

České vysoké učení technické v Praze  
Fakulta jaderná a fyzikálně inženýrská

Katedra softwarového inženýrství

Obor: Aplikace informatiky v přírodních vědách



# Analýza příčin vzniku shrinku produků společnosti na základě logistických dat

## Root Cause Analysis of Shrinkage Based on Logistics Data

DIPLOMOVÁ PRÁCE

Vypracoval: Bc. Anna Gruberová

Vedoucí práce: Ing. Martin Plajner, Ph.D.

Rok: 2023



České vysoké učení technické v Praze  
Fakulta jaderná a fyzikálně inženýrská

Katedra softwarového inženýrství

Akademický rok 2022/2023

## ZADÁNÍ DIPLOMOVÉ PRÁCE

**Student:** Bc. Anna Gruberová

**Studijní program:** Aplikace informatiky v přírodních vědách

**Název práce:** Analýza příčin vzniku shrinku produktů společnosti na základě logistických dat

**Název práce anglicky:** Root Cause Analysis of Shrinkage Based on Logistics Data

### Pokyny pro vypracování:

1. Seznamte se s procesy v dodavatelském řetězci. Pozornost věnujte štíhlé logistice a plýtvání v logistických procesech.
2. Proveďte přípravu dat – definujte shrink, rozpoznejte a označte jej v obdržených datech.
3. Na základě předchozího kroku najděte a analyzujte příčiny, které vedou ke vzniku shrinku produktů dané společnosti. Zjištěné výsledky vhodně vizualizujte.
4. Navrhněte automatizaci tohoto postupu, kterou by bylo možné aplikovat i na logistická data dalších společností.

**Doporučená literatura:**

- [1] JIRSÁK, P., MERVART M., VINŠ M. *Logistika pro ekonomy - vstupní logistika*. 1. vydání. Praha: Wolters Kluwer ČR, 2012. ISBN 978-80-7357-958-6.
- [2] SUTHERLAND J., BENNET B. *The Seven Deadly Wastes of Logistics: Applying Toyota Production System Principles to Create Logistics Value*. Bethlehem, PA: Lehigh University, 2007.
- [3] HASTIE, T., TIBSHIRANI R., FRIEDMAN J. H. *The elements of statistical learning: data mining, inference, and prediction*. 2nd ed. New York: Springer, c2009. Springer series in statistics. ISBN 978-0-387-84857-0.

**Jméno a pracoviště vedoucího práce:**

**Ing. Martin Plajner, Ph.D.**

Oddělení matematické teorie rozhodování, Ústav teorie informace a automatizace AV ČR, v.v.i.

.....  
vedoucí práce

**Datum zadání diplomové práce:** 12. 10. 2022

**Termín odevzdání diplomové práce:** 3. 5. 2023

Doba platnosti zadání je dva roky od data zadání.

.....  
garant oboru

.....  
vedoucí katedry

.....  
děkan

V Praze dne 12. 10. 2022

## **Prohlášení**

Prohlašuji, že jsem svou bakalářskou práci vypracovala samostatně a použila jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v přiloženém seznamu.

V Praze dne .....  
.....  
Bc. Anna Gruberová

## **Poděkování**

Chtěla bych poděkovat za vedení mé bakalářské práce, za cenné rady a připomínky k tvorbě této práce a za čas strávený touto pomocí.

Bc. Anna Gruberová

*Název práce:*

**Analýza příčin vzniku shrinku produktů společnosti na základě logistických dat**

*Autor:* Bc. Anna Gruberová

*Studijní program:* Aplikace přírodních věd

*Obor:* Aplikace informatiky v přírodních vědách

*Druh práce:* Diplomová práce

*Vedoucí práce:* Ing. Martin Plajner, Ph.D.

Oddělení matematické teorie rozhodování, Ústav teorie informace a automatizace AV ČR, v.v.i.

*Konzultant:* –

*Abstrakt:*

*Klíčová slova:* Datová analýza, Logistika

*Title:*

**Root Cause Analysis of Shrinkage Based on Logistics Data**

*Author:* Bc. Anna Gruberová

*Abstract:*

*Key words:* Data Analysis, Logistics



# Obsah

<b>Úvod</b>	<b>11</b>
<b>1 Logistika</b>	<b>13</b>
1.1 Definice logistiky . . . . .	13
1.2 Štíhlá logistika . . . . .	13
1.2.1 Toyota Production System . . . . .	14
1.2.2 Plýtvání v logistických procesech . . . . .	18
<b>2 Použité metody</b>	<b>19</b>
2.1 Redukce dimenzionality . . . . .	19
2.1.1 Analýza hlavních komponent . . . . .	19
2.1.2 Korespondenční analýza . . . . .	21
2.2 Korelační analýza . . . . .	23
2.2.1 Korelační koeficient . . . . .	23
2.2.2 Další způsoby měření závislosti . . . . .	24
2.3 Metoda GUHA . . . . .	26
2.3.1 Základní princip metody . . . . .	27
2.3.2 Důležité pojmy . . . . .	27
2.3.3 Procedury . . . . .	29
2.4 Nástroje . . . . .	32
2.4.1 Python – Jupyter Notebook . . . . .	32
2.4.2 Power BI . . . . .	32
<b>3 Shrink</b>	<b>37</b>
3.1 Definice . . . . .	37
3.2 Typy shrinků . . . . .	37
<b>4 Zpracování dat</b>	<b>41</b>
4.1 Popis obdržených dat . . . . .	41
4.2 Příprava vzorku dat pro další analýzy . . . . .	45
4.2.1 Výběr dat . . . . .	45
<b>5 Korelační analýza</b>	<b>55</b>
5.1 Postup . . . . .	55
5.2 Implementace . . . . .	57
5.2.1 Vstupy a výstupy . . . . .	57
5.2.2 Spuštění analýzy . . . . .	58
5.2.3 Popis funkcí a struktura kódu . . . . .	59

5.3	Výsledky . . . . .	66
<b>6</b>	<b>Analýza pomocí metody 4ftMiner</b>	<b>71</b>
6.1	Hypotézy . . . . .	71
6.2	Produkty nepopsané korelační analýzou . . . . .	77
<b>7</b>	<b>Vizualizace dat</b>	<b>79</b>
7.1	Popis řešení . . . . .	79
7.1.1	Metriky . . . . .	80
7.1.2	Reporting . . . . .	81
7.2	Výsledky . . . . .	86
	<b>Závěr</b>	<b>95</b>
	<b>Literatura</b>	<b>96</b>
	<b>Přílohy</b>	<b>101</b>
<b>A</b>	<b>Obsah přiloženého CD</b>	<b>101</b>

# Úvod

Tato diplomová práce se zabývá

První kapitola se věnuje

V následující kapitole

Ve třetí kapitole se nachází

Poslední kapitola

- Data - jak jsou data uložená v DB u zákazníka - provázané - SQL příkazy - výběr proměnných - target hodnoty - cost, množství - produktová Hierarchie - Vymazaní outlierů - outlier metody - businessově
- Co vysvetluje target - Miner - PCA
- Korelační analýza mezi produkty v rámci kategorie - korelace - Rozčlenění produktů
- Vizualizace dat - Jak funguje PBI - Seznam metri



# Kapitola 1

## Logistika

### 1.1 Definice logistiky

Logistika zahrnuje všechny operace, které se týkají doručení zboží nebo služeb od výrobce k zákazníkovi, s výjimkou samotné výroby zboží nebo provádění služby. Výrobou je naopak rozuměno vše, co mění podobu materiálu. Během výroby se však logistika uplatňuje, například jako přesun materiálu nebo polotovarů mezi jednotlivými výrobními zařízeními. Operace lze rozdělit do tří hlavních toků: materiálový, informační a finanční tok. Materiálový obsahuje všechny pohyby týkající se fyzického materiálu, tedy jeho získávání, přesuny a skladování, a to jak mezi zákazníky, dodavateli či výrobními areály a sklady, tak i vnitřní pohyby mezi produkčními linkami nebo skladovými pozicemi. Informační tok popisuje procesy vznikající během materiálového toku, dále se do něj řadí analýzy již proběhlých toků a plánování a předpovědi budoucích toků. Poslední kategorie, finanční tok mapuje náklady způsobené předešlými dvěma zmíněnými toky.[1]

Pojem logistika je úzce propojen s pojmem Supply Chain Management (SCM)<sup>1</sup>. Zatímco logistika se zabývá toky zboží, služeb či lidí, Supply Chain Management zahrnuje operace logistiky, navíc ale sleduje vztahy mezi procesory, které koordinuje a optimalizuje za účelem naplnění určitých cílů. Tímto cílem bývá často snížení nákladů v rámci částí procesu nebo zvýšení konkurenceschopnosti podniku [4]. Supply Chain Management se tedy prolíná s pojmem logistika a bývají často zaměňovány. Důvodem může být i to, že se jedná o nový pojem, který byl poprvé použitý v roce 1982.[2]

### 1.2 Štíhlá logistika

Štíhlost neboli *lean* je koncept neustálého vylepšování procesu vytváření produktu nebo služby pomocí odstranění jakéhokoli plýtvání. Plýtváním rozumíme jakoukoli

---

<sup>1</sup>Do češtiny lze Supply Chain Management přeložit jako řízení či správa dodavatelského řetězce. V českém prostředí se používá jak anglická tak česká podoba.

činnost, která v očích zákazníka nezvyšuje hodnotu produktu a tedy není ochotný za tuto činnost zaplatit ve formě vyšší prodejní ceny. Z této definice plýtvání je patrné, že pohled zákazníka hraje důležitou roli při vytváření hodnoty produktu ve štíhlých systémech.[21, 22]

Svůj původ nachází štíhlá logistika na začátku 20. století, kdy Henry Ford zavedl pohyblivou montážní linku při výrobě automobilu Ford modelu T. Tato linka měla za následek několikanásobné snížení výrobního času a odstartovala sériovou výrobu aut. Díky čemuž se snížila prodejní cena, a automobily tak byly dostupné nejen nejbohatší vrstvě společnosti. Po druhé světové válce navázala automobilová společnost Toyota Motor Company na Fordovu efektivní montážní linku a vytvořila systém nazvaný Toyota Production System (TPS), který je přímým předchůdcem štíhlé logistiky.[20]

### 1.2.1 Toyota Production System

Toyota Production System je založen na pěti základních principech. Nejdůležitějším krokem je odstranit plýtvání. Je třeba se soustředit na jednotlivé procesy a na vazby mezi nimi. Pomocí metody genchi genbutsu<sup>2</sup> se nasbírají data a informace o procesech přímo na místě, kde procesy probíhají, aby případné problémy a zdroje plýtvání mohly být přesně určeny. Po této analýze se aplikuje přístup řešení problémů zvaný kaizen<sup>3</sup>, jehož cílem je kontinuální zlepšování procesů. Posledním z principů je dodržování vzájemného respektu mezi všemi oddělení společnosti, jak vedoucími pracovníky, tak zaměstnanci u výrobních linek. [20]

V TPS je plýtvání rozděleno do tří kategorií - Muda (plýtvání), Mura (nevyyáženost) a Muri (přetěžování) [23]. V následující části jsou podrobněji popsány jednotlivé typy.

#### Muda

Japonské označení Muda v překladu znamená plýtvání, neužitečnost či marnost. Muda zahrnuje všechny činnosti, které nepřispívají ke zvyšování hodnoty produktu. Mudu lze rozdělit na dva podtypy – 1. typ zahrnuje aktivity, které jsou nezbytné pro koncového zákazníka, např. testování, zda je produkt nebo služba bezpečná. Druhý typ obsahuje ty procesy, které již zákazník nepotřebuje, či dokonce nechce, neboť mohou mít vliv na rychlosť výroby produktu (výkonu služby) nebo přímo na jeho kvalitu.

Taiichi Ohno, manažer ve společnosti Toyota, identifikoval sedm typů plýtvání, někdy nazývané *seven deadly wastes*. Klasifikace a popis včetně příkladů je uveden níže [20]:

1. **Nadprodukce** – Pokud je vyrobeno více produktů, než je možné expedovat k zákazníkovi, nebo více materiálu, než kolik je požadováno k další výrobě či okamžité spotřebě.

<sup>2</sup>Genchi v překladu znamená skutečná lokace a genbutsu skutečná věc.

<sup>3</sup>Kaizen je japonský překlad slova zlepšení.

2. **Zpoždění/čekání** – Jakákoli prodleva mezi dvěma na sebe navazujícími procesy, např. čekání jedné montážní linky na meziprodukty z jiné linky vlivem rozdílných výrobních časů nebo vlivem nedostatečné výrobní kapacity jednoho ze strojů, dále sem patří také čekání zaměstnanců z důvodu kontroly odvedené práce, pomalého načítání počítačového programu nebo čekání na konkrétní instrukce k výkonu práce [21].
3. **Transport** – Zbytečný přesun produktů, materiálů nebo informací. Tento transport navíc může vést k poškození produktu. Příkladem tohoto typu plýtvání může být situace, kdy materiál, který je nejvíce potřebný pro výrobu produktů je umístěn v největší vzdálenosti, nebo pokud přístup k jedné položce ve skladu je blokovaný jinými položkami.
4. **Pohyb** – Zbytečný pohyb lidí, vzniklý špatným rozmístěním objektů v prostoru, např. nepřiměřeně dlouhotrvající chůze, natahování se pro předměty, vyhýbání se lidem či předmětům.
5. **Skladování** – Pokud je naskladněno více surovin, rozpracovaných výrobků a hotových produktů, než kolik je požadováno, např. předčasná dovážka položek do skladu, chyba v dodávce, naskladnění položek do zásoby tzv. pro jistotu nebo z důvodu množstevní slevy.
6. **Nadbytečné zpracování** – Při výrobě dochází k použití více energie nebo prostředků než nutné, nebo je vytvořen koncový produkt, který má vyšší hodnotu, než jaký je dohodnutý a požadovaný standard.
7. **Defekty** – Produkty či meziprodukty, které je nutné přepracovat nebo odstranit z výroby z důvodu vady.

Tyto podoby plýtvání aplikované v TPS byly inspirací pro identifikaci sedmi typů plýtvání v logistice [20, 5]:

1. **Nadprodukce** – V případě logistiky je nadprodukce chápána jako doručení produktů dříve nebo ve větším množství než bylo požadováno.
2. **Zpoždění/čekání** – Jakákoli prodleva mezi dvěma na sebe navazujícími procesy, např. čekání na převoz meziproduktů mezi dvěma výrobními linkami, příjezd kamionu mimo časové okno, doba mezi příjezdem kamionu a jeho naložením nebo čas mezi přijetím objednávky a zahájením její realizace.
3. **Transport** – Zbytečný přesun produktů, materiálů nebo informací, např. materiál, který je nejvíce potřebný pro výrobu produktů je umístěn v největší vzdálenosti, nebo pokud přístup k jedné položce ve skladu je blokovaný jinými položkami.
4. **Pohyb** – Zbytečný pohyb lidí, např. vzniklý špatnou organizací předmětů ve skladu, kdy položky, ke kterým se nejčastěji přistupuje, jsou v méně přístupných pozicích skladu, nebo dokonce sklad není strukturovaný vůbec, nebo nutnost změnit trasu při převozu položek ve skladu kvůli nedostatečně širokým uličkám.

5. **Skladování** – Pokud je naskladněno více surovin, rozpracovaných výrobků a hotových produktů, než kolik je požadováno, např. předčasná dovážka položek do skladu, chyba v dodávce, naskladnění položek do zásoby tzv. pro jistotu.
6. **Prostor** – Neoptimální využití dostupného místa, např. nedostatečná výška regálů ve skladech, nevyužitá kapacita regálů, neoptimální naložení kamionu, přetížení dostupných kapacit.
7. **Defekty** – Činnosti, které způsobí nutnost opakovat určitý proces, znehodnocení produktu nebo zvýší náklady, např. špatné zavezení produktu, špatné nebo chybějící označení produktu, chyby v evidenci.

V devadesátých letech, kdy se metody TPS začaly aplikovat ve společnostech, byl mezi sedm typů plýtvání Muda začleněn osmý typ - Dovednosti. V tomto případě dochází k neefektivitě kvůli nevyužití lidského potenciálu a talentů jednotlivých zaměstnanců. K tomu může docházet například striktním rozdělením na manažery a zaměstnance, kde role zaměstnanců je poslouchat nařízení shora a vykonávat práci tak, jak byla navržena vedoucími pracovníky. Avšak právě zaměstnanci pracující přímo v terénu lépe identifikují případné problémy a snadněji naleznou řešení díky svým zkušenostem.[21]

## Mura

Mura lze přeložit jako nestejnoměrnost, nevyrovnanost a nepravidelnost. Jedná se o plýtvání vznikající špatnou provázaností jednotlivých procesů a to jak interních, tak externích. Následkem nevyváženosti je pak vznik plýtvání Muda. [23, 5]

Plýtvání v podobě Mura se rozlišuje jak v procesech informačního, tak hmotného toku. V případě informačního toku je nejvýznamnějším zdrojem plýtvání situace, kdy je chybně predikována poptávka mezi jednotlivými články logistického řetězce. Ignorování vztahů mezi procesy může vést k chybovosti i v řádu desítek procent. Informace, jejichž opomíjení způsobuje chybovost předpovídání poptávky, mohou být např. v jaké fázi životního cyklu se výrobek nachází, plánování promoakcí nebo výrobní a logistická omezení dodavatelů. Další zdroj Mura v informačním toku je nedostatečná znalost stavu zásob mezi dodavatelem a odběratelem. Následkem čehož dochází k méně častým zavázkám avšak s větším objemem, což vede k vyšším pojistným zásobám ve skladech. Většinu zmíněných situací lze eliminovat aplikováním konceptu „Just-in-Time“ do jednotlivých procesů. Plýtvání také vzniká při administrativě, pokud nejsou vhodně standardizované dokumenty používané v logistickém řetězci. Příkladem může být špatná evidence pohybů ve skladu či tvorba objednávek. Nesjednocenost v administrativních procesech vede ke zpomalení navazujících činností nebo dokonce k chybám, které způsobí nemožnost dokončení celého procesu. Pak je nutné vybrané procesy provést znovu a napravit chyby.[5]

Plýtvání v hmotném toku je přímým důsledkem chyb vznikajících v informačním toku. Lze identifikovat i takové zdroje plýtvání, které nesouvisejí přímo s informačním tokem, a to například dodržování různých standardů přepravních prostředků na

straně dodavatele a odběratele. To má pak za následek nadbytečné překládání materiálu do podoby, kterou druhá strana používá a se kterou je schopna následně efektivněji manipulovat.[5]

## Muri

Pojem Muri označuje přetěžování. Muri často vzniká při snaze zvýšit produktivitu a odstranit tak předešlé typy plýtvání, v konečném důsledku může ale vést k výrazně větší chybovosti i celkovému selhání. Přetížení mohou být zaměstnanci, ale i stroje. V obou případech vytížení na více než 100 % se může projevit na snížení kvality výstupu. Lidé mohou být méně pozorní a může docházet k nehodám, které mohou v menší či větší míře negativně ovlivnit i větší část logistického řetězce. Stroje mohou produkovat zmetkové výrobky, nebo může dojít k jejich poškození až zničení.[5, 23]

## Příklad plýtvání Muda, Mura a Mudi

Všechny tři zmíněné typy plýtvání Muda, Mura a Muri jsou navzájem propojené. Tuto skutečnost je třeba brát v potaz při řešení zefektivňování procesů a eliminaci plýtvání. Pro představu je uvedena následující situace. Společnost potřebuje zákazníkovi přivézt šest tun materiálu, uloženého ve stejných jednotunových kontejnerech. Možné způsoby řešení této úlohy jsou znázorněny na obr. 1.1.[23]

Nejjednodušší možností je naložit na jeden kamion veškerý požadovaný materiál. V takovém případě společnost ušetří na počtu vozidel a eliminuje tak plýtvání přepravními prostředky, ušetří čas při nakládce a vykládce, protože není nutné obsluhovat více vozidel, zároveň. Na druhou stranu ale hrozí přetížení kamionu. Následkem přetížení se může zvýšit riziko nehody vozidla, firma může být pokutována nebo vozidlu nemusí být umožněn vjezd na určitá místa.

Opačným extrémem je použít tři kamiony, každý se dvěma tunami materiálu. Potom ale není efektivně využita dostupná kapacita a je patrné, že dochází k mnoha druhům plýtvání typu Muda.

Třetí možností je využití dvou kamionů, kdy první je naložen čtyřmi a druhý dvěma tunami. Toto rozložení nepodléhá žádným pravidlům a patrně proces nakládky není dostatečně spjatý s ostatními procesy nebo neprobíhá správný přenos informací o požadavcích mezi jednotlivými procesy. Nakládka a vykládka prvního velmi naloženého kamionu vyžaduje více času než druhého kamionu. Z toho plyne, že budou možné v dostupném čase stihnout obsloužit první kamion a dochází k přetížení, anebo v případě druhého kamionu je zbude velké množství času a zaměstnanci zbytečně čekají. Z této volby plyne, že plýtvání typu Mura může způsobit Mudu i Muru.[23]

Optimální řešení je naložit dva kamiony po třech tunách, což je jejich ideální kapacita. V takovém případě společnost minimalizuje za daných podmínek všechny tři typy plýtvání. V reálném světě jsou situace mnohonásobně komplexnější a ne vždy existuje jednoznačné optimální řešení, které je navíc snadno dosažitelné. Důležité ale



Muri = overburdened



Mura = unevenness, fluctuation, variation



Muda = waste



No Muri, Mura, or Muda

Obrázek 1.1: Příklady plýtvání Muda, Mura a Muri při transportu šesti tun materiálu.[23]

je soustředit se na všechny tři typy současně, protože optimalizace pouze jednoho kritéria může způsobit jiný druh plýtvání nebo kolaps části systému.

V roce 2011 bylo realizováno dotazníkové šetření Vysokou školou ekonomickou v Praze, které mapovalo, kolik procent logistických expertů se zabývá odstraněním změňovaných tří typů plýtvání. Plýtvání Muda se snaží odstranit z logistických procesů 72 % respondentů, Murou se zabývá 39 % a plýtvání Muri řeší 30 % dotazovaných.[5]

### 1.2.2 Plýtvání v logistických procesech

Tato sekce se zabývá třinácti vybranými logistickými procesy z hlediska plýtvání, jak jsou uvedeny v knize *Logistika pro ekonomy – Vstupní logistika*. Analýza vychází z již zmíněného dotazníkového šetření z roku 2011.

# Kapitola 2

## Použité metody

### 2.1 Redukce dimenzionality

#### 2.1.1 Analýza hlavních komponent

Analýza hlavních komponent (anglicky *Principal component analysis*, dále jako PCA) je statistická metoda využívaná pro extrakci proměnných, redukci vícedimenziálních dat nebo vizualizaci dat. Lze ji aplikovat pouze na kvantitativní data s numerickými, spojitémi hodnotami, neboť metoda využívá lineární algebraické techniky, jako je například kovarianční matice, pro jejíž výpočet se předpokládají spojité hodnoty.

Jednotlivá pozorování obsažená v datech bývají popsána několika různými příznaky. Tyto příznaky jsou často vzájemně korelované a obsahují šum. Metoda PCA dovede extrahovat pouze důležité informace z proměnných a snížit šum. K tomu je třeba vypočítat nové ortogonální proměnné, nazývané hlavní komponenty, které se získají jako lineární kombinace původních proměnných [7]. Hlavní komponenty reprezentují směry největšího rozptylu původních dat a jsou řazeny podle své významnosti. Jinými slovy, první hlavní komponenta zachycuje co nejvíce variability v datech, druhá hlavní komponenta zachycuje co nejvíce variability, která nebyla zachycena první hlavní komponentou, pro zbylé komponenty analogicky. [9]

#### Princip

Předpokládáme množinu dat  $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_N)$ , kde  $N$  je počet pozorování a každý vektor  $\mathbf{x}_i$  přísluší jednomu pozorování popsanému  $M$  proměnnými.  $\mathbf{X}$  je potom matice rozměru  $N \times M$  vstupních dat. Dále je definovaný výběrový průměr  $\bar{\mathbf{x}}$  jako

$$\bar{\mathbf{x}} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i, \quad (2.1)$$

a výběrová kovarianční matice  $\mathbf{C}$

$$\mathbf{C} = \frac{1}{N} \sum_{i=1}^N (\mathbf{x}_i - \bar{\mathbf{x}}). \quad (2.2)$$

První hlavní komponentu, která popisuje největší rozptyl dat označíme  $y_{1i}$  a vypočteme následovně jako lineární kombinaci původních proměnných

$$y_{1i} = \mathbf{a}_1^\top (\mathbf{x}_i - \bar{\mathbf{x}}), \quad \text{pro } i = 1, \dots, N, \quad (2.3)$$

kde  $\mathbf{a}_1 = (a_{11}, \dots, a_{M1})^\top$  je vektor vah.

Optimální vektor  $\mathbf{a}_1$  je takový vektor, který maximalizuje výběrový rozptyl nové proměnné  $y_{1i}$  za podmínky  $\mathbf{a}_1^\top \mathbf{a}_1 = 1$ . Pakliže je výběrový rozptyl  $y_{1i}$  definován jako

$$D(y_{11}, \dots, y_{1N}) = \mathbf{a}_1^\top \mathbf{C} \mathbf{a}_1 \quad (2.4)$$

můžeme maximalizační úlohu vyřešit pomocí metody Lagrangeových multiplikátorů. Lagrangeova funkce s parametrem  $\lambda_1$  má následující tvar

$$\mathcal{L}(\mathbf{a}_1, \lambda_1) = \mathbf{a}_1^\top \mathbf{C} \mathbf{a}_1 - \lambda_1 (\mathbf{a}_1^\top \mathbf{a}_1 - 1). \quad (2.5)$$

Derivaci funkce položíme rovnou nule

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \mathbf{a}_1} &= 2\mathbf{C}\mathbf{a}_1 - 2\lambda_1 \mathbf{a}_1 \stackrel{!}{=} 0 \\ (\mathbf{C} - \lambda_1 \mathbf{I})\mathbf{a}_1 &\stackrel{!}{=} 0, \end{aligned}$$

kde  $\mathbf{I}$  je jednotková matice.

Řešíme soustavu lineárních rovnic pro neznámý parametr  $\mathbf{a}_1$ , která má řešení právě tehdy, když je matice  $\mathbf{C} - \lambda_1 \mathbf{I}$  singulární, tedy platí, že její determinant je roven nule.  $\lambda_1$  je pak největší vlastní číslo matice  $\mathbf{C}$  a  $\mathbf{a}_1$  vlastní vektor příslušný tomuto vlastnímu číslu. Toto tvrzení se matematicky zapíše následovně

$$\mathbf{C}\mathbf{a}_1 = \lambda_1 \mathbf{a}_1. \quad (2.6)$$

Po vynásobení vektorem  $\mathbf{a}_1$  zleva získáme řešení pro maximální rozptyl proměnné  $y_{1i}$

$$D(y_{11}, \dots, y_{1N}) = \mathbf{a}_1^\top \mathbf{C} \mathbf{a}_1 = \lambda_1. \quad (2.7)$$

Druhá hlavní komponenta

$$y_{2i} = \mathbf{a}_2^\top (\mathbf{x}_i - \bar{\mathbf{x}}), \quad \text{pro } i = 1, \dots, N, \quad (2.8)$$

se vypočte obdobným způsobem s přidanou podmínkou ortogonality vzhledem k první hlavní komponentě – druhá hlavní komponenta nesmí být korelovaná s předchozí, první hlavní komponentou. Potom popisuje druhý největší možný rozptyl v datech.



Obrázek 2.1: Znázornění dvou hlavních komponent na pro dvě proměnné. Zdroj: vlastní.

Znázornění dvou hlavních komponent ve dvoudimenzionálním prostoru je vyobrazeno na obrázku 2.1. Vektor  $\mathbf{a}_2$  se opět získá jako jednotkový vlastní vektor kovarianční matice  $\mathbf{C}$  příslušící druhému největšímu vlastnímu číslu  $\lambda_2$ .[7, 9]

Získání předpisů pro další hlavní komponenty je analogické. Obecně lze zapsat metodu PCA a převod původních proměnných následujícím maticovým zápisem

$$\mathbf{Y} = \mathbf{XA}, \quad (2.9)$$

kde  $\mathbf{Y}$  obsahuje komponenty  $\mathbf{y}_1, \mathbf{y}_2, \dots$ ,  $\mathbf{X}$  je matice vstupních dat,  $\mathbf{A}$  je matice vlastních vektorů kovarianční matice  $\mathbf{C}$ . Pro matici  $\mathbf{A}$  zároveň platí  $\mathbf{C} = \mathbf{A}\Lambda\mathbf{A}^\top$ , kde  $\Lambda$  je diagonální matice vlastních čísel  $\mathbf{C}$ .[8]

### 2.1.2 Korespondenční analýza

Vícenásobná korespondenční analýza (anglicky *Multiple correspondence analysis*, dále jako MCA) je metoda, která umožnuje popsat vztahy mezi daty, které jsou popsané kategorickými proměnnými, vytvořením kontingenční tabulky. V případě, že se popisuje vzájemná relace pouze dvou proměnných, se použije základní korespondenční analýza<sup>1</sup>. MCA je alternativou k PCA, pokud jsou analyzovanými daty kategorická data. [12]

#### Značení

Nechť  $\mathbf{N}$  je matice dat s rozměry  $I \times J$ , kde  $I$  odpovídá počtu pozorování a  $J$  je počet kategorií. Matice  $\mathbf{N}$  je převedena na korespondenční matici  $\mathbf{P}$  vydelením matice  $\mathbf{N}$  jejím celkovým součtem  $n = \sum_{i=1}^I \sum_{j=1}^J n_{ij} = \mathbf{1}_I^\top \mathbf{N} \mathbf{1}_J$ . To zaručuje, že součet prvků

---

<sup>1</sup>anglicky *correspondence analysis* (CA)

matice  $\mathbf{P}$  je roven jedné. Tyto kroky lze shrnout následujícím matematickým zápisem

$$\mathbf{P} = \frac{1}{n} \mathbf{N}, \quad \mathbf{P} = \{p_{ij}\}, \quad \sum_{i=1}^I \sum_{j=1}^J p_{ij} = 1. \quad (2.10)$$

Součet  $i$ tého řádku, resp. součet  $j$ tého sloupce je značen následovně

$$r_i = \sum_{j=1}^J p_{ij} \quad \text{pro } i = 1, \dots, I,$$

$$c_j = \sum_{i=1}^I p_{ij} \quad \text{pro } j = 1, \dots, J.$$

Vektor  $\mathbf{r} = \mathbf{P}\mathbf{1}_J$  obsahuje všechny řádkové součty matice  $\mathbf{P}$ , analogicky vektor  $\mathbf{c} = \mathbf{P}^\top\mathbf{1}_I$  obsahuje všechny sloupcové součty téže matice.

Pro další výpočty zavedeme značení pro diagonální matice, které mají na diagonále řádkový, resp. sloupcový součet

$$\mathbf{D}_r = \text{diag}(\mathbf{r}), \quad \text{resp.} \quad \mathbf{D}_c = \text{diag}(\mathbf{c}). \quad (2.11)$$

### Výpočetní algoritmus základní korespondeční analýzy [17, 13]

Označme  $\mathbf{S} = \{s_{ij}\}$  následující matici

$$\mathbf{S} := \mathbf{D}_r^{-\frac{1}{2}} (\mathbf{P} - \mathbf{r}\mathbf{c}^\top) \mathbf{D}_c^{-\frac{1}{2}}. \quad (2.12)$$

Po té proved'me singulární rozklad této matice

$$\mathbf{S} = \mathbf{U}\Delta\mathbf{V}^\top, \quad (2.13)$$

kde  $\Delta = \Delta^2$  je matice vlastních čísel  $\lambda_k$  pro  $k = 1, \dots, K$ , kde  $K = \min\{I-1, J-1\}$ . Potom rozměry matice  $\mathbf{U}$ , resp.  $\mathbf{V}$  jsou  $I \times k$ , resp.  $J \times k$ . Dále platí  $\mathbf{U}^\top\mathbf{U} = \mathbf{V}^\top\mathbf{V} = \mathbf{I}$ .

Korespondenční analýza měří míru váženého rozptylu, tzv. inercii pomocí vlastních čísel  $\lambda_k$  matice  $\mathbf{S}$ ,  $\lambda_k$  se pak nazývají hlavní inercie. Celková inercie je rovna

$$I = \sum_{k=1}^K \lambda_k = \sum_{i=1}^I \sum_{j=1}^J s_{ij}^2. \quad (2.14)$$

Hlavní komponenta řádků  $\mathbf{F}$  je rovna

$$\mathbf{F} = \mathbf{D}_r^{-\frac{1}{2}} \mathbf{U} \Delta. \quad (2.15)$$

Hlavní komponenta sloupců  $\mathbf{G}$  je rovna

$$\mathbf{G} = \mathbf{D}_c^{-\frac{1}{2}} \mathbf{V} \Delta \quad (2.16)$$

## Výpočetní algoritmus MCA

Předpokládejme, že původní matice kategorických dat má tvar  $N \times Q$ , tj.  $N$  pozorování a  $Q$  proměnných. Matice dat převedeme na indikátorovou matici. Indikátorová matice  $\mathbf{Z}$  je vytvořena tak, že kategorická data jsou rozepsána do pomocných proměnných. Pokud  $q$  třídy je má  $J_q$  typů kategorií, tak příslušná indikátorová matice bude mít  $J = \sum_{q=1}^Q J_q$  sloupců a  $N$ . Tzn. počet proměnných byl tímto rozepsáním rozšířen z počtu původních  $Q$  proměnných na  $J$  proměnných. První způsob MCA aplikuje základní algoritmus korespondenční analýzy na matici  $\mathbf{Z}$ , takto se získají souřadnice pro  $N$  pozorování a  $J$  kategorií.

## 2.2 Korelační analýza

### 2.2.1 Korelační koeficient

Pojem korelace obecně znamená vzájemný vztah mezi dvěma veličinami. Pokud se jedna veličina mění, pak se mění dle míry korelace i druhá veličina. Samotná korelace ale neurčuje míru vztahu, ani směr vztahu. Tedy která veličina je příčinou a která důsledkem. Tuto vlastnost popisuje kauzalita. Míra korelace mezi dvěma veličinami je určena pomocí korelačního koeficientu. Existuje více způsobů měření míry korelace, v následující části jsou popsány vybrané z nich.[31]

Nejčastěji používaným koeficientem pro měření korelace je *Pearsonův korelační koeficient*. Nechť  $X$  a  $Y$  jsou náhodné veličiny s realizacemi  $x_1, x_2, \dots$  a  $y_1, y_2, \dots$ , potom hodnota Pearsonova koeficientu se vypočítá jako:

$$r_p = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}} = \frac{\sum_{i=1}^n x_i y_i - n \bar{x} \bar{y}}{(n-1) s_x s_y} \quad (2.17)$$

kde  $\bar{x}, \bar{y}$  jsou výběrové průměry,  $s_x, s_y$  výběrové směrodatné odchyly.[31]

Tento koeficient měří lineární vztah mezi dvěma proměnnými. Hodnoty se pohybují v intervalu  $\langle -1, 1 \rangle$ . Krajní hodnoty znamenají dokonalou lineární závislost. Pokud je koeficient roven 1, pak pokud roste jedna veličina, roste i hodnota druhé veličiny. Pokud je koeficient roven -1, potom s rostoucí hodnotou jedné veličiny, klesá hodnota druhé. Zatímco je-li hodnota koeficientu rovna nule, veličiny jsou lineárně zcela ne-korelované. Pro výpočet tohoto koeficientu je předpokládána normalita zkoumaných dat.[31]

Další koeficient, který měří korelací mezi dvěma veličinami, je *Spearmanův korelační koeficient*. Tento neparametrický koeficient měří nelineární závislost dvou veličin, určuje, jak moc jejich vztah odpovídá monotónní funkci. Spearmanův koeficient je robustní vůči odlehlym hodnotám a nevyžaduje normalitu dat, protože pracuje se seřazenými hodnotami obou veličin. Hodnoty opět leží mezi -1 a 1 a platí pro mě analogická tvrzení jako Pearsonův korelační koeficient.[31, 46]

Nechť  $X$  a  $Y$  jsou náhodné veličiny s realizacemi  $x_1, x_2, \dots$  a  $y_1, y_2, \dots$  a číslo  $x_{ri}$  je pořadí čísla  $x_i$  v rámci všech hodnot veličiny  $X$ , číslo  $y_{ri}$  je pořadí čísla  $y_i$  v rámci

všech hodnot veličiny  $Y$ .  $\bar{x}_r, \bar{y}_r$  jsou průměrná pořadí a  $s_{x_r}, s_{y_r}$  příslušné směrodatné odchylky. Vztah pro výpočet Spearmanova koeficientu je:

$$r_s = \frac{\sum_{i=1}^n x_i y_i - n \bar{x}_r \bar{y}_r}{(n-1)s_{x_r}s_{y_r}}. \quad (2.18)$$

Pokud předpokládáme, že pořadí hodnot je unikátní, tj. neexistují v rámci jedné veličiny hodnoty realizace se stejnou hodnotou, pak lze vzorec pro výpočet Spearmanova korelačního koeficientu zjednodušit na:

$$r_s = 1 - \frac{6 \sum_{i=1}^n d_i^2}{n(n^2 - 1)}, \quad (2.19)$$

kde  $d_i = (x_{ri} - y_{ri})$  je diferenční pořadí hodnot veličin  $X$  a  $Y$ .[31, 46]

Jak je patrné ze vzorců pro oba korelační koeficienty tato míra lze aplikovat pouze na numerické veličiny. V případě kategorických veličin by bylo potřeba je převést na číselné hodnoty. K tomu slouží řada metod. Mezi dva nejznámější způsoby překódování kategorických proměnných patří one-hot kódování a label kódování. V případě one-hot kódování se ale může počet proměnných výrazně zvýšit, pokud v datech existují příznaky s větším počtem unikátních kategorií. Pro druhý zmíněný způsob kódování je nevýhodou fakt, že přiřazením čísel od 0 do  $n$ , kde  $n$  je počet kategorií v příznaků, se kategorickým hodnotám přiřadí pořadí, které ale v datech vůbec nemusí být a tudíž je tato nová informace v datech na obtíž. Další možností je předělat kategorické hodnoty na spojité hodnoty pomocí člověkova sloupu (tj. vysvětlované proměnné), tímto způsobem ale může dojít k zanesení informace o předpovídání sloupcu přímo do vysvětlujících proměnných. [26]. Proto jsou v další části této sekce uvedeny vybrané způsoby měření závislosti dvou kategorických proměnných.

## 2.2.2 Další způsoby měření závislosti

Pro měření míry závislosti dvou kategorických proměnných lze použít Cramerovo  $V$ , dále značeno jako  $V$ . Hodnota koeficientu se pohybuje mezi 0 a 1. 1 znamená dokonalou závislost mezi proměnnými, 0 neznamená žádnou závislost. Tento koeficient nemůže nabýt negativní hodnoty, tj. neexistuje negativní závislost. Stejně jako předchozí koeficienty pro korelace je  $V$  symetrické a nezáleží na pořadí veličin.[30, 31]

Pro dvě zkoumané veličiny  $X, Y$  s hodnotami  $x_1, x_2, \dots, x_r$  a  $y_1, y_2, \dots, y_s$  existuje kontingenční tabulka  $\mathbf{K}$  těchto veličin, jejíž prvky jsou četnosti hodnot proměnných  $n_{ij}$ , tj. kdy byly pozorovány hodnoty pro dvojici  $(x_i, y_j)$ .  $r$ , resp.  $s$  je počet řádků, resp. sloupců kontingenční tabulky  $\mathbf{K}$ . Vzorec pro Cramerovo  $V$  má tvar:

$$V = \sqrt{\frac{\chi^2/n}{\min(r-1, s-1)}}, \quad (2.20)$$

kde statistika  $\chi^2$  se výpočítá následovně

$$\chi^2 = \sum_{i=1}^n \frac{(n_{ij} - n_i n_j / n)^2}{n_i n_j / n}, \quad (2.21)$$

kde  $n_i$  je četnost výskytu hodnoty  $x_i$ ,  $n_j$  je četnost výskytu hodnoty  $y_j$ . Tedy platí  $n_i = \sum_{i=0}^r n_{ij}$  a  $n_j = \sum_{j=0}^s n_{ij}$ . [29, 30, 31]

Pro určení kolik informace o jedné proměnné nese druhá proměnná, je popsáno pomocí *vzájemné informace* [14]. Informací lze rozumět obsah jakéhokoli oznamení nebo údaje, který se přenáší v daném čase a prostoru. Podle Shannona, zakladatele teorie informace, je informace míra množství neurčitosti nebo nejistoty o nějakém náhodném jevu, která se odstraní realizací daného jevu [15]. Informací tak může být stanovení výsledku náhodného jevu, tedy se jedná o hodnotu náhodné veličiny [14]. Pro definování vzájemné informace je třeba definovat ještě *vlastní informace* a pojem *entropie*.

Dále jsou sepsány předpoklady pro výpočet množství informace. Pokud má náhodný jev  $X$   $n$  realizací, pak je množství informace funkcií  $n$ . Pakliže je  $n = 1$ , množství informace se rovná nule, neboť se jedná o jev jistý. Pokud jevy  $X$  a  $Y$  probíhají nezávisle, ale ve stejný čas, tj.  $p_{XY}(x, y) = p_X(x) \cdot p_Y(y)$ , potom množství informace obou jevů se tovna součtu jejich množství. Pokud jev  $X$  má  $n$  realizací a jev  $Y$   $m$  realizací, kde  $m > n$ , potom se očekává, že množství informace jevu  $Y$  je větší než množství informace jevu  $X$ . [15] Pokud je pravděpodobnost každé realizace stejná, tj.  $p_X(x) = 1/n$ , pak Hartleyho míra informace je definována jako funkce  $I : \mathbf{N} \leftarrow \mathbf{R}$  ve tvaru  $I(n) = \log n$ . Pro vlastní míru informace obsažené ve výsledku  $x$  pak platí: [15, 16]

$$I(x) = -\log p_X(x). \quad (2.22)$$

Množství informace celého jevu je popsáno entropií náhodné veličiny. Entropie  $H(X)$  náhodné veličiny  $X$  s hodnotami  $x_1, x_2, \dots$  s pravděpodobnostní funkcí  $p(x)$  je rovna: [15, 3]

$$H(X) = -\sum_x p_X(x) \log p_X(x). \quad (2.23)$$

Nechť je dán vektor  $(X, Y)$ , kde  $X$ , resp.  $Y$  je náhodná veličina nabývající hodnot  $x_1, x_2, \dots$ , resp.  $y_1, y_2, \dots$ . Náhodný vektor nabývá hodnot  $(x_1, y_1), (x_2, y_2), \dots$ . Sdružená entropie vektoru  $(X, Y)$  má tvar: [14, 16]

$$H(X, Y) = -\sum_{(x,y)} p_{XY}(x, y) \log p_{XY}(x, y). \quad (2.24)$$

Podmíněná entropie s předpokladem  $p_Y(y) > 0$ : [16]

$$H(X|Y = y) = -\sum_{(x,y)} p_{X|Y}(x|y) \log p_{X|Y}(x|y), \quad (2.25)$$

kde podmíněná pravděpodobnost je rovna  $p_{X|Y}(x|y) = p_{XY}(x, y)/P_Y(y)$ .

Pokles entropie se měří pomocí vzájemné informace, tj. platí věta [16]:

$$I(X; Y) = -H(X, Y) + H(X) + H(Y). \quad (2.26)$$

Vzájemná informace měří ztrátu informace v důsledku závislosti  $X$  a  $Y$ . Jinými slovy, kolik informace o jedné proměnné  $X$  nese druhá proměnná  $Y$ . Matematicky

je vzájemná informace definována následovně: [14, 16, 3]

$$I(X; Y) = \sum_{(x,y)} + \log \frac{p_{X|Y}(x|y)}{p_X(x)} \quad (2.27)$$

Míra, která dovede změřit asymetrickou závislost kategorických proměnných se nazývá *Thielovo U*, které se někdy označuje jako koeficient nejistoty. Pro jeho výpočet se používá podmíněná entropie, viz vztah 2.25. Thielovo  $U$  nabývá hodnot z intervalu  $\langle 0, 1 \rangle$ , kde 0 neznamená žásnou závislost a 1 dokonalou závislost. Hodnota není symetrická, tj.  $U(X, Y) \neq U(Y, X)$ , proto se může používat značení, které určuje směr závislosti –  $U(X, Y) = U(X|Y)$ . Vzorec pro výpočet koeficientu  $U$  je: [10, 30]

$$U(X, Y) = U(X|Y) = \frac{H(X) - H(X - Y)}{H(X)}. \quad (2.28)$$

## Ostatní použité pojmy

Při analýze dat lze narazit na problém multikolinearity. *Multikolinearita* je vzájemná lineární závislost vysvětlujících proměnných. Jeli  $\mathbf{A}$  matice dat (vysvětlujících proměnných bez předpovídaného sloupce), pak multikolinearita v datech existuje, pokud platí rovnice pro alespěj jedno nenulové  $c_i$ :  $c_1\mathbf{a}_1 + \dots + c_k\mathbf{a}_k$ , kde  $c_i$  jsou konstanty a  $\mathbf{a}_i$  sloupce matice reprezentující jednotlivé příznaky,  $k$  počet sloupců matice, tj. počet příznaků. V realních datech stačí, když je daná rovnice přibližně splněna.[31]

Měřítkem multikolinearity je *rozptylový inflační faktor* (zkratka VIF z anglického variance inflation factor). Možné hodnoty pro koeficient jsou 1 až libovolné číslo větší než jedna, 1 znamená nezávislost. Nad určitou hodnotu koeficientu, v literatuře [11] je uvedeno už číslo větší než 5, je značná multikolinearita již přítomna v datech. Koeficient má pro  $i$ -tý sloupec matice  $\mathbf{A}$  tvar:

$$VIF_i = \frac{1}{1 - R_i^2}, \quad (2.29)$$

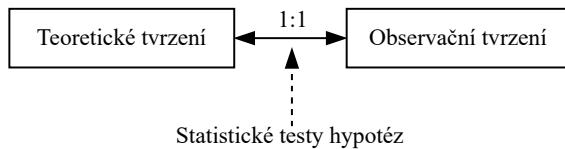
kde  $R_i^2$  je koeficient determinace  $i$ -tého sloupce. Ten říká, jak velkou část variability závislé proměnné je možné vysvětlit.[11]

## 2.3 Metoda GUHA

Metoda GUHA je původní česká metoda používaná pro nexplorační analýzu dat. První článek o této metodě vyšel v roce 1966. V současné době je jedním z nejrozšířejších implementací metody systém LISp-Miner. Jedná se o software vyvíjený na Fakultě informatiky a statistiky Vysoké školy ekonomické v Praze, kde se zároveň používá pro výuku a výzkum dobývaní znalostí z databází [44]. Zároveň je také implementována knihovna *CleverMiner* v jazyce Python, která disponuje částí funkcionality softwaru LISp-Miner.

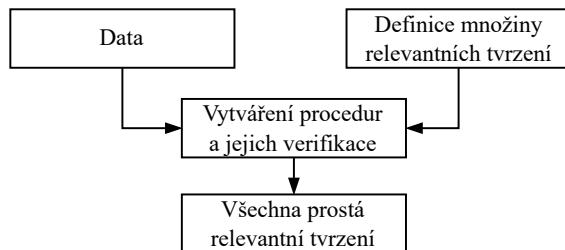
### 2.3.1 Základní princip metody

Cílem metody GUHA je získat z pozorovaných dat všechny vztahy, které jsou jsou pravdivé pro množinu objektů, ze které pochází zkoumaná data. Vy užívají se k tomu statistické testy hypotéz, které dovolují na základě platnosti určitého tvrzení o vzorku dat přijmout tvrzení o celé množině objektů. Pravdivé tvrzení o celé množině dat se nazývá *teoretické tvrzení*. Tvrzení o vzorku dat se nazývá *observační tvrzení*. Vztah 1 : 1 mezi těmito tvrzeními zprostředkovávají statistické testy, znázorněno na obr. 2.2.[44]



Obrázek 2.2: Vztah mezi tvrzeními vzorku dat a celých dat v metodě GUHA. Zdroj: vlastní.

Základní postup GUHA procedury je na obrázku 2.3. Vstupem pro procedury jsou vstupní data a parametry, které definují množinu relevantních tvrzení. Na základě definice jsou vytvořena všechna relevantní observační tvrzení, která jsou verifikována podle dat. Výstupem jsou pak všechna všechna prosté observační tvrzení vycházející ze vstupů. Prosté relevantní tvrzení je takové tvrzení, které je pravdivé ve vstupních datech a zároveň neplyne již z uvedeného jiného tvrzení ve výstupu.[44]



Obrázek 2.3: Základní postup procedury GUHA. Zdroj: vlastní.

### 2.3.2 Důležité pojmy

Pro podrobnější popis procedur je nejprve třeba definovat několik pojmu, se kterými se v procedurách pracuje. Metoda pracuje s následujícími pojmy [44]:

- **Matici dat a atributy** – Řádky matice jsou jednotlivá pozorování. Atributem se rozumí sledovaná vlastnost, jedná se o sloupec matice.
- **Základní booleovský atribut** – Jedná se o výraz  $\mathbf{A}(\alpha)$ , kde  $\mathbf{A}$  je atribut a  $\alpha$  je vlastní podmnožina  $\mathbf{A}$ .  $\alpha$  může obsahovat více prvků než jeden.
- **Booleovský atribut** – Každý základní booleovský atribut je booleovský atribut. Booleovské atributy jsou i negace, konjunkce a disjunkce základních booleovských atributů.

Pro každý řádek  $i$  matice  $\mathbf{M}$  nabývá boolovský atribut  $\mathbf{A}$  hodnotu 0, nebo 1.

$\mathbf{A}[i] = 1 \Rightarrow$  boolovský atribut  $\mathbf{A}$  je pravdivý pro řádek  $i$ .

$\mathbf{A}[i] = 0 \Rightarrow$  boolovský atribut  $\mathbf{A}$  je nepravdivý pro řádek  $i$ .

- **Literál** – Základní boolovský atribut nebo jeho negace.
- **Dílčí cedent** – Konjunkce nebo disjunkce literálů.
- **Cedent** – Jedná se o konjunkci dílčích cedentů. Příkladem cedentu je boolovský atribut, který vznikl konjunkcí a disjunkcí dalších atributů.

Další pojmy se týkají vztahů, se kterými procedury pracují [44]:

- **Asociační pravidlo** – Výraz  $X \rightarrow Y$ , kde X a Y jsou konjunkce dvojic atribut a jeho hodnota. Dále v textu je používáná pro tento pojem zkratka AP.
- **Konfidence AP** – Podíl počtu řádků, které splňují A a zároveň S a počtu řádků, které splňují pouze S.
- **Podpora AP** – Podíl počtu řádků, které splňují A a zároveň S a počtu řádků vstupní matice dat.

Častou úlohou pro dobývání AP je nalezení všech AP, u kterých je hodnota konfidence a podpory AP větší nebo rovna danému prahu. V rámci GUHA se AP zkoumají jako vztah dvou obecných boolovských atributů, které jsou odvozené ze sloupců vstupní matice. GUHA asociační pravidlo (GUHA AP) je výraz

$$\varphi \approx \psi, \quad (2.30)$$

kde  $\varphi, \psi$  jsou boolovské atributy, které nemají obsažený žádný společný boolovský atribut.  $\varphi$  se nazývá *antecedent* a  $\psi$  *sukcedent*<sup>2</sup>. Symbol  $\approx$  odpovídá *4ft-kvantifikátoru*, viz dále v této sekci. Existují také podmíněná GUHA AP, která mají tvar  $\varphi \approx \psi | \chi$ , kde  $\chi$  je boolovský atribut.[44]

Pravdivost GUHA AP v matici dat  $\mathbf{M}$  se určuje pomocí tzv. *4ft-tabulky*. Nechť je dána matice vstupních dat  $\mathbf{M}$ , antecedent  $\varphi$ , sukcedent  $\psi$ . Pak *4ft-tabulka*  $4ft(\varphi, \psi, \mathbf{M})$  je definována jako čtverice čísel  $(a, b, c, d)$ , pro které platí:

- $a$  je počet řádků matice  $M$ , které splňují oba boolovské atributy  $\varphi, \psi$ .
- $b$  je počet řádků matice  $M$ , které splňují  $\varphi$ , ale nesplňují  $\psi$ .
- $c$  je počet řádků matice  $M$ , které nesplňují  $\varphi$ , ale splňují  $\psi$ .
- $d$  je počet řádků matice  $M$ , které nesplňují ani jeden atribut  $\varphi, \psi$ .[44]

Reprezentace této tabulky je zobrazena v tab. 2.1.

*4ft-kvantifikátor*, symbol  $\approx$ , definuje podmínu, která se týká hodnot  $(a, b, c, d)$  v *4ft-tabulce*. Kvantifikátor je formálně definovaný pomocí funkce  $F_{\approx}$ , která každé

---

<sup>2</sup>Antecedent, jako cedent, který předchází a sukcedent, jako cedent, který následuje.

Tabulka 2.1: *4ft-tabulka* matice  $\mathbf{M}$  s asociačním pravidlem  $\varphi \approx \psi$ .

$\mathbf{M}$	$\psi$	$\neg\psi$
$\varphi$	$a$	$b$
$\neg\varphi$	$c$	$d$

čtveřici nezáporných čísel přiřazuje hodnotu 1, resp. 0 pokud je, resp. není podmínka splněna. Zapisujeme  $F_{\approx}(a, b, c, d)$  nebo zkráceně  $\approx(a, b, c, d)$ .[44]

GUHA AP  $\varphi \approx \psi$  je pravdivé v matici dat  $\mathbf{M}$

$$\Leftrightarrow \approx(a, b, c, d) = 1, \text{ formálně zapsáno jako } \text{Val}(\varphi \approx \psi) = 1.$$

(2.31)

GUHA AP  $\varphi \approx \psi$  je nepravdivé v matici dat  $\mathbf{M}$

$$\Leftrightarrow \approx(a, b, c, d) = 0, \text{ formálně zapsáno jako } \text{Val}(\varphi \approx \psi) = 0.$$

Pro podmíněné AP  $\varphi \approx \psi|\chi$  platí obdobné vztahy. Předpokládáme však, že booleovský atribut  $\chi$  nemá ani jeden společný atribut s atributy  $\varphi$  a  $\psi$ . Platí tvrzení [44]:

Nechť  $\mathbf{M}$  je matice vstupních dat,  $\varphi, \psi, \chi$  booleovské atributy,  $\approx$  kvantifikátor.

Podmíněné AP  $\varphi \approx \psi|\chi$  je pravdivé v  $\mathbf{M} \Leftrightarrow \varphi \approx \psi$  je pravdivé v matici  $\mathbf{M}|\chi$ .

(2.32)

### 2.3.3 Procedury

V dokumentaci [44] je popsáno sedm procedur – *4ft-Miner*, *SD4ft-Miner*, *CF-Miner*, *SDCF-Miner*, *KL-Miner*, *SDKL-Miner*, *Ac4ft-Miner* [44]. V knihovně v jazyce Python jsou implementované pouze metody *4ft-Miner*, *SD4ft-Miner*, *CF-Miner* [45]. V této práci jsem použila metodu pouze první metodu, proto další je další teoretický popis věnován pouze metodě *4ft-Miner*.

Tato procedura pracuje s AP  $\varphi \approx \psi$ , nebo s podmíněnými AP  $\varphi \approx \psi|\chi$ . V knihovně *Cleverminer* lze v hlavní funkci `cleverminer` předat vstupní DataFrame s daty, který reprezentuje vstupní matici dat, další parametr je jedna ze tří implementovaných procedur, dále seznam podmínek pro vyhodnocení tvrzení, vypnutí optimizace, limit pro výsledná tvrzení a seznam cedentů. Cedenty jsou rozděleny na antecedenty (parametr `ante`, tj. booleovský atribut  $\varphi$ ), sukcedenty (parametr `succ`, tj. atribut  $\psi$ ) a podmínky (parametr `cond`, tj. booleovský atribut  $\chi$ ). Každý z booleovských atributů libovolného typu cedentu může mít tyto atributy:

- `name` – Název příznaku matice, tj. název sloupce v DataFrame.
- `type` – Jakým pravidlem se řídí výběr více kategorií v příznaku. Jedna z hodnot `subset`, `lcut`, `rcut`, `seq`, `one`.
- `minlen` – Minimální počet kategorií v daném příznaku.

- **maxlen** – Maximální počet kategorií v daném příznaku.[45]

Příznaky musí být kategorické a musí být možné je seřadit. Druhá vlastnost je třeba pro vybírání více kategorií v jednom cedentu určitými způsoby selekce. Pro textové řetězce reprezentující kategorie jsou názvy kategorií řazeny podle abecedy.[45]

Pro názornost jsou dále uvedeny příklady pro jednotlivé druhy atributu **type**. Nechť je dán příznak **A** s kategoriemi 1, 2, 3, 4, 5 a parametry jsou definovány následovně: **minlen=1**, **minlen=3**. Pokud je typ **one**, bere se jedna z kategorií daného příznaku, tuto kategorii je třeba specifikovat. Pro typ **subset** jsou vybrány všechny následující možnosti:

- Délka je rovna 1 – **A(1)**, **A(2)**, **A(3)**, **A(4)**, **A(5)**.
- Délka je rovna 2 – **A(1, 2)**, **A(1, 3)**, **A(1, 4)**, **A(1, 5)**, **A(2, 3)**, **A(2, 4)**, **A(2, 5)**,  
**A(3, 4)**, **A(3, 5)**, **A(4, 5)**.
- Délka je rovna 3 – **A(1, 2, 3)**, **A(1, 2, 4)**, **A(1, 2, 5)**, **A(2, 3, 4)**, **A(2, 3, 5)**,  
**A(3, 4, 5)**.[44]

Pro typ sekvence, **seq** by se pak vybraly následující možnosti:

- Délka je rovna 1 – **A(1)**, **A(2)**, **A(3)**, **A(4)**, **A(5)**.
- Délka je rovna 2 – **A(1, 2)**, **A(2, 3)**, **A(3, 4)**, **A(4, 5)**.
- Délka je rovna 3 – **A(1, 2, 3)**, **A(2, 3, 4)**, **A(3, 4, 5)**.[44]

Pro typ **lcut** se vybírají možnosti:

- Délka je rovna 1 – **A(1)**.
- Délka je rovna 2 – **A(1, 2)**.
- Délka je rovna 3 – **A(1, 2, 3)**.[44]

Analogicky pro typ **rcut**.

Literály v rámci cedentů lze také kombinovat obdobnými způsoby. Opět lze přiřadit minimální a maximální délku, typ pro kombinování literálů je výběr konjunkce, nebo disjunkce. Tyto možnosti lze specifikovat pro antecedenty, sukcedenty i podmínky. Zadání podmínek není nezbytné v atrubutech funkce **cleverminer**.

Další parametry, které lze předat této funkci jsou:

- **Base** – Minimální počet řádků, které splňují antecedenty i sukcedenty (číslo *a* v tabulce 2.1).
- **RelBase** – Hodnota Base vydělená celkovým počtem řádků dat (případně počtem řádků v matici s aplikovanou podmínkou).

- **conf** – Konfidence, pravděpodobnost  $\text{Pr}(\psi|\varphi)$ . Jinými slovy procentuální zastoupení řádků, které vyhovují  $\psi$  (sukcendentům) z těch řádků, které vyhovují i  $\varphi$  (antecedentům).
- **aad** (nadprůměrná závislost) – Jak moc  $\varphi$  zvyšuje pravděpodobnost  $\psi$ . Kolikrát se zvýší pravděpodobnost splnění sukcedentů, když se vezmou pouze záznamy, které vyhovují antecedentům, oproti všem záznamům minus 1.
- **bad** (podprůměrná závislost) – Jak moc  $\varphi$  snižuje pravděpodobnost  $\psi$ .

Příklad volání funkce `cleverminer` je sepsaný v ukázce kódu č. 2.1.

```

01 cleverminer(df = data,
02             proc = "4ftMiner",
03             quantifiers = {"conf":0.6, "Base":1000},
04             ante = {
05               "attributes":
06               [
07                 {
08                   "name": "weekday",
09                   "type": "subset",
10                   "minlen":1, " maxlen":3
11                 },
12                 {
13                   "name": "quarter",
14                   "type": "lcut",
15                   "minlen":1, " maxlen":4
16                 }
17               ],
18               "minlen":1, " maxlen":3, "type": "con"
19             },
20             succ = {
21               "attributes":
22               [
23                 {
24                   "name": "L3",
25                   "type": "subset",
26                   "minlen":1, " maxlen":3
27                 }
28               ],
29               "minlen":1, " maxlen":1, "type": "con"
30             },
31             cond = {
32               "attributes":
33               [
34                 {
35                   "name": "promo",
36                   "type": "one",
37                   "value": "promo"
38                 }
39               ],
40               "minlen":1, " maxlen":1, "type": "con"
41             }
42           )

```

Kód 2.1: Příklad volání funkce `cleverminer`.

## 2.4 Nástroje

### 2.4.1 Python – Jupyter Notebook

Veškeré výpočty probíhaly v jazyce Python. Metoda GUHA ve verzi Pythonu 3.10, ostatní výpočty a příprava dat ve verzi 3.9. Kód byl napsán a spuštěn v nástroji *Jupyter Notebook*. Všechny informace o tomto nástroji jsem čerpala z dokumentace tohoto nástroje [37]. Jedná se o alternativu ke konzoli jazyka Python. Jupyter Notebooky jsou interaktivní a umožňují psát a spuštět blok po *buňkách*. Buňky jsou sdruženy v souboru s příponou *ipynb*, ve skutečnosti se jedná o JSON soubor. V souboru je uložený kód a zároveň i naposledy spuštěné výstupy jednotlivých buněk.

Výhodou Jupyter Notebooku oproti klasické konzoli je, že podporuje odsazování, zvýrazňování syntaxe, zobrazení obrázků, HTML prvků nebo L<sup>A</sup>T<sub>E</sub>Xvýrazů přímo ve výstupu pod kódem. Dále je možné soubor dobře dokumentovat pomocí jazyka Markdown. Tento značkovací jazyk není omezený jen na prostý text, jako jsou klasické komentáře v kódu. Díky němu lze soubor strukturovat do sekcí různých úrovní. Vytvořen je tak přehlednější kód.

Za zmínku také stojí, že dalšími základními programovacími jazyky, které je možné spuštět v Noteboocích jsou R a Julia. Další jazyky lze spuštět pomocí speciálního jádra pro příslušný jazyk.

S Jupyter Notebooky jsem pracovala v editoru Visual Studio Code, který podporuje řadu programovacích jazyků. Knihovnu Cleverminer jsem spuštěla v prostředí Google Colaboratory, které podporuje pouze Jupyter Notebooky.

## Databáze

Data společnosti jsou uložená v MySQL databázi, ke které jsem přistupovala pomocí nástroje HeidiSQL, což je open-sourcový nástroj pro práci s databázovými tabulkami. Z tohoto programu je data možné vyexportovat do formátu CSV. S exportovanými soubory jsem dále pracovala v Pythonu.

### 2.4.2 Power BI

Pro vizualizaci dat jsem použila nástroj Power BI Desktop, dále už jen Power BI. Tato aplikace umožňuje vytvořit business intelligence report pro sledovaná data. Data je možné nahrát z různých datových zdrojů, poté z nich vytvořit datový model. Na základě tohoto modelu pak lze vytvářet reporty s nejrůznějšími vizuály. Aplikace má rozsáhlou online dokumentaci, a to i v českém jazyce. Veškeré informace o Power BI jsou čerpány z této dokumentace [39].

Power BI má tři možná zobrazení – reporty, data, model. V reportovací části je možné vytvářet interaktivní vizualizace vstupních dat a sestavit tak i vícestránkový report. V sekci data jsou zobrazené všechny řádky aktuálně vybrané tabulky. Uživatel může v tabulce vyhledávat pomocí filtrů, může přidávat nové sloupce, měnit datové

typy sloupců, ale nemůže změnit hodnotu existujících dat v tabulce. V sekci model se nachází grafické znázornění datového modelu včetně vztahů mezi jednotlivými tabulkami a jejich sloupců. Uživatel může měnit – odebírat, přidávat, měnit kardinalitu vazeb mezi nimi.

Power BI pro práci s daty využívá dva jazyky – jazyk M a jazyk DAX (Data Analysis Expressions). První jmenovaný lze použít při nahrávání dat a jejich zpracování, jazyk se generuje na základě kroků v GUI aplikace, nebo je možné psát příkazy ve vestavěném editory. Druhý jmenovaný se používá přímo ve vizualizační části pro vytváření nových sloupců a metrik, obsahuje přes 200 předdefinovaných funkcí, které jsou podobné funkcím v aplikaci Microsoft Excel.

## Nahrání dat

Data lze do reportu nahrát tabulková data z mnoha typů zdrojů. Je možné se např. připojit přímo k databázi, získat data z webu, z cloudového úložiště, z textového souboru, souboru z nástroje Microsoft Excel nebo je také možné spustit Python či R kód, který vytváří data. Nahraná data je možné předzpracovávat v editoru Power Query, který je součástí Power BI Desktop. Dále jsou uvedeny příklady úprav v editoru. Je možné nastavovat záhlaví tabulek, vybírat relevantní řádky a sloupce, přidávat nové sloupce pomocí příkazu v jazyce M. Tabulku s daty je možné rozdělit na více tabulek, nebo naopak více tabulek sloučit do jedné, odstranit řádky s chybějícími hodnotami nebo hodnoty nahradit. V nástroji lze také vytvářet funkce a proměnné např. pro vygenerování tabulky kalendáře.

Editor zaznamenává provedené změny na datech. Jednotlivé kroky tak lze později případně přeskočit, upravit nebo lze mezi úpravy vložit nový krok. Posloupnost kroků je ale důležitá, neboť kroky se provádějí postupně. Vložený krok může tedy v některých případech způsobit chybné vykonání následujících kroků. Na obr. 2.4 se nachází ukázka z editoru Power Query.

Po uložení upravených dat v editoru se transformovaný model zobrazí v aplikaci Power BI, kde lze s daty dále pracovat. Úpravu dat, který model obsahuje lze provádět pouze v nástroji Power Query. Do editoru lze přistupovat opakováně i během vytváření reportu, může ale nastat sitace, kdy úprava vstupních dat změní model takovým způsobem, že vizuálně přestanou správně fungovat.

## Míry

*Míry*, někdy nazývané *metriky* v Power BI umožňují uživateli reportu sledovat ukazatele, které jsou relevantní pro zkoumaná data. Jedná se o výpočty na datech vytvářené pomocí jazyka DAX. Vypočítaná hodnota míry se mění podle toho, které konkrétní řádky tabulek vstupují do výpočtu, tj. jaké je vizuálně zvolená agregace a vstupní pole. K přepočítávání dochází automaticky při interakci s daty v reportu. Vytvořené metriky jsou zobrazeny vedle seznamu tabulek a sloupců, které se nachází v datovém modelu. Pro přehlednost jsou ale označeny ikonou. Z důvodu přehlednosti

The screenshot shows the Microsoft Power Query Editor interface. At the top, there's a ribbon with tabs like File, Home, Transform, Add Column, View, Tools, and Help. Below the ribbon is a toolbar with icons for Close & Apply, New Query, Data Sources, Parameters, Refresh, Manage, Sort, Split, Group, and Transform. A status bar at the bottom indicates '8 COLUMNS, 50 ROWS' and 'Column profiling based on top 1000 rows'. The main area displays a table titled 'Best States to Retire' with columns: State, Overall rank, Overall score, Affordability rank (40%), and Wellness. The table lists 17 US states from Georgia down to South Carolina. To the right of the table is a 'Query Settings' pane with sections for 'PROPERTIES' (Name: Best States to Retire) and 'APPLIED STEPS' (listing 'Source', 'Extracted Table From Html', 'Promoted Headers', and 'Changed Type'). Red circles numbered 1 through 4 point to specific elements: 1 points to the Home tab in the ribbon; 2 points to the 'Best States to Retire' query in the Queries list; 3 points to the table data; 4 points to the 'Properties' section in the Query Settings pane.

Obrázek 2.4: Ukázka nástroje Power Query.

1 – Možné interakce s daty. 2 – Seznam nahraných tabulek, případně proměných a funkcí. 3 – Ukázka vybraných dat. 4 – Seznam kroků a vlastností. Zdroj: [39].

je ale lepší míry přiřadit do samostatné tabulky, které neobsahuje vstupní data, ale pouze vytvořené míry.

Základní dvě možnosti jak vytvořit metriku jsou – napsat do řádku vzorců výraz v jazyce DAX nebo vytvořit tzv. rychlou míru pomocí dialogového okna. Rychlé míry mají ale tu nevýhodu, že nabízí pouze základní operace s daty jako např. průměr, rozptyl, extrémy, matematické operace nebo převody datumů. Při vkládání dat do vizuálu jsou k dispozici automatické míry, které se neukládají do seznamu měr v reportu, ale vstupují pouze do vybraného vizuálu. Velmi častou používanou je míra pro počet záznamů, nebo unikátní počet záznamů.

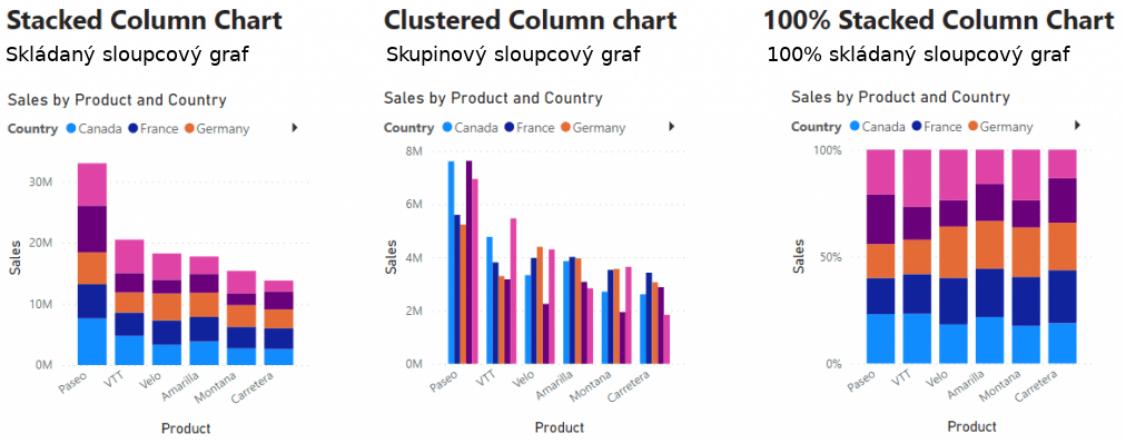
Míru lze bud' přímo vložit jako vstup do vizuálu nebo ji použít v definici jiné míry.

## Typy vizuálů

Aplikace nabízí přes dvacet základních vizuálů a stovky vizuálů dostupných ke stažení. V následující části jsou představeny vybrané vizuály, které jsou použité v reportu pro data analyzovaná v této práci.

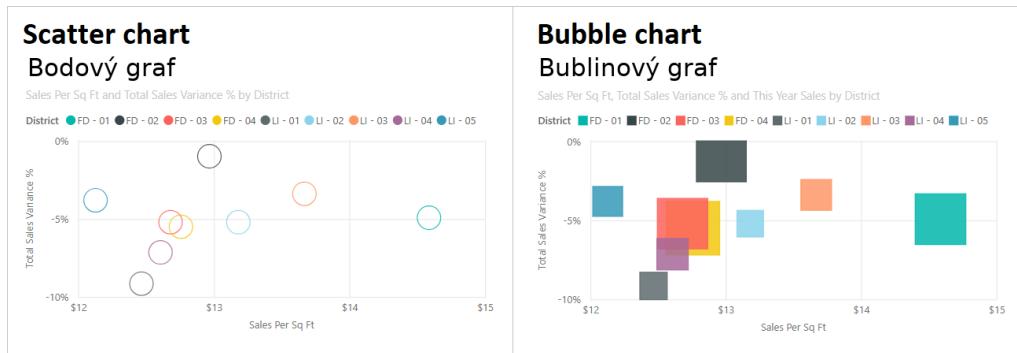
Základní graf je graf sloupcový, případně pruhový, které lze dále rozlišit na skládaný, skupoinový a 100% skládaný graf. Rozdíly mezi těmito grafy jsou na obr. 2.5. Se sloupcovým grafem souvisí i graf kombinovaný, který obsahuje jak sloupce s hodnotami, tak spojnice pro zobrazení jiných hodnot. Takový graf má tedy dvě rozdílné osy  $y$ , které mohou mít různá měřítka, ale pouze jednu společnou osu  $x$ .

Další klasický graf je graf bodový, který zobrazuje body v průsečíku číselných hodnot  $x$  a  $y$ . Osy mohou mít opět různá měřítka. Z bodového grafu vychází tzv. bublinový graf, který ale navíc může zobrazit ještě třetí rozměr v datech, a to v podobě velikosti bodů – bublin. Tvar a poměr velikostí zobrazených bodů, resp. bublin je možné



Obrázek 2.5: Základní typy sloupcových grafů. Zdroj: [40], upraveno.

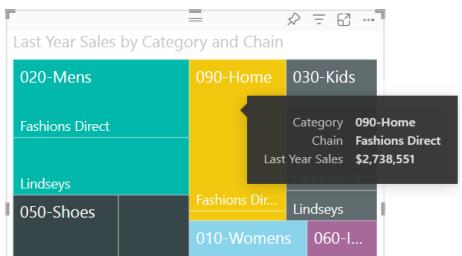
upravovat, stejně tak jejich barvu. Ukázka je na obr. 2.6.



Obrázek 2.6: Základní typy bodových grafů. Zdroj: [39], upraveno.

Zajímavým vizuálem je mapa stromové struktury. Díky tomuto vizuálu je možné zachytit hierarchická data a poměrové zastoupení kategorií v datech. Každá kategorie je reprezentována jako obdélník, označuje se pojmem větví. Každý obdélník může obsahovat své podkategorie, označené jako listy. Příklad je na obr. 2.7.

Formou vizuálu jsou i tabulky, matice a tzv. karty - jednočíselné nebo víceřádkové. Karty se používají pro zobrazení sledované celkové hodnoty, např. celkový počet produktů. Dále karta může obsahovat název sledované hodnoty. Příklad je uveden na obr. 2.8. Rozdíl mezi tabulkou a maticí je ten, že tabulce lze předávat pouze sloupce a případné číselné hodnoty se spočítají podle agregace v předchozích sloupcích. Tabulka může obsahovat záhlaví a řádek s celkovými součty. Matice na rozdíl od obyčejné tabulky umožňuje stupňovité nahlížení na data. Pokud definujeme více vstupních sloupců z dat jako řádky matice, lze pak záhlaví jendotlivých názvů řádků rozbalit pro větší detail. Příklad tabulky je na obr. 2.9



Obrázek 2.7: Mapa stromové struktury s tooltipem. Zdroj: [39].



Obrázek 2.8:  
Ukázka vizuálů  
jednořádková  
karta (vlevo)  
a  
víceřádková  
karta (vpravo).  
Zdroj:  
vlastní.

Obrázek 2.9: Ukázka vizuálu  
matice. Zdroj: [39].

# Kapitola 3

## Shrink

Cílem této práce je analyzovat shrinky produktů, které byly zaznamenány v datech dané společnosti, a zjistit příčiny jejich vzniku. V následující části je vysvětlen pojem shrink a popsány kategorie, které vybraná společnost rozeznává ve svých datech.

### 3.1 Definice

Definice pojmu shrink v oblasti retailu není jednotná. Problematikou, jak přesně označit ztrátu produktů, se zabývá Beck ve svém článku [43]. V této práci je slovem shrink označována ztráta zisku z neuskutečněného prodeje hotového produktu. Tento produkt je vyroben, či naskladněn, ale z nějakého důvodu nemohl být prodán zákazníkovi. Tímto důvodem může být například poničení produktu, jeho ztráta nebo prošlá doba spotřeby. Za shrink produktu lze označovat i stav, kdy cena produktu je neplánovaně snížena v důsledku zmíněných důvodů. Shrinkem je potom rozdíl plánované prodejní ceny a ceny, za kterou byl produkt skutečně prodán [24]. Tuto definici jsem zvolila vzhledem k povaze dat analyzované společnosti.

Často se v literatuře lze setkat s pojmem shrink, resp. anglickým *shrinkage* v retailových společnostech především v souvislosti se ztrátou zboží z důvodu krádeže – interní neboli zamšstnanecká, externí neboli zákaznická, dále z důvodu administrativního pochybení, nebo podvodem na straně dodavatele. [42, 43]

### 3.2 Typy shrinků

Shrinky je možné kategorizovat podle Becka do čtyř kategorií podle čtyř oblasti, kde se shrink odehrává. Kategorie se dále dělí na ztráty, které jsou známé a neznámé. Neznámé ztráty z povahy věci nejde evidovat, takže se dále již nedělí. Známou ztrátu lze dále rozdělit na úmyslné a neumýslné. Rozdělení je podrobně znázorněno na obrázku 3.1.

Vybraná společnost rozlišuje ve svých datech tři kategorie shrinku – shrinky způsobené inventurou, škodami a cenové snížení. Dále se text věnuje popisu jednotlivých typů



Obrázek 3.1: Topologie shrinků. Zdroj: [43]

v rámci těchto kategorií v analyzované společnosti. Každý typ má přiřazeno jednoznačné identifikační číslo, podle kterého je zaznamenáván v databázi. Z důvodů anonymizace dat v práci nejsou uvedené přesné hodnoty těchto ID, namísto toho jsou uvedeny pouze názvy, které definují shrinky.

**Shrinky způsobené inventurou** Tato kategorie sdružuje všechny shrinky týkající se změn ve stavech zásob. Tyto změny se projeví při inventuře. V tabulce 3.1 se nachází přehled všech evidovaných typů. Některé typy mají obdobný význam a jsou duplicitní. K tomu mohlo dajít patrnš tím, že některé subjekty používají dřívější značení pro inventuru, než jiné subjekty, které mohli přejít na nový, podrobnější způsob záznamu.

### Shrinky způsobené škodami

Do kategorie shrinků způsobených škodami jsou řazeny zbylé důvody k odstranění produktu z prodeje z důvodu degradace produktu. V následující tabulce 3.2 jsou vypsané všechny typy, které mohou být evidovány.

### Snížení ceny

Tento typ shrinku vzniká v důsledku snížení ceny na prodejně. Tento shrink není přímo evidovaný v datech, ale lze jej vypočítat ze záznamů prodejů. Jedná se o situaci, kdy přímo na prodejně je nějaký produkt zlevněný v důsledku blížící se expirace nebo z důvodu poničení obalu. Nejedná se tak o klasickou promoakci, ale o

zlevnění, které není evidováno systémem, protože se netýká všech produktů daného typu, ale pouze jednoho či několika konkrétních produktů na vybrané prodejně.

Postup pro zjištění velikosti shrinku pro jeden konkrétní produkt je následovný. Pro každou účtenku je třeba porovnat cenu každého prodaného produktu s ceníkovou cenou, případně promoční slevou. Pokud si tyto ceny nejsou rovné, pak rozdíl těchto cen je shrink daného produktu.

Vzhledem k tomu, že denně se na každé prodejně zaevidují stovky účtenek, bylo by toto postupné procházení velmi časově náročné. Zároveň tento shrink postihuje jen velmi malou část celkového prodaného objemu. Tento shrink jsem ve svých analýzách již dále nezkoumala, protože nebyl shledán prioritním. Určení příčin vzniku takového shrinku se může lišit v závislosti na konkrétních prodejnách, a to jak na zaměstnaných, které vytváří snížení cen, tak na spotřebitelích, kteří na konkrétních prodejnách nakupují.

Tabulka 3.1: Přehled jednotlivých typů shrinků způsobených inventurou.

Název	Popis
Inventura - příjem	Kladné připsání zboží během inventury.
Inventura - odpis	Záporné odepsání zboží během inventury.
Inventura - velká	Velká inventura skladu.
Inventury - oprava	Dodatečné opravy, které bylo třeba provést po dokončení velké inventury.
Inventura - částečná	Odpis, nebo naskladnění zboží při inventuře položek.
Neuznané reklamace centrálním skladem	Odpis zboží, které bylo fyzicky dodané z centrálního skladu na prodejnu, ale prodejna jej vrátila, ale vratka nebyla uznaná.
Inventura	Starší verze ID používaného pro inventuru.
Neexistující zboží	Odpis prokazatelně ukradeného zboží nebo i ztraceného zboží.

Tabulka 3.2: Přehled jednotlivých typů shrinků způsobených škodami.

Název	Popis
Poškození	Odpis zboží, které bylo poškozené. Např. nedopečené, spálené, špatně vyrobené nebo poškozené zaměstnancem nebo zákazníkem (kdy nelze uplatnit reklamaci na zákazníka.)
Prošlé a zkažené zboží	Odpis zboží, kterému prošla doba spotřeby (v případě výrobků, kde je datum uvedené), zkažené či shnilé zboží (ovoce, zelenina) nebo ztvrdlé pečivo.
Zákaznické reklamace	Odpis zboží, které zákazník reklamoval a reklamace byla uznána, ale zároveň nelze toto zboží reklamovat u dodavatele.
Reklamace centrálního skladu	Odpis zboží, které fyzicky nedorazilo z distribučního centra a nebylo možné ho reklamovat z důvodu nesplnění limitu pro vytvoření reklamace na distribučním centru. Také obsahuje odpisy neprodaných položek po ukončení výprodeje.
Kompostéry	Odpis zboží, které je prošlé nebo poškozené a které prodejna zlikviduje v kompostéru.
Potravinová banka	Odpis potravinářského zboží, které bylo darováno potravinovým bankám. Jedná se o produkty, které nebylo možné zařadit znovu do oběhu.
Zvířecí útulky	Odpis potravinářského zboží, které bylo darováno do útulků zvířat. Jedná se o produkty, které nebylo možné zařadit znovu do oběhu.
Poškození vnějšími vlivy	Odpis zboží, které bylo poškozeno nebo zničeno vlivem třetí strany (výbuch, vytopení, poškození majetku) nebo přírodními živly. Zboží se tedy na prodejně nenachází a nemůže proto být zlikvidováno.
Zničení	Jinak zničené zboží

# Kapitola 4

## Zpracování dat

Tato kapitola se zabývá popisem práce s konkrétní datovou sadou, kterou jsem obdržela. Z důvodu ochrany dat se v textu nevyskytují přesná pojmenování, ani není možné zobrazit přesnou strukturu uložení dat.

### 4.1 Popis obdržených dat

Všechna data poskytnutá společností jsou uložena v databázi, ke které byl zhotoven omezený přístup pro účely získání dat pro analýzy shrinku produktů společnosti. Zároveň s možností přístupu jsem obdržela i tabulkou, která stručně komentuje všechny tabulky v databázi a sloupce v jednotlivých tabulkách. Celkem se v databázi nachází přes čtyři sta tabulek, z nichž bylo potřeba vybrat pouze ty, které obsahují relevantní data pro úlohu shrinků.

Z důvodu ochrany dat nelze uvádět přesné názvy tabulek, nicméně pro lepší orientaci v textu, každé použité tabulce přiřadím název, který odpovídá obsaženým datům v tabulce.

#### Číselníky

Základní číselník s údaji o produktech, se nachází v tabulce *produkt* se 27 sloupcí. Pro analýzu vzniku shrinků jsem z této tabulky vybrala jako možné významné údaje následující sloupce:

- **ID produktu**
- **ID prodejní varianty** – Určuje o jaký typ balení daného produktu se jedná
- **Expirace** – Expirace produktu ve dnech (hodnoty 0, 999 a NULL označují neomezenou expiraci)
- **ID kategorie** – Kategorie produktu v číselné struktuře (pro lepší interpretaci, o jakou kategorii zboží se jedná, je vhodnější použít strukturu podle úrovní, kterou lze získat napojením na tabulkou *produkt\_kategorie*.)

- **Aktivní** – Zda je tento produkt stále aktivní v portfoliu, nebo se jedná o produkt, který se již neprodává

Tabulka *produkt\_kategorie* obsahuje převod z číselné struktury do struktury pomocí produktové hierarchie. V obdržených datech má produktová hierarchie šest úrovní. Hierarchie produktů tvoří tedy strom se šesti úrovněmi. Nejvyšší úroveň, tj. úroveň číslo 1 má šest kategorií.

V tabulce 4.1 jsou uvedeny počty podkategorií pro každou z kategorií z nejvyšší úrovně. Také je uvedeno procentuální zastoupení kategorií v nejvyšší úrovni v rámci produktového portfolia vybrané společnosti. Zastoupení je odvozeno podle počtu produktů v kategorii.

Tabulka 4.1: Počet podkategorií na jednotlivých úrovních a zastoupení nejvyšší kategorie v rámci produktového portfolia.

Úroveň:	Název kategorie	Počty kategorií						Zastoupení kategorie
		2	3	4	5	6		
	Nepotravinářské	1	7	27	76	179		76,12%
	Suché	3	13	33	147	494		7,28%
	Kosmetika a drogerie	1	4	21	59	193		7,07%
	Čerstvé	5	11	27	111	469		4,27%
	Velmi čerstvé	6	10	31	92	271		4,04%
	Ostatní	4	4	4	5	5		1,06%
	Tabák	1	1	1	3	8		0,17%

Poslední, šestá úroveň hierarchie je přímo napojená na hodnotu číselné struktury, která je uvedena v číselníku produktů (v tabulce *produkt*). Pro získání všech úrovní kategorizace po úrovních k danému produktu je třeba vyhledat v tabulce *produkt* číselné ID kategorie daného produktu a napojit jej na poslední úroveň v tabulce produktové hierarchie (*produkt\_kategorie*). V této tabulce je pak uvedena rodičovská kategorie z úrovně 5. Poté je potřeba opět vyhledat v tabulce *produkt\_kategorie* tuto hodnotu a zjistit její nadřazenou kategorii. Takto se postupuje dokud není dosaženo nejvyšší úrovně. Tyto operace jsem provedla SQL příkazem přímo nad databází. Použila jsem vnitřní spojení na každou úroveň hierarchii na sloupce kategorie a rodičovská kategorie.

Další tabulka, se kterou jsem pracovala obsahuje informace o velikosti a hmotnosti produktů. Tato tabulka je důležitá z toho důvodu, že některé položky jsou vážené. Pokud se udává jejich množství udává se v gramech, zatímco nevážené položky jsou uvedeny v kusech. Aby bylo možné porovnávat oba číselné údaje, ke každému váženému produktu existuje přepočet na počet kusů (ozn. SKU). K tomu jsou využity údaje o počtu kusů na jednu vychystávací jednotku (dále označeno jako *SKU\_VJ*) a hmotnost jedné vychystávací jednotky daného produktu (ozn. *m\_VJ*).

Vychystávací jednotka je jednotka množství používaná pro vychystávání produktů – jeho balení a transport. Postup pro přepočet hmotnosti produktu na počet kusů ( $SKU_v$ ) je následovný:

$$SKU = \frac{m}{m_{VJ}} \cdot SKU_{VJ},$$

kde  $m$  je hmotnost produktu. Ze vzorce vyplývá, že může vejít neceločíselný počet kusů. Vzhledem k tomu, že tento přepočet se použije k porovnávání velikosti objemů, nikoli k objednávání zboží, tak tato skutečnost není problém.

Číselník prodejen je obsažen v tabulce *prodejny*. Vybrala jsem z tabulky následující sloupce.

- **ID prodejny** – Označení prodejny nebo skladu
- **Název** – Název prodejny, který obsahuje název města, kde se prodejna nachází.
- **ID kategorie prodejny** – Do jaké kategorie prodejna nebo sklad patří - zda se jedná o malou nebo velkou prodejnu nebo o sklad.

S číselníkem prodejen souvisí číselník pro jejich zařazení do skupin *prodejny\_skupiny*. Skupiny se mohou v čase měnit. Pro analýzu jsou relevantní tyto sloupce:

- **ID prodejny** – Označení prodejny nebo skladu
- **ID skupiny prodejen** – Prodejny jsou sdruženy do skupin. Ty se například používají pro hromadné objednávání, nebo pro plánování promoakcí.

Promoakce se nachází v tabulce *promoakce*. Z této tabulky jsou pro následnou analýzu potřebné údaje o ID produktu, počátečním a koncovém datu promoakce a ID skupiny prodejen, na kterých promoakce platí. Promoakce nejsou přiřazeny na konkrétní prodejny, ale na skupiny prodejen. Pro další analýzy shrinků je třeba zjistit, zda byl konkrétní zaznamenaný shrink v době záznamu v promoakci, nebo ne. Z tohoto důvodu bylo potřeba tabulky spojit pomocí příkazu JOIN s číselníkem *prodejny\_skupiny* podle ID prodejny.

## Tabulky transakcí

V tabulce *transakce* se nachází údaje o všech provedených transakcích, a to jak skladové transakce, tak prodeje a další pohyby na prodejnách. V případě prodejů prodejen jsou údaje agregované podle prodejny, konkrétního produktu a dne transakce, tzn. v této tabulce nelze rozlišit konkrétní prodeje na jednotlivých pokladnách, ale pouze souhrn za jeden den. Tabulka obsahuje údaje za posledních dvacet měsíců.

Tabulka transakcí obsahuje 21 sloupců, jako možné podstatné sloupce pro analýzu jsem vybrala následující sloupce:

- **ID transakce** – Jedinečné pro každou transakci.
- **ID produktu** – Produkt kterého se transakce týká. Každá transakce obsahuje údaje pouze o jediném produktu.

- **ID prodejny** – Transakce je takto přiřazená prodejně, případně skladu.
- **Datum transakce** – Jedná se o obchodní datum, pokud samotná transakce proběhne až po půlnoci uvedeného dne, tak se posílá s datem z předchozího dne, neboť obchodně patří do toho dne.
- **ID promoce** – Příznak zda a v jaké promoční akci se produkt nacházel v čase uvedeném v datu transakce. V rámci zpracování dat vyplynulo, že tento příznak není zcela věrohodný
- **Typ transakce** – Příznak, zda se jedná např. o prodej, výdej, korekce pohybů a jiné.
- **ID shrinku** – Obsahuje označení jednotlivých typů shrinků viz sekce 3.2. Celkem je identifikováno sedmnáct typů shrinků. V databázi tento sloupec označuje i jiná ID než ta, která se týkají shrinků, z toho plyne, že bylo třeba filtrovat pouze ta data, která obsahují sedmnáct identifikačních čísel označující shrinky. V případě, že typem transakce jsou např. příjmy tento parametr nehráje roli.
- **Objem** – Množství produktu uvedené v transakci. U kusových produktů se jedná o celočíselný údaj u vážených to je desetinné číslo.
- Hodnota transakce v nákladové ceně (desetinné číslo).
- Hodnota transakce v prodejní ceně včetně DPH – v případě prodejů se jedná o skutečnou cenu, u zbylých transakcích je uvedena odpovídající cena podle ceníku.

Z transakční tabulky je možné získat tabulkou se záznamy shrinků a tržeb prodejen. Velikost tabulky se záznamy shrinků za jeden kalendářní rok je přibližně 3.5 GB.

Tabulku, která obsahuje údaje o jednotlivých, neagregovaných prodejích na prodejnách společnosti, jsem pro účely této práce nazvala *transakce\_prodeje*. Celkem obsahuje třináct sloupců. Tato tabulka je vhodná pro analýzu shrinků typu snížení ceny, analýzou tohoto typu se tato práce nezabývá. Pro ostatní typy, není tato tabulka relevantní. Stejně tak není třeba zkoumat ceník jednotlivých produktů, protože v souhrnné tabulce transakcí je již uvedená hodnota transakce v prodejní ceně.

## Další datové zdroje

Dále jsem pracovala s daty z databáze Českého statistického úřadu [41]. Na webové stránce úřadu je dostupný odkaz ke stažení souboru ve formátu xlsx. Soubor obsahuje údaje o 237 českých městech za posledních několik desítek let. Některá města obsahují záznamy až sto let nazpět, jiné nemají tak dávno zaznamenanou historii. Dataset obsahuje údaje o lokalitě, o počtu obyvatel, o sňatečných, rozvodech, stěhování obyvatel a další. V rámci přípravy dat bylo potřeba napojit prodejny k údajům o okresu, kraji a počtu obyvatel, kteří žijí v okolí prodejny. Soubor s demografickými údaji bylo třeba převést do tabulkové struktury, kde každý řádek patří jednomu městu, protože původní struktura byla nastavená, co list v souboru, to jedno město. Navíc stejné informace nejsou vždy umístěny stejně na každém listu.

## 4.2 Příprava vzorku dat pro další analýzy

Data z výše popsaných tabulek bylo potřeba sloučit do jedné tabulky, aby nad nimi bylo možné provést analýzy. Napojení dat bylo provedeno v Jupyter Noteboocích v jazyce Python. Na tabulku s evidovanými shrinky bylo třeba napojit číselníky. Podle názvů prodejen byly napojeny na prodejny demografické údaje. Údaje o počtu obyvatelích, jsem rozdělila do pěti kategorií. s Z data transakce bylo extrahováno datum pro den a na základě jeho hodnoty bylo určeno do jaké čtvrtiny měsíce patří. Z data transakce také bylo odvozeno o jaký den v týdnu se jedná pomocí funkce v Python knihovně *pandas*.

Zkoumaný dataset se záznamy shrinků produktů jsem rozšířila o další sledované sloupce, které dávají do srovnání hodnotu shrinku a objem tržeb. Vytvořila jsem takto sloupce: podíl shrinku na celkových tržbách prodejny, podíl shrinku na týdenních tržbách shrinkovaného produktu na prodejně, podíl shrinku a tržeb v kategorii úrovně 1. Tabulka tržeb byla získaná z tabulky *transakce*, pro ID transakcí, které odpovídají prodejům. Tabulka byla agregovaná podle sloupců: prodejna, datum transakce, čtvrtina měsíce, kategorie z úrovně 1.

Zkoumaná březnová data obsahují přes 1,6 milionů řádků. Každý řádek odpovídá jednomu záznamu v databázi shrinku daného produktu. Sledované údaje ve sloupcích jsou:

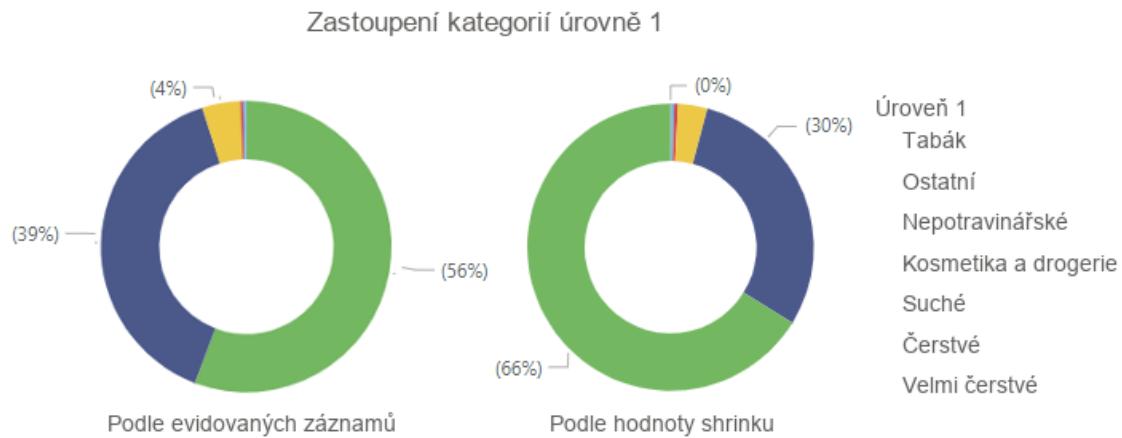
- ID prodejny – kategorická proměnná
- ID produktu – kategorická proměnná
- datum transakce – kategorická proměnná
- ID shrinku – kategorická proměnná
- 1 – kategorická proměnná
- 2 – kategorická proměnná
- 4 – kategorická proměnná
- 5 – kategorická proměnná
- 6 – kategorická proměnná
- Expirace – kategorická proměnná
- Een v týdnu – kategorická proměnná
- Číslo den – kategorická proměnná
- Čtvrt měsíce (rozdelení měsíce na čtyři části) – kategorická proměnná
- Množství – spojitá proměnná
- Ztracená nákladová cena – spojitá proměnná

Původní sloupec datum jsem rozdělila na tři jiné proměnné, a to den v týdnu, číslo dne a období v měsíci a sloupec datum jsem vyneschala.

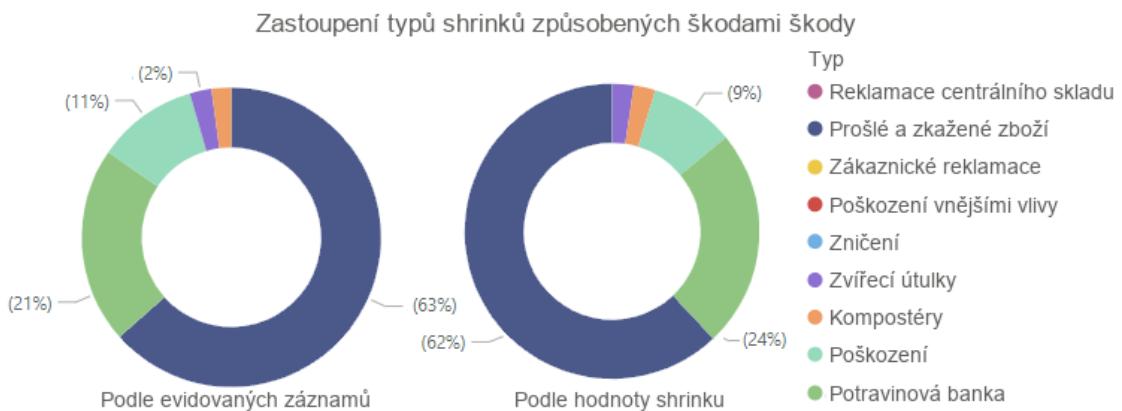
### 4.2.1 Výběr dat

Nejprve jsem graficky analyzovala zastoupení shrinků v závislosti na vybraných proměnných pomocí nástroje Power BI, viz obr. 4.1 a 4.2. Více o analýze v tomto

nástroji je v kapitole 7. V návaznosti na zjištěné zastoupení shrinků v datech jsem se rozhodla vybrat pouze nejvíce zastoupený typ shrinku, který tvoří více jak 62 % celkových nákladů. Ponechaný byl tedy pouze shrnk *prošlé a zkažené zboží*. Obdobně jsem přistupovala k záznamům i z hlediska kategorie produktu úrovně 1, jelikož z grafu je patrné, že majoritní zastoupení mají pouze dvě kategorie, a to kategorie Velmi čerstvé a Čerstvé. Všechny záznamy se zbylými kategoriemi jsem z datasetu pro další analýzy odstranila. Těmito kroky byl zredukován původní počet řádků datasetu na necelých jeden a půl milionu řádků.



Obrázek 4.1: Zastoupení kategorií úrovně 1 v datech z března roku 2023 podle hodnoty shrinku.

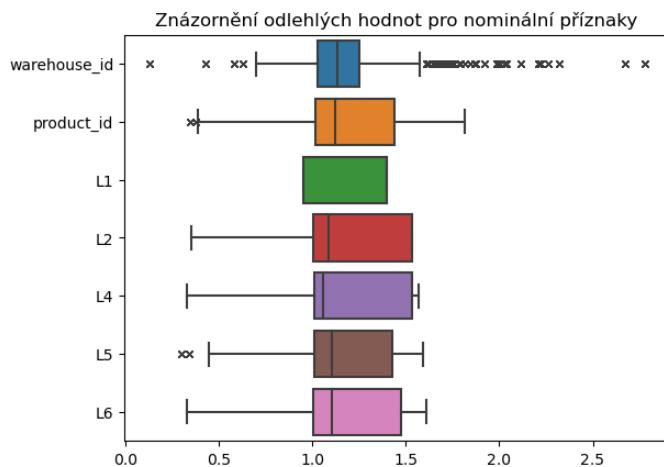


Obrázek 4.2: Zastoupení typů shrinků způsobených škodami v datech z března roku 2023 podle hodnoty shrinku.

Jako cílové sloupce (*target* sloupce) jsem určila sloupec s typem shrinku, množstvím produktu a nákladovou cenou. Zbylých jedenáct sloupců slouží jako vysvětlující proměnné, dále budou označovány jako příznaky pro cílový sloupec. Všechny vybrané příznaky jsou kategorické proměnné, které lze dále rozdělit na nominální a ordinální. Nominální proměnné jsou ID prodejny, ID produktu a kategorie produktové hierarchie. Ordinální proměnné jsou den v týdnu, číslo dne a období měsíce. Ordinální příznaky jsem přeznačila tak, aby každá obsahovala pouze hodnoty od nuly do  $n_p$ , kde  $n_p$  je počet kategorií v  $p$ -tému příznaku.

Pro další postup bylo vhodné přesunout se z nominálních kategorických hodnot na číselné hodnoty. Pro tyto účely jsem zvolila metodu kódování podle cílového sloupce. Neboť toto kódování na numerické hodnoty zachovává velikost datového souboru, to je klíčové vzhledem k tomu, že nominální proměnné ve zkoumaných datech obsahují velký počet kategorií. Např. počet unikátních produktů v datech je více než 19 tisíc, což odpovídá stejnemu počtu kategorií pro tuto proměnnou. Pokud bych použila one-hot kódování<sup>1</sup> mohlo by dojít k zásadnímu zvýšení počtu sloupců v datech, v tomto případě až o desítky tisíc. *Kódování podle cílového sloupce* je podobné převodu, který jsem použila pro ordinální proměnné. Avšak na rozdíl od něj, hodnota, která je kategorii přiřazena, souvisí se zastoupením této skupiny v cílovém sloupci a nesouvisí s uspořádáním hodnot uvnitř příznaku. Nevýhodou je, že takto upravená data mohou být náchylná na overfitting.

Dále jsem se zabývala identifikací odlehlych hodnot. Nejprve jsem vizualizovala hodnoty pomocí grafu pro vybrané proměnné, obrázky 4.3. Z grafu je patrné, že problémová je proměnná s ID prodejny (na grafu vystupuje pod názvem `warehouse_id`). Prodejny, které tvoří outliersy mohou být malé prodejny, které kvůli menšímu počtu celkových produktů nevidují větší počet shrinků.



Obrázek 4.3: Znázornění odlehlych hodnot pro vybrané nominální příznaky.

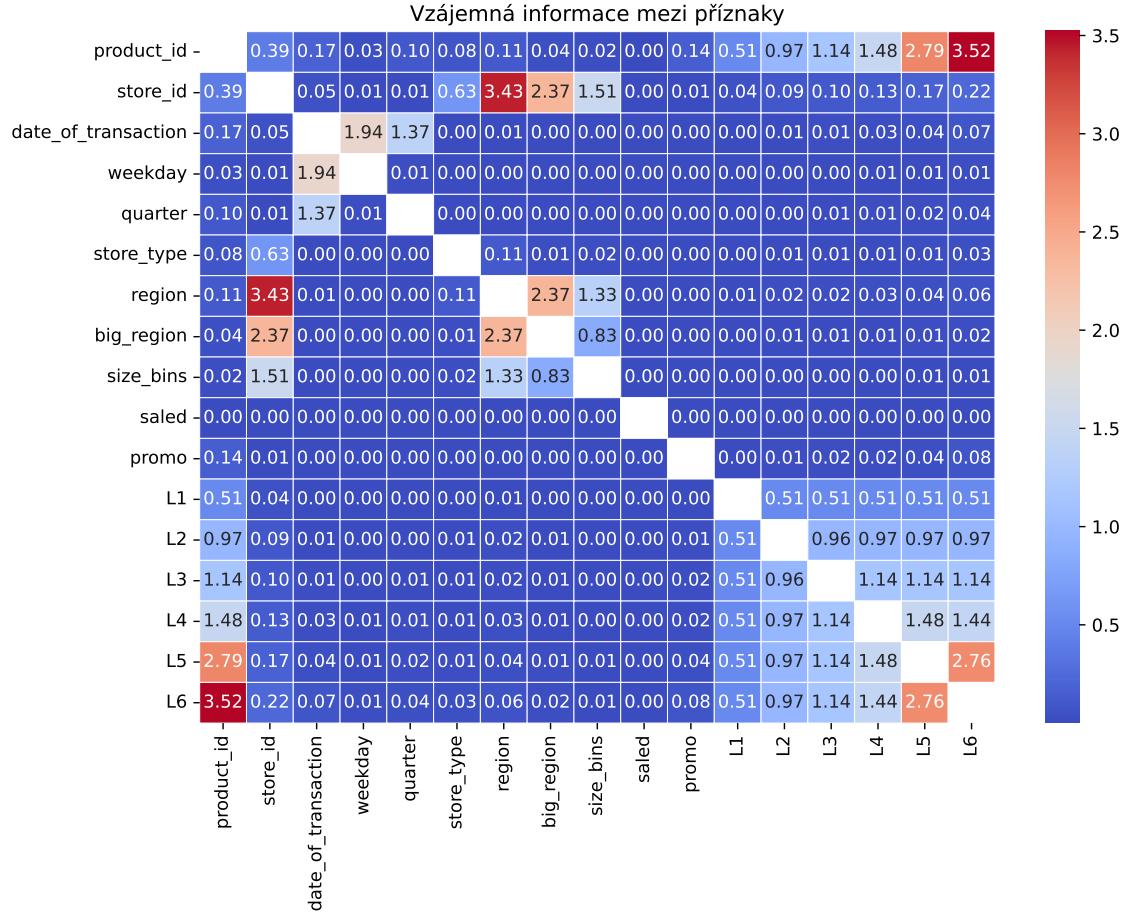
Pomocí Tukeyho testu jsem identifikovala přes 150 000 outlierů pro příznak ID prodejny, čímž se dataset zredukoval. S tímto krokem klesl i počet ostatních outlierů.

V dalším kroku jsem se zaměřila na míru závislosti mezi proměnnými. Použitým metodám je věnována sekce v teoretické části 2.2. Pro měření závislosti jsem již pracovala s kategorickými hodnotami proměnných, tj. bez převodu na spojité hodnoty. Důvodem je to, že když jsem provedla měření korelace na překódovaných datech, byla míra zavislosti ovlivněna cílovým sloupcem, který byl použity pro kódování.

Jako první metodu jsem zvolila  $\chi^2$  statistiku. Vzhledem k vysokému počtu dat je matice příliš řídká, a proto nejsou výsledné hodnoty vypovídající a test je tedy pro tuto úlohu nespolehlivý.

<sup>1</sup>One-hot kódování převádí kategorické hodnoty na numerické takovým způsobem že pro každou kategorii vytvoří samostatný sloupec s binárními hodnotami, kde 1 odpovídá dané kategorii a 0 zbylým kategoriím.

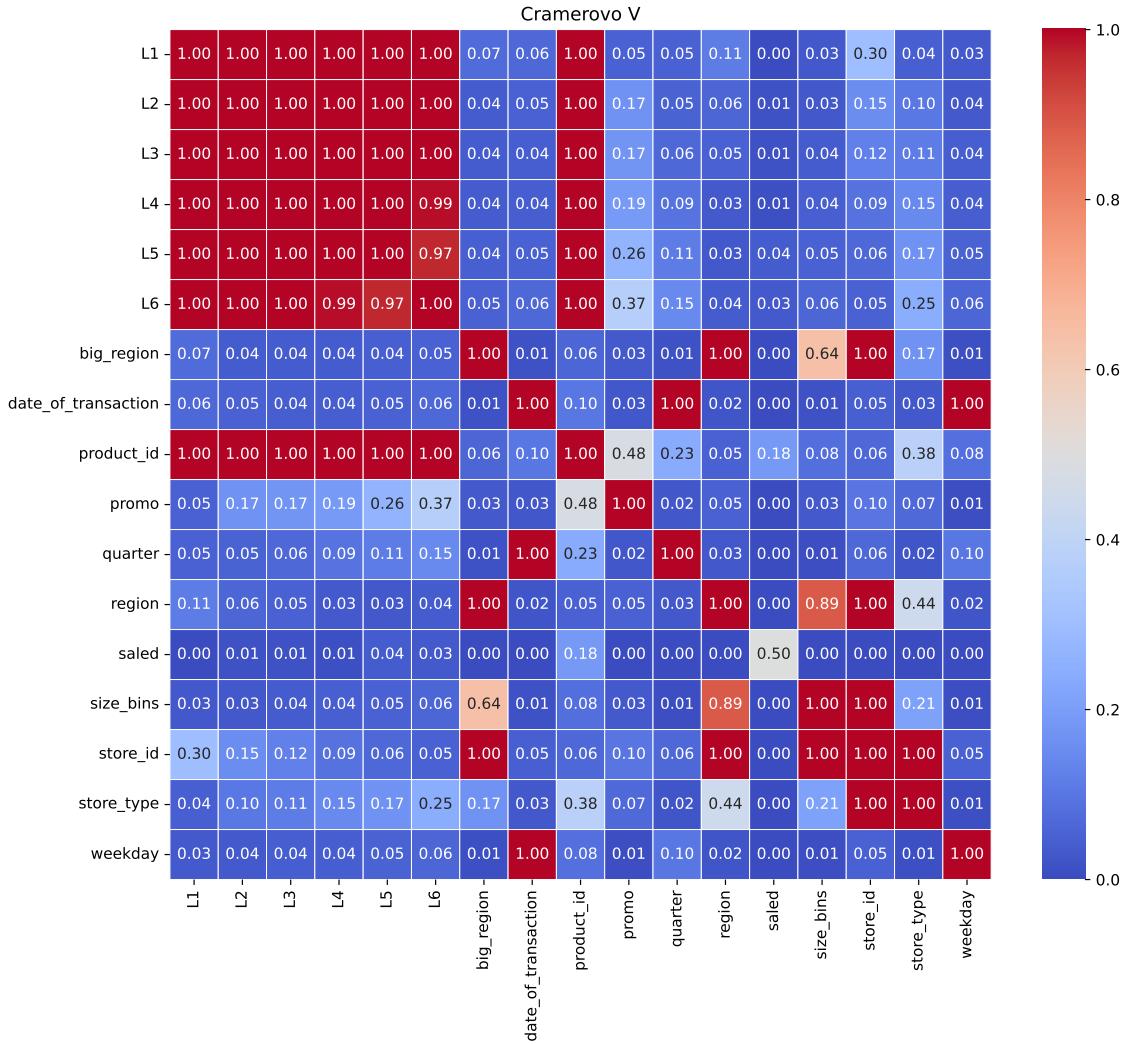
Použila jsem proto míru vzájemné informace, která říká, jaká je podobnost mezi dvěma proměnnými [27]. Matice vypočítaných koeficientů je na obr. 4.4. Jedná se o symetrickou vlastnost, proto jsou hodnoty pod a nad diagonálou stejné. Z výsledků je opět vidět, že ID produktu sdílí informaci s úrovněmi kategorizace tím více, čím je kategorizace jemnější.



Obrázek 4.4: Matice koeficientů vzájemné informace mezi příznaky.

Dále jsem pro znázornění vztahu mezi proměnnými použila koeficient Cramerovo  $V$ . Koeficient jsem postupně počítala pro každou dvojici příznaků. Koeficient nabývá hodnot mezi 0 a 1. Číslo blízké nule indikuje, že mezi proměnnými není asociace, číslo blízké jedničce vysokou závislost [29]. Na obr. 4.5 lze vidět, že pro kategorie 1 až 6 a ID produktu je hodnota koeficientu po zaokrouhlení vždy rovna jedné. Vysoká závislost je pak i mezi příznakem promoakce a ID produktu. Dále logicky mezi datem transakce a dnem v týdnu a obdobím v měsíci. ID prodejny je extrémně závislé s demografickými údaji o lokalitách a typem prodejny.

Další statistikou spočtenou na datech je Theilovo  $U$  (neboli koeficient nejistoty), který opět nabývá hodnot mezi 0 a 1 a měří vztah mezi dvěma proměnnými. Na rozdíl od předchozích statistik tento koeficient není symetrický a z výsledků lze vyvodit, ze které proměnné ze dvou zkoumaných můžeme vyvodit informaci o druhé proměnné [30]. Z výsledků zobrazených v matici na obr. 4.6 plyne, že z ID produktu lze vyvodit část informace o kategoriích. Zatímco úrovně 1 a 2 o ID produktu mnoho



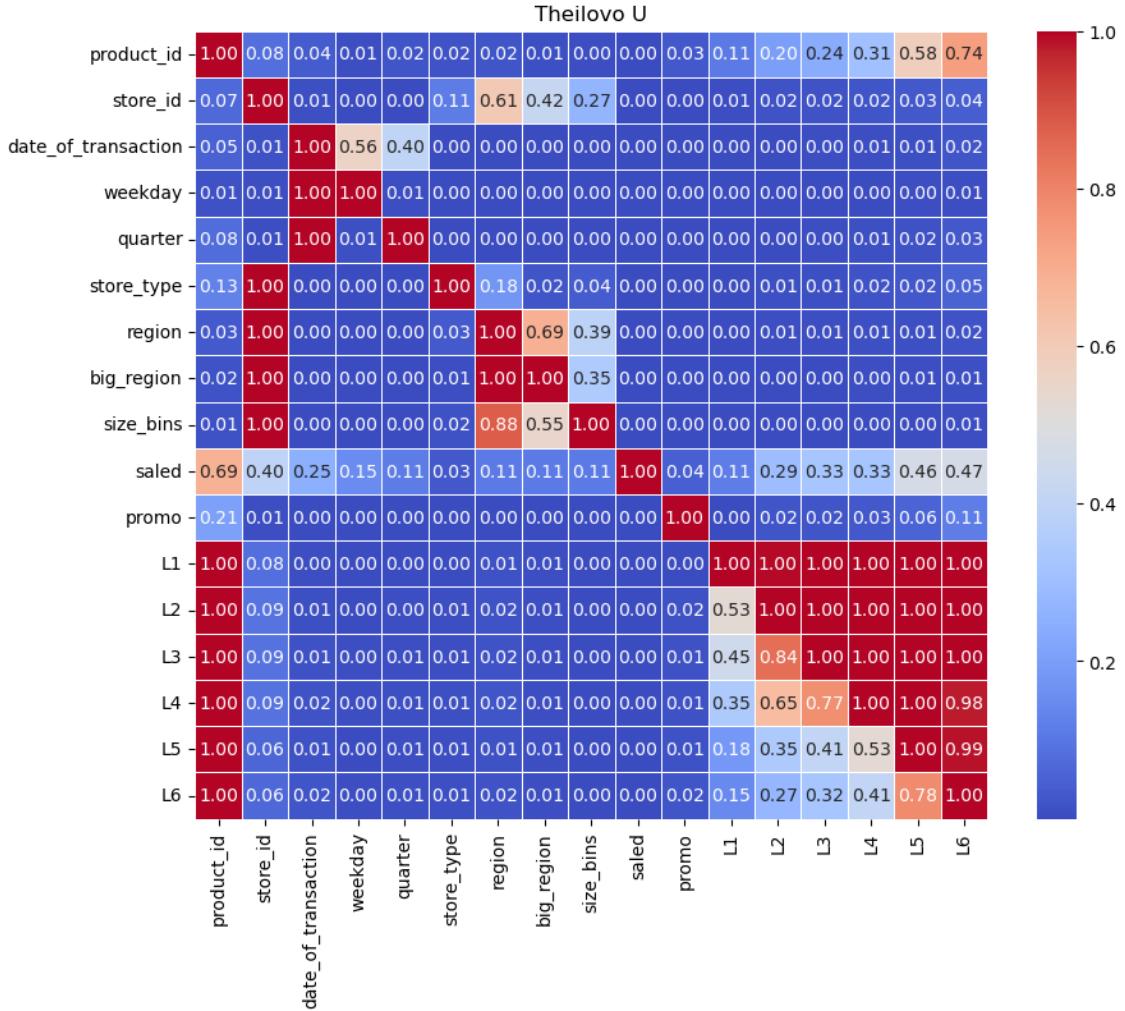
Obrázek 4.5: Matice koeficientů Cramerovo  $V$  pro kategorické příznaky.

informace nenesou. Jak bylo ukázáno i v předchozích statistikách a jak vyplývá z logiky pro získání dne v týdnu a období měsíce, číslo dne nese informaci o těchto dvou příznacích.

Z vypočítaných statistik na datasetu je patrné, že některé příznaky jsou významně závislé, a proto je třeba je z dat odstranit. Kandidáti na vynechání jsou kategorie 2, 4, 6 a číslo dne.

V dalším testu jsem otestovala multikolinearitu dat pomocí rozptylového inflačního faktoru (VIF). Jako hraniční faktor jsem zvolila hodnotu 40 VIF. Vysvětlující proměnné jsem odebírala z datasetu postupně a odebírání jsem ukončila až, když hodnota VIF nebyla nižší než hraniční. Tímto došlo k redukci příznaků z jedenácti na pět, a to na kategorii 1, číslo dne, období měsíce, ID prodejny a den v týdnu. Hodnoty koeficientu VIF na datech jsou na obr. 4.7.

Jako další metodu po výběru příznaků jsem vypočítala hodnotu koeficientů vzájemné informace mezi všemi příznaky s cílovým sloupcem - ID shrinku. Na obrázku 4.8 lze vidět, jak jednotlivé proměnné souvisí s cílovým sloupcem. Zde můžeme vidět,

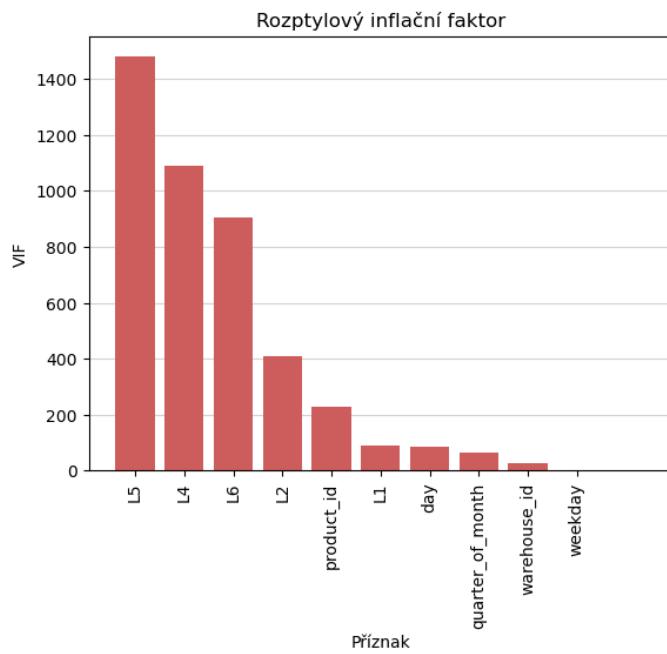


Obrázek 4.6: Matice koeficientů Theilovo  $U$  mezi příznaky.

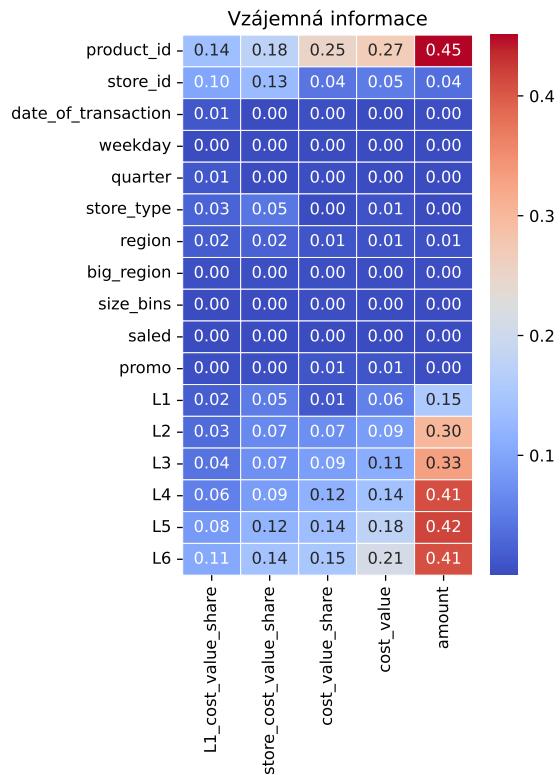
že významnost příznaku ID prodejny klesla o téměř polovinu. Nejvíce informace je sdíleno s ID produktu, kategorií 6, dále 5, 4, 2. Příznaky související s časovými údaji podle tohoto kritéria nenesou mnoho společné informace.

Jako hlavní metodu pro výběr proměnných jsem se rozhodla použít metodu PCA, případně MCA. Tuto metodu je možné použít, protože kategorické proměnné byly převedeny na číselné pomocí základního kódování kategorických hodnot na hodnoty 0 až  $n$ , kde  $n$  je počet kategorií v příznaku. Toto kódování má bohužel tu nevýhodu, že dává kategoriím pořadí, i když jedna kategorie není lepší než jiná. Alternativou je použití metody MCA, která se používá pro kategorické datasety. Výsledky pro metodu MCA jsou uvedeny dále v textu. Ve své práci jsem využila implementaci PCA v knihovně *Prince* v jazyce Python. Předtím než jsem metodu aplikovala jsem otěstovala předpoklad homoskedasticity, tedy shodnost rozptylů v datech, pomocí Bartlettova testu implementovaného v knihovně *factor\_analyzer*. Nulová hypotéza o shodnosti rozptylů nebyla vyvrácena ( $p$ -hodnota vyšla nulová). Metodu PCA je proto možné použít.

Na obrázcích 4.9 a 4.10 je znázorněno prvních deset komponent a rozptyl který v da-



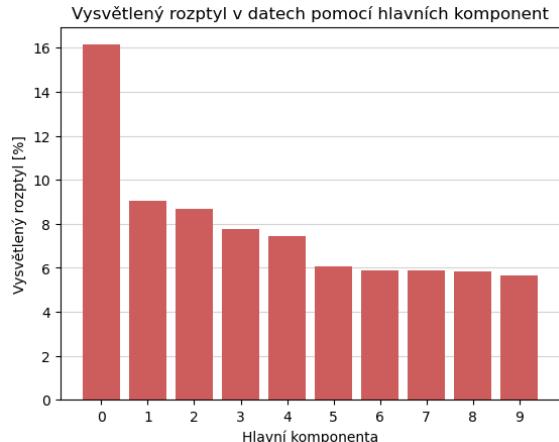
Obrázek 4.7: Rozptylový inflační faktor.



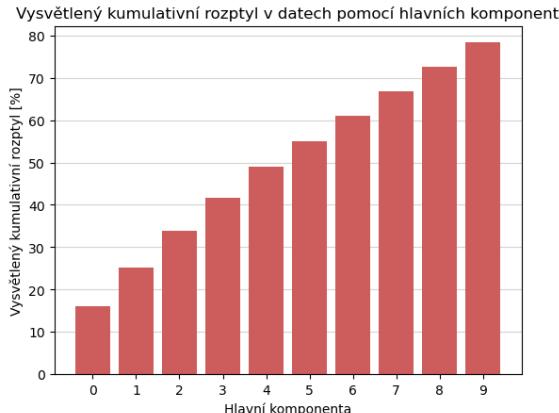
Obrázek 4.8: Vzájemná informace mezi příznaky cílovými sloupcí.

tech vysvětlují. Desátá komponenta (označená č. 9) spolu s předchozími vysvětluje téměř 80 % variability dat. V dalším kroku jsem vypočítala příspěvky příznaků k této komponentě a vybrala jsem ty příznaky, které přispívají nejvíce. Jejich příspěvek je znázorněný na obr. 4.11. Na základě výsledků analýzy hlavních kompo-

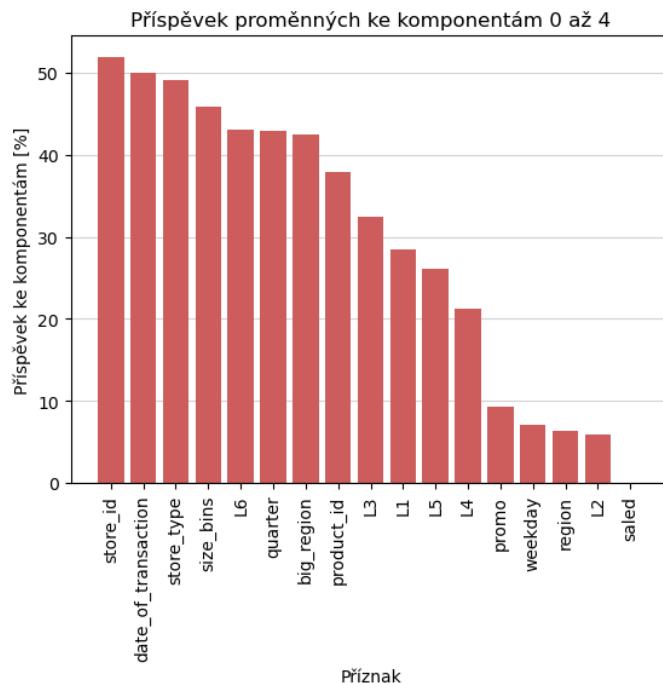
není lze říci, že nejvíce rozptylu v datech nesou příznaky – ID prodejny, den v týdnu, expirace, den a období v měsíci.



Obrázek 4.9: PCA - vysvětlený rozptyl hlavních komponent.



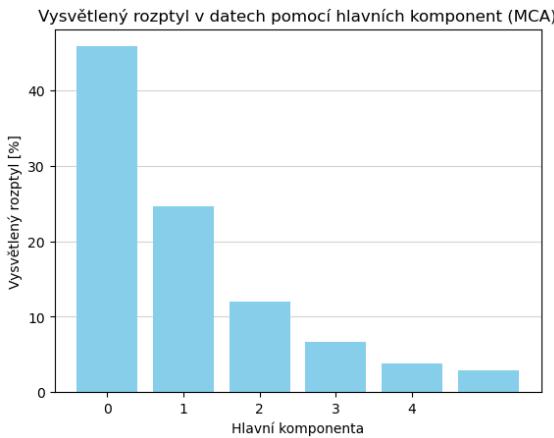
Obrázek 4.10: PCA - kumulativní vysvětlený rozptyl hlavních komponent.



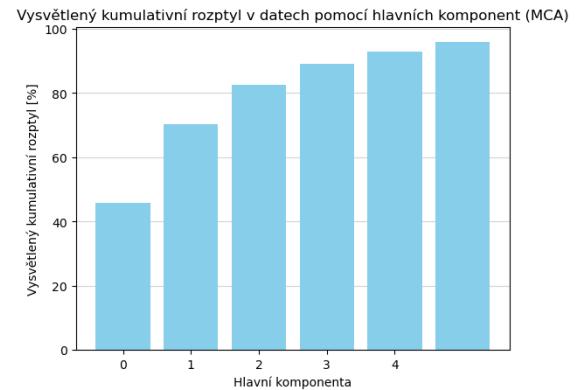
Obrázek 4.11: Příspěvek proměnných ke komponentám 0 až 10.

Jak již bylo zmíněno pro redukci dimenzionality, příp. výběr proměnných, u kategorických dat lze použít metodu MCA, opět jsem využila implementaci z knihovny *Prince*. V této implementaci jsou nominální kategorické hodnoty kódovány tak, že narůstá počet sloupců, a proto bylo nutné, vzhledem k nárokům na paměť k uložení matice, omezit množství dat. Vybrala jsem náhodný 20% vzorek dat, na které jsem MCA aplikovala. Vypočítala jsem prvních pět komponent, které dohromady popisují 79 % variability dat. Jelikož byla každá kategorie chápána jako samostatná proměnná příspěvky jednotlivých příznaků ke komponentám byly rozmístěny mezi

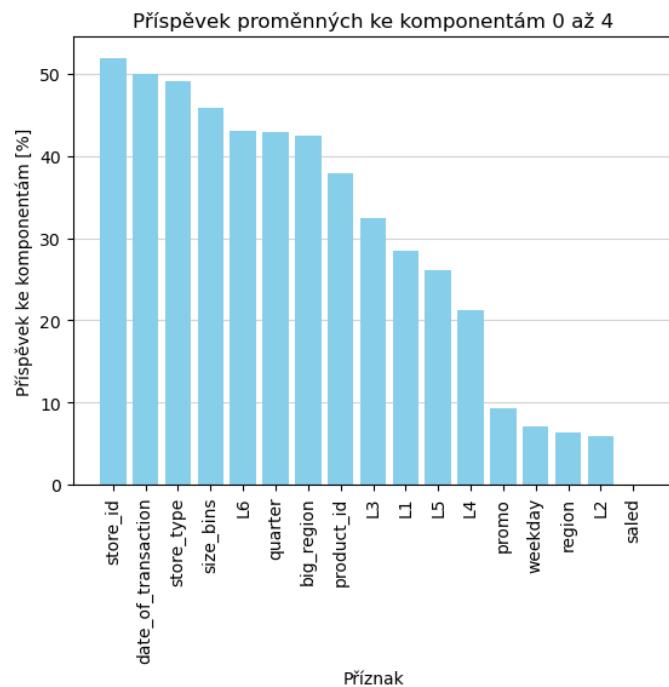
všechny kategorie, nikoli k jednotlivým příznakům. Z obrázků 4.12 až 4.14 je vidět, že výsledky jsou obdobné jako pro předchozí metodu.



Obrázek 4.12: MCA - vysvětlený rozptyl hlavních komponent.



Obrázek 4.13: MCA - kumulativní vysvětlený rozptyl hlavních komponent.



Obrázek 4.14: Příspěvek proměnných ke komponentám 0 až 5.

## Shrnutí pro výběr dat

Na základě předchozích metod bylo z původních příznaků datasetu vybráno několik příznaků, které popisují hodnotu shrinku. Vzhledem k tomu, že různé metody vybraly různé příznaky, níže je sepsáno shrnutí, které říká, jaké příznaku jsou na základě zkoumaných dat relevantní vzhledem k naměřeným hodnotám shrinku.

Korelované jsou hodnoty ID produktu, 6, 5, 4 a expirace. Dále také z označení dne lze určit období měsíce. Ze zmíněných korelovaných příznaků není proto vhodné začlenit více než jeden příznak. Pokud je tato myšlenka aplikována na výsledky metod PCA a MCA a výsledků zjištěných pomocí hraniční hodnoty VIF.

Následující sloupce byly získány podle hodnoty rozptylového inflačního faktoru. Touto metodou byl navržen i sloupec s číslem dne, ten však z důvodů korelace nebyl zahrnutý – úroveň 1, období měsíce, ID prodejny a den v týdnu.

Metodou PCA bylo zjištěno, které příznaky nejvíce přispívají ke komponentám, které popisují téměř 96 % rozptylu v původních datech - jedná se o příznaky ID prodejny, den v týdnu, období v měsíci a číslo dne. Naopak metoda MCA vybrala kategorie 4 až 6 jako důležité. Sloučením a přihlédnutím ke korelačním koeficientům byly vybrány pět příznaků – ID prodejny, den v týdnu, období v měsíci, 5.

1. Následující sloupce byly získány podle hodnoty rozptylového inflačního faktoru. Touto metodou byl navržen i sloupec s číslem dne, ten však z důvodů korelace nebyl zahrnutý

- 1.1. 1, období měsíce, ID prodejny a den v týdnu.

K této variantě existují i dvě alternativy, ve kterých je obměněna úroveň kategorizace produktu:

- 1.2. 5, období měsíce, ID prodejny a den v týdnu

- 1.3. 4, období měsíce, ID prodejny a den v týdnu

2. Metodou PCA bylo zjištěno, které příznaky nejvíce přispívají ke komponentám, které popisují téměř 96 % rozptylu v původních datech - jedná se o příznaky ID prodejny, den v týdnu, období v měsíci a číslo dne. Naopak metoda MCA vybrala kategorie 4 až 6 jako důležité. Sloučením a přihlédnutím ke korelačním koeficientům byly vybrány pět příznaků

- 2.1. ID prodejny, den v týdnu, období v měsíci, 5.

Tato varianta příznaků byla ještě rozšířena o příznaky, které se týkají produktů. Přidané příznaky jsou spolu korelované, přesto

- 2.2. ID prodejny, den v týdnu, období v měsíci, 5, 2

- 2.3. ID prodejny, den v týdnu, období v měsíci, 5, 2, ID produktu

- 2.4. ID prodejny, den v týdnu, období v měsíci, 2, ID produktu

- 2.5, období v měsíci, ID prodejny, den v týdnu, ID produktu, target

# Kapitola 5

## Korelační analýza

Tato kapitola se věnuje popisu korelační analýzy pro zjištění důvodu shrinků produktů. Tuto analýzu je možné spustit na data libovolné společnosti, pokud obsahují vstupy, které jsou definované dále. Analýza byla napsána v jazyce Python, jako sada funkcí sdružená do modulu. Ukázka volání funkcí pro spuštění analýzy je pak vytvořena v Jupyter Notebooku. V této kapitole je popsána implementace funkcí a princip analýzy. Analýza se hodnotí korelací mezi hodnotou shrinku a tržbami, na základě výsledku roztrídí produkty ve vstupních datech do několika kategorií.

### 5.1 Postup

V rámci analýzy se porovnávají pouze záznamy produktů, které se vyskytují ve stejné kategorii. Jedno pozorování je na agregované na produkt, prodejnu a den záznamu. Základní hypotéza je, že shrink produktu může být ovlivněn promoakcemi jiných produktů v kategorii.

Hodnotu shrinku jsem porovnávala s následujícími ukazateli.

- Tržby daného produktu.
- Tržby daného produktu, které byly v daný den v promoakci - ukázalo se, že takové, až na výjimky nejsou.
- Součet tržeb všech ostatních produktů v kategorii.
- Součet tržeb všech ostatních produktů v kategorii, které byly v daný den v promoakci.
- Součet tržeb všech ostatních produktů v kategorii, které byly v daný den v promoakci nebo byly v rozmezí jednoho týdne po promoakci.

Ke každému ukazateli, jsem ještě vytvořila analogický ukazatel, který uvažoval zpozdění shrinku. V takovém ukazateli, se nebrala hodnota prodeje ze stejného dne, jako byl den záznamu shrinku, ale hodnota z předchozího dne. Důvodem pro vytvoření takových ukazatelů byla hypotéza, že shrink se může projevit až další den po uskutečněných tržbách. Důvodem může být to, že

Na základě korelační analýzy je možné rozdělit produkty v kategorii do pěti skupin:

**Kategorie P** - Produkty, které si samy způsobují shrink.

**Kategorie O** - Produkty, jejichž shrink je způsoben tím, že ostatní produkty v kategorii jsou v promoakci.

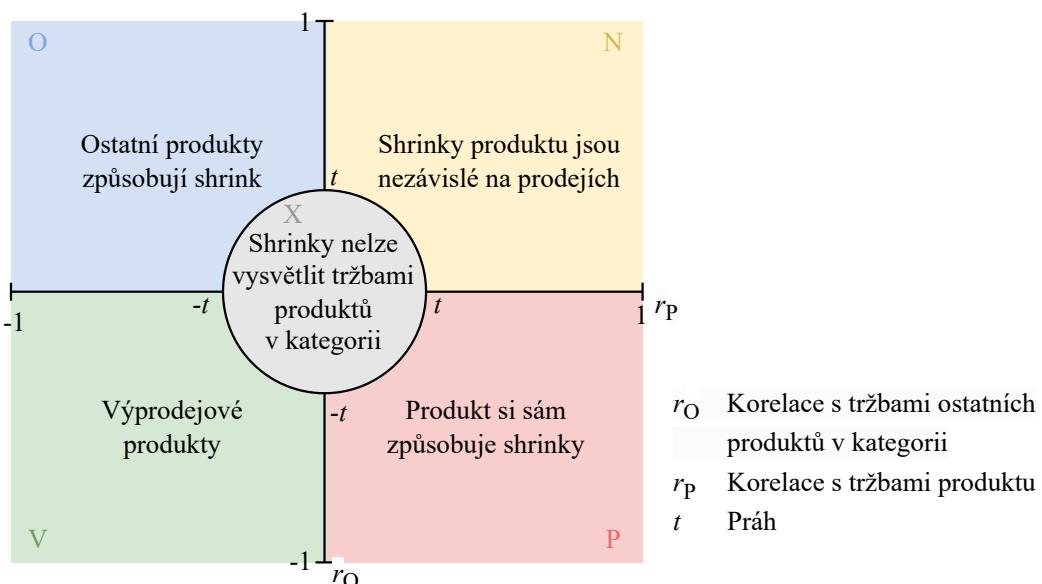
**Kategorie X** - Produkty, jejichž shrink se nepodařilo vysvětlit pomocí korelační analýzy.

**Kategorie V** - Produkty, které jsou úspěšné ve výprodejích. Produkt se hodně prodává a zároveň má malé shrinky.

**Kategorie N** - Produkty, jejichž shrink je nezávislý na svých tržbách i tržbách ostatních produktů v kategorii. Shrink zůstává v podobném poměru, tj. pokud jsou tržby vyšší je i shrink vyšší, pokud jsou tržby nižší, je nižší i shrink.

Na obrázku 5.1 je znázorněno rozdělení produktů vzhledem ke korelačnímu koeficientu. Kategorie jsou pro lepsí orientaci na obrázku oddělené i barevně, zároveň s popisem je u každé části i písmenné označení kategorie.

Korelace hodnoty shrinku produktu s tržbami



Obrázek 5.1: Kategorizace produktů podle korelace hodnoty shrinku produktu s tržbami.

Hypotéza pro zařazení do kategorie P je následující:

Pokud je korelační koeficient zaznamenaného shrinku s tržbami téhož produktu kladná, produkt si způsobuje shrinky sám. Abych mohla tuto hypotézu potvrdit, nebo vyvrátit, je třeba statisticky otestovat významnost korelačního koeficientu. Formulovala jsem nulovou hypotézu  $H_0$  a alternativní hypotézu  $H_A$  pro koeficient  $r_P$ , který měří korelací mezi hodnotou shrinku a tržbami produktu.

$$H_0 : r_P = 0 \quad \text{Výběry nejsou korelované.}$$

$$H_A : r_P \neq 0 \quad \text{Výběry jsou korelované.}$$

Hypotéza pro zařazení do kategorie O je následující:

Pokud jsou kladně korelované hodnoty zaznamenaného shrinku a tržby ostatních produktů a zároveň korelace shrinků produktu s vlastními tržbami je záporná, potom lze vyslovit hypotézu, že shrinky na produkту jsou způsobené ostatními produkty v promoakci. Pro toto tvrzení je opět nutné statisticky otestovat koeficienty korelace. Pro koeficient  $r_P$  je statistický test stejný jako v předchozím případě. Pro koeficient  $r_O$  měřící, jak jsou korelované shrinky a tržby ostatních produktů, je třeba otestovat následující hypotézy.

$$H_0 : r_O = 0 \quad \text{Výběry nejsou korelované.}$$

$$H_A : r_O \neq 0 \quad \text{Výběry jsou korelované.}$$

Pokud na zvolené hladině významnosti zamítнемe nulovou hypotézu pro zkoumané korelační koeficienty, můžeme tvrdit že s danou pravděpodobností je koeficient statisticky významný. Na základě hodnoty korelace lze pak produkt zařadit do příslušné kategorie. Produkty, u kterých nelze zamítnout, není možné zařadit do tří uvedených kategorií.

Pro výpočet korelačního koeficientu je ještě třeba ověřit předpoklady. Pro Pearsonův korelační koeficient se jedná o předpoklad normality dat, shodnost rozptylů a nezávislost dat. Pro Spearmanův korelační koeficient není třeba splňovat tyto předpoklady.

## 5.2 Implementace

V této části je uveden přesný postup pro získání kategorizace produktů. Kód je napsaný v jazyce Python. Součástí kódu je výběr kategorií, které jsou zkoumány, propojení dat shrinků, prodejů a promoakcí, výpočet korelace a ověření předpokladů, statistické testování a rozřazení produktů.

### 5.2.1 Vstupy a výstupy

Pro korelační analýzu zaznamenaných shrinků s tržbami dalších produktů je třeba zajistit data, které se týkají zaznamenaných prodejů, produktů a prodejen. V následující části jsou popsány tabulková data, která jsou nezbytná pro správné spuštění analýzy. Dále jsou definované i vstupy, které musí definovat uživatel pro specifikování názvů konkrétních sloupců v souborech a parametry pro analýzu.

Celkem jsou požadovány čtyři vstupní tabulky - *záznamy shrinků*, *záznamy prodejů*, *záznamy o promoakcích*, *číselník produktů s rozdelením produktové hierarchie*. Tabulka se zaznamenanými shrinky musí obsahovat sloupec s datem záznamu, ID produktu, ID prodejny, hodnotu zaznamenaného shrinku. Tabulka s prodeji potřebuje

stejné sloupce jako tabulka se shrinky s výjimkou že hodnota prodejů je celková prodaná částka, která byla zaznamenaná na dané prodejně v jeden den u daného produktu. Tabulka s údaji o promoakcích by měla obsahovat ID produktu, kterého se promoakce týká, začáteční a koncové datum promoakce a ID prodejny, pro kterou promoakce platí. Všechny záznamové tabulky musí pokrývat stejné časové období. Odbor může být libovolně dlouhé. Tabulka produktové hierarchie obsahuje ID produktu, jeho název a libovolně hluboký strom hierarchií. Každá úroveň stromu má vlastní sloupec. Všechny úrovně jsou vyplňené pro každý produkt, tato podmínka je nutná jen pro kategorie, které bude chtít uživatel využít při analýze. Tabulka s hierarchií produktů slouží k tomu, aby mohla být napojena na ostatní tabulky a data se pak mohla vyfiltrovat pouze na záznamy týkající se vybrané kategorie.

Před spuštěním hlavní výpočetní části musí uživatel vypsat konkrétní pojmenování sloupců v tabulce do proměnných. Sloupce, které v různých tabulkách označují tytéž hodnoty, musí mít stejný název. V následujícím kódu 5.1 je ukázka zadání. V komentářích je slovní popis o jaký sloupec se jedná. Sloupec by však měl být jasný přímo z názvu proměnné.

```

01 product_col      = "product_id"                      # Product ID column
02 product_name_col = "name"                           # Product name
03               column
04 whs_id_col      = "warehouse_id"                   # Store ID column
05 date_col         = "date_of_transaction"           # Date of
               transactions column - for sales and shrinks tables
06 value_col_shrink = "cost_value"                    # Column with value
               of shrinks (shrink table)
07 value_col_sales = "cost_value"                    # Column with value
               of total sales (sales table)
08 promo_col_from   = "promotion_date_from"          # Starting date of
               promotion (promotion table)
09 promo_col_to     = "promotion_date_to"            # Starting date of
               promotion (promotion table)
09 categories       = ["L3", "L4", "L5", "L6", "name"] # Categories
               that we want to map to product ID (product hierarchy)
```

Kód 5.1: Definice konkrétních názvů sloupců.

Uživatel dále zadefinuje formát data, který se používá v datumových sloupcích, aby se tyto sloupce mohly převést z textového řetězce na typ `datetime`. V proměnné `category_column` je třeba vybrat jednu kategorii (název sloupce). Na této úrovni se poté budou procházet jednotlivé kategorie, v rámci každé z nich se pak budou porovnávat a třídit produkty. V dalších proměnných může uživatel změnit umístění tj. název složky, kam se ukládají výsledky kategorizace a grafy. Složky s těmito názvy se vytvoří jako podsložky aktuální cesty.

## 5.2.2 Spuštění analýzy

Analýzu lze spustit pomocí předpřipraveného Jupyter Notebooku v jazyce Python. V první buňce notebooku se načítají potřebné balíčky a modul s definovanými funkcemi pro analýzu.

V dalším buňce jsou definovány vstupní parametry do funkcí - názvy sloupců a úrovně produktové hierarchie. V následující buňce se načítají potřebné datasety.

Přehled potřebných vstupů je v sekci 5.2.1. V závislosti na konkrétních datech je třeba specifikovat, jak se mají tabulková data načíst - jedná se např. o parametry pro oddělovač hodnot v řádku, nebo značení desetinné čárky v datech. Pokud nahrané datasety pro prodeje, shrinky a promoakce mají pouze sloupec ID produktu s nenapojenou produktovou hierarchií, je třeba ji připojit.

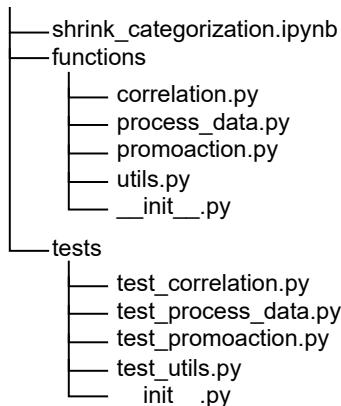
V další buňce se spouští samotná analýza. Nejprve se spustí funkce, která vrátí seznam kategorií, které jsou nejrizikovější. Je třeba definovat na které úrovni hierarchie se budou kategorie prohledávat a také, kolik kategorií budeme chtít prozkoumat. Nalezené kategorie se dále prochází v cyklu.

Z datasetů se vyfiltrují pouze záznamy dané kategorie. Pokud jsou v prodejích záznamy, kde je prodej kladný, tak se tyto záznamy vynechají. V dalším kroku se k údajům o prodejích navážou promoakce. Poté je spuštěna korelační analýza, lze definovat, jaká metoda se má použít s jakou alternativní hypotézou. Případně zda uživatel chce zkoumat shrinky oproti zpožděným prodejům a zda se má analýza zabývat pouze promočními prodeji, nebo i popromočními.

Vypočítané korelační koeficienty se kategorizují a výsledky se uloží do souboru. Zároveň se pro zkoumanou kategorii uloží i graf závislosti shrinků na promočních prodejích.

### 5.2.3 Popis funkcí a struktura kódu

Kód pro korelační analýzu je umístěn ve složce `shrink_categorization`, struktura složky je vidět na obrázku 5.2.



Obrázek 5.2: Struktura souborů pro kód zpracovávající korelační analýzu.

Funkce jsou rozčleněny do modulů podle toho, na jaký výpočet jsou zaměřené. Každá funkce má je zdokumentovaná pomocí docstring obsaženého ve své definici. Dokumentace funkce se skládá ze stručného popisu, co funkce dělá, jaké má vstupní parametry a jaký je jejich význam a co funkce vrací. Funkce jsou otestované pomocí unit testů.

Pro práci s tabulkovými daty, které jsou hlavním vstupem, jsem použila balíček *pandas* jazyka Python.

## Funkce pro přiřazení kategorií k produktům

Jak je uvedeno na začátku sekce 5.2.1, uživatel musí specifikovat názvy sloupců kategorií, které bude v analýze používat. Seznam těchto kategorií je pak parametrem pro funkci `assign_levels`. Další parametry jsou DataFrame, kam se mají kategorie napojit a DataFrame odkud se kategorie napojují. Tyto DataFramy musí mít společný sloupec, podle kterého se napojení provede. Defaultně se jedná o sloupec s ID produktu. Defauletně se provádí *left join*, aby nedošlo ke ztrátě dat, kdyby nějaký produkt neměl v DataFramu kategorií zastoupení. Funkci je také možné předat další argumenty, které se dají volat ve funkci `merge` knihovny *pandas*. V analýze shrinků jeden řádek dat odpovídá transakci jednoho produktu, proto byl zvoleno ID produktu jako propojovací sloupec.

## Funkce pro vytipování rizikových kategorií

Funkce `define_risk_categories` vybere prvních  $n$  kategorií v dané produktové hierarchii, kde suma hodnot v dané kategorii, je nejvyšší, resp. nejnižší. Funkce vrací seznam těchto kategorií. Prvním vstupním parametrem je DataFrame, který obsahuje minimálně tři sloupce. Tyto sloupce je třeba definovat jako další parametry funkce. Jedná se o sloupec `value_column`, ve kterém jsou hodnoty, které ohodnocují řádky DataFramu a kategorie. Další sloupec je jedna z úrovní produktové hierarchie, ve sloupci se nachází názvy, nebo jiné označení, kategorií. Posledním povinným parametrem je počet kategorií, které má funkce vrátit. Pokud je zadán tento počet tak, že je větší než je počet unikátních kategorií, vrátí se všechny kategorie seřazené od nejriskovější. Dale je funkci možné předat keyword argumenty, které se předají funkci `sort_values` z knihovny *pandas*. Jedná se např. o parametr pro vzestupné, nebo sestupné řazení. Defaultní řazení je vzestupné, což znamená, že se vezmou kategorie s nejnižší hodnotou. V této analýze sledujeme vyhozené množství, resp. peníze. Tento ukazatel je záporný, tedy vzestupné řazení vybere ty kategorie, jejichž ztráta byla nejvyšší. Vrácený seznam kategorií je tedy seřazen od nejriskovější kategorie.

## Funkce pro výběr pouze dané kategorie ze všech záznamů

Ve funkci `select_category` jsou vstupem DataFrame, název kategorie a úroveň, ve které se daná kategorie nachází. Funkce vrací DataFrame pouze s těmi řádky, kde je obsažena jmenovaná kategorie. V případě, že tato kategorie v datech není, je vrácen prázdný DataFrame.

Tato funkce je volána ve funkci `process_dataframes`. Vstupy jsou totožné, avšak namísto jednoho DataFramu je možné jich zadat více jako samostatné parametry. Funkce vrací seznam všech vstupních DataFramů, a to pouze řádky, které obsahují zadanou kategorii.

## Funkce pro přiřazení promoakcí

V rámci korelační analýzy bylo potřeba přiřadit k jednotlivým zaznamenaným transakcím, zda byl produkt v den záznamu v promoakci nebo nikoli. V ideálním případě by tento příznak mohl být již uvedený u každého záznamu. Pokud tomu tak, ale není, je nutné data o promocích provázat na základě data, produktu a prodejny podle číselníku promoakcí. Data vybrané společnosti, na jejichž datech analýza probíhá, nemají promoakce přímo napojené na již proběhlé, zaznamenané transakce. Data o promoakcích jsou uložena v číselníku promoakcí. Ten obsahuje ID produktu, prodejny, začátek a konec promoakce a prioritu promoakce. V jeden den může být více promoakcí, v takovém případě platí ta s nejvyšší prioritou.

Základní funkce pro přiřazování promoakcí k záznamům s transakcemi se nazývá `map_all_promotions`. Tato funkce propojí DataFrame s promoakcemi s druhým DataFrame s transakcemi. Může se jednat jak o záznamy shrinků, tak i o prodeje. Důležité je, že tento DataFrame má sloupec s datem, protože díky datu pak lze identifikovat správnou promoakci. Nalezení správné promoakce je implementováno až ve funkci `promo_`, viz dále v této sekci. Funkce `map_all_promotions` nejprve provede vnitřní spojení (neboli *inner join*) obou vstupních DataFramů podle definovaných sloupců v parametrech funkce. Tyto sloupce jsou vzhledem k datům společnosti - sloupce s ID produktu a ID prodejny. Tím je docíleno toho, že z promoakcí získáme pouze ty záznamy pouze těch produktů, které se prodali, a které zároveň byly v promoakci. Duplicity záznamy se vynechají. U každého takového záznamu spočítá, kolikrát byl k němu byla přiřazeno promoakce (tj. kolik bylo promoakcí ve sledovaném období pro daný produkt a prodejnu) a ke každému záznamu toto číslo přiřadí. Spolu s číslem se přiřadí i identifikátor promoakce pro dvojici produkt-prodejna (viz tabulka C), tím je určena skupina k sobě patřících záznamů. Takto označené záznamy se připojí k původnímu DataFrame s transakcemi. Záznamy, kde není žádná promoakce je počet promoakcí roven nule, zbylé hodnoty nejsou definované. Dále se identifikátor upraví tak, že dokáže rozlišit unikátní promoakci na trojici produkt-prodejna-datum (viz tabulka D). V tabulce 5.1 jsou umělá ukázková data, na kterých je znázorněno spojování dat.

Z ukázky a z popsaného postupu plyne, že výsledný DataFrame může mít více řádků než ten původní, ke kterému se přidávali promoakce. V dalším kroku je tedy potřeba určit, která z přiřazených promoakcí probíhala ve stejný čas jako je čas transakce. K tomu jsem vytvořila funkci `label_date_with_promo`. V této funkci je každý řádek promoakce označen jednou ze tří možností: `no_promo`, `promo`, `after_promo`. Tedy zda je datum transakce během promoakce, nebo nikoli, nebo zda je v rozmezí týden po evidované promoakci. Vzniklý příznak byl pojmenován jako typ promoakce. Ve funkci se pracuje pouze se záznamy u nichž byla nalezena alespoň jedna možná promoakce, tj. transakce, kde dvojice produkt-prodejna existuje i v promoakcích. Zbylé řádky tato funkce neoznačuje. V tabulce 5.2 jsou podle těchto pravidel označené jednotlivé řádky<sup>1</sup>.

V dalším kroku je třeba vybrat pouze jednu přiřazenou promoakci o to se stará funkce `find_duplicated_records`. Tato funkce vrací seznam indexů řádků Da-

---

<sup>1</sup>Zbylé sloupce jsou vynechané, protože pro ukázku příznaku nejsou podstatné.

Tabulka 5.1: Umělá data pro znázornění přiřazování promoakcí k transakcím.

Tab. A: Tabulka promoakcí

ID produktu	Prodejna	Začátek promoakce	Konec promoakce
0001	01	2023-03-01	2023-03-05
0001	01	2023-03-15	2023-03-25
0002	02	2023-03-15	2023-03-25
0003	10	2023-03-15	2023-03-25
0004	02	2023-03-15	2023-03-25

Tab. B: Tabulka transakcí

Produkt	Prodejna	Datum transakce
0001	01	2023-03-02
0001	01	2023-03-09
0002	02	2023-03-15
0003	02	2023-03-15
0004	10	2023-03-15
0004	11	2023-03-30

Tab. C: Tabulka souhlasných dvojic promoakce-produkt

Produkt	Prodejna	Začátek promoakce	Konec promoakce	Identifikátor dvojice	Počet promoakcí
0001	01	2023-03-01	2023-03-05	1	2
0001	01	2023-03-15	2023-03-25	1	2
0002	02	2023-03-15	2023-03-25	2	1

Tab. D: Tabulka souhlasných trojic promoakce-produkt-datum

Produkt	Prodejna	Datum transakce	Začátek promoakce	Konec promoakce	Identifikátor trojice	Počet promoakcí
0001	01	2023-03-02	2023-03-01	2023-03-05	1A	2
0001	01	2023-03-02	2023-03-15	2023-03-25	1A	2
0001	01	2023-03-09	2023-03-01	2023-03-05	1B	2
0001	01	2023-03-09	2023-03-15	2023-03-25	1B	2
0002	02	2023-03-15	2023-03-15	2023-03-25	2A	1
0003	10	2023-03-15	Nan	Nan	Nan	0
0004	10	2023-03-15	Nan	Nan	Nan	0
0004	11	2023-03-30	Nan	Nan	Nan	0

Tabulka 5.2: Tabulka transakcí a promoakcí s přidaným příznakem typ promoakce.

Produkt	Prodejna	Datum transakce	Začátek promoakce	Konec promoakce	Typ promoakce
0001	01	2023-03-02	2023-03-01	2023-03-05	promo
0001	01	2023-03-02	2023-03-15	2023-03-25	no promo
0001	01	2023-03-09	2023-03-01	2023-03-05	after promo
0001	01	2023-03-09	2023-03-15	2023-03-25	no promo
0002	02	2023-03-15	2023-03-15	2023-03-25	promo
0003	10	2023-03-15	NaN	NaN	no promo
0004	10	2023-03-15	NaN	NaN	no promo
0004	11	2023-03-30	NaN	NaN	no promo

taFramu, které se mohou zahodit. Algoritmus je znázorněný na obr. ?? *TBD: obrázek UML*. Postupně se prochází každý řádek DataFramu. v pomocné proměnné se zaznamenává aktuální identifikátor určující jednoznačnou trojici produkt-prodejna-datum. Nejdříve se do pomocného seznamu nahrají všechny indexy řádků, které mají aktuální identifikátor. Potom se iteruje přes všechny tyto vybrané řádky. Pokud je typ promoakce iterovaného řádku typu **promo**, běh se zastaví a tento řádek se vybere ze skupiny záznamů, uloží se a pokračuje se na další skupinu. Pokud typ promoakce nebyl **promo**, ale **after promo**, tak se vybere tato promoakce, následné kroky jsou analogické předchozímu případu. Pokud nenastala ani jedna z možností zbývá situace, kdy typ promoakce je **no promo**. Až jsou takto prohledané všechny záznamy, na základě seznamu vybraných řádkových indexů se vytvoří seznam indexů ke smazání jako rozdíl všech indexů v DataFramu a indexů s vybranými promoakcemi.

Funkce `match_promo_to_sales` sdružuje dříve popsané funkce, které zpracovávají promoakce. Vstupními parametry funkce jsou DataFramy transakcí a promoakcí a názvy sloupců. Názvy sloupců mají předdefinovanou hodnotu, kterou lze změnit. v dalším volitelném parametru je možné specifikovat formát datumu. Všechny sloupcy obsahující datumy se převedou na typ `datetime`. Poté se zavolá funkce `map_all_promotions`, která spojí transakce s promoakcemi. Může vzniknout DataFrame, který má více řádků než původní. Výsledný DataFrame se předá funkci `label_date_with_promo`, kde se označí u napojených promoakcích typ promoakce. Dále se pomocí funkce `find_duplicated_records` vyberou všechny řádky, které obsahují redundantní záznamy. Tyto řádky se odstraní z DataFramu s namapovanými promoakcemi. Ke všem řádkům, ke kterým neexistuje promoakce v číselníku promoakcí, je přiřazen příznak **no promo**. Na závěr funkce zobrazí souhrn o velikostech dílčích DataFramů, aby měl uživatel informaci o počtech duplicitních záznamů. Během výpočtu jsou procesy iterování sledovány pomocí knihovny `tqdm`.

## Funkce pro korelační analýzu

Funkce `aggregate_sum` je pomocná funkce použitá v kategorizaci produktů. Funkce zgreguje vstupní DataFrame podle uvedených sloupců a seče hodnoty ve všech numerických sloupcích. Ve výsledném DataFramu resetuje označení řádků a vrátí ho.

Hodnoty korelačních koeficientů se počítají ve funkci `correlation`. Funkci je předán DataFrame a sloupce, kterých se korelace týká. Tato analýza je zaměřena na korelaci hodnoty shrinku s dalšími ukzateli, proto je jedním vstupem název sloupce se shrinky a dalším vstupem je seznam sloupců ostatních ukazatelů. Obecně se nemusí jednat o sloupec shrinků, základní myšlenkou ale je, že korelace je počítána pro každý sloupec ze seznamu sloupců s právě tímto jedním shrink sloupcem. Jedním z volitelných parametrů funkce je určení metody pro získání korelačního koeficientu. Implementovány jsou dvě metody Pearsonův korelační koeficient a Spearmanův korelační koeficient. Pro výpočet jsou využité metody z knihovny *scipy*. Těmto metodám lze předat argument, zda se má uvažovat jednostranná nebo oboustranná alternativní hypotéza. Defaultní metodou je Pearsonův korelační koeficient a oboustranná alternativní hypotéza [36].

Před spuštěním výpočtů korelací jsou sloupce testované pro předpoklady IID. Pro testování, zda dva zkoumané sloupce patří do stejného rozdělení byl použitý Kolmogorov-Smirnovův test implementovaný v knihovně *scipy*. Pro nezávislost Ljung-Boxova metoda implementovaná v knihovně *statsmodels*.

Pro každý vypočtený koeficient je spočtena i  $p$ -hodnota, díky které lze hodnotu koeficientu označit za statisticky významnou, nebo ne. Pro určení významnosti byla implementována pomocná funkce `significance`. Ta vrací `True`, resp. `False` pro statisticky významné, resp. nevýznamné výsledky, tedy pokud je  $p$ -hodnota menší, resp. větší než  $\alpha$ . Předpokládaná hladina významnosti  $\alpha$  je 5 %. Výši hladiny lze změnit v parametru funkce pro výpočet korelace, odtud se předá funkci pro určení významnosti. Vypočtené koeficienty a booleovský příznak o jejich významnosti se ukládají do dvou seznamů, které funkce vrací. Oba seznamy mají takový počet hodnot, jaká je délka vstupního seznamu sloupců.

V parametru `days` funkce `correlation` lze specifikovat, zda se má korelace spočítat pouze mezi sloupcem shrinků se všemi sloupci ze seznamu sloupců anebo navíc se všemi sloupci ze seznamy, kde jsou ale hodnoty v tomto sloupci posunuté o parametr `days`. Pokud například `days=1`, pak k hodnotě shrinku zaznamenané v jistý den nebude náležet hodnota prodejů v téže dni, ale hodnota ze dne předchozího. Tato volba byla přidána na základě hypotézy, že shrink se může projevit se zpožděním. Pokud jsou data takto posunutá, je třeba nahradit data na začátku sledovaného období.

Funkce `product_sales_correlation` je zastřešující funkcí pro korelační analýzu na datových vstupech. Vstupními daty jsou DataFrame se záznamy shrinků a se záznamy prodejů včetně informace o promoakcích. K tomu je třeba definovat názvy sloupců potřebných pro analýzu. Jedná se o sloupec s hodnotami shrinků, hodnotou prodejů, ID produktů, ID prodejen a daty transakcí. Názvy sloupců mají de-

faultní hodnotu, kterou je samozřejmě možné změnit podle zkoumaných dat. Dále má funkce volitelný parametr `after_promo`, jehož defaultní hodnota je `False`, který zohledňuje, zda se pro analýzu s promočními prodeji použijí jen prodeje uskutečněné přímo během promoakce nebo i prodeje, které nastaly týden po promoakci. Další parametry jsou volitelné parametry, které se předávají funkcím, které jsou volány v rámci zastřešující funkce (metoda, alternativní hypotéza, hladina významnosti, počet dní posunu).

Funkce vrací tři proměnné. První je `DataFrame`, který obsahuje seznam produktů a ke každému z nich napočítané korelační koeficienty hodnoty shrinku s ukazateli a statistickou významností tohoto koeficientu. Dále je vrácen seznam produktů, které neměly žádný promoční prodej ve sledovaném období a případně i produktů, které neměly žádný prodej.

Funkce nejprve vytiskne hlášku, která metoda pro výpočet korelace se použije. Poté se inicializují názvy sloupců pro ukládání korelací a příznaku o statistické významnosti. Počet sloupců se liší v závislosti na tom, zda se v analýze zkoumá i varianta se zpožděním shrinku oproti prodejům. Sloupce jsou seřazeny tak, aby sloupce týkající se korelace s jedním ukazatelem byly vedle sebe v následujícím pořadí: korelační koeficient, statistická významnost, korelační koeficient se zpožděním, statistická významnost pro koeficient se zpožděním. Takto budou hodnoty uloženy ve výsledném `DataFramu`. Pro všechny ukazatele se čtverice (v případě zpoždění) nebo dvojice (bez zpoždění), opakuje. Dále se inicializuje prázdný `DataFrame` pro ukládání výsledků s názvem sloupce pro ID produktu spolu s nově vytvořenými názvy.

Dále je třeba ze vstupního `DataFramu` prodejů vybrat pouze záznamy produktů, které se prodaly během promoakce. Pokud je parametr `after_promo` je `True`, pak se kromě záznamů produktů v promoakci vyberou i ty, kde produkty byly prodány v rámci týdne po promoakci. Dále se inicializují prázdné seznamy pro uchování produktů, které nemají žádné prodeje, resp. promoční prodeje.

Následně probíhá iterace přes všechny unikátní produkty, pro které byl zaznamenaný shrink. Počet zkoumaných produktů se vytiskne. Na začátku každé iterace je třeba z `DataFramů` shrinků vybrat pouze záznamy s daným produktem. `DataFrame` se potom agreguje podle sloupců datum transakce a ID prodejny. Stejný postup se aplikuje pro `DataFrame` s prodeji. Navíc se obdobný postup aplikuje i na `DataFramy` s promočními záznamy a se všemi prodeji s tím rozdílem, že se vyhledají záznamy všech produktů kromě iterovaného produktu. Výdledné `DataFramy` se potom sloučí do jednoho podle sloupců ID prodejny a datumu. Jelikož může nastat situace, že ne všechny hodnoty jsou definované na každém řádku, nahradí se nedefinované hodnoty nulou.

Na složený `DataFrame` se použije funkce `correlation`, které se předají příslušné parametry. Výsledky se pak vloží jako nový řádek do `DataFramu` pro ukládání výsledků. Pokud nebylo možné spočítat korelace, z důvodu, že rozptyl hodnot byl nulový - nastane pokud produkt nemá žádné prodeje - nahradíme nedefinovanou korelací nulou, která indikuje, že mezi veličinami není závislost.

Výsledný `DataFrame` s korelací je vstupem do funkce `categorization`. Dalšími vstupy je název sloupce, který obsahuje korelačními koeficienty shrinků produktu

s jeho vlastními tržbami a sloupce s koeficienty shrinků produktu s prodeji ostatních produktů. Ve funkci se vytvoří nový DataFrame pro uložení výsledků kategorizace. Jeho indexem jsou ID produktů. Samotná kategorizace se získá spuštěním funkce `categorize_products`, která vrací seznam kategorií pro každý řádek vstupního DataFramu. Funkce `categorize_products` roztrídí produkty do pěti kategorií: `itself`, `other`, `sellout`, `independent`, `none`. V textu se o těchto kategoriích mluví jako o kategoriích P, O, V, N, X. Postup roztrídění produktů do těchto kategorií je popsáný v sekci ??.

Poté, co má každý produkt přiřazenou kategorii se ve funkci `categorization` označí každý produkt s kategorií, zda je výsledek statisticky reprezentativní, nebo ne. Rozhodující hodnota je získána pomocí funkce `unsignificant_rows`. Která vrací logickou hodnotu výroku:

$$\text{Významnost}(r_i) \vee ((\text{Koeficient(korelace produktu se sebou)} \leq 0)$$

$$\wedge \text{Významnost(korelace produktu s ostatními)})$$

Funkce pak vrátí DataFrame s takto označenými a kategorizovanými produkty. `categorization`, `categorize_products`, `unsignificant_rows`,

## Pomocné funkce

Funkce `create_folder` vytvoří složku se zadáným jménem v aktuální cestě, pouze pokud již taková složka neexistuje. Další pomocná funkce je `format_date`, která využívá funkci z knihovny *pandas to\_datetime*. Pro základní vizualizaci jsem vytvořila funkci, která pomocí knihovny *matplotlib* vytváří bodový graf dvou proměnných. Graf je buď uložen nebo zobrazený při spuštění funkce. Funkci lze předat DataFrame a názvy dvou sloupců, které reprezentují vstupy pro osy *x* a *y* grafu. Další vstupy jsou názvy os a grafu, případně název souboru, pokud uživatel graf uložit.

## Testování

Pro testování funkcí jsem použila knihovnu *pytest* jazyka Python. Testy lze spustit příkazem `python -m pytest tests` v kořenovém adresáři projektu.

## 5.3 Výsledky

Analýza se týká pouze dat jednoho měsíce a kategorií produktů první úrovně *Velmi čerstvé*, zastoupena 48 % a *Čerstvé*, zastoupena 52 % ve vybraných datech. Data obsahují pouze jeden typ shrinku – prošlé a zkažené zboží, který zaujímá téměř 65 % shrinků pro dané kategorie. Zastoupení typů shrinků, které zabírají v datech více jak dvě procenta se nachází v tabulce 5.3. Zaměřila jsem se na kategorie ze čtvrté úrovně, a to prvních deset kategorií s nejvyšší hodnotou shrinků (tj. s nejvyšší zaznamenanou ztrátou). V práci jsou popsány výsledky pouze tří kategorií – Masné výrobky – pultový prodej, Slané pečivo a Plodová zelenina. Na třetím místě byla kategorie Sladké pečivo, ale kvůli podobnosti s druhou kategorií, jsem zvolila následující

kategorii v pořadí vzhledem k hodnotě shrinku. V tabulce 5.4 jsou procentuální hodnoty zastoupení čtyř kategorií mezi ostatními kategoriemi úrovně 4 podle velikosti shrinku.

Tabulka 5.3: Zastoupení vybraných shrinků ve zkoumaných datech  
(kategorie Čerstvé a Velmi čerstvé).

Typ shrinku	Zastoupení v kategoriích [%]
Prošlé a zkažené zboží	64,97
Potravinová banka	23,72
Poškození	6,26
Zvířecí útulky	2,69
Kompostéry	2,36

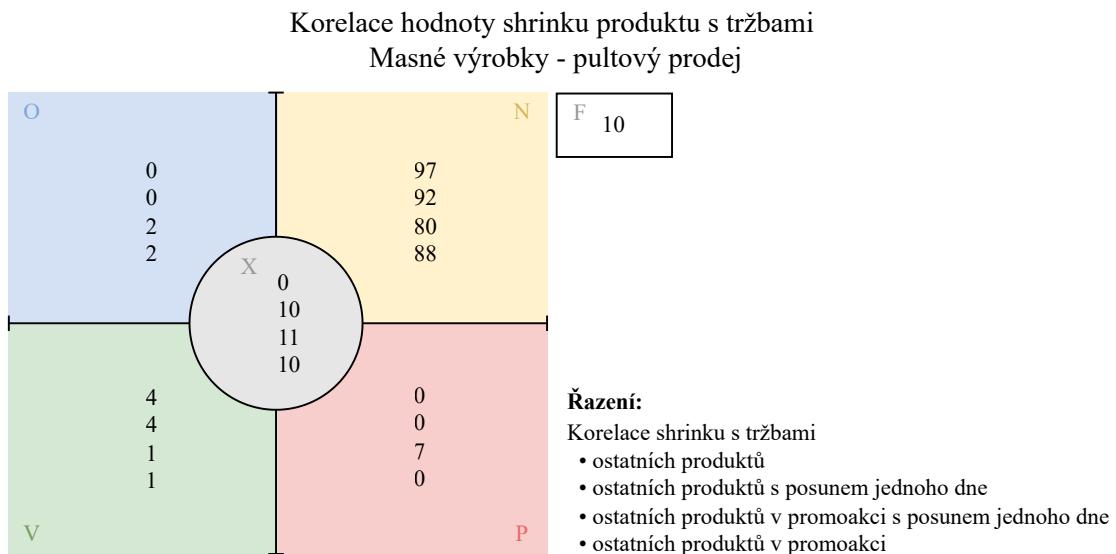
Tabulka 5.4: Tabulka čtyř kategorií ze čtvrté úrovně produktové hierarchie podle zastoupení zaznamenané hodnoty shrinku na všech evidovaných shrincích.

Kategorie	Zastoupení [%]
Masné výrobky – pultový prodej	26,27
Slané pečivo	12,12
Sladké pečivo	6,82
Plodová zelenina	5,65

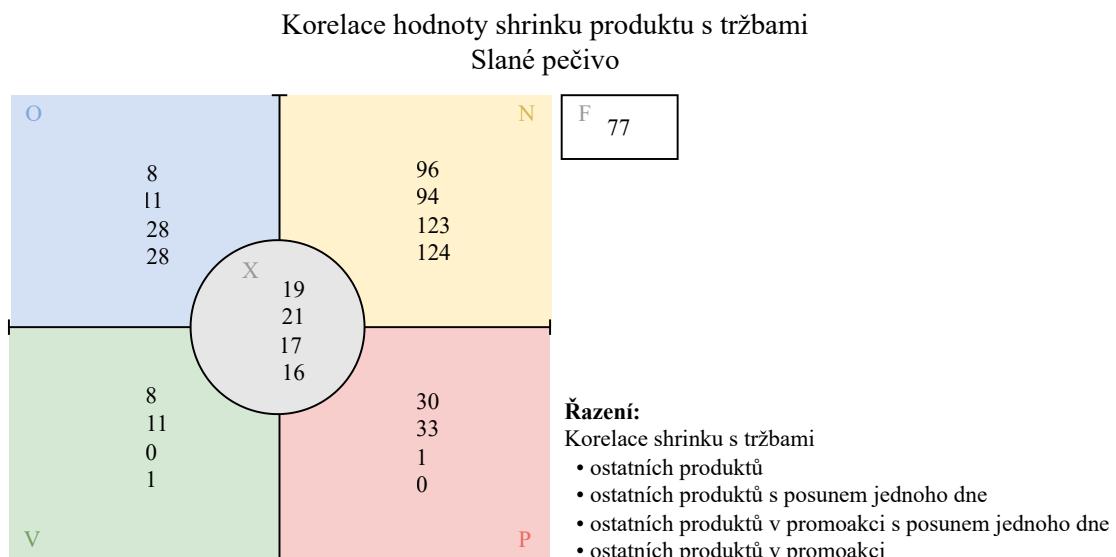
Měřila jsem postupně korelaci velikosti shrinku s různými ukazateli pro celkové tržby ostatních produktů. Pro určení míry korelace jsem zvolila Spearmanův korelační koeficient, jelikož data nesplňují předpoklady, které jsou nutné pro použití Pearsonova korelačního koeficientu - data nejsou nezávislá a stejně rozdělená. Data vybrané společnosti, také nesplňují podmínu normality, to může být dáno tím, že data pochází z reálného světa a zaznamenávají jev, který závisí na mnoha, těžce predikovatelných faktorech. Nejprve jsem zvolila 5% hladinu významnosti pro testování statistické významnosti koeficientů korelace  $r_P$  a  $r_O$ . Výsledky ovšem ukázaly, že alespoň hodnoty třetiny produktů ve zkoumaných kategoriích byly neprůkazné. Rozhodla jsem se tedy zvýšit hladinu významnosti na 10 %. Zvýšení hladiny významnosti zvýšilo pravděpodobnost vzniku chyby druhého druhu, nicméně případné zařazení produktu do špatné kategorie nemá z businessového hlediska fatální následky.

Na obrázcích 5.3 až 5.5 jsou porovnání výsledků kategorizace pro zmíněné tři kategorie. Pokaždé bylo spuštěno šest výpočtů. Korelace byla měřena mezi shrinky a tržbami ostatních produktů, ostatních produktů, kde prodeje byly posunuté o jeden den, dále mezi shrinky a tržbami produktů v promoakci a produktů v promoakci s posunem prodejů. Varianty s promoakcemi dále byly jak pro shrinky produktů během promoakce, tak pro během i po promoakci. Na obrázcích jsou zobrazené výsledky pro variantu během i po promoakci, protože zachytily stejně nebo více případů než varianta záznamů pouze během promoakce.

Z uvedených počtů produktů u jednotlivých kategoriích pro různé ukazatele, je patrné, že výsledky se příliš neliší. Pokud bychom se ale zaměřovali na celkové prodeje, nikoli promoční, tak získáváme větší množství produktů, u nichž nebylo možné vysvětlit shrink pomocí korelace. Avšak hypotézy pro rozřazení produktů uvažují právě promoční prodeje nikoli celkové prodeje. Další popis se věnuje výsledkům korelace mezi shranky a promočními a popromočními prodeji, které měly stejný den záznamu jako shrank, na obrázcích 5.3 až 5.5 se jedná o poslední řádek s výsledky.



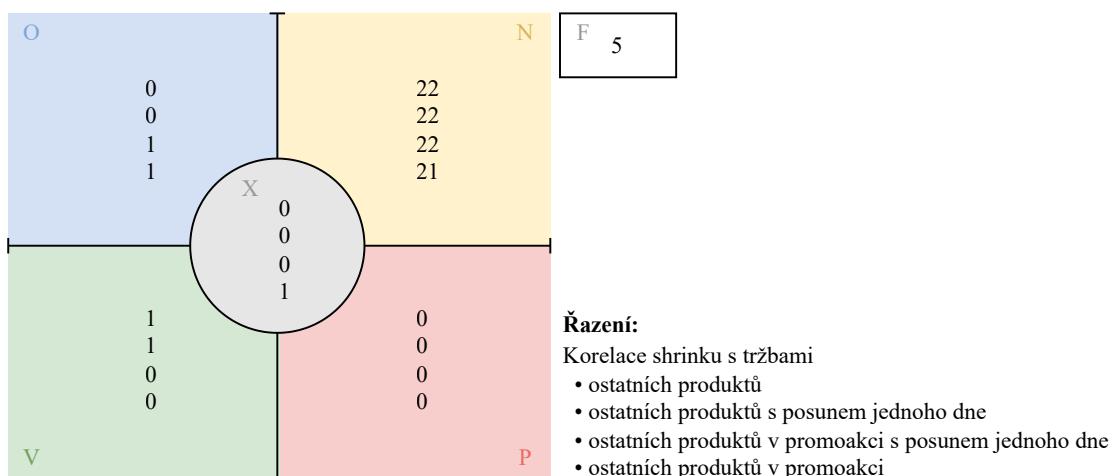
Obrázek 5.3: Počet produktů z kategorie Masné výrobky – pultový prodej roztríděné pomocí korelační analýzy v závislosti na různých ukazatelích.



Obrázek 5.4: Počet produktů z kategorie Slané pečivo roztríděné pomocí korelační analýzy v závislosti na různých ukazatelích.

## Masné výrobky – pultový prodej

### Korelace hodnoty shrinku produktu s tržbami Plodová zelenina



Obrázek 5.5: Počet produktů z kategorie Plodová zelenina roztríděný pomocí korelační analýzy v závislosti na různých ukazatelích.

Shrink byl zaznamenaný u 111 produktů v této kategorii úrovně 4. 88 produktů bylo klasifikováno jako kategorie N, deset jako kategorie X, dva jako kategorie O, jeden jako V. U zbylých deseti produktů nebyl koeficient korelace statisticky významný, a proto nejde u těchto produktů vyslovit hypotézu pro jejich zařazení. Korelace mezi hodnotou shrinku a promočními tržbami je na obr. 5.6.

Produkty, které patří do kategorie O: Velikonoční klobása a Velikonoční šunka - jedná se zcela jistě o sezónní výrobky. Produkt, který byl označen jako výprodejový jsou Párky (Kuřecí striptýzky). Šest produktů z kategorie nemělo během sledovaného období žádné evidované prodeje, všechny byly klasifikovány jako kategorie X, tedy hodnota koeficientu korelace neznamenala závislost.

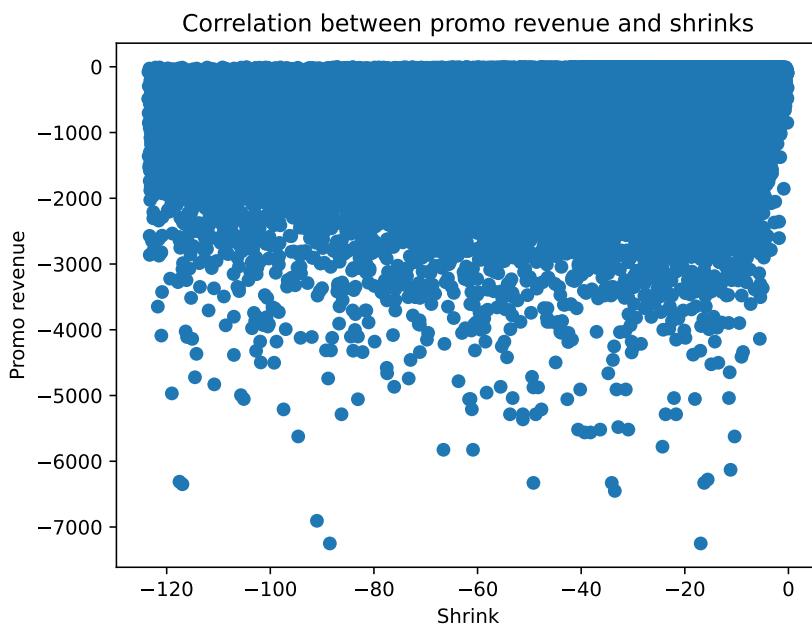
Dále jsem zkoumala podkategorie Masných výrobků. Porovnávala jsem prodeje v rámci kategorií na šesté úrovni produktové hierarchie. V podkategroii Salámy s krátkou dobou spotřeby se kategorizace potvrdila. Pro kategorie, do níž patří sezónní výrobky - Netučné masné výrobky, nově z této podkategorie byl jako kategorie O označen i produkt Kladenská pečeně.

#### Slané pečivo

Shrink byl zaznamenaný u 246 produktů. 124 produktů bylo klasifikováno jako kategorie N, 16 jako kategorie X, 28 jako kategorie O, jeden jako P. Pro 77 produktů nebyl koeficient korelace statisticky významný. Jako produkt, který si způsobuje shinky sám, byl označený obyčejný rohlík. Rohlík se tedy vyhazuje více čím vyšší jsou jeho vlastní tržby. Celkově patří tento produkt mezi ty s největšími shinky.

Produkty, které byly zařazeny do kategorie X, tj. takové, u kterých nebyl koeficient korelace dostatečně velký, byly produkty, které neměly během sledovaného období žádný prodej (promoční, či nepromoční).

#### Plodová zelenina



Obrázek 5.6: Závislost mezi tržbami produktu a tržbami ostatních produktů v kategorii během promoakce (Masné výrobky – pultový prodej).

Shrink byl zaznamenaný u 28 produktů v této kategorii. 21 produktů bylo klasifikováno jako kategorie N, jeden produkt jako X a jeden jako O. U ostatních pěti produktů nebyl koeficient dostatečně významný. Produkt, který v této kategorii neměl žádné prodeje byl pouze Lilek Bio, pro který koeficient korelace byl označen jako nevýznamný. Produkt, z kategorie O, byla Cherry rajčata. Zatímco produkt z kategorie X, byl Paprika barevná Mix.

# Kapitola 6

## Analýza pomocí metody 4ftMiner

Pomocí metody *4ftMiner*, která je jednou z metod procedury GUHA jsem provedla analýzu shrinků produktu. Metoda umožňuje odhalit zajímavé vzory chování, které jsou obsažené v datech ale lze je vztáhnout na celkovou zkoumanou množinu. Implementace metody se nachází v knihovně *Cleverminer* pro jazyk Python. Princip metod, které se používají v knihovně, a důležité pojmy týkající se GUHA procedur jsou popsány v sekci 2.3. Vstupními daty pro metodu GUHA byla tabulka zaznamenaných shrinků rozšířená o číselníky a sloupce s podíly zastoupení shrinků na tržbách. Tento dataset je popsán v sekci 4.2. Pracovala jsem pouze se vzorem dat jednoho měsíce a s kategoriemi produktů *Velmi čerstvé*, zastoupena 48 % a *Čerstvé*, zastoupena 52 % ve vybraných datech a se shranky typu prošlé a zkažené zboží.

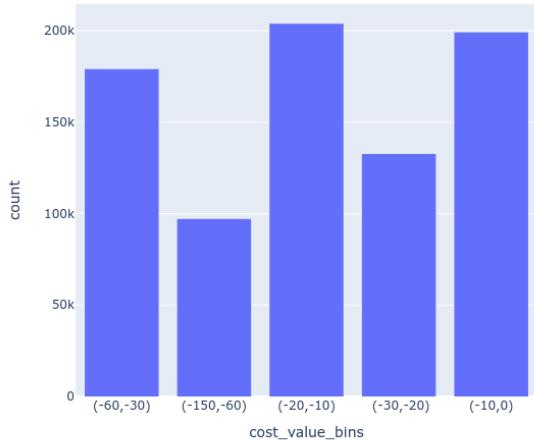
První část této kapitoly se věnuje hypotézám, které mohou platit o shrincích. Hypotéza je přeformulována jako asociační pravidlo, které je následně ověřeno metodou GUHA. Druhá část se věnuje zkoumání konkrétních produktů, u kterých pomocí korelační analýzy, popsané v kapitole 5, nebyla zjištěna možná příčina shrinku.

Metoda 4ftMiner pracuje pouze s kategorickými hodnotami, proto bylo nutné kategorizovat sloupce s hodnotou shranku, s množstvím shrinkovaných produktů a s jednotlivými podíly. Na obrázcích 6.1 až 6.5 jsou zobrazené četnosti záznamů v kategoriích.

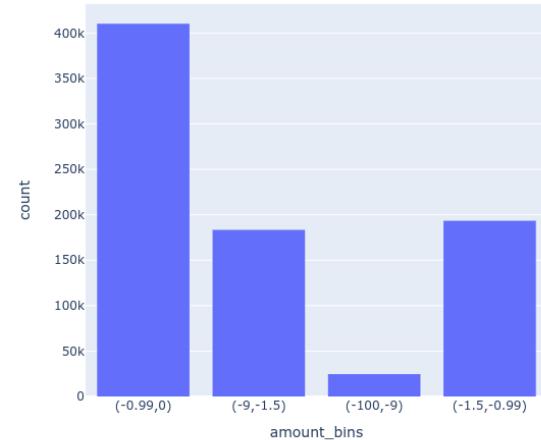
Pro první hypotézu je uvedeno volání funkce v jazyce Python včetně předaných parametrů. Rovněž je v tabulce uvedený celý výstup v obdobném formátu jako je zobrazen na konzoli po ukončení běhu funkce. Dále už kódy, ani přesné výstupy uvedené nebudou, ale bude uveden pouze popis vstupů a komentář k výstupům.

### 6.1 Hypotézy

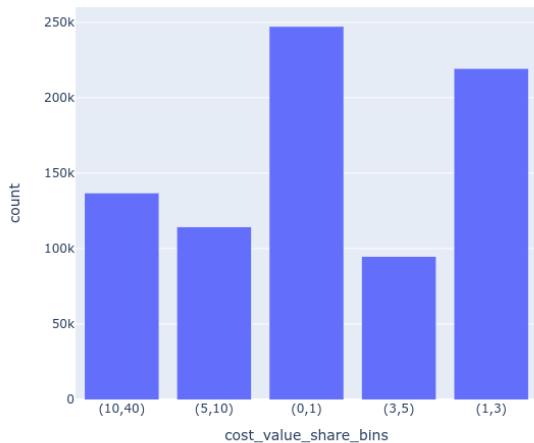
Před spuštěním metody bylo vždy třeba vznést hypotézu, která by mohla být pravdivá pro data týkající se shrinků. Tuto hypotézu pak přeformulovat do podoby asociačního pravidla, jehož pravdivost na vstupních datech ověřuje metoda *4ftMiner*. Tato metoda se předá jako parametr funkci `cleverminer`. Pravidlo se funkci zadává pomocí parametrů jako jednotlivé cedenty - antecedenty, sukcidenty,



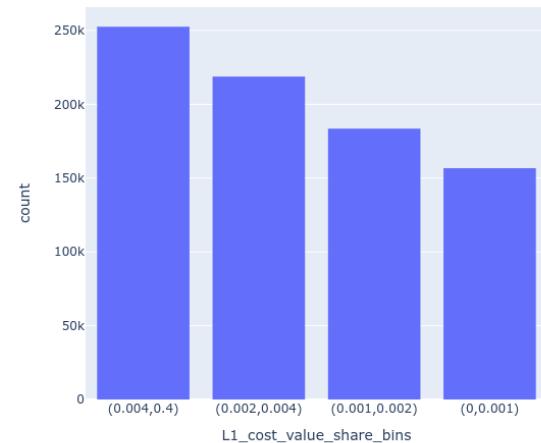
Obrázek 6.1: Histogram pro hodnoty velikosti shrinku v peněžních jednotkách.



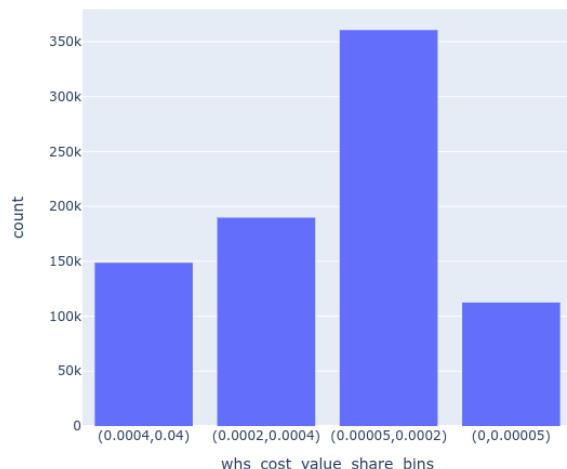
Obrázek 6.2: Histogram pro hodnoty objemu shrinku v kusech.



Obrázek 6.3: Histogram podílu shrinku na tržbách shrinkovaného produktu.



Obrázek 6.4: Histogram podílu shrinku a tržeb v kategorii úrovně 1.



Obrázek 6.5: Histogram podílu shrinku na celkových tržbách prodejny.

případně podmínky. Více o principu metody je ovedeno v teoretické části práce.

### Hypotéza č. 1: Objem prošlého zboží je závislý na typu promoakce a dni v týdnu

Ve zkoumaných datech je zboží bez promoakce zastoupeno 58,2 %, zboží týden po evidované promoakci 23,2 a zboží v promoakci 18,6 procentem.

Asociační pravidlo má tvar:

$$\varphi_{\text{Den v týdnu}} \wedge \varphi_{\text{Typ promoakce}} \Rightarrow \psi_{\text{Množství}} \quad (6.1)$$

V ukázce kódu 6.1 jsou uvedené parametry pro spuštění metody. Konfidence byla zvolena 80 %. Výsledky běhu jsou uvedené v tabulce 6.1. Z této tabulky lze vyčíst, že pro vybrané dny v týdnu – pondělí, úterý, středa, čtvrtok a neděle, tj. nikoli pro pátek a sobotu – a pro produkty, které byly v den záznamu týden po promoakci platí, že 80 % těchto záznamů bylo v množství do jednoho kusu. To znamená, že se jedná o produkty, které jsou vážené. Jejich přepočet na kusovou jednotku tedy může být menší než jeden celý kus. Podle dalšího zkoumání dat jsem zjistila, že se jedná především o kategorii *Masné výrobky* ze třetí úrovně hierarchie.

```

01 cleverminer(df = data,
02               proc = "4ftMiner",
03               quantifiers = {"conf":0.8, "Base":1000},
04               ante = {
05                 "attributes":
06                   [
07                     {
08                       "name": "weekday",
09                       "type": "seq",
10                       "minlen": 1, " maxlen": 3
11                     },
12                     {
13                       "name": "promo",
14                       "type": "sec",
15                       "minlen": 1, " maxlen": 1
16                     }
17                   ],
18                   "minlen": 2, " maxlen": 2, "type": "con"
19               },
20               succ = {
21                 "attributes":
22                   [
23                     {
24                       "name": "amount_bins",
25                       "type": "subset",
26                       "minlen": 1, " maxlen": 1
27                     }
28                   ],
29                   "minlen": 1, " maxlen": 1, "type": "con"
30               }
31           )
32

```

Kód 6.1: Hypotéza č. 1, funkce `cleverminer`.

### Hypotéza č. 2: Kategorie shrinkovaného zboží je závislá na typu promoakce a dni v týdnu

Tabulka 6.1: Výstup funkce `cleverminer` pro hypotézu 1.

Základ (a)	Konfidence	AAD	AP [%]
19765	0.821	+0.623	$\text{weekday}(0) \wedge \text{promo}(\text{after\_promo}) \Rightarrow \text{amount\_bins}((-0.99,0))$
39271	0.820	+0.622	$\text{weekday}(0, 1) \wedge \text{promo}(\text{after\_promo}) \Rightarrow \text{amount\_bins}((-0.99,0))$
63920	0.815	+0.613	$\text{weekday}(0, 1, 2) \wedge \text{promo}(\text{after\_promo}) \Rightarrow \text{amount\_bins}((-0.99,0))$
19506	0.820	+0.621	$\text{weekday}(1) \wedge \text{promo}(\text{after\_promo}) \Rightarrow \text{amount\_bins}((-0.99,0))$
44155	0.813	+0.608	$\text{weekday}(1, 2) \wedge \text{promo}(\text{after\_promo}) \Rightarrow \text{amount\_bins}((-0.99,0))$
68666	0.810	+0.603	$\text{weekday}(1, 2, 3) \wedge \text{promo}(\text{after\_promo}) \Rightarrow \text{amount\_bins}((-0.99,0))$
24649	0.808	+0.598	$\text{weekday}(2) \wedge \text{promo}(\text{after\_promo}) \Rightarrow \text{amount\_bins}((-0.99,0))$
49160	0.806	+0.595	$\text{weekday}(2, 3) \wedge \text{promo}(\text{after\_promo}) \Rightarrow \text{amount\_bins}((-0.99,0))$
24511	0.805	+0.593	$\text{weekday}(3) \wedge \text{promo}(\text{after\_promo}) \Rightarrow \text{amount\_bins}((-0.99,0))$
18864	0.813	+0.608	$\text{weekday}(6) \wedge \text{promo}(\text{after\_promo}) \Rightarrow \text{amount\_bins}((-0.99,0))$

Asociační pravidlo má tvar:

$$\varphi_{\text{Den v týdnu}} \wedge \varphi_{\text{Typ promoakce}} \Rightarrow \psi_{\text{Hierarchie3}} \vee \psi_{\text{Hierarchie4}}, \quad (6.2)$$

kde označením Hierarchie3 jsou myšleny kategorie na třetí úrovni produktové hierarchie, obdobně pro pojem Hierarchie4.

Parametry předané funkci jsou podobné jako u předchozí hypotézy. Ze záznamů, které se byly provedeny v pondělí, úterý nebo neděli a týkaly se produktů, které byly v rozmezí jednoho týdne po promoakci, bylo více než 75 % z kategorie Masné výrobky – pultový prodej ze čtvrté úrovně produktové hierarchie. Pokud je vynechána ze vstupních dat tato kategorie, pak maximální konfidence 31 % byla dosažena pro kategorii Slaného pečivo v záznamech, které byly provedeny v sobotu a týkaly se produktů zcela mimo promoakci. Jiné významné závislosti podle dat nebyly nalezeny.

### Hypotéza č. 3: Na některých lokalitách vyhazují často stejné produkty

Asociační pravidlo má tvar:

$$\varphi_{\text{Typ prodejny}} \wedge \varphi_{\text{Okres}} \Rightarrow \psi_{\text{Množství}} \quad (6.3)$$

60 % záznamů týkajících se okresů Jindřichův Hradec, Ústí nad Labem, Písek nebo Strakonice tvoří shrinky z kategorie *Masné výrobky*. Pro záznamy z okresu Kladno, které jsou zároveň evidovány velkými prodejními kategoriemi Masné výrobky byla zastoupena až téměř 70 %. Necelými 70 % je tato kategorie zastoupená také v záznamech v malých prodejnách v okrese Praha-východ.

Pokud úplně vynecháme kategorie Masné výrobky ze vstupních dat, pak se nejčastěji ve výsledcích objevovala kategorie *Pečivo*. Pro záznamy z velkých prodejen v okrese Pardubice nebo Plzeň-město Pečivo zaujímalo přes 60 % těchto záznamů. Nad 50 % záznamů pro okresy Bruntál, Olomouc, Příbram nebo Uherské Hradiště. 50 % záznamů náleželo kategorii Pečivo také v záznamech z malých prodejen v okrese Klatovy, Náchod nebo Přerov.

Po vynechání kategorie Pečivo již dostáváme maximální konfidenci 33 %, a to pro kategorii Zelenina ve zbylých záznamech z okresu Ostrava-město, Kroměříž, Hradec Králové nebo Karviná.

**Hypotéza č. 4: Některé produkty se vyhazují častěji než jiné, ale v malém množství.**

Asociační pravidlo pro úroveň produktové hierarchie 3 má následující tvar. Pro úroveň 4 je tvar AP analogický.

$$\varphi_{\text{Hierarchie}3} \Rightarrow \psi_{\text{Množství}} \quad (6.4)$$

Kategorie Masné výrobky byla zaznamenána téměř 300 tisíckrát, a v 94 procentech se jednalo o množství odpovídající do jednoho balení. Podkategorie Masné výrobky – pultový prodej má 99 % svých záznamů do jednoho kusu. Pokud se vyhazují čerstvé ryby, tak v 94 % svých záznamů je to množství do jednoho kusu. Kategorie Drobné občerstvení se vyhazuje v 89 % po jednom kusu (obvykle se jedná o sendviče a bagety) Kategorie Vejce se vyhazuje v 82 % po jednom kusu balení Kategorie Pečivo se vyhazuje v 56 % v počtu kusů do 10 kusů v až 94 tis. záznamech. Kategorie Jádroviny<sup>1</sup> se vyhazuje 74 % případech svých záznamů (14 000 záznamů) v množství do jednoho kusu. I zde se jedná přepočet váženého množství na kusy.

**Hypotéza č. 5: Některé vyhazované kategorie produktů jsou výrazně nákladnější.**

Asociační pravidlo má tvar:

$$\varphi_{\text{Hierarchie}4} \Rightarrow \psi_{\text{Shrink}} \quad (6.5)$$

Pokud se vyhazují čerstvé ryby, tak v téměř 80 % případech záznamů jsou ztracené náklady jednoho záznamu vyšší, a to v rozmezí 60-150 peněžních jednotek. Pokud se vyhazuje kategorie Červené maso, tak z téměř 60 % je ztráta v rozsahu 60-150 jednotek. Kategorie Chlazený pultový prodej, která obsahuje např. čerstvé chlebíčky, saláty a pochutiny, se v 50 % vyhazuje v hodnotě do 10 peněžních jednotek. Jedná se

---

<sup>1</sup>Jádroviny jsou druh ovoce, patří sem např. jablka a hrušky.

tedy o nižší částky, které jsou ale časté. Záznamů této kategorie bylo evidováno 12,5 tisíc. Cukrářské výrobky byly evidovány v 1835 záznamech. 66 % těchto záznamů mělo hodnotu mezi 10 a 20 peněžními jednotkami.

#### **Hypotéza č. 6: Shrink některých kategorií je v porovnání s tržbami těchto produktů na stejně prodejně velký.**

Asociační pravidlo má tvar:

$$\varphi_{\text{Hierarchie4}} \Rightarrow \psi_{\text{Podíl shrinku na svých tržbách}} \quad (6.6)$$

Nejedná se o porovnání s celkovými tržbami prodejny, ale pouze o týdenní tržbu těch produktů, které měly zaznamenaný v daném týdnu shrink. Kategorie Drobné občerstvení má podíl shrinku na svých tržbách v 84 % ze zaznamenaných případů mezi 10-40 %. Cukrářské výrobky mají podíl shrinku v 74 % zaznamenaných případech také mezi 10-40 %. Banány mají podíl shrinku na tržbách banánů v daném týdnu v 80 % ze svých zaznamenaných případech do 1 %. To znamená, že se jedná o malou část svého prodeje. Více než 30 tis. záznamů se týká kategorie Citrusů a kategorie Jádrovin. Přibližně 65 % těchto záznamů je podíl shrinku do 1 % na tržbách těchto produktů.

Dále pro tuto hypotézu bylo ověřováno podmíněné asociační pravidlo:

$$\varphi_{\text{Hierarchie3}} \Rightarrow \psi_{\text{Podíl shrinku na svých tržbách}} | \chi_{\text{Shrink}} \quad (6.7)$$

Následující tvrzení platí s více než 83% konfidencí. Pokud mezi produkty, kterým byl zaznamenán dražší shrink, tj. 30-60 peněžních jednotek, jsou produkty z kategorie Jogurty, tak podíl shrinku na jejich tržbách je mezi 10-40 %. Totéž tvrzení platí i pro kategorie Drobného občerstvení. Pokud mezi produkty, kterým byl zaznamenán levný shrink, tj. do 10 peněžních jednotek, je ovoce, tak jejich share shrinku na tržbách je do 1 %. To samé platí o pro kategorii Kořenová zelenina.

#### **Hypotéza č. 7: Kategorie má vliv na zastoupení shrinku na celkových tržbách prodejny v dané kategorii úrovně 1.**

S pravděpodobností vyšší než 50 % se toto tvrzení potvrdilo pouze u kategorie Bylinky z úrovně 4, kdy shrink této kategorie tvoří 0.002 % až 0.005 % tržeb na prodejnách v kategorii Velmi čerstvé v první úrovni produktové hierarchie.

#### **Hypotéza č. 8: Den v týdnu nebo čtvrtina měsíce mají vliv na záznamy.**

Asociační pravidlo je následovné:

$$\varphi_{\text{Den v týdnu}} \wedge \varphi_{\text{Čtvrtina měsíce}} \Rightarrow \psi_{\text{Typ prodejny}} \quad (6.8)$$

V případě antecedentu je možné uvažovat minimální délku jeden booleovský atribut, maximální dva. Je tedy možné, že nalezené pravidlo se může týkat pouze jednoho ze dvou booleovských atributů v antecedentu.

Záznamy uskutečněné ve středu, čtvrtok a pátek v poslední čtvrtině měsíce, se ze 67% konfidencí týkají malých prodejen.

### Další hypotézy

Dále byly uvažovány hypotézy:

- Ve větších prodejnách ve velkých městech se vyhazuje více typů produktů.
- Velké prodejny vyhazují širší spektrum produktů než malé prodejny.
- Na některých v některých lokalitách mají často velký shrink.

Pro tyto hypotézy ale nebylo nalezeno žádné dostatečně silné, tj. s konfidencí vyšší než 40 %, asociační pravidlo.

## 6.2 Produkty nepopsané korelační analýzou

Pomocí korelační analýzy korelační analýzy lze produkty z vybrané kategorie rozdělit do pěti skupin podle toho, zda hodnota shrinku produktů koreluje s tržbami jiných produktů. Jedna ze zmíněných skupin je přiřazena produktům, u kterých se nepodařilo touto metodou shrink vysvětlit. Také vzhledem k tomu, že je metoda založena na výpočtu korelace, je nutné provést na vypočtené koeficienty statistické testy významnosti. Pro některé produkty tak nelze vyslovit hypotézu o jejich zařazení do skupiny, neboť obdržený koeficient není statisticky významný. Popis metody a výsledků pro vybrané kategorie je v kapitole 5.

V této části jsem nástroji Cleverminer předala data týkající se pouze produktů, pro které nebyl koeficient korelace statisticky významný, nebo nebyla nalezena žádná souvislost s tržbami ostatních produktů v rámci kategorie.

Antecedent asociačního pravidla obsahuje boolovské atributy:  $\varphi$  Produkt,  $\varphi$  Typ promoakce a  $\varphi$  Prodej. Z těchto atributů mohlo být vybráno jeden až tři atributy pro vytvoření asociační pravidla. Sukcedent byl tvořen všemi možnými sloupcí ve vstupních datech a skládat se mohl z jednoho až čtyř boolovských atributů těchto sloupců.

Výsledky zkoumání produktů, u kterých nebyla pomocí korelační analýzy odhalena závislost, jsou popsány na kategorii čtvrté úrovně Masné výrobky – pultový prodej. Jedná se celkem o dvacet produktů. Všechny produkty měly zaznamenaný shrink do jednoho kusu. Z produktů, které neměly statisticky významný koeficient, sedm z nich bylo evidovaných pouze v okrese hlavní město Praha a jedná se o produkty, které nebyly v promoakci, ale zároveň měli evidované prodeje během sledovaného období. Pro pět z nich dále platí, že s více než 80% konfidencí pochází záznamy z menších prodejen. Pro produkt Klobása ostravská platí, že pokud byl v období po promoakci byl vyhazován na malých prodejnách (97% konfidence), zatímco na velkých prodejných byl vyhazován, když v promoakci nebyl (89% konfidence). O produkту Slanina uzená lze tvrdit z dat, že s 63% konfidencí se vyhazuje na malých prodejnách. Všechny záznamy se týkají nepromočního období produktu. 40% dat bylo zaznamenáno v poslední čtvrtině sledovaného měsíce. Produkt Salám točený

pikantní bylo zjištěno, že byl vyhazován se 73% konfidencí na malých prodejnách, a to jak během probíhající promoakce, tak po ní i v období, kdy v promoakci nebyl.

Co se týče deseti produktů, u kterých nebyla zjištěna závislost na prodejích ostatních produktů, až na jeden produkt, všechny tyto produkty neměly ve sledovaném období promoakci, ale měly záznamy o prodejích v tomto období. Čtyři produkty byly zaznamenány na velkých prodejnách, jeden z nich pouze na prodejnách v Praze. 81% záznamů produktu Párky královské, byly zaznamenány v první čtvrtině v měsíci, kdy nebyly v promoakci. Naopak pro Šunku prosciutto platí, že v 60% záznamů byla vyhazována pouze na konci měsíce a z 94% pouze na malých prodejnách. Pro zbylé produkty nebylo nalezeno žádné pravidlo s vysokou konfidencí z důvodu velmi malého počtu záznamů – méně než pět záznamů.

## Shrnutí

Pomocí metody Cleverminer bylo prozkoumáno jedenáct hypotéz týkajících se dat se záznamy shrinků. Tři hypotézy se pomocí metody 4ftMiner nepodařilo potvrdit, zbylé hypotézy našly, alespoň pro část záznamů oporu v datech. Dále se tato kapitola zabývala hledáním pravdivých tvrzení pro produkty, u kterých nebyla nalezena závislost pomocí korelační analýzy. V tomto případě asociační pravidlo předpokládalo ID těchto produktů a typ promoakce a údaj o existenci prodeje. Sukcedentem pak mohl být jakýkoli jiný sloupec vstupních dat. Výsledky byly diskutovány pro jednu ze zkoumaných kategorií čtvrté úrovně. Pro jiné kategorie by byl postup analogický. Je ale důležité zmínit, že produkty, u kterých nebyl výsledek korelační analýzy statisticky významný, bylo často velmi málo záznamů – v řádu jednotek, maximálně nízkých desítek.

# Kapitola 7

## Vizualizace dat

Očištěná data vybrané společnosti obsahující záznamy shrinků, které byly způsobeny škodami, jsem vizualizovala v nástroji Power BI, který se používá pro business intelligence analýzu. Vytvořila jsem report, který umožňuje pomocí interaktivních grafů analyzovat data. První část této kapitoly se věnuje technickému popisu reportu, zatímco druhá část shrnuje výsledky analýzy plynoucí z reportu.

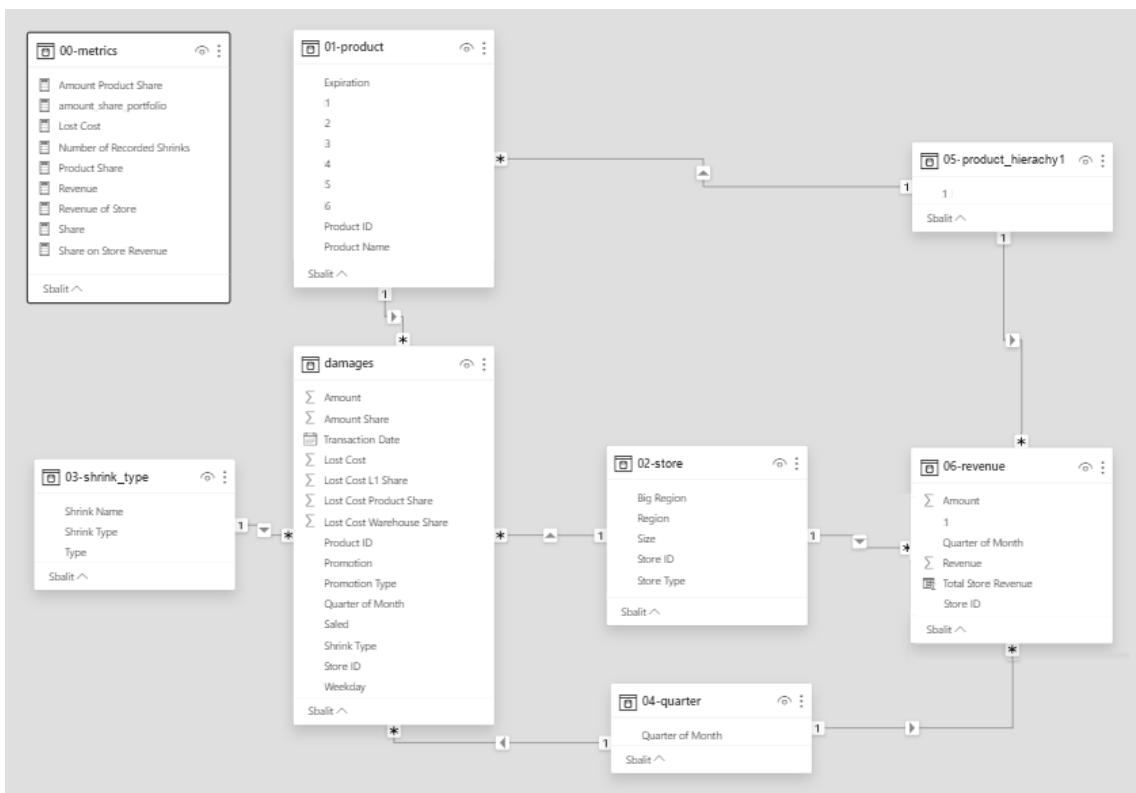
### 7.1 Popis řešení

Report obsahuje pět stránek. První stránka nabízí základní přehled, dashboard, týkající se všech shrinků. Druhá stránka je věnována prodejnám a údajům o lokalitách. Další stránky se již věnují pouze shrinkům zaviněných škodami a kategoriím Velmi čerstvé a Čerstvé z produktové hierarchie úrovně 1. Třetí stránka zobrazuje hodnoty ukazatelů hodnoty shrinku a odvozených podílů. Na čtvrté a páté stránce jsou další přehledy z pohledu konkrétních produktů, kategorií a typů promoakce. Poslední stránka týkající se reportingu je z pohledu vybraného konkrétního produktu.

Do Power BI souboru jsem pomocí integrovaného nástroje PowerQuery nahrála upravená data z databáze vybrané společnosti. Hlavní faktickou tabulkou je tabulka *damages*, která obsahuje všechny zaznamenané shrinky z kategorie shrinků, které byly způsobeny škodami. Druhá faktická tabulka má název *revenue* a obsahuje tržby v pozorovaném měsíci pro všechny prodejny, tržby jsou dále rozdělené podle kategorie z úrovně 1 a do čtvrtin měsíce. Doménové tabulky jsou číselník produktů, číselník shrinků a číselník prodejen, dále také číselníkové tabulky, které spojují faktické tabulky – čtvrtina měsíce a seznam kategorií úrovně 1.

Datový model tabulek, které jsou vstupem do reportu je znázorněný na obrázku 7.1. Mezi jednotlivými tabulkami jsou znázorněny vazby – jejich mohutnost a směr. Tabulka *metrics*, která není navázaná na žádnoujinou tabulkou obsahuje výpočetní metriky, které vychází z dat v modelu, metriky se dále používají ve vizualizacích.

Reporting je zpracován v angličtině, takže i dříve popsané názvy kategorií nebo shrinků jsou přeložené. Překlad je uvedený v příloze práce. Uvedené tržby v re-



Obrázek 7.1: Datový model tabulek v Power BI reportu.

portingu odpovídají nekonkrétní peněžní jednotce – z důvodu ochrany dat vybrané společnosti byla skutečná čísla vynásobena jistým koeficientem. Poměry zobrazené v reportu ale vstupním datům společnosti odpovídají.

### 7.1.1 Metriky

Power BI nabízí uživatelům reportu širokou interakci s vizuály. Díky metrikám se zobrazené hodnoty přepočítávají podle aktuálních filtrů nebo podle vybraných dat.

- **Lost Cost** – Základní metrika s hodnotou shrinku ze vstupních dat.
- **Product Share** – Základní metrika, obsahuje podíl shrinku na tržbách produktu (přímo ze vstupních dat). Pokud je ve vizuálu agregovaná např. na kategorii nebo prodejnu, vypočítá se její průměr.
- **Share** – Základní metrika, obsahuje podíl shrinku na tržbách produktu. Pokud je ve vizuálu agregovaná na prodejnu jedná se celkový evidovaný shrink prodejny dělený tržbou dané prodejny. Pokud je agregovaný podle typu shrinku jedná se o podíl součtu hodnot všech záznamů daného typu a tržeb všech prodejen. Bude-li aggregace probíhat zároveň na typu shrinku a na kategorii z úrovně 1, pak je podíl spočítaný vzhledem k této kategorii typu shrinku zároveň.
- **Share on Store Revenue** – Základní metrika, obsahuje podíl shrinku na celkových tržbách produktu. Tj. pokud je aggregace např. podle prodejny a

podle kategorie jedná se o podíl všech shrinků produktu z dané kategorie vydělený celkovými tržbami vybrané prodejny. (Zatímco v předchozí metrice by se jednalo o tržby pouze za vybranou kategorii.)

- **Revenue of Stores** – Celková tržba prodejny za celé sledované období
- **Revenue** – Základní metrika s hodnotou tržeb prodejny rozdělená podle části měsíce podle kategorií z úrovně 1 a ze vstupních dat.
- **Number of Recorded Shrinks** – Počet záznamů shrinků.

### 7.1.2 Reporting

#### Přehled

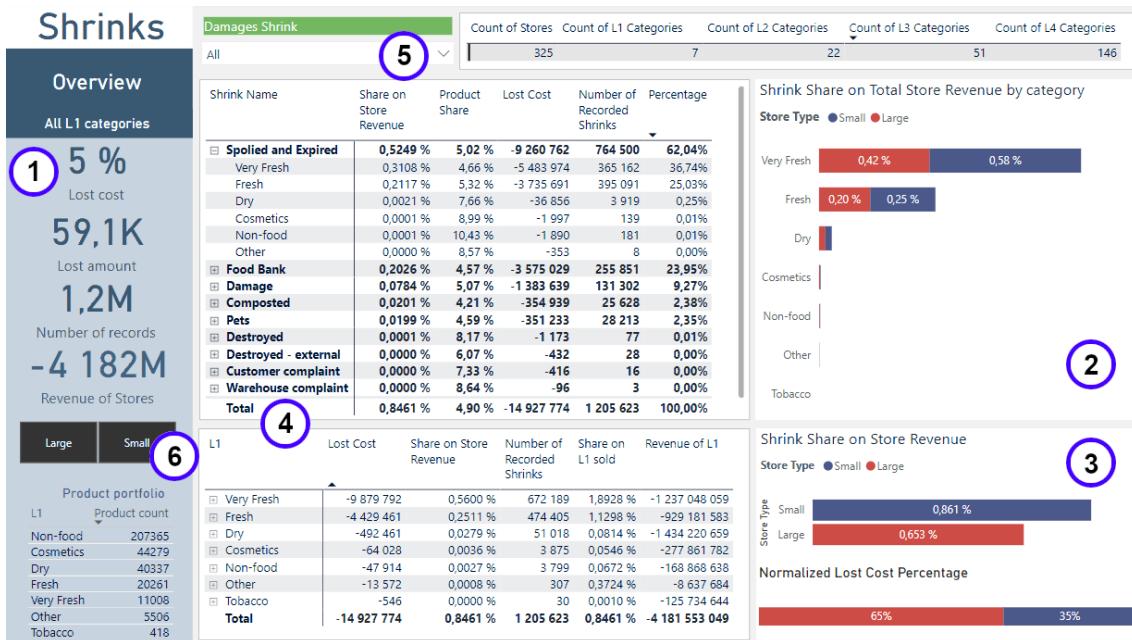
První stránka obsahuje přehled týkající se všech typů shrinků v rámci kategorie shrinky způsobené škodami, přehled je v tabulce 3.2 v kapitole 3. Na obr. 7.2 je snímek této stránky, pro lepší orientaci při popisu jsem na snímek přidala označující čísla. Na této stránce jsou vyfiltrované všechny kategorie, shrinky i prodejny. V levé části jsou uvedeny souhrnné informace pro vyfiltrované záznamy (tj. implicitně nic vyfiltrováno není, jedná se o celkové hodnoty) (č. 1).

Na grafu označeném č. 2, je znázorněno jak velký podíl shrinku na tržbách mají jednotlivé kategorie. Zároveň je barevně označeno, jakou měrou je hodnota zastoupená na malých či velkých prodejnách. Defaultní nastavení vizuálu je zobrazení kategorií úrovně 1, díky funkcionalitě nástroje Power BI je možné postoupit v hierarchii kategorií níže viz 7.3. První graf zobrazuje defaultní pohled, na druhém je vidět výsledek pokud uživatel klikne na ikonu  $\downarrow$  *přechod k podrobnostem všech polí*. V takovém případě se postoupí na další úroveň hierarchie napříč všemi kategoriemi. Třetí graf ukazuje stav, kterého uživatel docílí, pokud zaklikne ikonu  $\downarrow$  *přechod k podrobnostem jednoho pole*<sup>1</sup>. V takovém případě, poté co uživatel klikne na jednu z kategorií v grafu (její název nebo příslušný datový pruh), se zobrazí nižší úroveň hierarchie, ale pouze takové kategorie, které jsou podkategorií vybrané kategorie. Další graf (č. 3) ukazuje jaký je podíl shrinku na tržbách pro malé a velké prodejny. V tomto vizuálu po zvolení přechodu k podrobnostem se rozbalí hodnoty podílu pro jednotlivé typy shrinků. Opět jako v předchozím případě lze postupovat buď pouze pro jeden typ prodejen nebo oba. Pokud uživatel nezvolí *přechod k podrobnostem jednoho pole* ve vizuálu, ale klikne na datový element ve vizuálu, všechny vizuály na stránku se křížově vyfiltrují nebo křížově zvýrazní. Rozdíl mezi těmito dvěma akcemi je v sekci 2.4.2.

Přehledová stránka dále obsahuje dvě tabulky. První tabulka sleduje vybrané ukazatele pro typy shrinků, které lze dále prozkoumat z pohledu kategorií první úrovně. Druhá tabulka zobrazuje ukazatele z pohledu kategorií, a to od nejvyšší úrovně po nejnižší, případně až na detail samotných produktů a jejich ID. Vzhledem k tomu, že tabulka může při detailním procházení zabírat více místa je možné přejít na její detail, který se zobrazí přes celou aktuální stránku. Ukázka této tabulky je na obr. 7.4.

---

<sup>1</sup>Anglicky se přechod k podrobnostem jednoho pole v nástroji Power BI označuje jako *drill down*.



Obrázek 7.2: Power BI reporting pro zobrazení údajů o shrincích.

U čísel 5 a 6 jsou umístěny filtry – pro typ prodejny a pro typ shrinku. Vyfiltrováním příslušného typu se hodnoty v reportingu automaticky upraví. Všechny nevybrané kategorie nejsou zahrnuty do vizuálů, ani výpočtů hodnot. Např. pokud uživatel vybere pouze velké prodejny, celkové tržby se týkají již pouze všech velkých prodejen, nejde o celkové tržby všech prodejen z datasetu.



Obrázek 7.3: Ukázka interakce grafu záznamů shrinku pro přístupy k různým úrovním produktové hierarchie.

## Prodejny

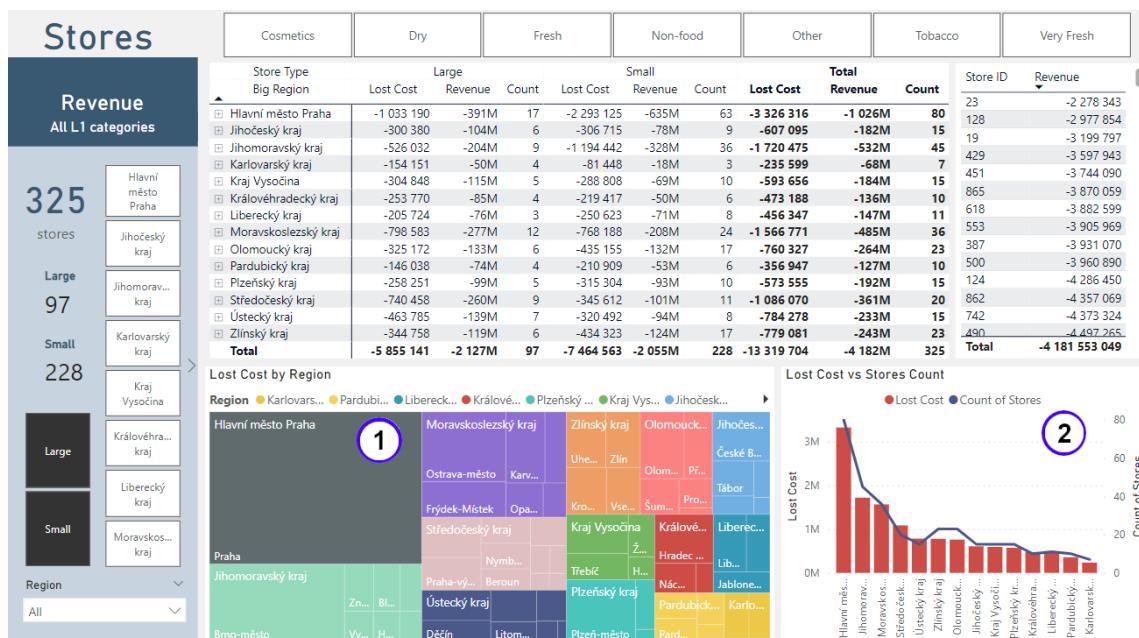
Další stránka zobrazuje prodejny a k nim příslušné ukazatele. Uživatel může filtrovat prodejny podle typu, kraje nebo okresu. Dále je možné filtrovat také podle kategorií. V tabulce lze zobrazit údaje agregované podle lokalit nebo přímo pro jednotlivé prodejny. Sloupce tabulky jsou rozdělené na hodnoty týkající se malých nebo velkých prodejen a poté celkové hodnoty pro oba typy. Graf stromová mapa pod číslem 1, obsahuje kraje, resp. vybrané okresy, ve kterých se nachází prodejny. Plocha

[Back to report](#)

L1	Lost Cost	Share on Store Revenue	Number of Recorded Shrinks	Share on L1 sold	Revenue of L1
Very Fresh	-9 879 792	0,5600 %	672 189	1,8928 %	-1 237 048 059
Fresh	-4 429 461	0,2511 %	474 405	1,1298 %	-929 181 583
FF CM MEAT PRODUCTS	-2 950 184	0,1672 %	335 871	0,7525 %	-929 181 583
PROC. MEAT SERVICE	-2 719 454	0,1541 %	316 852	0,6936 %	-929 181 583
PROC. MEAT SELF SERVICE	-220 106	0,0125 %	17 719	0,0561 %	-929 181 583
ŠUNKA STANDARDNÍ	-5 936	0,0003 %	479	0,0015 %	-929 181 583
SUNKA DUSENA	-5 224	0,0003 %	567	0,0013 %	-929 181 583

Obrázek 7.4: Detail tabulky vybrané ukazatele pro jednotlivé kategorie v produktové hierarchii.

příslušné lokality zabírá tolik procent obsahu grafu, kolik procent tvoří hodnota shrinků v této lokalitě. Graf č. 2 porovnává zaznamenaný shrink s počtem prodejen v regionu. Ukázka této stránky je na obr. 7.5.



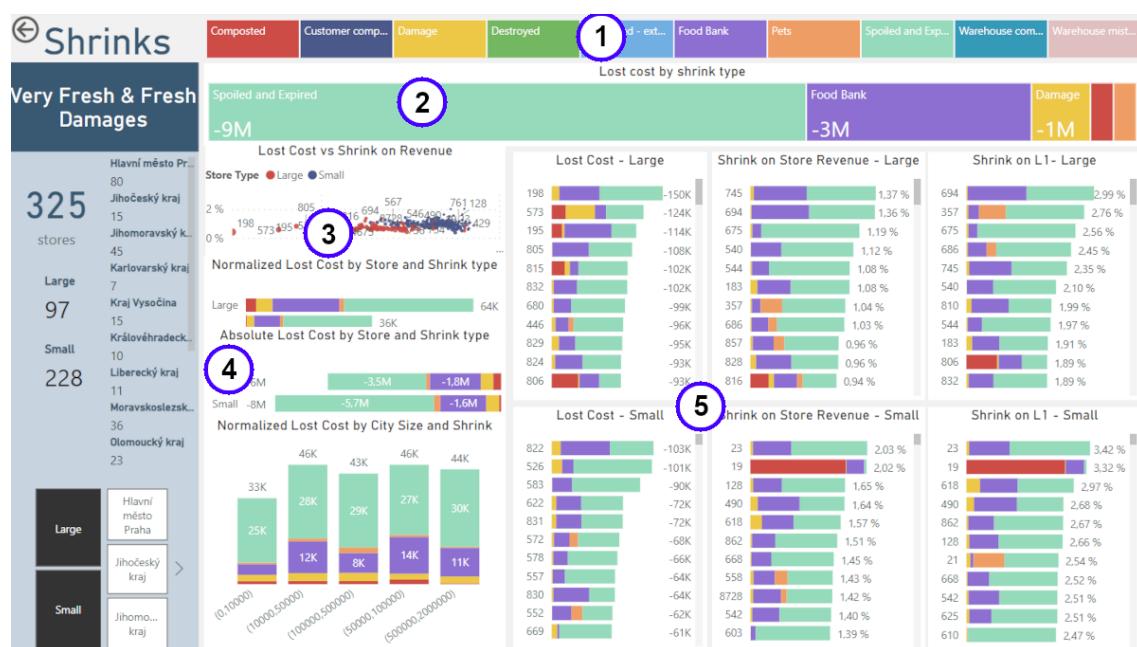
Obrázek 7.5: Power BI reporting pro zobrazení údajů o shrincích z pohledu prodejen.

## Prodejny – Velmi čerstvé a Čerstvé

Na této stránce reportu jsou vizuály pro analýzu chování shrinků z hlediska prodejen již pouze pro kategorie Čerstvé a Velmi čerstvé viz obr. 7.6. V nastavení reportu v sekci filtrů je možné vybrat i další kategorie, nicméně tato práce se věnuje především analýze těchto dvou kategorií, a tak jsou již předfiltrované tyto kategorie. Opět je možné filtrovat prodejny podle jejich atributů. Také je možné v horní části stránky vybrat sledovaný shrink (č. 1) – shrinky jsou pro lepší přehlednost barevně odlišené. Stránka dále obsahuje graf č. 2, který vizualizuje zastoupení shrinků podle

hodnoty shrinku (tj. ztracené náklady). Graf porovnání velkých a malých prodejen podle absolutního zaznamenaného shrinku na všech prodejnách a graf průměrné hodnoty shrinku na prodejně pro oba typy prodejen (č. 3). Pod tímto grafem jsou prodejny porovnané podle jejich ztracených nákladů. Zároveň je datový pruh barvne rozdelený podle typu shrinku (barva je shodná s barvou shrinku, kterou má přiřazenou nahoře na stránce).

Graf označený č. 4 zobrazuje ztrátu vlivem shrinku podle velikosti měst, ve kterých se prodejny nachází. Graf umožnuje přejít k podrobnostem, a to typu prodejny a konkrétním prodejnám. Zbylé grafy na stránce zobrazují konkrétní prodejny podle ukazatelů - hodnota shrinku, podíl shrinku na celkových tržbách prodejny, podíl shrinku na tržbách kategorií Velmi čerstvé a čerstvé na sledované prodejně. Grafy jsou rozdelené podle typu prodejen, datové pruhy jsou opět poměrově rozdelené podle zastoupení typů shrinků.

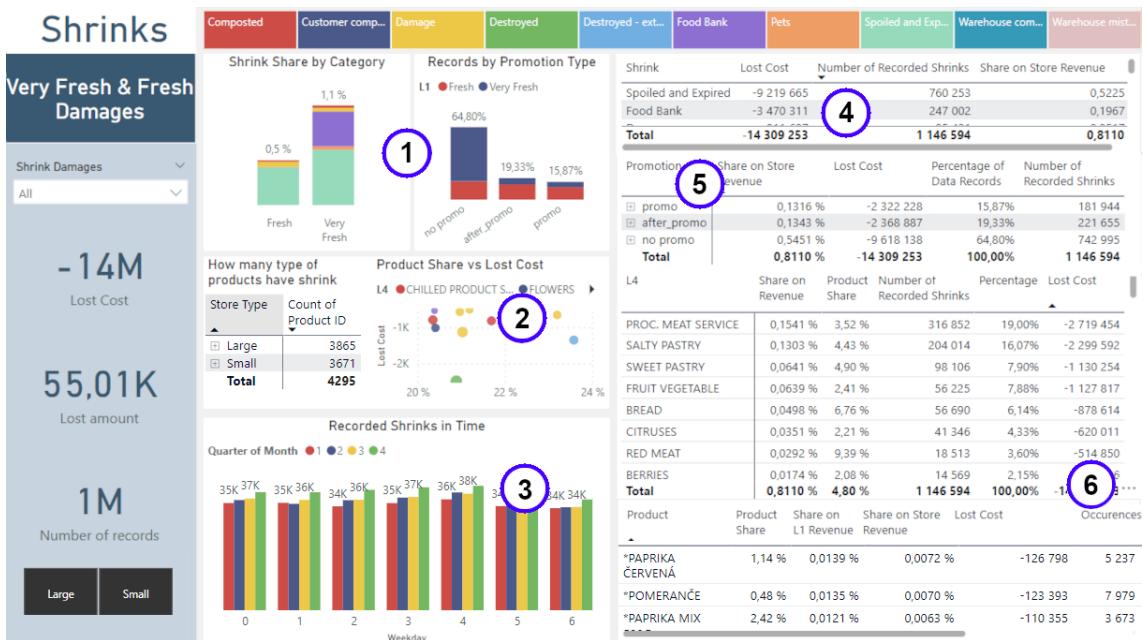


Obrázek 7.6: Power BI reporting pro zobrazení údajů o shrincích kategorie Čerstvé a Velmi čerstvé z pohledu prodejen.

## Kategorie – Velmi čerstvé a Čerstvé

Další stránka zobrazuje data kategorií Velmi čerstvé a Čerstvé podle dalších příznaček. Grafy označené jedničkou zobrazují podíl shrinku na celkových tržbách podle kategorie (z libovolné úrovně až k detailu produktu) a také zastoupení typu promokace produktů v záznamech, zároveň je datovým pruhem přiřazeno zastoupení typu shrinku. Graf označený dvojkou zobrazuje závislost ztracených nákladů a podílu shrinku na tržbách pouze daného produktu. Zobrazeny jsou pouze ty produkty, kde je ztráta vyšší než daná hodnota a podíl vyšší než 20 %. Produkty jsou barvne odlišené podle kategorie, do které patří. Tabulka vlevo od grafu 2 říká, kolik unikátních produktů bylo shrinkováno podle typu prodejny, od typu jde dále přejít

přes lokaci k samotným prodejnám. Graf č. 3. ukazuje počet záznamů evidovaných v daný den v týdnu, legenda zároveň určuje, v které části měsíce to bylo.



Obrázek 7.7: Zobrazení údajů o shrincích kategorie Čerstvé a Velmi čerstvé se zaměřením na kategorii a produkty.

Tabulka č. 4 přiřazuje vybrané ukazatele k jednotlivým typům shrinku. Další tabulka má tyto údaje ale přiřazené podle typu promoakce produktu, který je evidován v záznamu shrinku. Následující tabulky se týkají již konkrétních kategorií a produktů, z produktu se lze přesunout na další stránku věnující se detailu pouze jednoho produktu (viz obr. 7.8). Sledován je podíl shrinku na tržbách prodejny, na tržbách produktu na prodejne, výskyt v záznamech, ztracená tržba a její procentuální zastoupení. Na stránce jsou i jako v předešlých případech filtry a souhrnné údaje.

Product	Product Share	Share on L1 Revenue	Share on Store Revenue	Lost Cost
*PAPRIKA ČERVENÁ	1,10 %	0,0154 %	0,0080 %	-140 322
*POMERANČE		Show as a table		
*PAPRIKA MIX		Include	0,0077 %	-136 059
		Exclude	0,0070 %	-122 633
Promotion		Drill through	>	Product detail

Obrázek 7.8: Proklik na stránku s detailem produktu.

## Čas – Velmi čerstvé a Čerstvé

Na další stránce jsou data porovnávána vzhledem k datumu záznamu – ke dni v týdnu a části měsíce. Na této stránce, kromě filtrování typů jako v předchozích případech, může uživatel určit pro který ukazatel budou grafy zobrazeny. Vybrat lze ze ztracených nákladů, podílu shrinku na celkových tržbách prodejny a z podílu

shrinku na tržbách v dané kategorii a případně v části měsíce, viz č. 1 na obr. 7.9. Zbylé grafy jsou rozděleny na tři části – podle typu shrinku, podle umístění prodejen a podle typu prodejny. V každé části je přehled, který říká v jakém poměru tyto příznaky jsou (v závislosti na zvoleném ukazateli). Dále jsou pro každou část je ukazatel z pohledu dne v týdnu nebo části měsíce, kdy byl shrink zaznamenán.



Obrázek 7.9: Zobrazení údajů o shrincích kategorie Čerstvé a Velmi čerstvé se zaměřením na časové údaje.

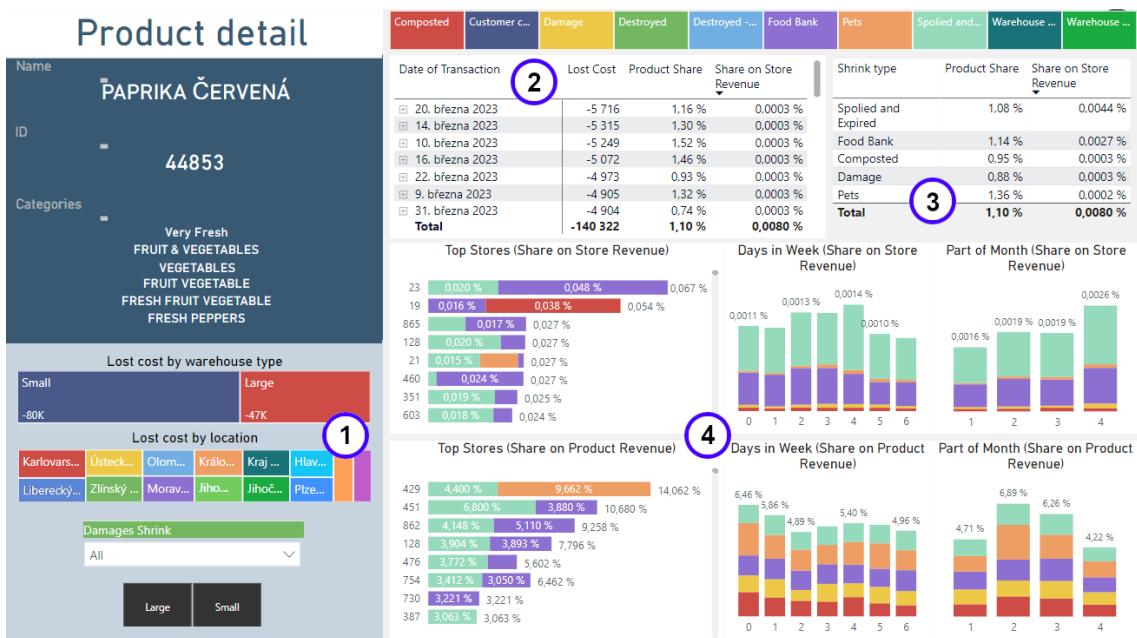
## Detail produktu

Poslední stránka je věnovaná analýze konkrétního produktu, snímek je na obr. 7.10. Je zobrazené zastoupení produktu podél typu prodejny a podle kraje z pohledu ztracených nákladů (č. 1). Dále je na této stránce tabulka (č. 2) se záznamy agregovaná podle data záznamu. Každý řádek s datem lze dále rozbalit pro detail o jaký typ shrinku se jednalo, k jednotlivým řádkům jsou napočítané vybrané ukazatele. Další tabulka (č. 3) ukazuje, jaký podíl na tržbách produktu a celkových tržbách má který typ shrinku.

Zbylé grafy ukazují konkrétní prodejny, které měli největší podíl shrinku tohoto produktu na svých tržbách, opět celkových i produktových. Dále jak je shrink tohoto produktu rozložený do dní v týdnu, resp. do částí v měsíci (č. 4).

## 7.2 Výsledky

Díky Power BI reportu je možné snadno zjistit, které kategorie či produkty jsou zastoupené více než jiné, nebo které prodejny mají vysoký podíl shrinku na svých tržbách a v jakém okrese ke shrinkům dochází nejčastěji. Tato sekce obsahuje popis



Obrázek 7.10: Power BI report – Detail produktu.

zjištěných informací z dat , a to včetně ukázků konkrétních vizualizací, ze kterých pozorování vychází. První část se věnuje pozorování na celých datech. tj. pozorovaná data za měsíc březen roku 2023, všech evidovaných shrinků způsobených škodami. Druhá část popisuje chování produktů v kategorii Čerstvé a Velmi čerstvé a typu shrinku prošlé a zkažené zboží.

## Pozorování na celých datech

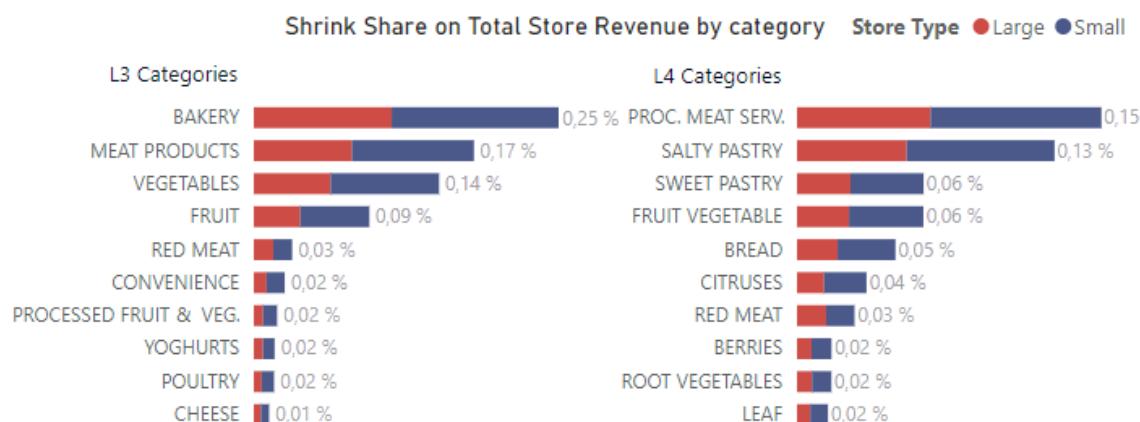
Pořadí zastoupení shrinků jednotlivých kategorií první úrovně hierarchie na celkových tržbách v datech lze vidět na obrázku 7.11. Je vidět, že kategorie Velmi čerstvé a Čerstvé jsou výrazně více zastoupeny než zbylé kategorie. Na této obrázku se nachází i porovnání hodnot vzhledem k velikosti prodejen. Malé prodejny mají v kategoriích s nejvyšším podílem shrinků na tržbách větší podíl než velké prodejny. Na následujícím obrázku 7.12 je zobrazeno zastoupení kategorií třetí a čtvrté úrovně opět s porovnáním pro oba typy prodejen. Všechny zobrazené kategorie jsou podkategoriemi skupin Velmi čerstvé nebo Čerstvé.

Tabulka 7.1 obsahuje agregované důležité metriky podle sedmi hlavních kategorií. Kategorie Velmi čerstvé má nejvíce evidovaných záznamů, rovněž nejvyšší oba podíly na tržbách i největší celkovou ztrátu způsobenou shrinkem. Hodnota shrinku činí téměř dvě procenta tržeb kategorie. Co se týče tržeb za tuto kategorii, jedná se o druhou kategorii s nejvyššími tržbami ve sledovaném období. Vyšší tržby má pouze kategorie Suché, kde ale ztráta u tohoto typu zboží je více než dvacetkrát menší. Kategorie Čerstvé se vyskytuje téměř v půl milionu záznamů. Hodnota shrinků produktů je více než jedno procento tržeb této kategorie. Zbylé kategorie mají velmi malé zastoupení v datech a význam vzhledem ke svým tržbám.

Další tabulka 7.2 obsahuje hodnoty ukazatelů k jednotlivým typům shrinků. Lze



Obrázek 7.11: Power BI report – Zastoupení kategorií na celkových tržbách ve sledovaném období (vlevo) a porovnání vzhledem k velikosti prodejen (vpravo).



Obrázek 7.12: Power BI report – Zastoupení kategorií 3. a 4. úrovně na celkových tržbách ve sledovaném období.

Tabulka 7.1: Tabulka základních metrik pro kategorie první úrovně produktové hierarchie.

Kategorie	Hodnota shrinku	Počet záznamů	Podíl na tržbách kategorie [%]	Podíl na celkových tržbách [%]	Tržby kategorie
Velmi čerstvé	9 879 792	672 189	1,8928	0,5600	1 237 mil.
Čerstvé	4 429 461	474 405	1,1298	0,2511	929 mil.
Suché	492 461	51 018	0,0814	0,0279	1 434 mil.
Kosmetika a drogerie	64 028	3 875	0,0546	0,0036	277 mil.
Nepotravinářské	47 914	3 799	0,0672	0,0027	168 mil.
Ostatní	13 572	307	0,3724	0,0008	8 mil.
Tabák	546	30	0,0010	0,0000	125 mil.

vidět, že Prošlé a zkažené zboží tvoří 62 % všech shrinků z pohledu ztracených nákladů. Necelými 24 % jsou zastoupené produkty, které byly věnovány potravinným bankám. Jedná se sice o druhý nejčastější shrink v záznamech, nicméně toto zboží není vyhozeno zcela, ale je předáno dále. Pro společnost se jedná stále ztracený

zisk, ale zboží je dál efektivně využito a nedochází tak k plýtvání jako takovému. Další typy shrinků nejsou dohromady netvoří ani 15 % všech ztracených nákladů. Potravinová banka je evidována z 96 % u produktů z kategorie Velmi čerstvé a jedná se především o podkategoriю Pečivo (62 %), Zelenina (19 %) a Ovoce (12,5 %).

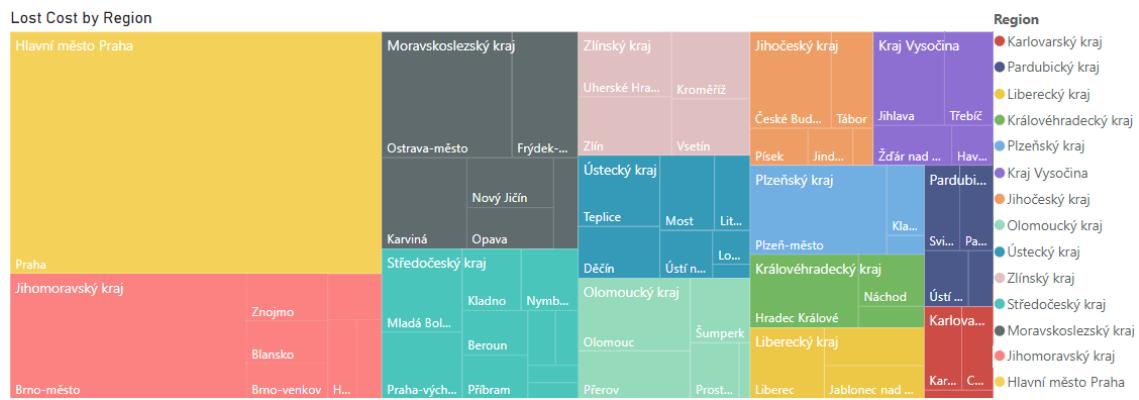
Tabulka 7.2: Tabulka základních metrik pro jednotlivé typy shrinků.

Typ shrinku	Hodnota shrinku	Počet záznámů	Průměr. podíl na tržbách produků [%]	Podíl na celkových tržbách [%]	Hodnota shrinku [%]
Prošlé a zkažené zboží	9 260 762	764 500	5,02	0,5249	62,04
Potravinová banka	3 575 029	255 851	4,57	0,2026	23,95
Poškození	1 383 639	131 302	5,07	0,0784	9,27
Kompostéry	354 939	25 628	4,21	0,0201	2,38
Zvířecí útulky	351 233	28 213	4,59	0,0199	2,35
Zničení	1 173	77	8,17	0,0001	0,01
Poškození vnějšími vlivy	432	28	6,07	0,0000	0,00
Zákaznické reklamace	416	16	7,33	0,0000	0,00
Reklamace centrálního skladu	96	3	8,64	0,0000	0,00

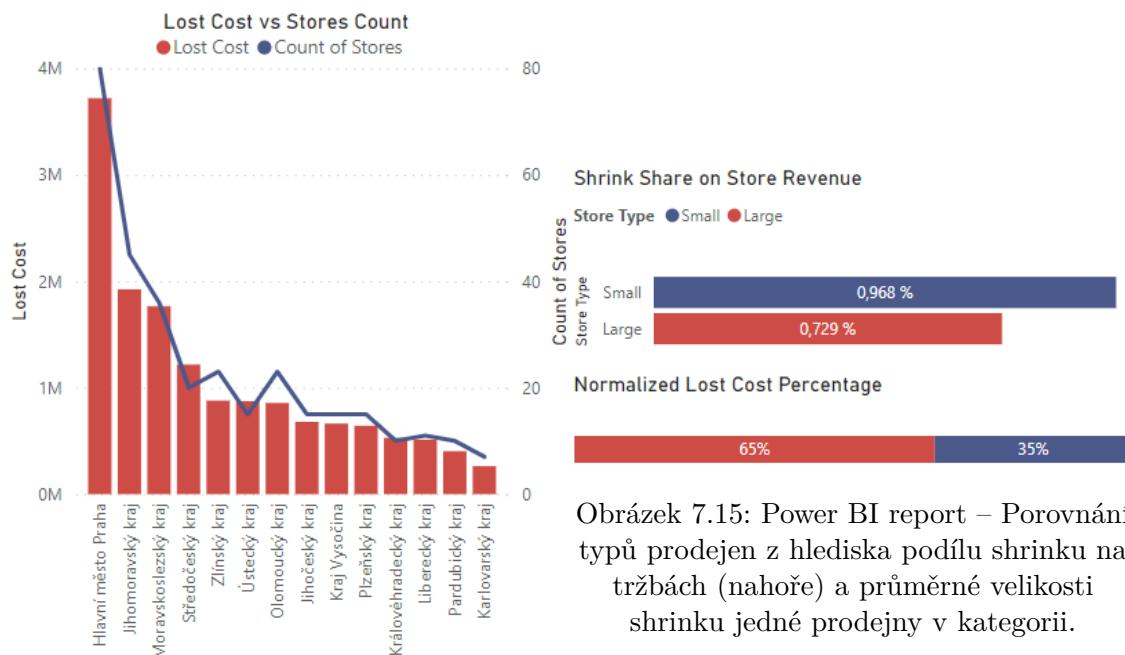
Obrázek 7.13 obsahuje graf typu stromová mapa. Na grafu jsou zobrazeny kraje České republiky, případně okresy s nejvyššími hodnotami shrinku. Pole příslušného kraje zabírá tolik procent grafu, kolik zaujímá evidovaný shrink. Necelých 25 % z celkového hodnoty všech zaznamenaných shrinků patří do kraje Hlavní město Praha. Dalších téměř 25 % tvoří kraje Jihomoravský a Moravskoslezský v podobném poměru. Největší zastoupení v těchto krajích mají okresy příslušející jejich krajským městům. Záznamy ze Středočeského kraje tvoří 8,2 % ztracených nákladů. Každý z krajů Zlínský, Ústecký, Olomoucký tvoří necelých 6 %. Zbylé kraje jednotlivě zaujímají méně jak 5 % na celkovém shrinku. Na grafu 7.14 je porovnání hodnoty shrinku a počtu prodejen pro kraje, je vidět, že tyto dva ukazatele spolu souvisí.

Na obrázku 7.15 nahoře je zobrazen jaký je podíl celkového shrinku na tržbách všech prodejen daného typu. V dolní části je poměr normalizované hodnoty shrinku na prodejnách obou typů – tj. průměrně jedna velká prodejna tvoří 65 % evidovaných shrinků, zatímco malé prodejny zbylých 35 %.

Pro většinu prodejen platí, že největší část hodnoty zaznamenaných shrinků tvoří



Obrázek 7.13: Power BI report – Zobrazení krajů podle velikosti hodnoty shrinku (vlevo) a počet prodejen v regionu (vpravo).



Obrázek 7.14: Power BI report – Porovnání hodnoty shrinku a počtu prodejen pro jednotlivé kraje.

prošlé a zkažené zboží, případně zboží darované potravinové bance. Neobvyklé začlenění shrinků ale vykazují malé prodejny s ID 19 (Brno) a 126 (okres Třebíč), kdy největší část zboží je kompostována a téměř žádné není vyhozeno jako shrink prošlého zboží. Ukázka grafů je na obrázku 7.16. V levé části obrázku jsou dva grafy. Jeden zobrazuje průměrnou hodnotu shrinku pro jednu prodejnu, a tedy i průměrné zastoupení typů shrinku. Druhý obsahuje součet hodnot shrinků přes všechny prodejny.



Obrázek 7.16: Power BI report – Hodnota shrinku pro typ prodejny). Prodejny s nejvyšší hodnotou ukazatelů (vpravo) – ztracené náklady a podíl shrinku na tržbách.

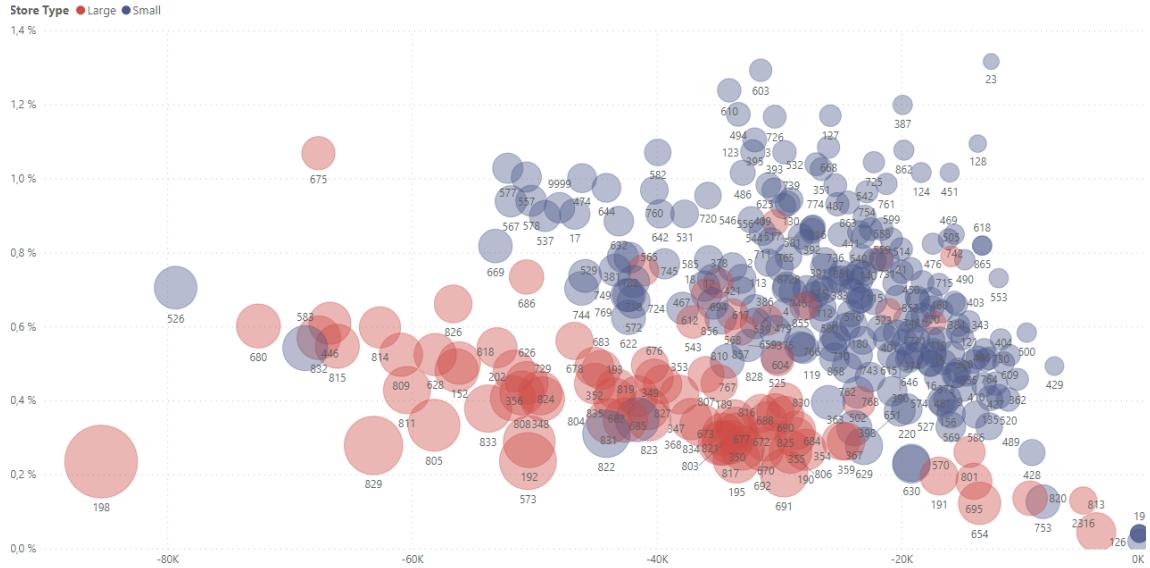
## Pozorování pro vybraná data

Další část se věnuje pouze datům, která se týkají kategorií Velmi čerstvé a Čerstvé a shrinku prošlé a zkažené zboží. Evidovaný shrink má pouze na těchto kategoriích hodnotu 9,2 mil. peněžních jednotek. Na grafu 7.17 je vidět, že malé prodejny mají ve většině nižší hodnotu shrinku, ale podíl na shrinku na tržbách mají vyšší než velké prodejny. Prodejny s velmi nízkými shrinky mají i nízký podíl shrinku na tržbách. Největší ztracené náklady byly evidovány u velké prodejny s ID 198 v Praze, nicméně podíl tohoto shrinku na celkových tržbách je pouhých 0,23 %, což je jeden z nejnižších. Zároveň se jedná o prodejnu s nejvyššími tržbami. Zatímco velká prodejna 675 v okresu Uherské Hradiště má třetí nejvyšší hodnotu shrinku a zároveň i nejvyšší podíl shrinku na tržbách 1,68 % mezi velkými prodejny. Nejvyšší podíl shrinku na svých tržbách byl evidován u malé prodejny 23, které se nachází v Brně, nicméně hodnota shrinku je nízká. Druhý nejvyšší podíl má malá pražská prodejna. Z grafu je patrné, že čtyři malé prodejny mají podobné tržby jako velké prodejny. Jedná se o dvě prodejny v Praze a prodejny v okresu Litoměřice a Jablonec nad Nisou, tyto pražské prodejny mají vyšší podíl shrinku než zbylé dvě prodejny. Z hlediska umístění prodejen do krajů a okresů nebyl v datech na této úrovni detailu objeven žádný vzor.

Graf 7.18 zobrazuje průměrnou hodnotu shrinků na prodejně podle obydlenosti města, kde se nachází. Pro malé prodejny velikosti shrinku nejsou příliš rozdílné vzhledem k počtu obyvatel, kde se prodejna nachází. Pro velké prodejny platí, že ve městech do deseti tisíc obyvatel bývá hodnota shrinku vyšší než ve městech s více obyvateli.

Z pohledu času záznamu, tj. dne v týdnu a čtvrtiny měsíce platí, že nejvyšší shrinky jsou evidovány poslední čtvrtinu měsíce. Důvodem může být to, že část výrobků má datum expirace uvedené ve formátu měsíc-rok, což znamená, že zboží prochází posledním dnem v měsíci. Dalším důvodem může být, že před začátkem nového měsíce zaměstnanci evidují více záznamů. Například se může jednat i o shrinky, které se uskutečnili dříve, ale až s koncem měsíce byly nahrány do systému. Nejvyšší hodnota shrinků je evidována v pátek, zatímco nejnižší v neděli, pondělí a úterý. Viz graf 7.19.

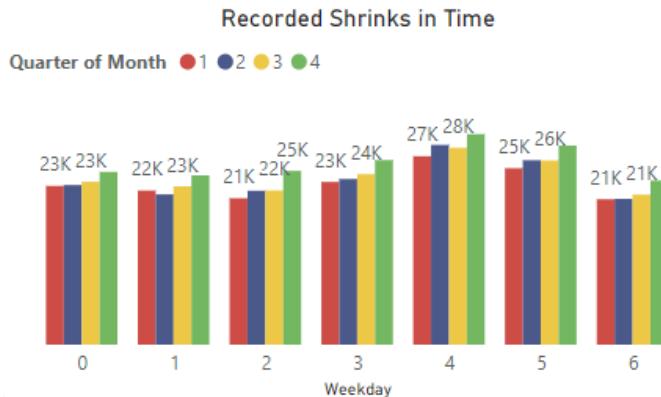
Na grafu 7.20 je procentuální zastoupení evidovaných shrinků vzhledem k typu promoakce. Nejvíce záznamů se týká produktů bez promoakce, zároveň lze vidět, že



Obrázek 7.17: Power BI report – Velikost shrinku na prodejně versus podíl shrinku na tržbách. Velikost zobrazeného bodu ukazuje výši tržeb na prodejně

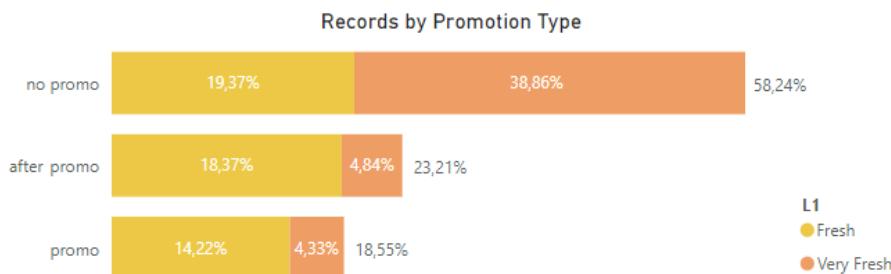


Obrázek 7.18: Power BI report – Průměrná hodnota shrinku prodejen podle velikosti měst, ve které se nachází prodejna.



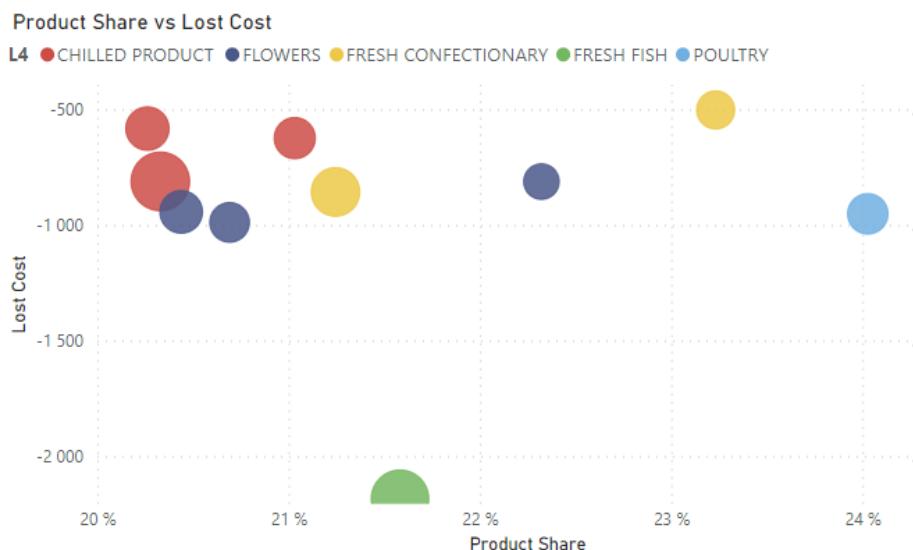
Obrázek 7.19: Power BI report – Počet záznamů vzhledem ke dni v týdnu a čtvrtině měsíce.

Velmi čerstvé produkty byly ve sledovaném období v promoakci méně než Čerstvé produkty.



Obrázek 7.20: Power BI report – Počet záznamů podle typu promoakce shrinkovaného produktu.

Nejvíce záznamů mají kategorie (úroveň 3) – Masné výrobky – pultový prodej (250 tis. záznamů s hodnotou shrinku 2 mil. peněžních jednotek), Slané a sladké pečivo (127 tis. záznamů s hodnotou shrinku 1,5 mil. peněžních jednotek), Plodová zelenina (29 tis. záznamů), Chléb (24 tis. záznamů) a Citrusy (22 tis. záznamů).



Obrázek 7.21: Power BI report – Ztracené náklady (hodnota shrinku) vs Podíl shrinku produktu na tržbách produktu. Velikost bodů odpovídá počtu záznamů.

Produktem, který má nejvyšší zaznamenanou celkovou hodnotu shrinku je balená šunka Nejvyšší jakosti z kategorie Masných produktů, většina záznamů pochází z období během promoakce. Druhý nejvyšší shrink měla Dušená šunka nejvyšší jakosti také s nejvíce záznamy v promoakci. Oba produkty měly podíl shrinku na svých tržbách též 3 %. Nejvyšší podíl na svých tržbách se týká části výrobků z kategorie sýrů (třetí úroveň hierarchie), podíl se pohybuje okolo 27 %. Nicméně ztracené náklady nedosahují ani 50 peněžních jednotek a mají jen velmi málo záznamů, podíl na celkových tržbách je tedy zanedbatelný. Na obr. 7.21 jsou vyfiltrované produkty, které mají hodnotu shrinku větší než 500 peněžních jednotek a zároveň podíl shrinku na svých tržbách více jak 20 %. Jedná se o tři chlazené produkty, dva produkty z kategorie cukrovinek, tři druhy řezaných květin, jeden prémiový drůbeží steak a

čerstvá treska. Nejvyšší podíl na celkových tržbách měly opět tyto dva produkty, dále pak pomeranče, které byly během záznamů shrinků v promoakci nebo těsně po promoakci.

# Závěr

Cílem práce bylo



# Literatura

- [1] BAUDIN, Michel. *Lean Logistics: The Nuts and Bolts of Delivering Materials and Goods*. New York: Productivity Press, 2005. ISBN 978-1563272967.
- [2] CHRISTOPHER, Martin. *Logistics & Supply Chain Management*. 5th ed. Harlow: Pearson Education Limited, 2016. ISBN 9781292083797.
- [3] HASTIE, T., TIBSHIRANI R., FRIEDMAN J. H. *The elements of statistical learning: data mining, inference, and prediction*. 2nd ed. New York: Springer, 2009. Springer series in statistics. ISBN 978-0-387-84857-0.
- [4] What is the difference between Logistics and Supply Chain Management. In: *IIM Udaipur Chronicles* [online]. 11. 10. 2019. [cit. 2022-11-07] Dostupné z: <https://www.iimu.ac.in/blog/what-is-the-difference-between-logistics-and-supply-chain-management/>
- [5] JIRSÁK, Petr, MERVART, Michal, VINŠ, Marek. *Logistika pro ekonomy – vstupní logistika*. 1. vydání. Praha: Wolters Kluwer ČR, 2012.
- [6] JONES, Daniel T., HINES Peter a RICH Nick. Lean logistics. *International Journal of Physical Distribution & Logistics Management*. 1997, **27**(3/4), 153-173. ISSN 0960-0035. Dostupné z: doi:10.1108/09600039710170557
- [7] KURITA, Takio. *Principal component analysis (PCA)*. *Computer Vision: A Reference Guide*. 2019, 1-4. [cit. 2022-11-07] Dostupné z: [https://link.springer.com/content/pdf/10.1007/978-3-030-03243-2\\_649-1.pdf](https://link.springer.com/content/pdf/10.1007/978-3-030-03243-2_649-1.pdf)
- [8] TONHAUSEROVÁ, Zuzana. *Metoda hlavních komponent a její aplikace*. Diplomová práce. Olomouc: UPOL. 2013 [cit. 2023-12-18]. Dostupné z: [https://theses.cz/id/iwan2b/Zuzana\\_Tonhauserov\\_-\\_Metoda\\_hlavnch\\_komponent.txt](https://theses.cz/id/iwan2b/Zuzana_Tonhauserov_-_Metoda_hlavnch_komponent.txt)
- [9] JAADI, Zakaria. *A Step-by-Step Explanation of Principal Component Analysis (PCA)* [online]. [cit. 2023-03-04]. Dostupné z: <https://builtin.com/data-science/step-step-explanation-principal-component-analysis>
- [10] MILLS, Peter. *Efficient statistical classification of satellite measurements*. In: *International Journal of Remote Sensing*. Informa UK Limited, 2011, 32(21): 6109–6132. [cit. 2023-12-18]. Dostupné z: doi:10.1080/01431161.2010.507795

- [11] ZAMAZAL, Petr. *Statistická analýza rozsáhlých dat z průmyslu*. Diplomová práce, vedoucí Šomplák, Radovan. Vysoké účení technické v Brně, 2010.
- [12] DI FRANCO, Giovanni. *Multiple correspondence analysis: one only or several techniques?*. Quality & Quantity, 2016, 50.3: 1299-1315. [cit. 2023-03-05]. Dostupné z: doi:10.1007/s11135-015-0206-0
- [13] ABDI, Hervé, VALENTIN, Dominique. *Multiple correspondence analysis*. In: *Encyclopedia of measurement and statistics*. 2007, 2.4: 651-657. [cit. 2023-03-05]. Dostupné z: <https://personal.utdallas.edu/~Herve/Abdi-MCA2007-pretty.pdf>
- [14] NAVARA, Mirko. *Teorie informace*. [online]. 3. 1. 2017 [cit. 2023-12-15]. Dostupné z: [https://cmp.felk.cvut.cz/~navara/psi/TI\\_ebook.pdf](https://cmp.felk.cvut.cz/~navara/psi/TI_ebook.pdf)
- [15] PRŽICHYSTAL, Jan. *Úvod do teorie informace*. [online]. 3. 1. 2007 [cit. 2023-12-15]. Dostupné z: <https://akela.mendelu.cz/~jprich/predn/teoinf.pdf>
- [16] KROUPA, Tomáš. *Úvod do teorie informace: Matematické základy komprese a digitální komunikace*. [online]. [cit. 2023-12-15]. Dostupné z: [https://math.fel.cvut.cz/en/people/gollova/tik/TI\\_prednasky.pdf](https://math.fel.cvut.cz/en/people/gollova/tik/TI_prednasky.pdf)
- [17] GREENACRE, Michael. *Correspondence analysis in practice*. chapman and hall/crc, 2017. [cit. 2023-03-05].
- [18] Correspondence analysis. In *Wikiwand* [online]. [cit. 2023-03-06]. Dostupné z: [https://www.wikiwand.com/en/Correspondence\\_analysis](https://www.wikiwand.com/en/Correspondence_analysis)
- [19] WRONKA, Anna. LEAN LOGISTICS. *Journal of Positive Management*. 2017, 7(2), 55-63. ISSN 2392-1412. Dostupné z: doi:10.12775/JPM.2016.012
- [20] SUTHERLAND Joel, BENNETT Bob. *The Seven Deadly Wastes of Logistics: Applying Toyota Production System Principles to Create Logistics Value*. Bethlehem, PA: Lehigh University, 2007. Dostupné z: <https://www.researchgate.net/publication/265356600>
- [21] SKHMOT, Nawras. *The Lean Way Blog: The 8 Wastes of Lean. The Lean Way* [online]. 5. 8. 2017 [cit. 2022-11-17]. Dostupné z: <https://theleanway.net/The-8-Wastes-of-Lean>
- [22] SKHMOT, Nawras. *The Lean Way Blog: What is Lean?. The Lean Way* [online]. 5. 8. 2017 [cit. 2022-11-17]. Dostupné z: <https://theleanway.net/what-is-lean>
- [23] SKHMOT, Nawras. *The Lean Way Blog: What is Muda, Mura, and Muri?. The Lean Way* [online]. 5. 8. 2017 [cit. 2022-11-17]. Dostupné z: <https://theleanway.net/muda-mura-muri>
- [24] Learning the Lingo: 3 definitions related to unsold food inventory. In: *Blog - Spoiler Alert* [online]. 17. 06. 2019. [cit. 2022-02-07] Dostupné z: <https://blog.spoileralert.com/3-definitions-unsold-food-inventory>

- [25] HALFORD, M. Prince. [online]. [cit. 2023-03-08]. Dostupné z: <https://github.com/MaxHalford/prince>
- [26] BAIJAYANTA, Roy. *All about Categorical Variable Encoding* [online]. [cit. 2023-03-13]. Dostupné z: <https://towardsdatascience.com/all-about-categorical-variable-encoding-305f3361fd02>
- [27] scikit-learn Machine Learning in Python [online]. 2023 [cit. 2023-03-21]. Dostupné z: <https://scikit-learn.org/stable/>
- [28] Multiclass and multioutput algorithms. scikit [online]. [cit. 2023-04-10]. Dostupné z: <https://scikit-learn.org/stable/modules/multiclass.html>
- [29] BOBBITT, Zach. *How to Interpret Cramer's V In Statology* [online]. 2021 [cit. 2023-03-21]. Dostupné z: <https://www.statology.org/interpret-cramers-v/>
- [30] ZYCHLINSKI, Shaked. *The Search for Categorical Correlation* [online]. 2018 [cit. 2023-03-14]. Dostupné z: <https://towardsdatascience.com/the-search-for-categorical-correlation-a1cf7f1888c9>
- [31] HOLČÍK, Jiří, KOMENDA, Martin (eds.) a kol. *Matematická biologie: e-learningová učebnice* [online]. 1. vydání. Brno: Masarykova univerzita, 2015. [cit. 2023-03-14]. ISBN 978-80-210-8095-9.
- [32] FROST, Jim. *Choosing the correct type of regression analysis.* [online]. [cit. 2023-04-10]. Dostupné z: <https://statisticsbyjim.com/regression/choosing-regression-analysis/>
- [33] Multiclass logistic regression. In *Refactored* [online]. [cit. 2023-04-10]. Dostupné z: <https://refactored.ai/>
- [34] BIAU, Gérard; SCORNET, Erwan. *A random forest guided tour.* In: *Test* [online]. 2016, 25: 197-227. [cit. 2023-04-10].
- [35] LOK, Leon. *Decision trees, random forests and gradient boosting: What's the difference?* [online]. 5. 1. 2022. [cit. 2023-04-10]. Dostupné z: <https://leonlok.co.uk/blog/decision-trees-random-forests-gradient-boosting-whats-the-difference/>
- [36] Scipy.stats.pearsonr. In *scipy.stats.pearsonr - SciPy v1.11.4 Manual* [online]. [cit. 2023-12-10]. Dostupné z: <https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.pearsonr.html>
- [37] Jupyter Team. *Jupyter Notebook Documentation.* [online]. [cit. 2023-12-27]. Dostupné z: <https://jupyter-notebook.readthedocs.io/>
- [38] HeidiSQL. [online]. [cit. 2023-12-27]. Dostupné z: <https://www.heidisql.com/>
- [39] Dokumentace ke službě Power BI. [online]. 2023. [cit. 2023-12-28]. Dostupné z: <https://learn.microsoft.com/cs-cz/power-bi/>

- [40] KUMAR, Bijay. *Power BI Column Chart – Complete tutorial*. [online]. 2021. [cit. 2023-12-28]. Dostupné z: <https://www.enjoysharepoint.com/power-bi-column-chart/>
- [41] BILÍK, Jan. *Databáze demografických údajů za vybraná města ČR* [online]. 30. 05. 2023 [cit. 2023-07-12]. Dostupné z: <https://www.czso.cz/csu/czso/databaze-demografickych-udaju-za-vybrana-mesta-cr>
- [42] HUBER, Nicholas, MICHAEL, Katina, *Minimizing Product Shrinkage across the Supply Chain using Radio Frequency Identification: a Case Study on a Major Australian Retailer*,. In: *International Conference on the Management of Mobile Business (ICMB 2007)*. Toronto, ON, Canada, 2007, 45-45. [cit. 2023-12-16]. Dostupné z: doi: 10.1109/ICMB.2007.43.
- [43] BECK, Adrian. *Moving beyond shrinkage: developing a definition and typology of total retail loss*. 2018, 93–110. <https://doi.org/10.1057/s41284-017-0090-5>
- [44] RAUCH, Jan, ŠIMŮNEK, Milan *Metoda GUHA a Systém LISp-Miner* [online]. [cit. 2023-12-15]. Dostupné z: [lispminer.vse.cz](http://lispminer.vse.cz)
- [45] MASA, Petr. *CleverMiner – Beyond apriori*. [online]. [cit. 2023-12-15]. Dostupné z: [www.cleverminer.org](http://www.cleverminer.org)
- [46] de WINTER, Joost, GOSLING, Samuel, POTTER, Jeff. *Comparing the Pearson and Spearman Correlation Coefficients Across Distributions and Sample Sizes: A Tutorial Using Simulations and Empirical Data*. *Psychological Methods*. 2016, 21: 273-290. Dostupné z: doi: 10.1037/met0000079.

# **Příloha A**

## **Obsah přiloženého CD**

BP\_Gruberova.pdf – soubor s elektronickou verzí této bakalářské práce.  
PreprocessData – složka se zdrojovými kódy naimplementovaného balíku  
PreprocessData.jl.