloučená doba v databázi STATMIN VZ je uvedena ve dnech.

Pro účely spojení databází je délka pracovní neschopnosti v hodinách přepočtena na celé dny a to tak, že uvažujeme 8hodinovou pracovní dobu. Tedy délka pracovní neschopnosti ve dnech je Délka pracovní neschopnosti v hodinách / 8.

10) **Filtrování TYP_PRIPADU**
Protože v databázi SEE22 jsou jak dávkové případy, tak případy ošetřování, filtrujeme databázi SEE22 tak, že TYP_PRIPADU = „D“ neboli ponecháváme v databázi pouze dávkové případy.

Výsledná struktura databází VZ, SEE a ISPV

Databáze VZ

VZ_IDOS	ID osoby, původně ID_OSOBA_AN
VZ_POHLAVI	Pohlaví osoby, původně Pohlavi
VZ_ROKNAR	Rok narození osoby, původně ROKNAR
VZ_MESNAR	Měsíc narození osoby, vzniká dle bodu 1
VZ_PSC	PSČ zaměstnance, původně PSC
VZ_OKRES	Okres, vzniká dle bodu 2
VZ_OD	Datum od, původně OD
VZ_DO	Datum do, původně DO
VZ_VDOBA	Vyloučená doba, vzniká dle bodu 4
VZ_VDOBASUM	Roční součet vyloučené doby, vzniká dle bodu 4
VZ_ELDPCNT	Počet odevzdaných ELDP, vzniká dle bodu 7
VZ_ROK	Rok záznamu, původně IXYEAR

Databáze SEE

SEE_IDOS	ID osoby, původně ID_OSOBY
SEE_POHLAVI	Pohlaví osoby, původně Pohlavi
SEE_ROKNAR	Rok narození osoby, rok z pole DATUM_NAROZENI
SEE_MESNAR	Měsíc narození osoby, měsíc z pole DATUM_NAROZENI
SEE_LAU	LAU neschopného/LAU_lekare/LAU zaměstnavatele
SEE_OKRES	Okres, vzniká dle bodu 2
SEE_OD	Datum od, původně Neschopen_Od, resp. OD
SEE_DO	Datum do, původně Neschopen_Do, resp. DO
SEE_VDOBA	Vyloučená doba, vzniká dle bodu 3 a 4
SEE_VDOBASUM	Roční součet vyloučené doby, vzniká dle bodu 4
SEE_ELDPCNT	Počet stejných neschopenek, vzniká dle bodu 7
SEE_ROK	Rok záznamu, rok z pole SEE_OD

Databáze ISPV

VZ_IDOS	ID osoby, původně ID_OSOBA_AN
VZ_ROK	Rok záznamu, původně ROK
VZ_DNYPN	Délka pracovní neschopnosti ve dnech, vzniká dle bodu 7

4. Spojení databází VZ, SEE a ISPV

Spojení databáze VZ a ISPV

Nejprve je k databázi VZ připojena databáze ISPV. Platí, že se musí rovnat záznamy v polích VZ_IDOS a VZ_ROK.

Tam, kde se nepodařilo najít odpovídající záznamy, je vyplněna do sloupce VZ_DNYPN hodnota -1.

Spojení databáze VZ a SEE

V rámci tohoto kroku jsou propojeny databáze SEE s VZ a ISPV. Jako relevantní pro spojení přichází v úvahu záznamy, které splňují následující podmínky:

1) SEE_ROK není prázdný
2) SEE_POHLAVI není prázdné
3) VZ_ROK není prázdný
4) VZ_POHLAVI není prázdné
5) VZ_VDOBA > 0

Propojení databází je pak provedeno na základě několika polí:

- pokud platí **SEE_ROK = VZ_ROK**
- a zároveň platí **SEE_POHLAVI = VZ_POHLAVI**
- a zároveň platí **SEE_ROKNAR = VZ_ROKNAR**
- a zároveň platí **SEE_MESNAR = VZ_MESNAR** nebo pokud **SEE_MESNAR = 0** nebo **VZ_MESNAR = 0**
- a zároveň platí **SEE_OKRES = VZ_OKRES** nebo pokud **SEE_OKRES = 0** nebo **VZ_OKRES = 0**
- a zároveň platí **SEE_VDOBASUM <= VZ_VDOBASUM + vdoba_tolerance**

Poslední podmínka vychází z předpokladu, že VZ_VDOBASUM by měla být vždy větší nebo rovna SEE_VDOBASUM (výjimkou je ukončení zaměstnání při pracovní neschopnosti).

Vzhledem k tomu, že kvalita záznamů nemusí být příliš vysoká, uvažujeme parametr vdoba_tolerance, která umožňuje situaci, kdy SEE_VDOBASUM je menší než VZ_VDOBASUM, a to právě o hodnotu parametru vdoba_tolerance.

Tento princip demonstruje obrázek níže

Databáze SEE

SEE_ROK	SEE_POHLAVI	SEE_ROKNAR	SEE_MESNAR	SEE_OKRES	SEE_VDOBASUM
2015	1	1989	3	1	20

Databáze VZ

VZ_ROK	VZ_POHLAVI	VZ_ROKNAR	VZ_MESNAR	VZ_OKRES	VZ_VDOBASUM
2015	1	1989	3	1	18

Pokud vdoba_tolerance = 0, pak nedojde ke spojení těchto záznamů. Pokud vdoba_tolerance = 5, pak dojde ke spojení těchto záznamů.

Pomocné databáze SEE_YEARS a VZ_YEARS

V rámci spojení databází SEE a VZ se mohlo stát, že se spojili jen některé roky, jak demonstruje příklad níže.

Databáze SEE

SEE_IDOS	SEE_ROK	SEE_VDOBASUM
123456	2013	50
123456	2014	10
123456	2015	30
123456	2016	20

Databáze VZ

VZ_IDOS	VZ_ROK	VZ_VDOBASUM
ABCDEF	2012	10
ABCDEF	2013	50
ABCDEF	2014	18
ABCDEF	2015	10
ABCDEF	2016	18

(předpokládáme, že SEE_POHLAVI = VZ_POHLAVI, SEE_ROKNAR = VZ_ROKNAR, SEE_MESNAR = VZ_MESNAR, SEE_OKRES = VZ_OKRES)

Spojená databáze

SEE_IDOS	SEE_ROK	SEE_VDOBASUM	VZ_IDOS	VZ_ROK	VZ_VDOBASUM
123456	2013	50	**ABCDEF**	2013	50
123456	2014	10	**ABCDEF**	2014	18
123456	2016	20	**ABCDEF**	2016	18

(Vdoba_tolerance = 5)

Vidíme, že nedošlo ke spojení let 2012 a 2015 protože SEE_IDOS 123456 nemá záznam pro rok 2012 a v roce 2015 není splněna podmínka VZ_VDOBASUM + vdoba_tolerance >= SEE_VDOBASUM.

Znalost případů, které se nepodařilo spojit se nám bude hodit v rámci počítání skóre jednotlivých spojení. Pro tento účel jsme vytvořili pomocné databáze **SEE_YEARS** a **VZ_YEARS,** a to následujícím způsobem:

VZ_YEARS

Spočteme VĚK osoby jako VZ_ROK – VZ_ROKNAR a připojíme databázi **maternity_and_care**, která obsahuje pravděpodobnosti mateřské a ošetřování člena rodiny. Tam, kde ke spojení nedošlo dosadíme 0.

Pro každý rok v intervalu [2009, max_year[2]] si ukládáme do sloupců hodnotu VZ_VDOBASUM, dále pak PROB_MATERNITY a PROB_CARE_WOMEN / PROB_CARE_MEN (v závislosti na VZ_POHLAVI), pokud VZ_VDOBASUM > 0, VZ_OKRES, VZ_OD, VZ_DO, VZ_DNYPN, VZ_ELDPCNT.

Pokud je hodnota VZ_OD, resp. VZ_DO prázdná vložíme 1.1.1900, jinak vkládáme minimum VZ_OD v daném roce a maximum VZ_DO v daném roce (může se stát, že je v daném roce více záznamů).

V příkladu níže předpokládáme, že se jedná o ženu.

Databáze VZ

VZ_IDOS	VZ_ROK	VZ_VDOBASUM
ABCDEF	2012	10
ABCDEF	2013	50
ABCDEF	2014	18
ABCDEF	2015	10
ABCDEF	2016	18

VZ_YEARS

VZ_IDOS	ABCDEF
VZ_Y_2009	0
VZ_Y_2010	0
VZ_Y_2011	0
VZ_Y_2012	10
VZ_Y_2013	50
VZ_Y_2014	18
VZ_Y_2015	10
VZ_Y_2016	18
VZ_Y_2017	0
VZ_Y_MATERSKA_2009	0,0000
VZ_Y_MATERSKA_2010	0,0000
VZ_Y_MATERSKA_2011	0,0023
VZ_Y_MATERSKA_2012	0,0012
VZ_Y_MATERSKA_2013	0,0005
VZ_Y_MATERSKA_2014	0,0000
VZ_Y_MATERSKA_2015	0,0000
VZ_Y_MATERSKA_2016	0,0003
VZ_Y_MATERSKA_2017	0,0000
VZ_Y_OSETROVNE_2009	0,0000
VZ_Y_OSETROVNE_2010	0,0000
VZ_Y_OSETROVNE_2011	0,1665
VZ_Y_OSETROVNE_2012	0,1426
VZ_Y_OSETROVNE_2013	0,1153
VZ_Y_OSETROVNE_2014	0,0000
VZ_Y_OSETROVNE_2015	0,0000

[2] Definujeme proměnnou max_year jako maximální společný rok databází SEE a VZ

VZ_Y_OSETROVNE_2016	0,0475
VZ_Y_OSETROVNE_2017	0,0000
VZ_Y_OKRES_2009	0
VZ_Y_OKRES_2010	0
VZ_Y_OKRES_2011	22
VZ_Y_OKRES_2012	22
VZ_Y_OKRES_2013	22
VZ_Y_OKRES_2014	0
VZ_Y_OKRES_2015	0
VZ_Y_OKRES_2016	22
VZ_Y_OKRES_2017	0
VZ_Y_OD_2009	01.01.1900
VZ_Y_OD_2010	01.01.1900
VZ_Y_OD_2011	18.07.2011
VZ_Y_OD_2012	01.01.2012
VZ_Y_OD_2013	01.01.2013
VZ_Y_OD_2014	01.01.2014
VZ_Y_OD_2015	01.01.2015
VZ_Y_OD_2016	01.01.2016
VZ_Y_OD_2017	01.01.1900
VZ_Y_DO_2009	01.01.1900
VZ_Y_DO_2010	01.01.1900
VZ_Y_DO_2011	31.08.2011
VZ_Y_DO_2012	31.08.2012
VZ_Y_DO_2013	31.12.2013
VZ_Y_DO_2014	01.01.1900
VZ_Y_DO_2015	01.01.1900
VZ_Y_DO_2016	31.12.2016
VZ_Y_DO_2017	01.01.1900
VZ_Y_DNYPN_2009	0
VZ_Y_DNYPN_2010	0
VZ_Y_DNYPN_2011	-1
VZ_Y_DNYPN_2012	-1
VZ_Y_DNYPN_2013	-1
VZ_Y_DNYPN_2014	0
VZ_Y_DNYPN_2015	0
VZ_Y_DNYPN_2016	18
VZ_Y_DNYPN_2017	0
VZ_Y_ELDPCNT_2009	0
VZ_Y_ELDPCNT_2010	0
VZ_Y_ELDPCNT_2011	1
VZ_Y_ELDPCNT_2012	1
VZ_Y_ELDPCNT_2013	2
VZ_Y_ELDPCNT_2014	0
VZ_Y_ELDPCNT_2015	0
VZ_Y_ELDPCNT_2016	1

VZ_Y_ELDPCNT_2017	0

(Pro lepší zobrazení jsme tabulku transponovali.)

SEE_YEARS

Analogicky jako VZ_YEARS vytvoříme i tabulku SEE_YEARS, přičemž PROB_MATERNITY a PROB_CARE budou nulové.

5. Výpočet podobnostního skóre

Výpočet podobnostního skóre pro jednotlivá ID probíhá na třech úrovních, a to úroveň jednotlivých záznamů neboli jednotlivá ELDP a jednotlivé neschopenky, dále pak na úrovni ID x ROK a konečně na úrovni ID x ID. Výsledné podobnostní skóre je hodnota mezi 0 a 1 pro každou relevantní dvojici z SEE a STATMIN VZ. Toto skóre je spočteno agregací několika dílčích skóre, které vždy hodnotí podobnost jedinců v jedné dimenzi.

Níže popisujeme výpočet dílčích skóre.

SCORE_ROKNAR

Toto skóre vyjadřuje podobnost jedinců z pohledu roku narození a je spočteno následovně:

- pp_roknar = parametr, vyjadřující počet různých roků narození v databázi SEE (50)
- SCORE_ROKNAR nabývá hodnoty 1, pokud VZ_ROKNAR = SEE_ROKNAR
- SCORE_ROKNAR nabývá hodnoty 1/pp_roknar, pokud VZ_ROKNAR = 0 nebo SEE_ROKNAR = 0
- SCORE_ROKNAR nabývá hodnoty 1/pp_roknar^2, pokud VZ_ROKNAR = 0 a SEE_ROKNAR = 0

SCORE_MESNAR

Toto skóre vyjadřuje podobnost jedinců z pohledu měsíce narození a je spočteno následovně:

- pp_mesnar = počet měsíců v roce (12)
- SCORE_MESNAR nabývá hodnoty 1, pokud VZ_MESNAR = SEE_MESNAR
- SCORE_MESNAR nabývá hodnoty 1/pp_mesnar, pokud VZ_MESNAR = 0 nebo SEE_MESNAR = 0
- SCORE_MESNAR nabývá hodnoty 1/pp_mesnar ^2, pokud VZ_MESNAR = 0 a SEE_MESNAR = 0

SCORE_POHLAVI

Toto skóre vyjadřuje podobnost jedinců z pohledu jejich pohlaví a je spočteno následovně:

- pp_pohlavi = počet různých pohlaví (2)
- SCORE_POHLAVI nabývá hodnoty 1, pokud VZ_POHLAVI = SEE_POHLAVI
- SCORE_POHLAVI nabývá hodnoty 1/pp_pohlavi, pokud VZ_POHLAVI = 0 nebo SEE_POHLAVI = 0
- SCORE_POHLAVI nabývá hodnoty 1/pp_pohlavi ^2, pokud VZ_POHLAVI = 0 a SEE_POHLAVI = 0

Příklad: Výše zmíněná skóre mohou vypadat následovně, viz tabulka níže.

Vidíme, že pro rok 2009 je skóre pro měsíc narození 1/12, což odpovídá pravděpodobnosti, že měsíc, který v databázi SEE neznáme je právě ten stejný měsíc jako v databázi VZ. Stejně tak pro rok narození v roce 2011 vidíme skóre 1/50, což odpovídá pravděpodobnosti, že rok narození v databázi VZ je tentýž rok jako v databázi SEE.

SEE IDOS	SEE ROK	SEE ROKNAR	SEE MESNAR	SEE POHLAVI	VZ IDOS	VZ ROK	VZ ROKNAR	VZ MESNAR	VZ POHLAVI
123456	2009	1975	5	1	ABCDEF	2009	1975	5	1
123456	2009	1975	0	1	ABCDEF	2009	1975	5	1

123456	2010	1975	5	1	ABCDEF	2010	1975	5	1
123456	2011	1975	5	1	ABCDEF	2011	0	5	1
123456	2011	1975	5	1	ABCDEF	2011	1975	5	1

SEE IDOS	SEE ROK	VZ IDOS	SCORE ROKNAR	SCORE MESNAR	SCORE POHLAVI
123456	2009	ABCDEF	1	1	1
123456	2009	ABCDEF	1	1/12	1
123456	2010	ABCDEF	1	1	1
123456	2011	ABCDEF	1/50	1	1
123456	2011	ABCDEF	1	1	1

Výše zmíněná skóre (SCORE_ROKNAR, SCORE_MESNAR, SCORE_POHLAVI) jsou dále agregována průměrem na granularitu ID x ROK a dále pak stejným způsobem na úroveň ID x ID.

Z tabulky výše pak dostáváme následující hodnoty

SEE IDOS	SEE ROK	VZ IDOS	SCORE ROKNAR	SCORE MESNAR	SCORE POHLAVI
123456	2009	ABCDEF	1	0,54	1
123456	2010	ABCDEF	1	1	1
123456	2011	ABCDEF	0,51	1	1

A dále pak

SEE IDOS	VZ IDOS	SCORE ROKNAR	SCORE MESNAR	SCORE POHLAVI
123456	ABCDEF	0,84	0,85	1

Průměr, jako způsob agregace, byl zvolen z toho důvodu, že zohledňuje míru nejistoty, která plyne z neznámých nebo nesprávně uvedených údajů, ale zároveň tuto míru nejistoty drží na rozumné úrovni. Nestane se tak, že by pozorování, kde nemáme úplnou informaci o např. měsíci narození, byla diskvalifikována.

Pro výpočet následujících skóre použijeme pomocné databáze VZ_YEARS a SEE_YEARS.

SCORE_OKRES

Toto skóre vyjadřuje podobnost jedinců z pohledu okresu bydliště a je spočteno následovně.

Uvažujeme následující tabulku VZ_YEARS a SEE_YEARS. (Tabulky jsou redukovány tak, aby obsahovaly pouze sloupce užité pro výpočet tohoto skóre. Tabulky jsou transponovány pro jednodušší zobrazení)

VZ_IDOS	ABCDEF
VZ_Y_2009	0
VZ_Y_2010	0
VZ_Y_2011	0
VZ_Y_2012	10
VZ_Y_2013	50

SEE_IDOS	12345
SEE_Y_2009	0
SEE_Y_2010	0
SEE_Y_2011	0
SEE_Y_2012	0
SEE_Y_2013	50

VZ_Y_2014	18
VZ_Y_2015	10
VZ_Y_2016	18
VZ_Y_2017	0
VZ_Y_OKRES_2009	0
VZ_Y_OKRES_2010	0
VZ_Y_OKRES_2011	22
VZ_Y_OKRES_2012	22
VZ_Y_OKRES_2013	22
VZ_Y_OKRES_2014	0
VZ_Y_OKRES_2015	0
VZ_Y_OKRES_2016	22
VZ_Y_OKRES_2017	0

SEE_Y_2014	18
SEE_Y_2015	10
SEE_Y_2016	18
SEE_Y_2017	0
SEE_Y_OKRES_2009	0
SEE_Y_OKRES_2010	0
SEE_Y_OKRES_2011	20
SEE_Y_OKRES_2012	22
SEE_Y_OKRES_2013	22
SEE_Y_OKRES_2014	0
SEE_Y_OKRES_2015	0
SEE_Y_OKRES_2016	20
SEE_Y_OKRES_2017	0

Nejdříve spočteme dílčí skóre pro každý rok SCORE_Y_OKRES_RRRR a to tak, že porovnáváme postupně sloupce VZ_Y_OKRES_RRRR a SEE_Y_OKRES_RRRR (RRRR je označení roku):

- SCORE_Y_OKRES_RRRR nabývá hodnoty 0, pokud VZ_Y_OKRES_RRRR > 0 a SEE_Y_OKRES_RRRR > 0 a zároveň VZ_Y_OKRES_RRRR <> SEE_Y_OKRES_RRRR
- SCORE_Y_OKRES_RRRR nabývá hodnoty 1, pokud VZ_Y_OKRES_RRRR = SEE_Y_OKRES_RRRR nebo pokud VZ_Y_OKRES_RRRR = 0 nebo SEE_Y_OKRES_RRRR = 0
- SCORE_Y_OKRES_RRRR nabývá hodnoty 0, pokud VZ_Y_RRRR = 0 nebo SEE_Y_RRRR = 0

Výsledné hodnoty u výše uvedeného příkladu pak jsou:

SCORE_Y_OKRES_2009	0
SCORE_Y_OKRES_2010	0
SCORE_Y_OKRES_2011	0
SCORE_Y_OKRES_2012	0
SCORE_Y_OKRES_2013	1
SCORE_Y_OKRES_2014	1
SCORE_Y_OKRES_2015	1
SCORE_Y_OKRES_2016	0
SCORE_Y_OKRES_2017	0

SCORE_OKRES je následně počítáno tak, že sečteme všechny SCORE_Y_OKRES_RRRR a tento součet vydělíme počtem let, ve kterých je SEE_Y_RRRR > 0. Pokud je výsledkem číslo menší než 1 a počet let, ve kterých máme pro dané SEE_IDOS záznam je větší než nebo rovno 3 (neboli délka historie pro dané SEE_IDOS je delší nebo rovna třem rokům), pak SCORE_OKRES = 0,95. Pokud máme pro dané SEE_IDOS pouze dvouletou historii, pak SCORE_OKRES = 0,5.

Touto úpravou je dosaženo toho, že snižujeme váhu tohoto skóre, pokud máme dostatečně bohatou historii neboli přikládáme větší váhu jiným skóre, pokud pro dané SEE_IDOS máme dostatek záznamů a potenciálně dobré spojení tak nepenalizujeme tím, že se v některých letech okresy v databázích nerovnají. Tento způsob výpočtu skóre je motivován hlavně tím, že kvalita záznamů pro okres bydliště není příliš dobrá.

Výsledné skóre u výše uvedeného příkladu pak je:

	Před úpravou	Po úpravě
SCORE_OKRES	0,75	0,95

SCORE_OD

Předpokládáme, že datum začátku neschopenky musí být stejné nebo větší než datum počátku pojištění. Vzhledem k možné špatné kvalitě dat připouštíme, že tento předpoklad nemusí platit zcela a zmírňujeme tento předpoklad definováním proměnné see_od_tolerance vyjádřené ve dnech.

Uvažujeme následující tabulku VZ_YEARS a SEE_YEARS. (Tabulky jsou redukovány tak, aby obsahovaly pouze sloupce užité pro výpočet tohoto skóre. Tabulky jsou transponovány pro jednodušší zobrazení.)

VZ_IDOS	ABCDEF
VZ_Y_2009	0
VZ_Y_2010	0
VZ_Y_2011	0
VZ_Y_2012	10
VZ_Y_2013	50
VZ_Y_2014	18
VZ_Y_2015	10
VZ_Y_2016	18
VZ_Y_2017	0
VZ_Y_OD_2009	01.01.1900
VZ_Y_OD_2010	01.01.1900
VZ_Y_OD_2011	18.07.2011
VZ_Y_OD_2012	01.01.2012
VZ_Y_OD_2013	01.01.2013
VZ_Y_OD_2014	01.01.2014
VZ_Y_OD_2015	01.01.2015
VZ_Y_OD_2016	01.05.2016
VZ_Y_OD_2017	01.01.1900

SEE_IDOS	12345
SEE_Y_2009	0
SEE_Y_2010	0
SEE_Y_2011	0
SEE_Y_2012	0
SEE_Y_2013	50
SEE_Y_2014	18
SEE_Y_2015	10
SEE_Y_2016	18
SEE_Y_2017	0
SEE_Y_OD_2009	01.01.1900
SEE_Y_OD_2010	01.01.1900
SEE_Y_OD_2011	01.01.1900
SEE_Y_OD_2012	01.01.1900
SEE_Y_OD_2013	10.03.2013
SEE_Y_OD_2014	12.11.2014
SEE_Y_OD_2015	09.08.2015
SEE_Y_OD_2016	03.01.2016
SEE_Y_OD_2017	01.01.1900

Nejdříve spočteme dílčí skóre SCORE_Y_OD_RRRR tak, že porovnáváme sloupce VZ_Y_OD_RRRR a SEE_Y_OD_RRRR.

- SCORE_Y_OD_RRRR nabývá hodnoty 1, pokud SEE_Y_OD_RRRR >= VZ_Y_OD_RRRR
- SCORE_Y_OD_RRRR nabývá hodnoty 0,9, pokud VZ_Y_OD_RRRR > SEE_Y_OD_RRRR >= VZ_Y_OD_RRRR – see_od_tolerance
- SCORE_Y_OD_RRRR nabývá hodnoty 0, pokud SEE_Y_OD_RRRR < VZ_OD_RRRR – see_od_tolerance
- SCORE_Y_OD_RRRR nabývá hodnoty 0, pokud SEE_Y_RRRR = 0 nebo VZ_Y_RRRR = 0.

Výsledné hodnoty u výše uvedeného příkladu pak jsou:

SCORE_Y_OD_2009	0
SCORE_Y_OD_2010	0
SCORE_Y_OD_2011	0

SCORE_Y_OD_2012	0
SCORE_Y_OD_2013	1
SCORE_Y_OD_2014	1
SCORE_Y_OD_2015	1
SCORE_Y_OD_2016	0
SCORE_Y_OD_2017	0

SCORE_OD je následně počítáno tak, že sečteme všechny SCORE_Y_OD_RRRR a tento součet vydělíme počtem let, ve kterých je SEE_Y_RRRR > 0.

Výsledné skóre u výše uvedeného příkladu pak je:

SCORE_OD	0,75

SCORE_DO

Předpokládáme, že trvání pracovní neschopnosti by mělo být v intervalu pojištěné doby dotyčné osoby. Existují případy, kdy tomu takto není (například dojde k ukončení pracovního poměru, ale pracovní neschopnost běží dál), a proto skóre není vypočteno pomocí ostré hranice, ale naopak lineárně klesá k nule s tím, jak doba neschopnosti přesahuje dobu pojištění.

Vzhledem k možné špatné kvalitě dat dále uvažujeme volitelný parametr see_do_tolerance (ve dnech), který zmírňuje penalizaci.

Uvažujeme následující tabulku VZ_YEARS a SEE_YEARS. (Tabulky jsou redukovány tak, aby obsahovaly pouze sloupce užité pro výpočet tohoto skóre. Tabulky jsou transponovány pro jednodušší zobrazení.)

VZ_IDOS	ABCDEF
VZ_Y_2009	0
VZ_Y_2010	0
VZ_Y_2011	0
VZ_Y_2012	10
VZ_Y_2013	50
VZ_Y_2014	18
VZ_Y_2015	10
VZ_Y_2016	18
VZ_Y_2017	0
VZ_Y_DO_2009	01.01.1900
VZ_Y_DO_2010	01.01.1900
VZ_Y_DO_2011	31.08.2011
VZ_Y_DO_2012	31.08.2012
VZ_Y_DO_2013	31.12.2013
VZ_Y_DO_2014	31.08.2014
VZ_Y_DO_2015	31.12.2015
VZ_Y_DO_2016	31.12.2016
VZ_Y_DO_2017	01.01.1900

SEE_IDOS	12345
SEE_Y_2009	0
SEE_Y_2010	0
SEE_Y_2011	0
SEE_Y_2012	0
SEE_Y_2013	50
SEE_Y_2014	18
SEE_Y_2015	10
SEE_Y_2016	18
SEE_Y_2017	0
SEE_Y_DO_2009	01.01.1900
SEE_Y_DO_2010	01.01.1900
SEE_Y_DO_2011	01.01.1900
SEE_Y_DO_2012	01.01.1900
SEE_Y_DO_2013	01.04.2013
SEE_Y_DO_2014	30.11.2014
SEE_Y_DO_2015	19.08.2015
SEE_Y_DO_2016	21.01.2016
SEE_Y_DO_2017	01.01.1900

(Hodnoty OD jsou uvedeny v tabulce výše.)

Nejdříve spočteme dílčí skóre SCORE_Y_DO_RRRR tak, že porovnáváme sloupce VZ_Y_DO_RRRR a SEE_Y_DO_RRRR.

- SCORE_Y_DO_RRRR nabývá hodnoty 1, pokud SEE_Y_DO_RRRR <= VZ_Y_DO_RRRR
- SCORE_Y_DO_RRRR nabývá hodnoty 0,9, pokud VZ_Y_DO_RRRR < SEE_Y_DO_RRRR <= VZ_Y_DO_RRRR + see_do_tolerance
- SCORE_Y_DO_RRRR nabývá hodnoty 0, pokud SEE_Y_DO_RRRR a VZ_Y_DO_RRRR = 01.01.1900

- SCORE_Y_DO_RRRR = (SEE_Y_DO_RRRR – VZ_Y_OD_RRRR) / (VZ_Y_DO_RRRR – VZ_Y_OD_RRRR)

Výsledné hodnoty u výše uvedeného příkladu pak jsou:

SCORE_Y_DO_2009	0
SCORE_Y_DO_2010	0
SCORE_Y_DO_2011	0
SCORE_Y_DO_2012	0
SCORE_Y_DO_2013	1
SCORE_Y_DO_2014	91/333
SCORE_Y_DO_2015	1
SCORE_Y_DO_2016	1
SCORE_Y_DO_2017	0

SCORE_DO je následně počítáno tak, že sečteme všechny SCORE_Y_DO_RRRR a tento součet vydělíme počtem let, ve kterých je VZ_Y_RRRR > 0. Výsledné skóre u výše uvedeného příkladu pak je:

SCORE_DO	0,66

SCORE_ELDPCNT

Předpokládáme, že pokud má dotyčná osoba více zaměstnavatelů a máme tedy pro tuto osobu více ELDP, bude i případná pracovní neschopnost pro tuto osobu uvedena vícekrát.

Uvažujeme následující tabulku VZ_YEARS a SEE_YEARS. (Tabulky jsou redukovány tak, aby obsahovaly pouze sloupce užité pro výpočet tohoto skóre. Tabulky jsou transponovány pro jednodušší zobrazení.)

VZ_IDOS	ABCDEF
VZ_Y_2009	0
VZ_Y_2010	0
VZ_Y_2011	0
VZ_Y_2012	10
VZ_Y_2013	50
VZ_Y_2014	18
VZ_Y_2015	10
VZ_Y_2016	18
VZ_Y_2017	0

SEE_IDOS	12345
SEE_Y_2009	0
SEE_Y_2010	0
SEE_Y_2011	0
SEE_Y_2012	0
SEE_Y_2013	50
SEE_Y_2014	18
SEE_Y_2015	10
SEE_Y_2016	18
SEE_Y_2017	0

VZ_Y_ELDPCNT_2009	0
VZ_Y_ELDPCNT_2010	0
VZ_Y_ELDPCNT_2011	1
VZ_Y_ELDPCNT_2012	1
VZ_Y_ELDPCNT_2013	2
VZ_Y_ELDPCNT_2014	0
VZ_Y_ELDPCNT_2015	0
VZ_Y_ELDPCNT_2016	1
VZ_Y_ELDPCNT_2017	0

SEE_Y_ELDPCNT_2009	0
SEE_Y_ELDPCNT_2010	0
SEE_Y_ELDPCNT_2011	1
SEE_Y_ELDPCNT_2012	0
SEE_Y_ELDPCNT_2013	2
SEE_Y_ELDPCNT_2014	0
SEE_Y_ELDPCNT_2015	0
SEE_Y_ELDPCNT_2016	2
SEE_Y_ELDPCNT_2017	0

Nejdříve spočteme dílčí skóre SCORE_Y_ELDPCNT_RRRR tak, že porovnáváme sloupce VZ_Y_ELDPCNT_RRRR a SEE_Y_ELDPCNT_ RRRR (připomínáme, že hodnoty těchto sloupců vyjadřují počet souběžných neschopenek/ELDP – 1 znamená, že žádné souběžné záznamy neexistují, 2 znamená, že existuje 1 souběžný záznam atd.).

- SCORE_Y_ELDPCNT_RRRR nabývá hodnoty 1, pokud VZ_Y_ELDPCNT_RRRR >= SEE_Y_ELDPCNT_RRRR
- SCORE_Y_ELDPCNT_RRRR nabývá hodnoty 0, pokud VZ_Y_ELDPCNT_RRRR < SEE_Y_ELDPCNT_RRRR
- SCORE_Y_ELDPCNT_RRRR nabývá hodnoty 0 v případě, že VZ_Y_RRRR = 0 nebo SEE_Y_RRRR = 0

Výsledné hodnoty u výše uvedeného příkladu pak jsou:

SCORE_Y_DO_2009	0
SCORE_Y_DO_2010	0
SCORE_Y_DO_2011	0
SCORE_Y_DO_2012	0
SCORE_Y_DO_2013	1
SCORE_Y_DO_2014	1
SCORE_Y_DO_2015	1
SCORE_Y_DO_2016	0
SCORE_Y_DO_2017	0

SCORE_ELDPCNT je následně počítáno tak, že sečteme všechny SCORE_Y_ELDPCNT_RRRR a tento součet vydělíme počtem let, ve kterých je SEE_Y_RRRR > 0. Výsledné skóre u výše uvedeného příkladu pak je:

SCORE_ELDPCNT	0,75

SCORE_VDOBA

Předpokládáme, že vyloučená doba v databázi VZ by měla být minimálně tak dlouhá, jako vyloučená doba v databázi SEE (tak, jak byla definována v kapitole 3). V databázi VZ může být vyloučená doba delší než v SEE, a to z toho důvodu, že (na rozdíl od SEE) zahrnuje i ošetřování člena rodiny a mateřskou dovolenou. Informace o mateřské dovolené a ošetřovném nemáme ve zdrojích k dispozici, proto pracujeme s pravděpodobností, že daný jedince mohl mít mateřskou dovolenou nebo ošetřovné a tuto pravděpodobnost zohledňujeme při výpočtu skóre.

Uvažujeme následující tabulku VZ_YEARS a SEE_YEARS. (Tabulky jsou redukovány tak, aby obsahovaly pouze sloupce užité pro výpočet tohoto skóre. Tabulky jsou transponovány pro jednodušší zobrazení.)

VZ_IDOS	ABCDEF
VZ_Y_2009	0
VZ_Y_2010	0
VZ_Y_2011	0
VZ_Y_2012	10
VZ_Y_2013	50
VZ_Y_2014	18
VZ_Y_2015	10
VZ_Y_2016	18
VZ_Y_2017	0
VZ_Y_MATERSKA_2009	0,0000
VZ_Y_MATERSKA_2010	0,0000
VZ_Y_MATERSKA_2011	0,0023
VZ_Y_MATERSKA_2012	0,0012
VZ_Y_MATERSKA_2013	0,0005
VZ_Y_MATERSKA_2014	0,0000
VZ_Y_MATERSKA_2015	0,0000
VZ_Y_MATERSKA_2016	0,0003
VZ_Y_MATERSKA_2017	0,0000
VZ_Y_OSETROVNE_2009	0,0000
VZ_Y_OSETROVNE_2010	0,0000
VZ_Y_OSETROVNE_2011	0,1665
VZ_Y_OSETROVNE_2012	0,1426
VZ_Y_OSETROVNE_2013	0,1153
VZ_Y_OSETROVNE_2014	0,0000
VZ_Y_OSETROVNE_2015	0,0000
VZ_Y_OSETROVNE_2016	0,0475
VZ_Y_OSETROVNE_2017	0,0000

SEE_IDOS	12345
SEE_Y_2009	0
SEE_Y_2010	0
SEE_Y_2011	0
SEE_Y_2012	0
SEE_Y_2013	50
SEE_Y_2014	18
SEE_Y_2015	10
SEE_Y_2016	18
SEE_Y_2017	0
VZ_Y_MATERSKA_2009	0,0000
VZ_Y_MATERSKA_2010	0,0000
VZ_Y_MATERSKA_2011	0,0000
VZ_Y_MATERSKA_2012	0,0000
VZ_Y_MATERSKA_2013	0,0000
VZ_Y_MATERSKA_2014	0,0000
VZ_Y_MATERSKA_2015	0,0000
VZ_Y_MATERSKA_2016	0,0000
VZ_Y_MATERSKA_2017	0,0000
VZ_Y_OSETROVNE_2009	0,0000
VZ_Y_OSETROVNE_2010	0,0000
VZ_Y_OSETROVNE_2011	0,0000
VZ_Y_OSETROVNE_2012	0,0000
VZ_Y_OSETROVNE_2013	0,0000
VZ_Y_OSETROVNE_2014	0,0000
VZ_Y_OSETROVNE_2015	0,0000
VZ_Y_OSETROVNE_2016	0,0000
VZ_Y_OSETROVNE_2017	0,0000

Nejdříve si definujeme pomocný sloupec VDOBA_DIFF jako VZ_Y_RRRR – SEE_Y_RRRR. Dále také používáme:

- volitelný parametr vdoba_tolerance z kapitoly **Spojení databází VZ, SEE a ISPV**
- materska_delka_od – volitelný parametr udává spodní hranici délky mateřské dovolené
- materska_delka_do – volitelný parametr udává horní hranici délky mateřské dovolené
- osetrovacka_delka_od – volitelný parametr udává spodní hranici délky ošetřování člena rodiny
- osetrovacka_delka_do – volitelný parametr udává horní hranici délky ošetřování člena rodiny

Nejdříve spočteme dílčí skóre SCORE_Y_VDOBA_RRRR tak, že porovnáváme sloupce VZ_Y_RRRR a SEE_Y_RRRR.

- SCORE_Y_VDOBA_RRRR nabývá hodnoty 0,9 v případě, že 0 < ABS(VDOBA_DIFF) <= vdoba_tolerance
- SCORE_Y_VDOBA_RRRR nabývá hodnoty 0, pokud VDOBA_DIFF < 0 a zároveň ABS(VDOBA_DIFF) > vdoba_tolerance
- V případě, že neplatí ani jedna z podmínek výše, tedy ABS(VDOBA_DIFF) > vdoba_tolerance a VDOBA_DIFF > 0, poté testujeme, zda tato diference není způsobena tím, že je v databázi VZ v rámci vyloučené doby zahrnuta i mateřská dovolená, případně ošetřování člena rodiny
    - SCORE_Y_VDOBA_RRRR nabývá hodnoty VZ_Y_MATERSKA_RRRR, pokud VDOBA_DIFF > = materska_delka_od a zároveň VDOBA_DIFF < = materska_delka_do
    - SCORE_Y_VDOBA_RRRR nabývá hodnoty VZ_Y_OSETROVNE_RRRR, pokud VDOBA_DIFF > = osetrovacka_delka_od a zároveň VDOBA_DIFF < = osetrovacka_delka_do a zároveň neplatí VDOBA_DIFF > = materska_delka_od a zároveň VDOBA_DIFF < = materska_delka_do
    - SCORE_Y_VDOBA_RRRR nabývá hodnoty VZ_Y_MATERSKA_RRRR násobené VZ_Y_OSETROVNE_RRRR, pokud VDOBA_DIFF > = osetrovacka_delka_od a zároveň VDOBA_DIFF < = materska_delka_do + osetrovacka_delka_do
- V případě, že neplatí ani jedna z podmínek výše, tak SCORE_Y_VDOBA_RRRR nabývá hodnoty dané vzorcem ((1+vdoba_tolerance)/(VDOBA_DIFF + vdoba_tolerance+1))^2
  Zlomek je umocněn z důvodu větší penalizace významnějších rozdílů mezi dobami. Body výše byly postihnuty možné odchylky, které vyplývají z logiky evidence vyloučené doby a doby pracovní neschopnosti. Ostatní odchylky mohou být způsobeny horší kvalitou dat nebo nesprávností přiřazení. Proto je nutné tyto diference dostatečně penalizovat.

Výsledné hodnoty u výše uvedeného příkladu pak jsou:

SCORE_Y_VDOBA_2009	0
SCORE_Y_VDOBA_2010	0
SCORE_Y_VDOBA_2011	0
SCORE_Y_VDOBA_2012	0
SCORE_Y_VDOBA_2013	1
SCORE_Y_VDOBA_2014	1
SCORE_Y_VDOBA_2015	1
SCORE_Y_VDOBA_2016	1
SCORE_Y_VDOBA_2017	0

SCORE_VDOBA je následně počítáno tak, že sečteme všechny SCORE_Y_VDOBA_RRRR a tento součet vydělíme počtem let, ve kterých je SEE_Y_RRRR > 0. Výsledné skóre u výše uvedeného příkladu pak je:

SCORE_VDOBA	1

SCORE_PN

Předpokládáme, že informace ve sloupci DNYPN, která vznikla spojením databáze STATMIN VZ a databáze ISPV, a která udává počet dní v pracovní neschopnosti by měla odpovídat vyloučené době (tak, jak byla definována v kapitole 3) v databázi SEE.

Uvažujeme následující tabulku VZ_YEARS a SEE_YEARS. (Tabulky jsou redukovány tak, aby obsahovaly pouze sloupce užité pro výpočet tohoto skóre. Tabulky jsou transponovány pro jednodušší zobrazení.)

VZ_IDOS	ABCDEF
VZ_Y_2009	0
VZ_Y_2010	0
VZ_Y_2011	0
VZ_Y_2012	10
VZ_Y_2013	50
VZ_Y_2014	18
VZ_Y_2015	10
VZ_Y_2016	18
VZ_Y_2017	0
VZ_Y_DNYPN_2009	0
VZ_Y_DNYPN_2010	0
VZ_Y_DNYPN_2011	-1
VZ_Y_DNYPN_2012	-1
VZ_Y_DNYPN_2013	-1
VZ_Y_DNYPN_2014	0
VZ_Y_DNYPN_2015	0
VZ_Y_DNYPN_2016	18
VZ_Y_DNYPN_2017	0

SEE_IDOS	12345
SEE_Y_2009	0
SEE_Y_2010	0
SEE_Y_2011	0
SEE_Y_2012	0
SEE_Y_2013	50
SEE_Y_2014	18
SEE_Y_2015	10
SEE_Y_2016	18
SEE_Y_2017	0
SEE_Y_2009	0
SEE_Y_2010	0
SEE_Y_2011	0
SEE_Y_2012	0
SEE_Y_2013	50
SEE_Y_2014	18
SEE_Y_2015	10
SEE_Y_2016	18
SEE_Y_2017	0

Dále také používáme volitelný parametr vdoba_tolerance z kapitoly 4 Spojení databází VZ, SEE a ISPV.

Nejdříve spočteme dílčí skóre SCORE_Y_PN_RRRR tak, že porovnáváme sloupce VZ_Y_DNYPN a SEE_Y_RRRR.

- SCORE_Y_PN_RRRR nabývá hodnoty SCORE_Y_VDOBA_RRRR, pokud neznáme VZ_Y_DNYPN_RRRR (tam, kde VZ_Y_DNYPN_RRRR = -1)
- **SCORE_Y_PN_RRRR nabývá hodnoty 0,9, pokud VZ_Y_DNYPN_RRRR – SEE_Y_RRRR <> 0 a zároveň VZ_Y_DNYPN_RRRR – SEE_Y_RRRR <= vdoba_tolerance**
- Pokud platí, že VZ_Y_DNYPN_RRRR >= SEE_Y_RRRR, pak SCORE_Y_PN_RRRR nabývá hodnoty dané vzorcem ((1+ vdoba_tolerance)/(ABS(VZ_Y_DNYPN_RRRR - SEE_Y_RRRR) + vdoba_tolerance + 1)) ^2

SCORE_Y_PN_RRRR nabývá hodnoty 0, pokud VZ_Y_RRRR a SEE_Y_RRRR = 0.

Výsledné hodnoty u výše uvedeného příkladu pak jsou:

SCORE_Y_PN_2009	0
SCORE_Y_PN_2010	0
SCORE_Y_PN_2011	0
SCORE_Y_PN_2012	0
SCORE_Y_PN_2013	1
SCORE_Y_PN_2014	0,14
SCORE_Y_PN_2015	0,27
SCORE_Y_PN_2016	1
SCORE_Y_PN_2017	0

Zde byla aplikovávána vdoba_tolerance = 10

SCORE_PN je následně počítáno tak, že sečteme všechny SCORE_Y_PN_RRRR a tento součet vydělíme počtem let, ve kterých je SEE_Y_RRRR > 0. Výsledné skóre u výše uvedeného příkladu pak je:

SCORE_PN	0,6

SCORE_VDOBACOMB

SCORE_VDOBA a SCORE_PN testují ve své podstatě to samé a pro výpočet finálního skóre by tato informace byla duplicitní. Proto definujeme SCORE_VDOBACOMB, které kombinuje SCORE_VDOBA a SCORE_PN.

Pro tento účel definujeme volitelný parametr ispv_confidance, který nabývá hodnoty z intervalu <0;1> Vzhledem k tomu, že hodnoty ve sloupci VZ_DNYPN vychází z nejednoznačného propojení databází ISPV a STATMIN VZ, a navíc se jedná o přepočet z hodin na dny, nemusí být tato hodnota vždy plně validní. Parametrem ispv_confidance určujeme jistotu, se kterou k hodnotám ve sloupci VZ_DNYPN přistupujeme.

SCORE_VDOBACOMB nabývá hodnoty SCORE_VDOBA, pokud je SCORE_VDOBA >= SCORE_PN * ispv_confidance, jinak nabývá hodnoty SCORE_PN.

SCORE_SEEROK

Předpokládáme, že pokud má daný člověk v daném roce záznam v databázi SEE, je důvod se domnívat, že by měl mít záznam ve stejném roce i v databázi VZ.

Uvažujeme následující tabulku VZ_YEARS a SEE_YEARS. (Tabulky jsou redukovány tak, aby obsahovaly pouze sloupce užité pro výpočet tohoto skóre. Tabulky jsou transponovány pro jednodušší zobrazení.)

VZ_IDOS	ABCDEF
VZ_Y_2009	0
VZ_Y_2010	0
VZ_Y_2011	0
VZ_Y_2012	10
VZ_Y_2013	50
VZ_Y_2014	18
VZ_Y_2015	10
VZ_Y_2016	18
VZ_Y_2017	0

SEE_IDOS	12345
SEE_Y_2009	0
SEE_Y_2010	0
SEE_Y_2011	0
SEE_Y_2012	0
SEE_Y_2013	50
SEE_Y_2014	18
SEE_Y_2015	10
SEE_Y_2016	18
SEE_Y_2017	0

Nejprve definujeme dílčí skóre 1SEE1VZ_Y_RRRR tak, že porovnáváme VZ_Y_RRRR a SEE_Y_RRRR. 1SEE1VZ_Y_RRRR nabývá hodnoty 1, pokud VZ_Y_RRRR > 0 a zároveň SEE_Y_RRRR > 0, v ostatních případech 1SEE1VZ_Y_RRRR nabývá hodnoty 0.

Výsledné hodnoty u výše uvedeného příkladu pak jsou:

1SEE1VZ_Y_2009	0
1SEE1VZ_Y_2010	0
1SEE1VZ_Y_2011	0

1SEE1VZ_Y_2012	0
1SEE1VZ_Y_2013	1
1SEE1VZ_Y_2014	1
1SEE1VZ_Y_2015	1
1SEE1VZ_Y_2016	1
1SEE1VZ_Y_2017	0

SCORE_SEEROK je spočítáno tak, že sečteme všechna dílčí skóre 1SEE1VZ_Y_RRRR a tento součet vydělíme počtem let, ve kterých je SEE_Y_RRRR > 0 a umocníme na druhou, čímž více penalizujeme větší odchylky, které jsou velmi pravděpodobně způsobené nesprávným přiřazením jedinců.

Pokud je ale rozdíl mezi sumou 1SEE1VZ_Y_RRRR a počtem let, ve kterých je SEE_Y_RRRR > 0 větší než 3 roky, pak SCORE_SEEROK nabývá hodnoty 0. Výsledné skóre u výše uvedeného příkladu pak je:

SCORE_SEEROK	1

SCORE_VZROK

Předpokládáme, že pokud má daný člověk v daném roce záznam v databázi VZ (s vyloučenou dobou > 0), je důvod se domnívat, že by měl mít záznam ve stejném roce i v databázi SEE až na případy mateřské dovolené a ošetřovného.

Uvažujeme následující tabulku VZ_YEARS a SEE_YEARS. (Tabulky jsou redukovány tak, aby obsahovaly pouze sloupce užité pro výpočet tohoto skóre. Tabulky jsou transponovány pro jednodušší zobrazení.)

VZ_IDOS	ABCDEF
VZ_Y_2009	0
VZ_Y_2010	0
VZ_Y_2011	0
VZ_Y_2012	10
VZ_Y_2013	50
VZ_Y_2014	18
VZ_Y_2015	10
VZ_Y_2016	18
VZ_Y_2017	0
VZ_Y_MATERSKA_2009	0,0000
VZ_Y_MATERSKA_2010	0,0000
VZ_Y_MATERSKA_2011	0,0023
VZ_Y_MATERSKA_2012	0,0012
VZ_Y_MATERSKA_2013	0,0005
VZ_Y_MATERSKA_2014	0,0000
VZ_Y_MATERSKA_2015	0,0000
VZ_Y_MATERSKA_2016	0,0003
VZ_Y_MATERSKA_2017	0,0000
VZ_Y_OSETROVNE_2009	0,0000
VZ_Y_OSETROVNE_2010	0,0000
VZ_Y_OSETROVNE_2011	0,1665

SEE_IDOS	12345
SEE_Y_2009	0
SEE_Y_2010	0
SEE_Y_2011	0
SEE_Y_2012	0
SEE_Y_2013	50
SEE_Y_2014	18
SEE_Y_2015	10
SEE_Y_2016	18
SEE_Y_2017	0
VZ_Y_MATERSKA_2009	0,0000
VZ_Y_MATERSKA_2010	0,0000
VZ_Y_MATERSKA_2011	0,0000
VZ_Y_MATERSKA_2012	0,0000
VZ_Y_MATERSKA_2013	0,0000
VZ_Y_MATERSKA_2014	0,0000
VZ_Y_MATERSKA_2015	0,0000
VZ_Y_MATERSKA_2016	0,0000
VZ_Y_MATERSKA_2017	0,0000
VZ_Y_OSETROVNE_2009	0,0000
VZ_Y_OSETROVNE_2010	0,0000
VZ_Y_OSETROVNE_2011	0,0000

VZ_Y_OSETROVNE_2012	0,1426
VZ_Y_OSETROVNE_2013	0,1153
VZ_Y_OSETROVNE_2014	0,0000
VZ_Y_OSETROVNE_2015	0,0000
VZ_Y_OSETROVNE_2016	0,0475
VZ_Y_OSETROVNE_2017	0,0000

VZ_Y_OSETROVNE_2012	0,0000
VZ_Y_OSETROVNE_2013	0,0000
VZ_Y_OSETROVNE_2014	0,0000
VZ_Y_OSETROVNE_2015	0,0000
VZ_Y_OSETROVNE_2016	0,0000
VZ_Y_OSETROVNE_2017	0,0000

Nejprve definujeme dílčí skóre 1VZ1SEE_Y_RRRR tak, že porovnáváme VZ_Y_RRRR a SEE_Y_RRRR. Dále také používáme volitelné parametry:

- materska_delka_od = volitelný parametr udává spodní hranici délky mateřské dovolené
- materska_delka_do = volitelný parametr udává horní hranici délky mateřské dovolené
- osetrovacka_delka_od = volitelný parametr udává spodní hranici délky ošetřování člena rodiny
- osetrovacka_delka_do = volitelný parametr udává horní hranici délky ošetřování člena rodiny

Skóre 1VZ1SEE_Y_RRRR se pak spočte následovně:

- 1VZ1SEE_Y_RRRR nabývá hodnoty 1, pokud VZ_Y_RRRR > 0 a zároveň SEE_Y_RRRR > 0
- 1VZ1SEE_Y_RRRR nabývá hodnoty VZ_Y_MATERSKA_RRRR, pokud VZ_Y_RRRR >= materska_delka_od a zároveň VZ_Y_RRRR <= materska_delka_do
- 1VZ1SEE_Y_RRRR nabývá hodnoty VZ_Y_OSETROVNE_RRRR, pokud VZ_Y_RRRR >= osetrovacka_delka_od a zároveň VZ_Y_RRRR <= osetrovacka_delka_do
- **1VZ1SEE_Y_RRRR nabývá hodnoty VZ_Y_MATERSKA_RRRR * VZ_Y_OSETROVNE_RRRR, pokud VZ_Y_RRRR >= osetrovacka_delka_od a zároveň VZ_Y_RRRR <= osetrovacka_delka_do + materska_delka_do**
- Pokud není splněna ani jedna podmínka výše, pak 1VZ1SEE_Y_RRRR nabývá hodnoty 0.

Výsledné hodnoty u výše uvedeného příkladu pak jsou:

1VZ1SEE_Y_2009	0
1VZ1SEE_Y_2010	0
1VZ1SEE_Y_2011	0
1VZ1SEE_Y_2012	0,1426
1VZ1SEE_Y_2013	1
1VZ1SEE_Y_2014	1
1VZ1SEE_Y_2015	1
1VZ1SEE_Y_2016	1
1VZ1SEE_Y_2017	0

*materska_delka_od = 42, materska_delka_do = 56, osetrovacka_delka_od = 5, osetrovacka_delka_do = 16

SCORE_VZROK je spočítáno tak, že sečteme všechna dílčí skóre 1VZ1SEE_Y_RRRR a tento součet vydělíme počtem let, ve kterých je VZ_Y_RRRR > 0 a umocníme na druhou.

Pokud je ale rozdíl mezi sumou 1VZ1SEE_Y_RRRR a počtem let, ve kterých je VZ_Y_RRRR > 0 větší než 3 roky, pak SCORE_VZROK nabývá hodnoty 0. Výsledné skóre u výše uvedeného příkladu pak je:

SCORE_VZROK	0,69

SCORE_TOTAL

Celkové skóre je pak definováno jako součin všech výše uvedených skóre. Agregace pomocí násobku bylo zvoleno z toho důvodu, že požadujeme, aby ideálně platily všechny podmínky najednou.

SCORE_TOTAL =

SCORE_ROKNAR * SCORE_MESNAR * SCORE_POHLAVI * SCORE_OKRES * SCORE_OD * SCORE_DO * SCORE_VDOBACOMB * SCORE_SEEROK * SCORE_VZROK * SCORE_ELDPCNT

6. Výpočet unikátnostního skóre

I po výpočtu podobnostního skóre může nastat situace, kdy na jedno SEE_IDOS napárujeme více kandidátů z databáze VZ s vysokým podobnostním skóre (zejména pro jedince s krátkou nemocenskou historií). Stejně tak může nastat situace, kdy jedno VZ_IDOS je přiřazeno více jedincům z databáze SEE. Naopak mohou nastat situace, kdy podobnostní skóre sice není maximální, ale nalezení přiřazení je jediné možné (ať už z pohledu SEE nebo z pohledu VZ).

Více potenciálních kandidátů (s vyšším podobnostním skóre) zvyšují nejistotu, se kterou jsme schopni prohlásit, že dané SEE_IDOS patří k danému VZ_IDOS. Z tohoto důvodu je v rámci tohoto kroku počítáno skóre unikátnostní, které bere v úvahu jednoznačnost přiřazení jednotlivých SEE/VZ_IDOS.

Unikátnostní skóre je spočteno jako pravděpodobnost toho, že dané SEE_IDOS patří k danému VZ_IDOS a zároveň nepatří k ostatním kandidátům z VZ anebo (naopak), že dané VZ_IDOS patří k danému SEE_IDOS a zároveň nepatří k ostatním kandidátům z SEE.

Postup budeme ilustrovat na následujícím příkladu. Uvažujme tyto výsledky podobnostního skóre.

SEE_IDOS	VZ_IDOS	PODOBNOSTNÍ SKÓRE
F	8	1,00
C	8	0,64
B	1	0,90
D	1	0,80
A	1	1,00
C	1	0,26
F	1	0,10
D	8	0,000568
B	8	0,0005
A	8	0,000165

Nejprve spočteme sumu PODOBNOSTNÍHO SKÓRE pro každé SEE_IDOS

SEE_IDOS	SUM(PODOBNOSTNÍ SKÓRE)
A	1,000
B	0,901
C	0,900
D	0,801
F	1,100

Dále spočteme analogicky sumu PODOBNOSTNÍHO SKÓRE pro každé VZ_IDOS

VZ_IDOS	SUM(PODOBNOSTNÍ SKÓRE)
1	3,060
8	1,641

V dalším kroku vydělíme PODOBNOSTNÍ SKÓRE příslušnými sumami, které jsme spočetli v předchozích krocích. Tím dostaneme normalizovaná (centrované) skóre.

SEE_IDOS	VZ_IDOS	PODOBNOSTNÍ SKÓRE	CENTROVANÉ_SCORE_SEE	CENTROVANÉ_SCORE_VZ
F	8	1,00	0,909	0,609
C	8	0,64	0,711	0,390
B	1	0,90	0,999	0,294
D	1	0,80	0,999	0,261
A	1	1,00	1,000	0,327
C	1	0,26	0,289	0,085
F	1	0,10	0,091	0,033
D	8	0,00	0,001	0,000
B	8	0,00	0,001	0,000
A	8	0,00	0,000	0,000

Tato centrovaná skóre můžeme považovat za pravděpodobnosti, že dané SEE_IDOS patří ke konkrétnímu VZ_IDOS. Odečtením těchto pravděpodobností od 1 dostaneme pravděpodobnost, že dané SEE_IDOS nepatří ke konkrétnímu VZ_IDOS.

SEE_IDOS	VZ_IDOS	PODOBNOSTNÍ SKÓRE	PRAVDĚP., ŽE SEE_IDOS PATŘÍ K VZ_IDOS	PRAVDĚP., ŽE VZ_IDOS PATŘÍ K SEE_IDOS	PRAVDĚP., ŽE SEE_IDOS NEPATŘÍ K VZ_IDOS	PRAVDĚP., ŽE VZ_IDOS NEPATŘÍ K SEE_IDOS
F	8	1,00	0,909	0,609	0,091	0,391
C	8	0,64	0,711	0,390	0,289	0,610
B	1	0,90	0,999	0,294	0,001	0,706
D	1	0,80	0,999	0,261	0,001	0,739
A	1	1,00	1,000	0,327	0,000	0,673
C	1	0,26	0,289	0,085	0,711	0,915
F	1	0,10	0,091	0,033	0,909	0,967
D	8	0,00	0,001	0,000	0,999	1,000
B	8	0,00	0,001	0,000	0,999	1,000
A	8	0,00	0,000	0,000	1,000	1,000

Dále spočteme pravděpodobnost, že dané VZ_IDOS patří ke konkrétnímu SEE_IDOS a zároveň nepatří k jiným SEE_IDOS. Analogicky pak spočítáme pravděpodobnost, že dané SEE_IDOS patří ke konkrétnímu VZ_IDOS a zároveň nepatří k žádnému jinému VZ_IDOS.

Níže uvedená tabulka obsahuje:

- „PST SEE_IDOS -> VZ_IDOS“ = Pravděpodobnost, že SEE_IDOS patří k VZ_IDOS
- „PST VZ_IDOS -> SEE_IDOS“ = Pravděpodobnost, že VZ_IDOS patří k SEE_IDOS
- „PST SEE_IDOS x> VZ_IDOS“ = Pravděpodobnost, že SEE_IDOS nepatří k VZ_IDOS
- „PST VZ_IDOS x> SEE_IDOS“ = Pravděpodobnost, že VZ_IDOS nepatří k SEE_IDOS
- „PST SEE_IDOS = VZ_IDOS“ = Pravděpodobnost, že toto SEE_IDOS patří k tomuto VZ_IDOS a zároveň toto SEE_IDOS nepatří k jiným potenciálním VZ_IDOS
- „PST VZ_IDOS = SEE_IDOS“ = Pravděpodobnost, že toto VZ_IDOS patří k tomuto SEE_IDOS a zároveň toto VZ_IDOS nepatří k jiným potenciálním SEE_IDOS

SEE IDOS	VZ IDOS	PODOBNOSTNÍ SKÓRE	PST SEE_IDOS -> VZ_IDOS	PST VZ_IDOS -> SEE_IDOS	PST SEE_IDOS x> VZ_IDOS	PST VZ_IDOS x> SEE_IDOS	PST SEE_IDOS = VZ_IDOS	PST VZ_IDOS = SEE_IDOS
F	8	1,00	0,909	0,609	0,091	0,391	0,826	0,371
C	8	0,64	0,711	0,390	0,289	0,610	0,506	0,152
B	1	0,90	0,999	0,294	0,001	0,706	0,999	0,129
D	1	0,80	0,999	0,261	0,001	0,739	0,999	0,110
A	1	1,00	1,000	0,327	0,000	0,673	1,000	0,151
C	1	0,26	0,289	0,085	0,711	0,915	0,083	0,029
F	1	0,10	0,091	0,033	0,909	0,967	0,008	0,010
D	8	0,00	0,001	0,000	0,999	1,000	0,000	0,000
B	8	0,00	0,001	0,000	0,999	1,000	0,000	0,000
A	8	0,00	0,000	0,000	1,000	1,000	0,000	0,000

Přičemž „PST SEE_IDOS = VZ_IDOS“ je pravděpodobnost té konkrétní dvojice vynásobená pravděpodobností ostatních dvojic neboli pravděpodobnost „PST SEE_IDOS -> VZ_IDOS“ vynásobíme součinem „PST SEE_IDOS x> VZ_IDOS“ pro dané SEE_IDOS.

„PST SEE_IDOS = VZ_IDOS“ = 0,826
(0,826 = 0,909 * 0,909)

„PST VZ_IDOS = SEE_IDOS“ = 0,371
(0,371 = 0,609 * 0,610 * 1,000 * 1,000 * 1,000)

Unikátnostní skóre je pak spočteno jako průměr „PST SEE_IDOS = VZ_IDOS“ a „PST VZ_IDOS = SEE_IDOS“. Průměr je zde zvolen z toho důvodu, že požadujeme, aby byla unikátní alespoň jedna strana, ne nutně oboje zároveň.

SEE_IDOS	VZ_IDOS	PODOBNOSTNÍ SKÓRE	PST SEE_IDOS = VZ_IDOS	PST VZ_IDOS = SEE_IDOS	UNIKÁTNOSTNÍ SKÓRE
F	8	1,00	0,826	0,371	0,599
C	8	0,64	0,506	0,152	0,329
B	1	0,90	0,999	0,129	0,564
D	1	0,80	0,999	0,110	0,554
A	1	1,00	1,000	0,151	0,575
C	1	0,26	0,083	0,029	0,056
F	1	0,10	0,008	0,010	0,009
D	8	0,00	0,000	0,000	0,000
B	8	0,00	0,000	0,000	0,000
A	8	0,00	0,000	0,000	0,000

7. Finální přiřazení

Na základě podobnostního a unikátnostního skóre je spočteno finální skóre, na základě kterého vybíráme nejvhodnější kandidáty.

Finální skóre spočteme jako kombinaci PODOBNOSTNÍHO SKÓRE a UNIKÁTNOSTNÍHO SKÓRE, a to následujícím způsobem:

PODOBNOSTNÍ SKÓRE^vaha_podobnost * UNIKÁTNOSTNÍ SKÓRE^vaha_unikatnost,

kde vaha_podobnost a vaha_unikatnost jsou volitelné parametry – v našem případě nabývaly hodnot vaha_podobnost = 0.8 a vaha_unikatnost = 0.4.

	Unikátnost																					
Základní skóre	0	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100	
0	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	
5	0%	3%	4%	4%	5%	5%	6%	6%	6%	7%	7%	7%	7%	8%	8%	8%	8%	9%	9%	9%	9%	
10	0%	5%	6%	7%	8%	9%	10%	10%	11%	12%	12%	12%	13%	13%	14%	14%	14%	15%	15%	16%	16%	
15	0%	7%	9%	10%	12%	13%	14%	14%	15%	16%	17%	17%	18%	18%	19%	20%	20%	21%	21%	21%	22%	
20	0%	8%	11%	13%	14%	16%	17%	18%	19%	20%	21%	22%	22%	23%	24%	25%	25%	26%	26%	27%	28%	
25	0%	10%	13%	15%	17%	19%	20%	22%	23%	24%	25%	26%	27%	28%	29%	29%	30%	31%	32%	32%	33%	
30	0%	12%	15%	18%	20%	22%	24%	25%	26%	28%	29%	30%	31%	32%	33%	34%	35%	36%	37%	37%	38%	
35	0%	13%	17%	20%	23%	25%	27%	28%	30%	31%	33%	34%	35%	36%	37%	38%	39%	40%	41%	42%	43%	
40	0%	14%	19%	22%	25%	28%	30%	32%	33%	35%	36%	38%	39%	40%	42%	43%	44%	45%	46%	47%	48%	
45	0%	16%	21%	25%	28%	30%	33%	35%	37%	38%	40%	42%	43%	44%	46%	47%	48%	49%	51%	52%	53%	
50	0%	17%	23%	27%	30%	33%	35%	38%	40%	42%	44%	45%	47%	48%	50%	51%	53%	54%	55%	56%	57%	
55	0%	19%	25%	29%	33%	36%	38%	41%	43%	45%	47%	49%	51%	52%	54%	55%	57%	58%	59%	61%	62%	
60	0%	20%	26%	31%	35%	38%	41%	44%	46%	48%	50%	52%	54%	56%	58%	59%	61%	62%	64%	65%	66%	
65	0%	21%	28%	33%	37%	41%	44%	47%	49%	51%	54%	56%	58%	60%	61%	63%	65%	66%	68%	69%	71%	
70	0%	23%	30%	35%	39%	43%	46%	49%	52%	55%	57%	59%	61%	63%	65%	67%	69%	70%	72%	74%	75%	
75	0%	24%	32%	37%	42%	46%	49%	52%	55%	58%	60%	63%	65%	67%	69%	71%	73%	74%	76%	78%	79%	
80	0%	25%	33%	39%	44%	48%	52%	55%	58%	61%	63%	66%	68%	70%	73%	75%	77%	78%	80%	82%	84%	
85	0%	26%	35%	41%	46%	50%	54%	58%	61%	64%	67%	69%	72%	74%	76%	78%	80%	82%	84%	86%	88%	
90	0%	28%	37%	43%	48%	53%	57%	60%	64%	67%	70%	72%	75%	77%	80%	82%	84%	86%	88%	90%	92%	
95	0%	29%	38%	45%	50%	55%	59%	63%	67%	70%	73%	76%	78%	81%	83%	86%	88%	90%	92%	94%	96%	
100	0%	30%	40%	47%	53%	57%	62%	66%	69%	73%	76%	79%	82%	84%	87%	89%	91%	94%	96%	98%	100%	

Násobek je zvolen z toho důvodu, abychom vybírali takové dvojice, které jsou (pokud možno) zároveň podobné i unikátní. Mocnitelé (v našem případě 0,8 a 0,4) byli stanoveni expertně a to tak, aby byla upřednostňována především taková spojení, která mají vysoké podobnostní skóre. Unikátnostní skóre má nižší váhu, protože i přes to, že může existovat více vhodných kandidátů, lze zpravidla určit, který je vhodnější a relativně malé unikátnostní skóre by způsobilo nežádoucí pokles finálního skóre.

Parametry jsou ale volitelné a záleží pouze na posouzení uživatele, jak je nastaví.

Finální skóre ve výše uvedeném příkladě pro jednotlivé kandidáty s váhami 0,8 a 0,4 je uvedeno v následující tabulce:

SEE_IDOS	VZ_IDOS	PODOBNOSTNÍ SKÓRE	UNIKÁTNOSTNÍ SKÓRE	FINÁLNÍ SKÓRE
F	8	1,00	0,599	0,815
C	8	0,64	0,329	0,449
B	1	0,90	0,564	0,731
D	1	0,80	0,554	0,661
A	1	1,00	0,575	0,802
C	1	0,26	0,056	0,108
F	1	0,10	0,009	0,024
D	8	0,00	0,000	0,000
B	8	0,00	0,000	0,000
A	8	0,00	0,000	0,000

Mechanismus přiřazení funguje tak, že se iterativně určuje maximální skóre pro danou kombinaci SEE_IDOS a VZ_IDOS, přičemž může dojít ke dvěma situacím:

1. Je nalezena jednoznačná kombinace SEE_IDOS a VZ_IDOS s maximálním finálním skóre (neexistuje tedy jiná dvojice, která by měla stejné nebo vyšší skóre).
   Pokud nastane tato situace, pak je tato jednoznačná kombinace označena ve sloupci VÝSLEDEK hodnotou 1 a všechny ostatní dvojice které obsahují stejné SEE_IDOS nebo VZ_IDOS dostanou do sloupce VÝSLEDEK hodnotu 0.
2. Je nalezena kombinace SEE_IDOS a VZ_IDOS, ale stejné maximální skóre má i jiná dvojice. Obě tyto dvojice jsou označeny ve sloupci VÝSLEDEK hodnotou 2 a všechny ostatní dvojice, které obsahují stejné SEE_IDOS nebo VZ_IDOS dostanou do sloupce VÝSLEDEK hodnotu 0.

Finální propojení v našem modelovém příkladu vypadá následovně:

SEE_IDOS	VZ_IDOS	PODOBNOSTNÍ SKÓRE	UNIKÁTNOSTNÍ SKÓRE	FINÁLNÍ SKÓRE	VÝSLEDEK
F	8	1,00	0,599	0,815	1
C	8	0,64	0,329	0,449	0
B	1	0,90	0,564	0,731	0
D	1	0,80	0,554	0,661	0
A	1	1,00	0,575	0,802	1
C	1	0,26	0,056	0,108	0
F	1	0,10	0,009	0,024	0
D	8	0,00	0,000	0,000	0
B	8	0,00	0,000	0,000	0
A	8	0,00	0,000	0,000	0

8. Výsledky propojení

V rámci projektu jsme měli k dispozici následující databáze k propojení:

- SEE20 se záznamy od roku 2009 do poloviny roku 2019,
- SEE21 se záznamy od roku 2018 do poloviny roku 2019,
- SEE22 se záznamy od roku 2019 do poloviny roku 2019,
- STATMIN VZ se záznamy od roku 2004 do roku 2017,

Dále pak pomocné databáze:

- Databázi propojující databázi STATMIN VZ s databází ISPV,
- Doplněný INEP do roku 2017.

V databázích jsme identifikovali následující počty osob:

Počet osob v SEE20	4 303 363
Počet osob v SEE21	46 349
Počet osob v SEE22	2 504
Počet osob ve STATMIN VZ	7 309 989

Maximálním společným rokem, pro který bylo možné provést propojení databází, byl rok 2017.

Po úpravách uvedených v kapitole Změna struktury zdrojových databází a vytvoření databází SEE a VZ jsme dostali následující počty osob, které jsou vhodné pro spojení.

Počet osob v SEE	4 002 228
Počet osob ve VZ	4 124 651

V rámci propojení byly aplikovány další podmínky tak, jak jsou uvedeny v kapitole Spojení databází VZ, SEE a ISPV. Databáze, která vznikla spojením databází VZ a SEE obsahuje následující počet osob:

Počet osob po spojeni	3 982 055

Pro takto spojenou databázi bylo vypočteno podobnostní skóre a následně odebrána taková spojení, která nedosáhla podobnostního skóre > 0, viz kapitola Výpočet podobnostního skóre.

Počet osob po podobnostním skórováni	3 963 775

Následně bylo spočteno unikátnostní skóre a pak také finální skóre s následující statistikou:

- Propojení s maximálním skóre, které zároveň tvořilo jedinou takovouto kombinaci s maximálním skóre.

Počet jedinečných propojení s maximálním skóre	3 311 165

- Propojení, u kterých sice bylo spočteno maximální skóre, ale zároveň existuje takových propojení více.

Počet jedinců v SEE s nejednoznačným propojením	503 025

Počet nejednoznačných propojení s maximálním skóre	2 029 748

- Ostatní propojení (která mají jiné než maximální skóre)

Počet ostatních ne-maximálních propojení:	3 935 818

Následující tabulka je shrnutím výše uvedeného. Z této tabulky je následně možné spočítat úspěšnost propojení.

Počet osob v databázi SEE vhodných pro spojení	4 002 228
Počet osob po spojení databází SEE a VZ	3 982 055
Počet osob po podobnostním skórováni	3 963 775
Počet jedinečných propojení s maximálním skóre	3 311 165

Uvažujeme-li pouze ty osoby, u kterých známe kompletní informaci (pohlaví, rok narození, měsíc narození, okres) a zároveň jsou v databázi SEE uvedeny v minimálně 3 letech, dostáváme následující počty osob.

Počet osob v databázi SEE vhodných pro spojení	4 002 228
Počet osob po odebrání osob s neúplnými daty a krátkou historií	1 361 633
Počet osob po spojení databází SEE a VZ	1 361 325
Počet osob po podobnostním skórováni	1 360 281
Počet jedinečných propojení s maximálním skóre	1 290 907

Bereme-li v úvahu pouze osoby, u kterých je možné provést spojení, dostaneme úspěšnost propojení 82,7%. Pro ty osoby, u kterých známe kompletní informaci a délka jejich historie v databázi SEE je alespoň 3 roky, dosahujeme úspěšnosti propojení 94,8%.

Následující tabulka ukazuje rozpad jednoznačných propojení dle úrovně finálního skóre.

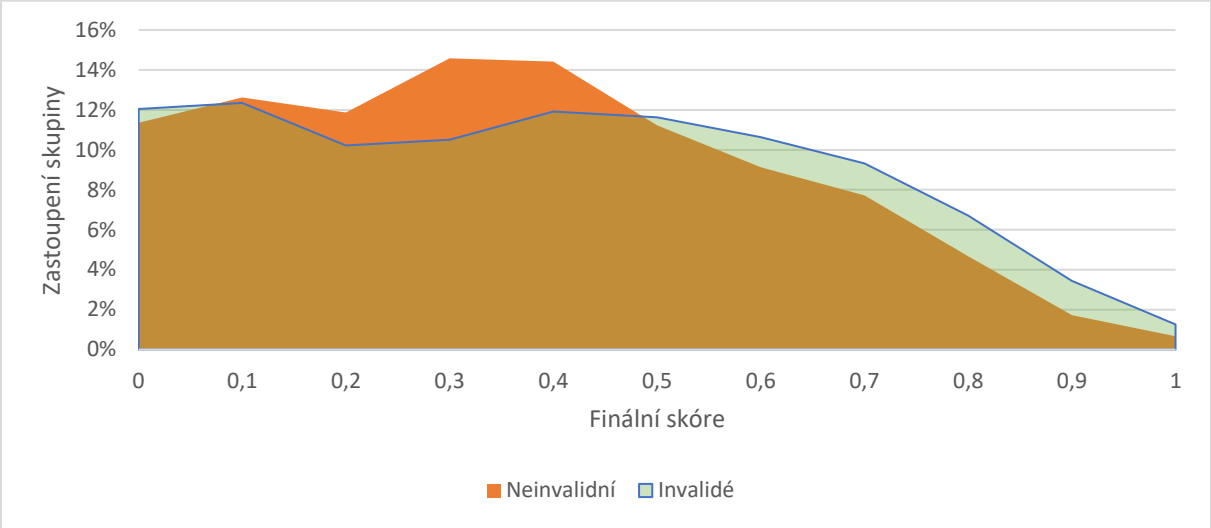
FINÁLNÍ SKÓRE	POČET PROPOJENÍ	PODÍL PROPOJENÍ
[0,50; 1,00)	983 549	29,7%
[0,25; 0,50)	1 143 228	34,5%
[0,00; 0,25)	1 184 388	35,8%

U osob, u kterých známe kompletní informaci (pohlaví, rok narození, měsíc narození, okres) a zároveň jsou v databázi SEE uvedeny v minimálně 3 letech, dostáváme následující rozpad dle finálního skóre.

FINÁLNÍ SKÓRE	POČET PROPOJENÍ	PODÍL PROPOJENÍ
[0,50; 1,00)	584 020	45,2%
[0,25; 0,50)	258 706	20,0%
[0,00; 0,25)	448 181	34,7%

Následující graf dále ukazuje rozložení finálního skóre a zastoupení ID osob, které byly napárovány z databáze STATMIN ANOD z pohledu invalidity. První skupina „Invalidé“ měla uveden v databázi STATMIN ANOD jako druh_hlavni IP, ID, IT nebo IM. Druhá skupina, „Neinvalidní“ měla uveden jiný druh_hlavni než IP, ID, IT nebo IM.

Z grafu lze vyčíst, že pro invalidy dosahovalo finální skóre neboli jistota propojení vyšší úrovně než v případě „neinvalidů“.



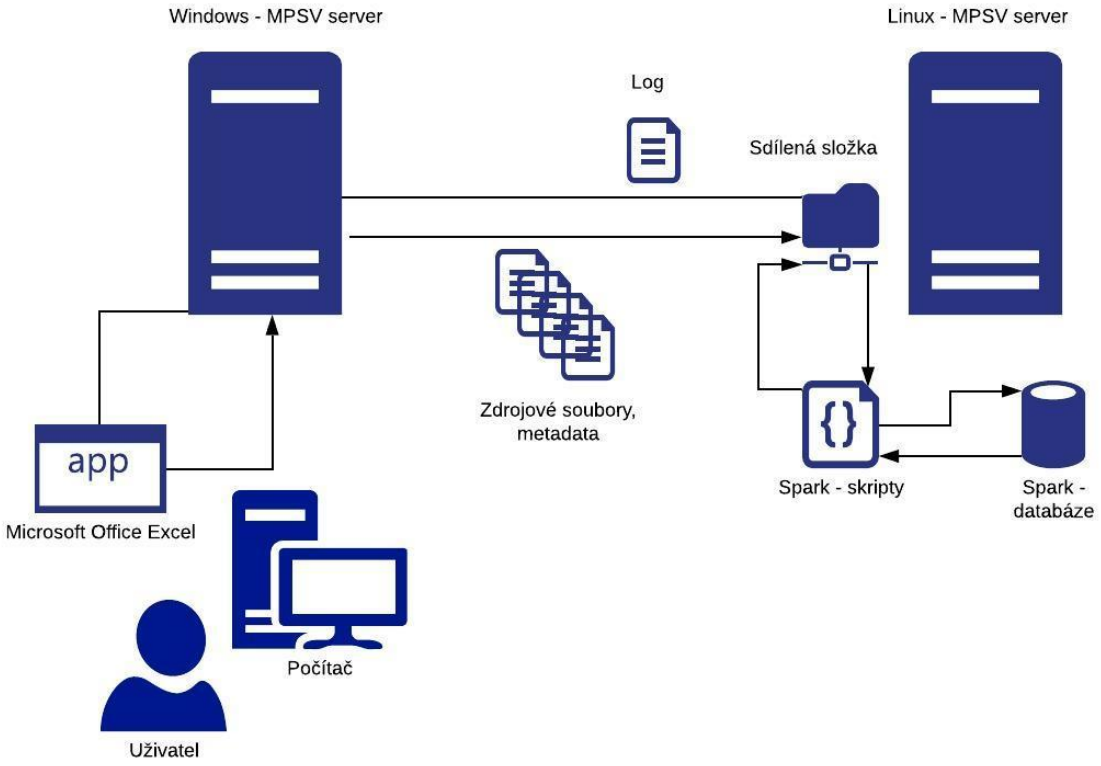
9. Ovládání nástroje

Smyslem této kapitoly je seznámit uživatele s ovládáním nástroje pro spojení databází STATMIN VZ, SEE20/21/22.

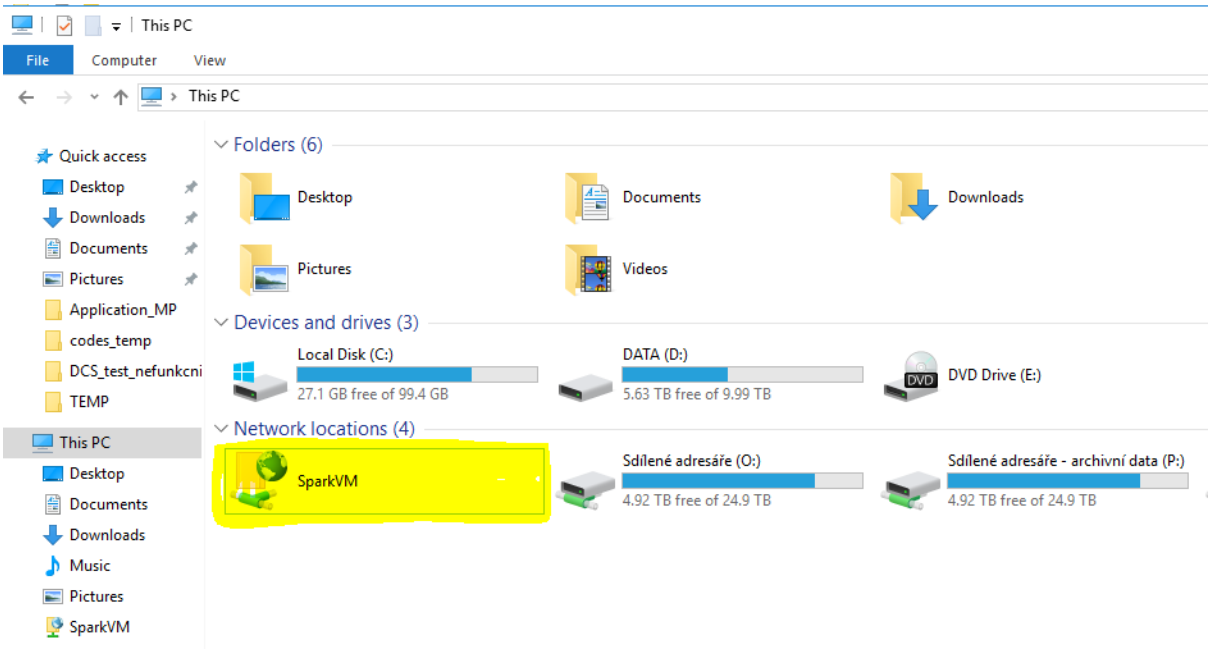
Hlavním vstupem do nástroje jsou databáze STATMIN_VZ, SEE20, SEE21, SEE22, INEP, STATMIN_ANOD a ISPV. Výstupem je databáze propojených identifikátorů z SEE a STATMIN VZ.

Ovládací nástroj je soubor vytvořený v MS Excel a slouží k jednoduchému a přehlednému spouštění výpočetních skriptů, jakož i k získání informací o průběhu výpočtů (dále Ovládací excel).

Celé řešení běží na dvou serverech MPSV. Ovládací excel je spouštěn na Windows Serveru („prophet1“), výpočty pak probíhají na Linux Serveru („prophet2“), kde jsou také uložena data potřebná pro chod aplikace. Architektura řešení je znázorněna na následujícím schématu:



Linuxový disk, na kterém musí být uložena vstupní data a na který jsou následně ukládány výsledky propojení, je namapován do prostředí Windows a lze tak k datům přistupovat a ukládat je jednoduše pomocí Průzkumníku. Linuxový disk se z pohledu Windows prostředí chová jako FTP server. Přenos souborů a ukládání vstupních dat na linuxový probíhá skrz Ovládací excel (odstavec výše má tedy spíš informativní charakter).



Složka s řešením má následující strukturu:

DELNEZ je hlavní adresář, který má tyto podadresáře/soubory:

- INPUT
- OUTPUT
- DelNez.xlsm

Do složky INPUT je třeba vložit všechny potřebné soubory (viz kapitola Zdrojová data a jejich příprava). Složka OUTPUT obsahuje výstupy propojení (viz kapitola Export výsledků). AppSEE.xlsm je Ovládací excel.

Celá složka DELNEZ může být umístěna kdekoliv, pro správnou funkcionalitu musí být zachována pouze struktura této složky.

Ovládací excel

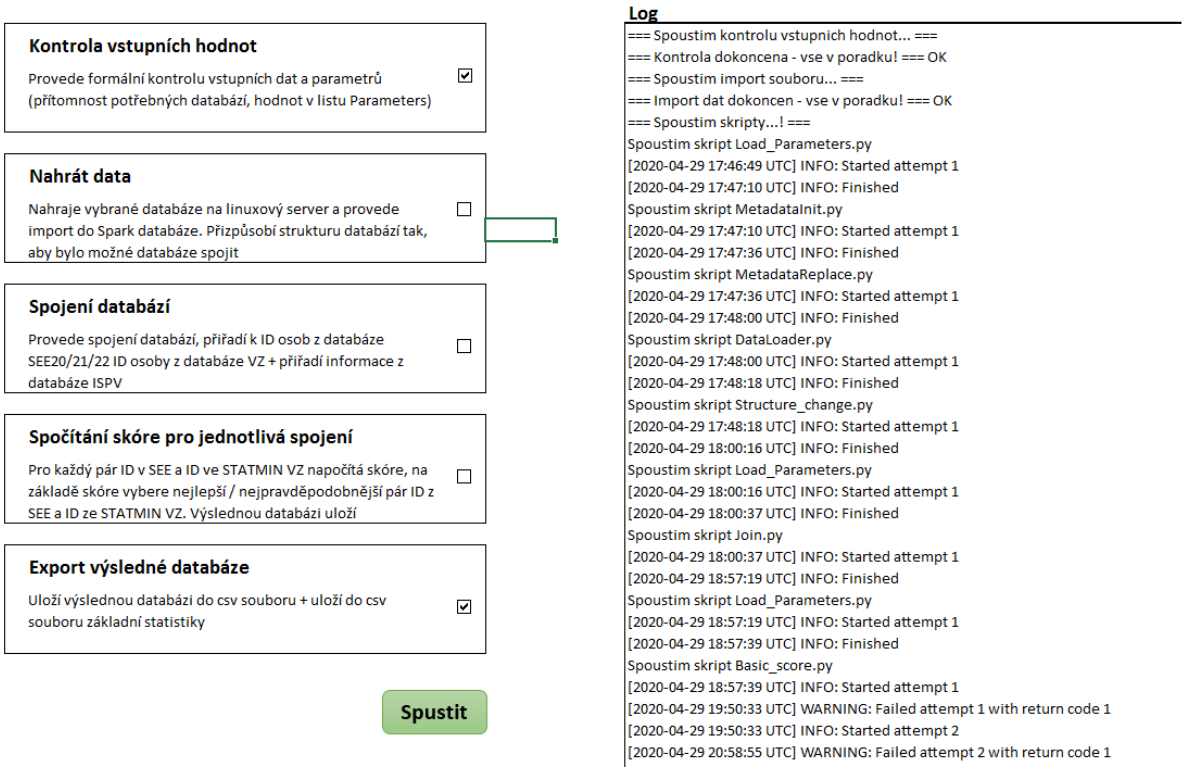
Slouží k jednoduchému a přehlednému spouštění výpočetních skriptů, přenosu potřebných souborů mezi servery, jakož i k získání informací o průběhu výpočtů.

Ovládací excel obsahuje následující listy:

- Dashboard
- Parameters
- Data_set
- Data_object
- Data_objec_field

Níže uvádíme detailní popis jednotlivých listů.

1. Dashboard



Tento list představuje spouštěcí panel celého řešení.

Vlevo jsou sekce se zaškrtávacím polem a dále pak tlačítko Spustit. Pomocí zaškrtávacího tlačítka lze zvolit, které kroky výpočtu se budou provádět a tlačítkem Spustit se pak spustí příslušné výpočty.

Vždy platí, že pro bezchybný průběh daného kroku musí být předtím spuštěny všechny předcházející výpočty. Např. pokud zaškrtneme „Spočítání skóre pro jednotlivá spojení“, program očekává, že již v minulosti proběhly kroky „Nahrát data“ a „Spojení databází“. Pokud tomu tak není, pak program skončí chybou.

„Kontrola vstupních hodnot“ probíhá vždy bez ohledu na to, která sekce je zaškrtnuta.

Popis jednotlivých kroků výpočtu:

a. Kontrola vstupních hodnot

Spouští se vždy.

Provede formální kontrolu parametrů na listech Parameters, Data_set a Data_object a zkontroluje, zda jsou ve složce INPUT přítomny všechny potřebné soubory.

b. Nahrát data

Provede přesun potřebných souborů ze složky INPUT do příslušné složky na linuxový disk.

Databáze INEP, STATMIN ANOD, STAMIN VZ, SEE20, SEE21, SEE22 a ISPV následně nahraje do Sparkové databáze a provede potřebnou změnu struktury nutnou k následnému spojení databází. Tato změna struktury se vztahuje ke kapitole Změna struktury zdrojových databází

c. Spojení databází

Provede spojení databáze ISVP v databází STATMIN VZ a vzniklou databázi spojí s databází SEE.

Tento krok se vztahuje ke kapitole Spojení databází VZ, SEE a ISPV.

d. Spočítání skóre pro jednotlivá spojení

Provede skórování jednotlivých propojení a vytvoří výslednou databázi s následující strukturou.

Tento krok se vztahuje ke kapitolám Výpočet podobnostního skóre (sloupec basic_score), Výpočet unikátnostního skóre(sloupec score_unique), Finální přiřazení(sloupec score_final).

see_idos	vz_idos	basic_score	score_unique	score_final
102419975	1006441740	0.012	0.663	0.0244

Tento výstup je uložen do složky OUTPUT ve formátu csv. Další výstupy jsou uvedeny v kapitole Export výsledků.

e. Export výsledné databáze

V tomto kroku se spočtou základní statistiky o počtech lidí ve vstupních a výstupních databázích a uloží se do složky OUTPUT.

Pravá část listu Dashboard je vyhrazena pro Log.

Log obsahuje informace o tom, zda jednotlivé kroky proběhly v pořádku. Je zde vždy uveden čas spuštění skriptu, čas jeho ukončení, název skriptu a také informace o tom, zda skript proběhl v pořádku anebo zda se vyskytla při výpočtu chyba.

V případě výskytu chyby je zde uvedena cesta k souboru, který obsahuje podrobné informace o chybě.

2. Parameters

Na listu Parameters lze nastavit některé parametry propojení a výpočtu skóre:

a) ref_year

Parametr, který udává maximální společný rok pro databáze STATMIN VZ a SEE20/21/22.

V době vytváření řešení byl maximálním společným rokem rok 2017, protože v databázi STATMIN VZ byly k dispozici pouze data do roku 2017.

b) see_od_tolerance

Tolerance ve dnech mezi dobou OD v databázi SEE a dobou OD v databázi STATMIN VZ (vycházíme z toho, že OD v SEE by neměla být menší než OD minus tolerance z databáze STATMIN VZ).

Tento parametr se vztahuje ke kapitole 5 Výpočet podobnostního skóre.

c) see_do_tolerance

Tolerance ve dnech mezi dobou DO v databázi SEE a dobou DO v databázi STATMIN VZ (vycházíme z toho, že DO v SEE by neměla být větší než DO plus tolerance z databáze STATMIN VZ).

Tento parametr se vztahuje ke kapitole 5 Výpočet podobnostního skóre.

d) vdoba_tolerance

Tolerance ve dnech, kdy platí, že délka neschopenky v SEE = doba vyloučená +/- tolerance ve STATMIN VZ.

Tento parametr se vztahuje ke kapitole 4 Spojení databází VZ, SEE a ISPV, 5 Výpočet podobnostního skóre.

e) materska_delka_od

Předpokládaný spodní interval trvání mateřské dovolené.

Tento parametr se vztahuje ke kapitole 5 Výpočet podobnostního skóre.

f) materska_delka_do

Předpokládaný horní interval trvání mateřské dovolené.

Tento parametr se vztahuje ke kapitole 5 Výpočet podobnostního skóre.

g) osetrovacka_delka_od

Předpokládaný spodní interval trvání OČR.

Tento parametr se vztahuje ke kapitole 5 Výpočet podobnostního skóre.

h) osetrovacka_delka_do

Předpokládaný horní interval trvání OČR.

Tento parametr se vztahuje ke kapitole 5 Výpočet podobnostního skóre.

i) ispv_confidance

Udává, jakou váhu chceme dát datům z ISPV neboli jak moc jim věříme (min = 0, max = 1).

j) vaha_podobnost

Udává, jakou váhu chceme přisoudit podobnostnímu skóre při výpočtu finálního skóre.

Finální skóre je vypočteno pomocí vztahu:

FINÁLNÍ SKÓRE = PODOBNOSTNÍ SKÓRE^vaha_podobnost * UNIKÁTNOSTNÍ SKÓRE^vaha_unikatnost.

k) vaha_unikatnost

Udává, jakou váhu chceme přisoudit unikátnostnímu skóre při výpočtu finálního skóre.

Finální skóre je vypočteno pomocí vztahu:

FINÁLNÍ SKÓRE = PODOBNOSTNÍ SKÓRE^vaha_podobnost * UNIKÁTNOSTNÍ SKÓRE^vaha_unikatnost.

l) export_result1

Udává, zda se do složky OUPUT vygeneruje tato výsledná databáze

- 0 – výstup se nevygeneruje,
- 1 – výstup se vygeneruje a uloží do složky OUTPUT.

Je třeba myslet na to, že vygenerování výstupu a následný export trvá určitý čas (řádově nízké desítky minut) a proto je na zvážení, zda je tento výstup vždy potřeba.

Výstup obsahuje jedinečné dvojice SEE_IDOS a VZ_IDOS, dále pak podobnostní skóre, unikátnostní skóre a finální skóre, které bylo pro tuto dvojici spočteno.

Jakkoliv je finální skóre vysoké, je maximální volné ze všech možný dvojic, které byly vytvořeny.

Příklad výstupu:

SEE_IDOS	VZ_IDOS	BASIC_SCORE	SCORE_UNIQUE	SCORE_FINAL
102419975	1006441740	0.0118	0.6634	0.0244
102520819	1006452740	0.0075	1.0	0.0201
103785473	1002485387	0.0385	0.2568	0.0429

m) export_result2

Udává, zda se do složky OUPUT vygeneruje tato výsledná databáze.

- 0 – výstup se nevygeneruje,
- 1 – výstup se vygeneruje a uloží do složky OUTPUT.

Je třeba myslet na to, že vygenerování výstupu a následný export trvá určitý čas (řádově nízké desítky minut) a proto je na zvážení, zda je tento výstup vždy potřeba.

Tento výstup obsahuje všechny dvojice, které byly vytvořeny, přičemž každé takové dvojici je přidělen příznak (sloupec Result), který udává, zda se jedná o nejlepší možnou a zároveň unikátní dvojici (hodnota 1), zda bylo nalezeno víc dvojic se stejným (nejvyšším) skóre (hodnota 2) a ostatní dvojice, které nemají nejvyšší skóre (hodnota 0).

Příklad výstupu:

SEE_IDOS	VZ_IDOS	BASIC_SCORE	SCORE_UNIQUE	SCORE_FINAL	RESULT
101603152	1006241490	0.0140	0.8031	0.0301	1
101603152	1003736671	0.0007	0.0005	0.0009	0
101603152	1006388459	0.0002	0.0654	0.0001	0

3. Data_set

Tento list obsahuje 3 sloupce DATA_SET_CODE, DATA_SET_DESCRIPTION a IS_ACTIVE. Informace obsažené v těchto sloupcích slouží ke správnému importu příslušných databází:

- DATA_SET_CODE – označení databáze (povoleny jsou hodnoty VZ, SEE20, SEE21, SEE22, INEP, ISPV, ANOD)
- DATA_SET_DESCRIPTION – informativní popis databází
- IS_ACTIVE – pole udává, zda se daná databáze bude nahrávat (1) nebo je již nahraná, a tudíž se již znovu nahrávat nebude (0). Pole IS_ACTIVE tedy umožňuje postupné nahrávání souborů. Není tedy třeba vždy dokola nahrávat již nahrané databáze.

Příklad:

DATA_SET_CODE	DATA_SET_DESCRIPTION	IS_ACTIVE
VZ	data z databáze STATMIN VZ	1
SEE20	data z databáze SEE20	1
SEE22	data z databáze SEE22	1
SEE21	data z databáze SEE21	1
INEP	data z databáze INEP	1
ISPV	data z databáze ISPV	0
ANOD	data z databáze STATMIN ANOD	1

Pro správné propojení databází STATMIN VZ a SEE20/21/22 jsou potřeba následující databáze

- STATMIN VZ – klíčová databáze
- SEE20 – klíčová databáze
- SEE21 – klíčová databáze
- SEE22 – klíčová databáze
- INEP – pro zjištění měsíce narození pro ID z databáze STATMIN VZ (stačí nahrát pouze jednou).
- ISPV – pro spočítání sloupce DNYPN viz kapitola Spojení databází VZ, SEE a ISPV. Databázi stačí nahrát pouze jednou. V případě, že nebude již dále aktualizována, nijak to zásadně neovlivní další fungování.
- ANOD – pro výpočet statistik propojení, viz další kapitola**.**

4. Data_object

Tento list slouží k nadefinování jednotlivých souborů, které se vztahují k té konkrétní databázi. Je zde tedy počítáno s případy, kdy máme např. dva soubory STATMIN VZ do roku 2017 a pak např. STATMIN VZ za rok 2018. Oba tyto soubory chceme nahrát do stejné databáze VZ.

Definice probíhá pomocí následujících metadat:

- DATA_OBJECT_CODE – přesný název souboru (rozlišujíc se malá a velká písmena) bez přípony (očekáváme soubory *.csv a *.txt)
- DATA_SET_CODE – označení cílové databáze (povoleny jsou hodnoty VZ, SEE20, SEE21, SEE22, INEP, ISPV, ANOD a zároveň jsou povoleny jen hodnoty, které se vyskytují ve sloupci DATA_SET_CODE na listu Data_set)
- COLUMN_DELIMITER – určuje oddělovač sloupců ve zdrojovém souboru
- QUOTE_CHAR – označuje typ uvozovek, které jsou použity pro obalení hodnot ve sloupcích typu String (textový řetězec)
- INSERT_MODE – povoleny jsou hodnoty append a overwrite:

    - Append použijeme, pokud chceme do cílové databáze data pouze přidat s tím, že zachováme data, která jsou v databázi již obsažena.
    - Overwrite nejdříve smaže obsah cílové databáze a poté nahraje požadovaná data.
    - Příklad: uvažujme, že máme dva soubory STATMIN VZ do roku 2017 a dále např. STATMIN VZ za rok 2018. Oba tyto soubory chceme nahrát do stejné cílové databáze s názvem VZ. U prvního souboru bude INSERT_MODE = overwrite, protože máme kompletní historii před rokem 2017 včetně. U druhého souboru bude INSERT_MODE = append, protože v rámci nahrání prvního souboru došlo k naplnění databáze VZ daty do roku 2017 včetně a druhý soubor pouze doplňuje tuto informaci o rok 2018. Overwrite u druhého souboru by způsobil smazání data z prvního souboru a nahrání dat za rok 2018.
- IS_ACTIVE – tento sloupec pouze označuje hodnotou 1 ty soubory, které chceme nahrát a hodnotou 0 soubory, které nahrát nechceme.

Příklad:

DATA_OBJECT_CODE	DATA_SET CODE	COLUMN DELIMITER	QUOTE CHAR	INSERT MODE	IS_ACTIVE
VZ_DO_2017	VZ	,	"	overwrite	1
SEE20_DO_2017	SEE20	,	"	overwrite	1
INEP_DO_2017	INEP	,	"	overwrite	1
ISPV_DO_2017	ISPV	,	"	overwrite	0
SEE22_DO_2018	SEE22	,	"	overwrite	1
SEE21_DO_2018	SEE21	,	"	overwrite	1
ANOD_DO_2017	ANOD	,	"	overwrite	1

5. Data_object_field

V rámci tohoto listu definujeme pro každý soubor (DATA_OBJECT_CODE) jednotlivé sloupce, jejich pořadí a typ, a to pomocí následujících metadat:

- DATA_OBJECT_CODE – obsahuje stejné hodnoty jako sloupec DATA_OBJECT_CODE na listu Data_object neboli název souboru bez přípony.
- FIELD_NAME – název sloupce ve zdrojovém souboru.
- POSITION – pozice sloupce ve zdrojovém souboru.
- DATA_TYPE – datový typ sloupce. Povolené hodnoty jsou LongType – dlouhé číslo, StringType – text, IntegerType – celé číslo, DoubleType – dlouhé desetinné číslo a DateType – datum.
- DATA_FORMAT – udává formát data v případě, že DATA_TYPE je DateType. Formát může být např. dd.MM.YYYY nebo dd/MM/YYYY atp.

Příklad:

DATA_OBJECT_CODE	FIELD_NAME	POSITION	DATA_TYPE	DATA_FORMAT
VZ_DO_2017	ID_VZZAM_AN	1	LongType	
VZ_DO_2017	ID_ELDP	2	LongType	
VZ_DO_2017	ID_OSOBA_AN	3	LongType	

VZ_DO_2017	POHLAVI	4	IntegerType

Export výsledků

Finálním výstupem celého řešení jsou statistiky a databáze obsahující ID osob z databáze SEE20/21/22 a k nim napárovaná ID osob z databáze STATMIN VZ.

1. **Statistiky**

   stats.csv – soubor obsahuje informace o počtech osob v různých fázích tvorby finální databáze.

   Příklad:

```
POCTY OSOB VE ZDROJOVYCH DATABAZICH
Pocet osob v SEE20:      4303363
Pocet osob v SEE21:      46349
Pocet osob v SEE22:      2504

Pocet osob ve STATMIN VZ:        7309989

==========

POCTY OSOB PO UPRAVACH NUTNYCH PRO SPOJENI
Pocet osob v SEE:        4002228
Pocet osob ve VZ:        4124651

==========

POCET OSOB VE SPOJENE DATABAZI
Pocet osob po spojeni:   1361325

==========

POCET OSOB PO ZAKLADNIM SKOROVANI
Pocet osob po zakladnim skorovani:       1360281

==========

CELKOVY POCET PROPOJENYCH OSOB
Celkovy pocet propojenych osob:          1360281
```

   skore_vek.csv – obsahuje informaci o rozložení skóre dle věkových skupin. SCORE_TOTAL je pro tyto účely zaokrouhleno na jedno desetinné místo.

   Příklad:

A	B	C	D	E	F	G	H	I	J	K
SCORE_TOTAL	VEK_1_10	VEK_11_20	VEK_21_30	VEK_31_40	VEK_41_50	VEK_51_60	VEK_61_70	VEK_71_80	VEK_81_90	VEK_91_100
0	0	88	20205	36262	35956	34555	21698	2503	172	0
0.1	0	117	24111	52055	45266	31134	18490	1727	98	0
0.2	0	57	16808	42743	33947	19016	10573	568	32	0
0.3	0	78	14850	35555	27698	16109	10611	390	14	0
0.4	0	103	14000	31488	26083	17438	11336	364	14	0
0.5	0	89	16252	29807	28934	23800	11739	171	12	0
0.6	0	59	19045	29610	33497	34577	16520	110	8	0
0.7	0	28	18105	25443	30819	38112	16810	93	11	0
0.8	0	33	21000	29870	37681	42294	14831	38	2	0
0.9	0	24	8980	10859	25426	41012	15167	22	0	0
1	0	2	838	939	3219	6994	3704	8	1	0

   skore_invalidita.csv – obsahuje informaci o rozložení skóre v závislosti na invaliditě. Data o invaliditě pocházejí z databáze STATMIN ANOD. SCORE_TOTAL je pro tyto účely zaokrouhleno na jedno desetinné místo.

   Příklad:

A	B	C	D	E
SCORE_TOTAL	POCET_NEINV	PROCENTO_NEINV	POCET_INV	PROCENTO_INV
0	136227	0.117742591	47272	0.114901571
0.1	154307	0.133369346	57707	0.140265378
0.2	111491	0.096362976	37708	0.091654858
0.3	95492	0.082534853	29773	0.072367669
0.4	91488	0.079074149	28011	0.068084868
0.5	100007	0.086437221	32549	0.079115147
0.6	118667	0.102565277	44559	0.108307224
0.7	113744	0.098310271	47990	0.116646776
0.8	132960	0.114918884	40366	0.098115519
0.9	90075	0.077852877	35788	0.086988014
1	12532	0.010831554	9690	0.023552975

2. Výsledné databáze

Výsledkem propojení databází jsou 2 výstupy (přičemž je možné zvolit, který se má vypsat; je možné vypsat i oba dva).

První z výstupů obsahuje jedinečné dvojice SEE_IDOS a VZ_IDOS, dále pak podobnostní skóre, unikátnostní skóre a finální skóre, které bylo pro tuto dvojici spočteno.

Jakkoliv je finální skóre vysoké, je maximální ze všech možný dvojic, které byly vytvořeny.

Příklad:

SEE_IDOS	VZ_IDOS	BASIC_SCORE	SCORE_UNIQUE	SCORE_FINAL
102419975	1006441740	0.0118	0.6634	0.0244
102520819	1006452740	0.0075	1.0	0.0201
103785473	1002485387	0.0385	0.2568	0.0429

Druhý z výstupů je podobný prvnímu výstupu, ale je obecnější. Tento výstup obsahuje všechny dvojice, které byly vytvořeny, přičemž každé takové dvojici je přidělen příznak (sloupec Result), který udává, zda se jedná o nejlepší možnou a zároveň unikátní dvojici (hodnota 1), zda bylo nalezeno víc dvojic se stejným (nejvyšším) skóre (hodnota 2) a ostatní dvojice, které nemají nejvyšší skóre (hodnota 0).

Příklad:

SEE_IDOS	VZ_IDOS	BASIC_SCORE	SCORE_UNIQUE	SCORE_FINAL	RESULT
101603152	1006241490	0.0140	0.8031	0.0301	1
101603152	1003736671	0.0007	0.0005	0.0009	0
101603152	1006388459	0.0002	0.0654	0.0001	0

V případě, že bude v rámci vstupních parametrů zvolen jen jeden z výstupů, vytvoří se pro každý z výstupů soubor, nicméně soubor pro nezvolený výstup bude prázdný.