Vysoká škola ekonomická v Praze

Fakulta informatiky a statistiky

Vliv komplementů na dynamickou cenotvorbu

DIPLOMOVÁ PRÁCE

Studijní program: Aplikovaná informatika

Studijní obor: Informační systémy a technologie

Autor: Bc. Karel Poplštein

Vedoucí diplomové práce: doc. Ing. Miloš Maryška, Ph.D.

Praha, 2019

Prohlášení

Prohlašuji, že jsem diplomovou práci Vliv komplementů na dynamickou cenotvorbu zpracoval samostatně a že jsem uvedl všechny použité prameny a literaturu, ze které jsem čerpal.

V Praze dne 29. dubna 2019 . . . . . .

Karel Poplštein

Poděkování

Především bych velmi rád poděkoval svému vedoucímu práce doc. Ing. Miloši Maryškovi, Ph.D. za jeho ochotu i věnovaný čas při vedení této práce. Mé velké díky zároveň patří Mgr. Filipu Trojanovi, Ph.D. za podnět ke zpracování tohoto tématu i mnoho cenných rad.

Abstrakt

Tato diplomová práce se zabývá tématem dynamické cenotvorby, respektive algoritmického počítání optimální ceny produktů na e-shopech z vymodelovaných poptávkových křivek pomocí strojového učení. Práce zkoumá, jaký vliv na tyto poptávkové křivky mají produkty, které zákazníci často nakupují společně a jejím cílem je rozšířit modely o tyto nové prediktory a prozkoumat jejich vliv na přesnost modelů a tím i na výpočet výsledné optimální ceny. K identifikaci těchto asociačních pravidel v transakční historii e-shopu je v práci využito postupů analýzy nákupního košíku, pro jehož implementaci v jazyce R práce poskytuje návod. Zároveň poskytuje pohled na architekturu současného řešení, a jak do něj nové prediktory jsou implementovány. Nakonec poskytuje porovnání rozšířených modelů a modelů původních, následované diskuzí nad těmito výsledky, které potvrzují původní hypotézu o zpřesnění modelů vlivem komplementů.

Klíčová slova

Dynamická cenotvorba, regresní analýza, analýza nákupního košíku

Abstract

This thesis addresses the topic of dynamic pricing, or more specifically finding optimal prices of products on e-shop utilizing demand curves approximated by machine learning models. The thesis examines the effect of complementary products that customers often buy together and aims to extend these models with new predictors relating to these products and try to reduce the prediction error this way thus improving the optimized price output. The thesis contains an R implementation of market basket analysis to identify these association rules in the transaction history as well as instructions on how to use it. Afterwards it describes a high-level architecture of the current solution and specifies how the new predictors are added into it. Finally, it provides a comparison between the original and extended models followed by a discussion of the results that supports the original hypothesis that dynamic pricing models can be made more accurate by adding the effect of complements.

Keywords

Dynamic pricing, regression analysis, market basket analysis

Obsah

1 Úvod

Tato diplomová práce se zabývá aktuální problematikou dynamické cenotvorby, což je jedna ze strategií oceňování produktů či služeb, podle které je možné produkty přeceňovat flexibilně podle aktuální poptávky na trhu. Díky použití algoritmů, do kterých je začleněno mnoho faktorů, jako je cena konkurence, nabídka a poptávka, případně další externí faktory na trhu, je možné optimalizovat cenu produktů či služeb v reálném čase a zvýšit tak tržby i zisk, případně se soustředit na maximalizaci jednoho z nich.

Tato práce se soustředí na dynamickou cenotvorbu v prostředí e-commerce, tedy ideální prostředí, kde lze takovou technologii snadno implementovat a dosáhnout výhod, které z ní plynou. Na rozdíl od kamenných prodejen lze v případě využití algoritmu reagovat na měnící se podmínky okamžitě a ceny upravovat libovolně často, aniž by to přinášelo neúměrné zvýšení nároků na provoz. Zároveň jsou data nezbytná pro správné nastavení a fungování modelů, které počítají optimální cenu, v prostředí e-commerce mnohem snadněji a levněji dostupná, než v případě tradičního maloobchodu.

Jelikož samotná technologie a algoritmus pro dynamickou cenotvorbu nejsou novinkou, práce se zabývá rozšířením již existujících modelů, nasazených na jednom z předních českých e-shopů, o vliv komplementů jednotlivých produktů, tedy produktů, které zákazníci nakupují společně, na jejich cenovou elasticitu. Stávající modely nastavují optimální cenu pomocí sestrojení poptávkových křivek produktů modely strojového učení. V této poptávkové křivce lze následně nalézt cenu, při které dochází k maximalizaci požadovaného parametru, tedy marže či celkového obratu.

Cílem této práce je tedy komplementy jednotlivých produktů nalézt, čehož je docíleno pomocí analýzy nákupního košíku, díky které je vydolován seznam asociačních pravidel identifikující produkty, které jsou zákazníky nakupovány společně a následně tyto nové prediktory začlenit do modelů počítajících poptávkové křivky vybraných produktů a porovnat, jestli došlo ke zpřesnění těchto modelů.

Práce je určena komukoliv, koho problematika dynamické cenotvorby zajímá, případně sám provozuje e-shop a měl by zájem zjistit více o této technologii, jejích výhodách i úskalích a zvážit, zda by měla v jeho případě odpovídající přínos. Zároveň může posloužit i vývojářům, kteří mají zájem podobnou technologii vyvinout, či se vývoji podobné technologie již věnují

a hledají inspiraci pro zpřesnění aktuálních modelů. Další cílovou skupinou této práce mohou být čtenáři, kteří chtějí v prostředí e-commerce blíže prozkoumat vzájemný vliv produktů, které jsou nakupovány společně, na elasticitu jejich poptávky.

1.1 Cíle práce

Tato podkapitola obsahuje definici hlavního cíle práce a na něj navázaných dílčích cílů.

1.1.1 Hlavní cíl práce

Hlavním cílem práce je rozšířit existující implementaci modelů pro dynamickou cenotvorbu o vliv komplementů jednotlivých produktů a dosáhnout tak větší přesnosti, než mají modely původní. Porovnání přesnosti modelů bude provedeno na vybrané datové množině tvořené roční historií objednávek jednoho z předních českých e-shopů. Primárním cílem je tedy rozšířit modely pro produkty, u kterých budou komplementy nalezeny a těmito novými a rozšířenými modely dosáhnout nižších hodnot MASE (mean absolute scaled error) tedy hlavní metriky, která je blíže vysvětlena v teoretické části práce.

1.1.2 Dílčí cíle

Pro dosažení tohoto cíle jsou dále definovány následující dílčí cíle:

- Analyzovat a popsat možnosti a postupy analýzy nákupního košíku.
- Pomocí analýzy nákupního košíku vydolovat z dat o transakční historii asociační pravidla a identifikovat jejich pomocí v katalogu jednotlivé komplementární produkty.
- Rozšířit existující modely o vliv komplementů na elasticitu poptávky vybraných produktů.
- Porovnat přesnost původních modelů s novými, rozšířenými modely.

1.2 Přínosy práce

Hlavním přínosem práce je ověření hypotézy, že zapojením ceny komplementárních produktů lze docílit zpřesnění výpočtu poptávkových křivek, čímž lze zároveň zpřesnit výpočet optimální ceny, a ještě více tak zvětšit přínos, tedy maximalizovat tržby a zisk, který e-shopy

využitím tohoto řešení získají. Během rešerše blíže popsané v následující kapitole nebyl nalezen zdroj, který by se průzkumu vlivu komplementů v podobném kontextu algoritmické dynamické cenotvorby věnoval.

Dílčím přínosem spojeným s cílem analyzovat možnosti a postupy analýzy nákupního košíku je také detailně rozepsaný postup, jak lze tuto analýzu provést v jazyce R i s komentovanými výpisy kódu, včetně následné transformace dat s vydolovanými pravidly do podoby, ve které je lze dále použít.

Další přidanou hodnotou je i samotný seznam vydolovaných asociačních pravidel, tedy jeden z dílčích cílů této práce. Ten může provozovateli e-shopu poskytnout další informace o tom, které produkty jsou nakupovány společně, a je tedy možné využít tento seznam například pro lepší doporučování produktů zákazníkům na e-shopu či optimalizaci marketingových kampaní či slevových akcí.

1.3 Omezení a předpoklady práce

Práce předpokládá u čtenáře základní znalosti z oblasti ekonomické terminologie, nepopisuje tedy například rozdíl mezi obratem a ziskem či nemá za cíl rozebrat podrobně definici obchodní marže.

Dále předpokládá jistou vstupní znalost datové analýzy a zároveň základní znalosti z oblasti data miningu, jelikož se v teoretické části například věnuje pouze popisu asociačních pravidel či poissonovy regrese, které jsou v části praktické následně využívány, ale ne již například pojmům jako co jsou to data, databáze, či veškeré základní terminologii v této oblasti.

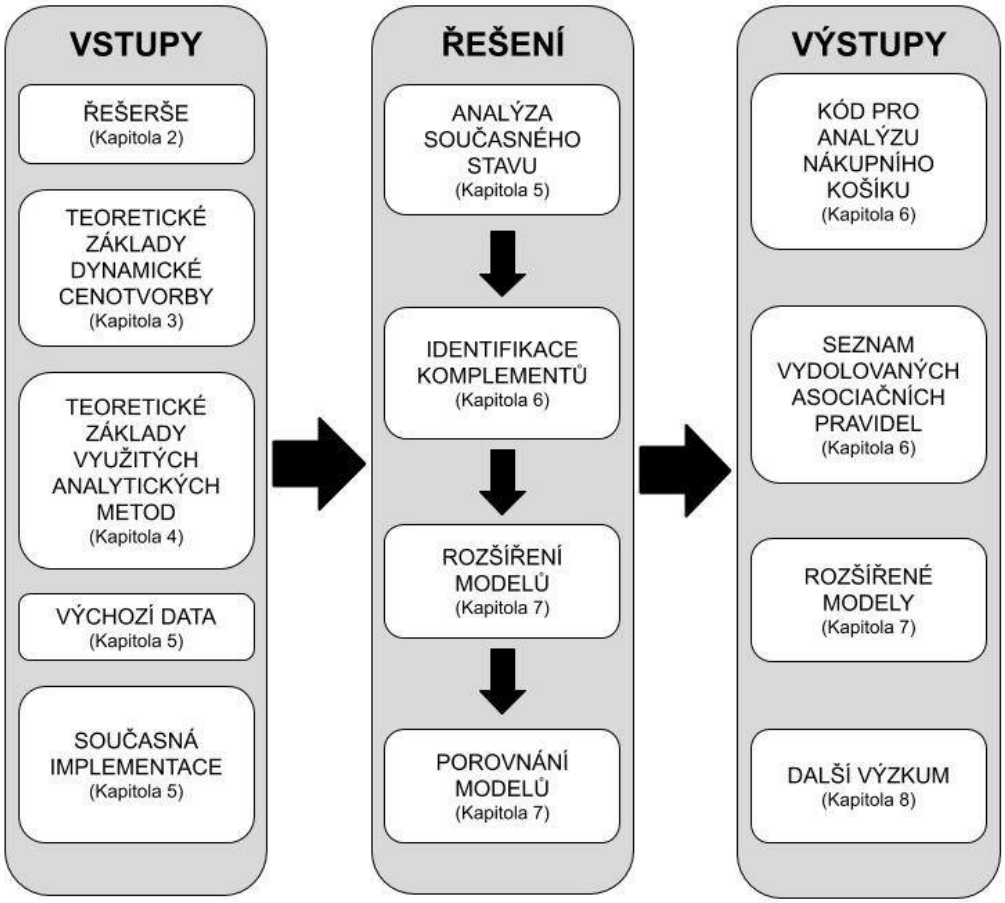
Práce si nedává za cíl ani popsat kompletní postup implementace modelů dynamické cenotvorby na libovolný e-shop, jelikož se jedná o rozsáhlé a složité téma, které je mimo rámec této práce a z důvodu heterogenního prostředí i datových modelů většiny e-shopů i poměrně složité na zobecnění. Soustředí se tedy na tu část celkového řešení, která je nezbytná pro dosažení cíle práce a demonstraci jejího přínosu.

V práci je zároveň hojně využíváno termínů komplement či komplementární produkt, ve spojení s produkty vydolovanými asociačními pravidly. Zde práce do jisté míry zjednodušuje fakt, že ne všechna vydolovaná asociační pravidla musí striktně splňovat ekonomickou definici komplementů a bylo by poměrně komplikované u všech vydolovaných pravidel ověřovat, zda je křížová elasticita poptávky skutečně záporná a naplňuje tak mikroekonomickou

definici tohoto pojmu, což je mimo rámec této diplomové práce. Nicméně v práci je na ně i tak obvykle odkazováno právě jako na komplementy či komplementární produkty, místo na možná přesnější označení „produkty, které zákazníci často nakupují společně“.

1.4 Struktura práce

Práce je pomyslně rozdělena na dva celky – teoretickou a praktickou část, kde teoretická uvádí do problematiky, používaných termínů, technologií a postupů, které jsou v praktické části využity k naplnění cíle rozšíření modelů dynamické cenotvorby a dosažení výstupů práce v podobě kódu pro vydolování asociačních pravidel, tedy identifikaci komplementů, rozšířených modelů a jejich porovnání s modely původními. Obecnou strukturu práce lze vidět na obrázku 1 níže, přičemž teoretická část představuje především vstupy práce, zatímco praktická pak část řešení a výstupů.



Obrázek 1: *Struktura práce (zdroj: autor)*

První kapitolou je tato úvodní, která uvádí čtenáře do tématu této práce, a následně definuje cíle, dílčí cíle, přínosy práce s důrazem na její praktickou část, dále také popisuje její omezení, předpoklady a strukturu.

Kapitola 2 se věnuje zkoumání současného stavu problematiky, tedy rešerši literatury a podrobněji popisuje nejvýznamnější články, publikace a vědecké práce, ze kterých práce čerpá. Je rozdělena do několika podkapitol podle typu zdrojů.

Kapitola 3 uvádí čtenáře do problematiky dynamické cenotvorby, stěžejního tématu této práce. Přibližuje krátce její historii, její využití v prostředí kamenných obchodů i e-commerce. Věnuje se zároveň identifikaci hlavních úskalí, které jsou s implementací dynamické cenotvorby spojeny nebo rozlišení pojmů personalizovaná a dynamická cenotvorba.

Kapitola 4 popisuje z teoretického hlediska metody analýzy, které jsou v této práci využity. První popisuje market basket analýzu, tedy analýzu nákupního košíku. Uvádí čtenáře do historie této analýzy a popisuje přidružený termín asociačních pravidel. Zároveň teoreticky

i prakticky na příkladech demonstruje klíčové metriky spolehlivosti a podpory. Obsahuje i popis apriori algoritmu, který je pro provedení této analýzy následně použit v praktické části práce. Následně poskytuje teoretický úvod do regresní analýzy a popisuje v praktické části využitou poissonovu regresi i prozkoumává možnosti, jak lze měřit kvalitu spočítaných modelů. Zde je i blíže popsáno MASE, které je hlavní metrikou, podle které je v kontextu této práce porovnávána kvalita mezi původními a rozšířenými modely. Nakonec též čtenáře seznamuje s problémem přeučení neboli overfittingu.

Kapitola 5 se již věnuje popisu současného stavu praktického řešení. Popisuje klienta, výchozí data a zároveň obsahuje bližší rozbor současného stavu implementace, včetně definice účelové funkce či obecného pohledu na celou logiku hledání optimalizované ceny od začátku do konce.

Kapitola 6 se věnuje analýze nákupního košíku a samotnému vydolovaní komplementárních produktů z dat o historii transakcí, které jsou popsané v předchozí kapitole. Obsahuje komentovaný kód v jazyce R, který byl použit pro implementaci, představuje možnosti následné analýzy vydolovaných pravidel skrze funkce i grafy i popis transformace vydolovaných pravidel do formátu vhodného pro rozšíření modelů.

V kapitole 7 již dochází k začlenění vydolovaných pravidel do stávajících modelů. Popisuje tedy přípravu vstupních dat nezbytných pro rozšířené modely i jejich samotné rozšíření o nové prediktory a následné přepočítání poptávkových křivek a analýzu dopadu tohoto rozšíření na jejich přesnost.

V kapitole 8 je prezentována diskuze nad výsledky práce, ve které jsou kromě konkrétních výsledků zároveň diskutovány i možnosti využití této práce. Zároveň jsou zde popsány případné oblasti, na které je nezbytné se zaměřit při nasazení tohoto řešení na jiného klienta.

Poslední kapitola je závěrem práce, kde jsou shrnuty výsledky práce. Je zde prezentováno splnění cílů a popsáno, jakých přínosů bylo dosaženo a jsou zde zmíněny oblasti, ve kterých je možné navázat dalším výzkumem a zlepšením dosavadních výsledků práce. Nastiňuje tak například možnost kromě komplementů začlenit do modelů též substituty, validace výsledků práce v ostrém provozu či nasazení na různé e-shopy s rychloobrátkovým zbožím a analyzovat vliv komplementů v různých segmentech.

2 Komentovaná rešerše informačních zdrojů

Tato kapitola práce obsahuje rešerši literatury. Rešerše je soustředěna na dvě hlavní témata, a to zdroje věnující se dynamické cenotvorbě jako takové a následně zdroje věnující se teorii kolem data miningu, které jsou využity při sestrojování jednotlivých modelů v praktické části. Dále je rešerše rozdělena podle povahy zdroje na publikace, odborné články a nakonec akademické práce.

2.1 Publikace

Kvůli aktuálnosti pramenů práce nečerpá z velkého množství knih a hlavním těžištěm zdrojů, ze kterých především teoretická část práce vychází, jsou odborné a internetové články či příspěvky v odborných časopisech. V určitých tématech však využití pouze nejnovějších pramenů není nutné, jelikož i dnešní metody vycházejí ze stejného teoretického základu a práce tedy cituje i mnohem tradičnější a starší zdroje.

I proto v teoretické části práce věnující se problematice data miningu, tato práce hojně čerpá z publikace *Dobývání znalostí z databází* od profesora Berky (2003). Ačkoliv se na poměry IT, kde dochází každoročně k velmi rychlému vývoji, může zdát, že jde o starší publikaci, metody, algoritmy a principy, které jsou v praktické části práce využity, stojí na stejných základech, jedná se tedy stále o relevantní literaturu. Kniha podrobně a názorně vysvětluje problematiku asociačních pravidel a algoritmu apriori, který je využíván v praktické části práce k analýze nákupního košíku, ale poskytuje například i užitečný úvod do tématu regresní analýzy, to jsou tedy kapitoly, které z této publikace čerpají především.

Důležitým knižním zdrojem této práce je také publikace *Data Mining* od Charu C. Aggarwala (2015). Ta se věnuje mnoha oblastem spojených s dolováním dat z databází a tato práce z ní čerpá zejména v teoretických kapitolách, které popisují asociační pravidla či regresní analýzu včetně zobecněných lineárních modelů. Mnoho z těchto témat práce nepopisuje zdaleka do takové hloubky a podrobností jako tato rozsáhlá publikace, v případě většího čtenářova zájmu o problematiku dolování dat z databází se tedy jedná o užitečný zdroj dalších informací.

2.2 Odborné články

Relevantních odborných článků na téma dynamické cenotvorby lze najít celou řadu, mnoho se jich věnuje i konkrétně oblasti e-commerce, nejčastěji ve spojitosti s největšími e-shopy, jako je Amazon, který dynamickou cenotvorbu využívá již delší dobu.

Druhou oblastí, na kterou se tato rešerše zaměřuje je oblast metod data miningu, které jsou využity v praktické části práce pro implementaci samotné dynamické cenotvorby. Těch je samozřejmě obrovské množství, jelikož se především v poslední době jedná o velice často skloňované téma, rešerše se tedy soustředí pouze články věnující se konkrétním metodám, které jsou v praktické části práce využity.

Tabulka 1: *Rešerše – odborné články*

Název publikace	Popis publikace
An Empirical Analysis of Algorithmic Pricing on Amazon Marketplace (Chen a další, 2016)	Článek se věnuje analýze algoritmického oceňování na Amazon Marketplace. Autoři prezentují metodologii, jak dynamickou cenotvorbu odhalit a zkoumají její chování.
How Retailers Use Personalized Prices to Test What You're Willing to Pay (Rafi, 2017)	Článek se podrobně věnuje problematice personalizované cenotvorby
Support vs Confidence in Association Rule Algorithms (Lai a Cerpa, 2001)	Článek poskytuje dobré názorné příklady výpočtu podpory a spolehlivosti, které jsou v práci částečně využity pro lepší ilustraci.
Another Look at Measures of Forecast Accuracy (Hyndmand a Koehler, 2006)	Návrh nové metriky pro měření chyby odhadu modelů mean absolute scaled error (MASE) a porovnání s vybranými existujícími metrikami

Prvním příkladem je článek *An Empirical Analysis of Algorithmic Pricing on Amazon Marketplace* od Chena a dalších (2016), který blíže zkoumá algoritmickou cenotvorbu v kontextu Amazon Marketplace. Jedná se tak o jeden ze zdrojů pro kapitolu, která do problematiky algoritmicky řešené dynamické cenotvorby čtenáře uvádí.

Dalším článkem je *How Retailers Use Personalized Prices to Test What You're Willing to Pay* od Mohammeda Rafiho (2017), který podrobně rozebírá problematiku cenotvorby personalizované, je z něj tedy čerpáno především v podkapitole 3.2, která se personalizované cenotvorbě věnuje a vůči dynamické cenotvorbě tento termín vymezuje.

Článek Laie a Cerpy *Support vs Confidence in Association Rule Algorithms* z roku 2001 je obdobně jako v publikace profesora Berky z předchozí podkapitoly již poměrně starým zdrojem, stejně jako v případě výše zmíněné knihy však slouží pro lepší ilustraci spolehlivosti a podpory asociačních pravidel na ukázkových datech v kapitole 4.2.

V článku *Another Look at Measures of Forecast Accuracy* od Hyndmana a Koehlerové (2006) byla poprvé představena metrika mean absolute scaled error neboli MASE. Ta je stěžejní pro porovnání přesnosti modelů v praktické části práce, z článku je tedy hojně citováno především v kapitole 4.6, věnující se různým metrikám hodnocení přesnosti modelů a jejich specifickým problémům.

2.3 Akademické práce

Akademických prací, které by se zabývaly přímo dynamickou cenotvorbou lze najít několik, nicméně ty se jí většinou věnují v kontextu hotelnictví či leteckých společností, tedy odvětví, kde má tato praktika mnohem větší tradici. Mezi takové práce patří například práce *Beyond customer perception of price discrimination: A consumer behavior analysis and its implications on aviation revenue management* od Katariny Kusch (2017), napsaná na Fakultě mezinárodních vztahů VŠE. Ta se ve své práci, jak již název napovídá, věnuje právě leteckému odvětví a zkoumá mimo jiné, jak spotřebitelé vnímají cenovou diskriminaci. Ačkoliv se věnuje jinému odvětví, jsou v práci zdroje i některá zjištění, které lze využít i pro zkoumání dynamické cenotvorby v prostředí e-commerce, jako například závěr o přesunu důvěry od značek k jednotlivcům, což je třeba mít při dynamickém přeceňování na paměti.

Naopak práce, které se více zabývají e-commerce či retailem obecně se dynamické cenotvorbě věnují obvykle velmi okrajově. Příkladem takové práce je *Business intelligence application in online retail* od Petra Halouna (2017). Ve své práci se věnuje využití business intelligence právě v oblasti e-commerce a zmiňuje pak i její aplikaci při dynamické cenotvorbě. V jeho praktickém řešení pak ale nechává samotné změny v cenách na obchodníkovi, který tak činí na základě dat získaných z business intelligence řešení. Cílem této práce je

však tento proces automatizovat a přenechat na rozhodnutí algoritmu, spíše než nutnosti manuálního rozhodování obchodníka v každém konkrétním případě.

3 Dynamická cenotvorba

Následující kapitola se věnuje tématu dynamické cenotvorby. Tato kapitola tento termín definuje, popisuje jeho vznik a historii, klíčové výhody, které jeho implementace přináší a na druhé straně také identifkuje hlavní problémy, které s dynamickou cenotvorbou mohou být spojeny. Nakonec definuje též příbuzný termín personalizované cenotvorby a vysvětluje hlavní rozdíly mezi těmito často zaměňovanými termíny.

Účtovat si za stejný produkt od různých zákazníků v různém čase různou cenu není při pohledu do historie žádnou novinkou. Právě naopak, spíše fenomén cen fixních je poměrně novou záležitostí, která přišla až po průmyslové revoluci a velkovýrobě (Sahay, 2007). V době dřívější bylo běžné, že každý zákazník zaplatil tolik, kolik si sám usmlouval. Z části lze podobně pohlížet na nastupující trend i dnes, akorát zde již nesmlouváme svými vyjednávacími schopnostmi jako dříve, ale spíše svým předchozím chováním, respektive daty o něm nasbíranými obchodníkem, časem, kdy se rozhodneme nakoupit a podobně.

Před samotným popisem dynamické cenotvorby je třeba zmínit nadřazený termín „revenue management“. Jedná se o termín, který zastřešuje celou řadu taktik, které zahrnují mimo jiné i různé přístupy k cenotvorbě, ale i další metody jako segmentace, správné předpovídání poptávky a podobně. Rozvinuly se především jako snaha reagovat na poptávku v případě „dočasných“ produktů, tedy například letenek, hotelů či vstupenek na koncerty a jiné akce. Jelikož například prodat letenku lze pouze do vzletu letadla a poté je již naprosto bezcenná, revenue management se snaží prodat v pravý čas ten správný produkt tomu správnému zákazníkovi (Baker, 2018).

Dynamická cenotvorba je pak právě jedním z přístupů, které lze v rámci revenue managementu využít k optimalizaci poptávky a umožňuje v závislosti na měnícím se vývoji poptávky ze strany zákazníků prodávat například letenky či lístky za různé ceny tak, aby došlo k maximalizaci revenue – tedy tržeb.

Dramatický růst trendu, kdy jsou ceny nastavovány počítačovými algoritmy i v dalších odvětvích mimo letecké společnosti a hotely, které s tímto přístupem začaly, byl umožněn až nedávné době masivním růstem e-commerce. Zatímco celkové tržby v odvětví maloobchodu rostly v posledních letech v řádu jednotek procent (Novotný, 2018), tržby na internetových obchodech rostly dvouciferným tempem, a tak získávají z celkových tržeb stále větší podíl.

Ten má v roce 2018 již přesáhnout 11 % (Dostál, 2018). Česká republika bývá často označována za jakousi e-commerce velmoc, kde se tato oblast rozvíjí velkým tempem a v roce 2018 tak mohli tuzemští zákazníci nakoupit již zhruba na 42 000 e-shopech (Uďan, 2018), potenciální trh dynamické cenotvorby v e-commerce je tedy značný, a to i v případě omezení se pouze na Českou republiku.

Právě v prostředí internetových obchodů se teprve stává dynamická cenotvorba opravdu praktickou. V tradičním maloobchodním odvětví totiž dynamické přeceňování naráží na řadu problémů. Chen a další (2016) zmiňují například chybějící data, především pak ta o cenách konkurence i třeba problémy s neustálým manuálním přelepováním cenovek v kamenném obchodě. Přelepování by sice šlo nahradit jednoduchými obrazovkami, kde je možné cenu libovolně nastavovat automaticky, a v mnohých řetězcích se tak i děje (Horáček, 2018), nicméně i s tím je pochopitelně spojena investice a usnadnil to teprve nedávný rozmach internetu věcí, umožňující snadnější centrální správu takových zařízení. V kontrastu s tím lze v prostředí internetového obchodu snadno data o konkurenci strojově sbírat a analyzovat je v reálném čase. Stejně tak lze v okamžiku přecenit třeba celý katalog zboží bez výrazného zvýšení provozních nákladů.

3.1 Hlavní problémy dynamické cenotvorby

Ačkoliv má dynamická cenotvorba mnoho potenciálu pro oslovení více zákazníků a dosažení vyšších tržeb a zisků, jsou s ní zároveň spojeny některé problémy a výzvy, ke kterým je třeba také přihlédnout a které následující kapitola adresuje.

3.1.1 Snížení marže

Především v případě nastavování ceny podle konkurence může při vidině zvýšených objemů prodaného zboží a snaze o maximalizaci tržeb model začít zlevňovat a snižovat tak obchodníkovu marži. To nemusí nutně představovat problém, ale při nastavení jednodušších modelů, aby upravovaly cenu podle konkurence, může dojít ke zlevňování až za únosnou mez, případně dokonce do marže záporné. Vždy je tedy třeba mít na paměti správné nastavení modelu a případnou definici mantinelů, mezi kterými by se měl model se svým cenovým návrhem pohybovat a nenechat mu neomezenou volnost (Prisync, 2018).

V praktické části práce je tento problém adresován kromě mnoha kontrol nad vypočtenou cenou i parametrem agresivity, pomocí kterého lze nastavit, jestli je cílem implementované

dynamické cenotvorby maximalizovat hrubou marži, celkový obrat, či určitou kombinaci obou. Model pak následně nastavuje ceny odpovídající zadání. Bližší popis fungování tohoto parametru lze najít v kapitole 5.

3.1.2 Neočekávané interakce

Jedním z problémů přenechání procesu cenotvorby na algoritmu je jeho původně neočekávané interakce či potenciální technické problémy. Je sice samozřejmě možné implementovat dynamickou cenotvorbu zcela manuálně a svépomocí, v případě katalogu čítajícího již několik stovek položek se však jedná o velmi nepraktické řešení a v případě těch největších e-shopů pak o řešení zcela nesmyslné a nereálné. Abychom od upravování cen skutečně získali očekávané výhody, je potřeba implementovat automatizaci algoritmem alespoň do nějaké míry. Ten však může v závislosti na použité technologii i složitosti použitého modelu vykazovat chování, které jsme na začátku nepředpokládali.

Je například možné, že algoritmus produkt ocení zcela nesmyslně, jako se stalo například s vědeckou publikací o mouchách na Amazonu, která byla v jednu chvíli na prodej za neuvěřitelných 23 milionů dolarů plus poštovné, kvůli neočekávané interakci mezi dvěma modely na dynamické oceňování, které mezi sebou začaly cenovou válku a neustále přeceňovaly jako násobek ceny toho druhého (Sutter, 2011).

Pochopitelně však není třeba jít do takového extrému, a naopak ještě horší může být případ, kdy model sice nenacení zboží zcela nesmyslně, nicméně přesto neoptimálně, a chyba pak může být velmi těžko identifikovatelná, případně místo extrémně vysoké ceny nastaví extrémně nízkou, čímž lze napáchat ještě větší škody i za krátkou dobu, než si takového problému provozovatel všimne.

Praktická část práce adresuje i tento problém, a to řadou kontrol, o kterých se blíže zmiňuje opět kapitola 5.

3.1.3 Datová kvalita

Dalším potenciálním problémem je datová kvalita – v momentě kdy je rozhodování o ceně přenecháno algoritmu, který tak činí na základě získaných dat, je nezbytné dbát na jejich kvalitu, pokud mají být výsledné ceny správné. Problém může být především v aktuálnosti dat v případě, že má model přeceňovat produkty velmi často, tedy v řádu hodin nebo i minut. Pokud takto aktuální data nelze jednoduše získat, nemá smysl ceny měnit v takové frekvenci.

Ačkoliv se ekosystém poskytovatelů dat i technologie k jejich rychlému zpracování i využití neustále vylepšují, nelze nedostatečnou kvalitu dat nikdy zcela vyloučit a je třeba ji neustále sledovat (Lake, 2017). V závislosti na tom, odkud data získáváme, a jaké transformace jsou na nich následně prováděny, může totiž i drobná změna v jejich schématu či struktuře zcela vyřadit systém přeceňování z provozu.

K natrénování modelů v prostředí e-commerce jsou navíc obvykle potřeba historická data a z důvodu sezónnosti ideálně za období několika let, jejich dostupnost však pochopitelně není vůbec samozřejmá (e-shop nemusí vůbec tak dlouho na trhu působit, či nesbírá nebo neuchovává data za tak dlouhou historii). V případě, že taková data k dispozici nemáme, budou během roku nastávat období, kdy model nebude schopen spočítat optimální cenu, typicky v obdobích kolem Vánoc, kdy je poptávka ve většině odvětví dramaticky odlišná od zbytku roku.

Na druhou stranu je také nutné vzít v potaz, že v průběhu existence daného e-shopu na trhu se pravděpodobně vyvíjela jeho popularita a celkový obrat, což je také do modelu důležité začlenit, jinak data stará několik let, kdy mohl mít e-shop například řádově nižší obrat, mohou vyústit v nepřesné modely a neoptimální nastavování cen.

3.1.4 Zákaznická nespokojenost

Pro navýšení marže na produktech lze využít dvou postupů, a to snížení nákladů na kus či zvýšení výnosů na kus. V případě zvyšování marže pomocí dynamické cenotvorby se vydáváme druhou cestou, zákazník si tedy musí připlatit za produkt, který by před využitím takového algoritmu koupil levněji. Takové srovnání je samozřejmě spíše hypotetické, když je již model implementovaný, a tedy cenu produktu před přeceněním může zákazník s aktuální cenou těžko porovnat. Zákazníci však mohou špatně nést i příliš časté úpravy cen, kterých si všimnou sami, případně odpozorují z grafu historie cen některého srovnávače, či náhlé prudké zdražení při zvýšené poptávce. V dnešní době sociálních sítí se pak může poměrně jednoduše stát, že zákazník zjistí, jak jeho známý ve stejném obchodě za stejnou věc zaplatil jinou cenu, jen proto, že nakoupil o den dříve, což může snadno vytvořit negativní publicitu.

Vzhledem k tomu, že loajalita zákazníků a její zachování je především v e-commerce prostředí, kde jsou retence a customer lifetime value bedlivě sledovány, velmi důležitý faktor, rozhněvání věrných zákazníků je vysoce nežádoucí a je potřeba i na něj při nastavování mo-

delu pro cenotvorbu pamatovat. V závislosti na elasticitě poptávky zároveň při zdražení mohou i loajální zákazníci náhle přejít ke konkurenci a konečný efekt může být záporný (Prisync, 2018).

S tím, jak bude více a více využíváno dynamického přeceňování v reálném čase je také možnost, že zákazníci začnou více spekulovat a případně čekat s nákupem s vidinou zmenšení ceny. Jelikož čím více je nákup odkládaný, tím se zvyšuje šance, že jej zákazník nakonec neprovede vůbec, je možné, že v některých případech můžeme v důsledku podobných spekulací o některé nákupy přijít (Lake, 2017).

Lake (2017) zároveň zmiňuje, že velmi časté změny cen mohou v zákaznících vyvolat obavu, že pokud nakoupí zrovna teď, zaplatí více jen na základě například denní doby. V případě nedostatečné transparentnosti, kdy někteří zákazníci nemusí chápat, co se děje, může opět dojít ke ztrátě jejich důvěry.

3.1.5 Legalita a etika

Ačkoliv se může mnoho lidí domnívat, že účtovat různým zákazníkům různé ceny není legální, ve valné většině případů se mýlí. Obvykle je ilegální pouze rozlišování na základě faktorů jako rasa, národnost, náboženství či pohlaví, nebo pokud cena porušuje například antimonopolní zákony (Coble, 2015). Právě poslední ze zmíněných může představovat problém v momentě, kdy je tak snadné upravovat cenu podle konkurence, jak dokazuje například kauza, ve které došlo mezi dvěma prodávajícími k dohodě o zafixování cen vůči sobě na platformě Amazon Marketplace a vyústila až v soudní řízení a vysoké pokuty (Stambor, 2015).

Dalším potenciálním problémem, který při využití dynamické cenotvorby vyvstává, je problém s etikou takových praktik. Podle průzkumu Retail System Research 71 % spotřebitelů tento přístup oceňování nijak zvlášť nevítá. V generaci mileniálů jsou tato čísla poněkud optimističtější, 14 % mladých spotřebitelů tuto myšlenku dokonce kvituje s nadšením, ani tam však stále nejde o většinovou podporu (Walker, 2017). Jak Walker v článku potvrzuje, důležitá je pro zákazníky v tomto ohledu transparentnost. V některých případech, například pokud zbývá poslední kus na skladě a zákazník jej nutně potřebuje, nemá obvykle se zaplacením vyšší ceny problém. Podle průzkumů Retail Systems Research (2017) vidělo v dynamické cenotvorbě příležitost 28 % respondentů z řad obchodníků v roce 2016 a dokonce

pouze 22 % v roce následujícím, jako důvod průzkum zmiňuje právě strach z reakce zákazníků.

Nesprávné užití dynamické cenotvorby neetickým, případně necitlivým způsobem může snadno způsobit vlnu negativní publicity, která v konečném důsledku může znamenat, že využití této metody přinese více problémů než užitku. Jako příklad může posloužit aféra Sony Music, které na internetovém obchodě s hudbou iTunes po smrti Whitney Houston zdražilo její album Ultimate Collection o 60 % ze 4,99 $ na 7,99 $ v očekávání zvýšené poptávky. Ačkoliv zvýšení poptávky skutečně přišlo a v tomto případě se jednalo o logický krok směrem ke zvýšení tržeb, zákazníci zdražení v takové chvíli vnímali jako snahu o vyšší výdělek za každou cenu a necitlivé parazitování na smutné události (Hiotis, 2018).

Je tedy nezbytné mít na paměti, že důvěra zákazníků je stále jedním z nejdůležitějších faktorů, a i dynamickou cenotvorbu nastavovat citlivě a eticky, i když to nemusí vždy znamenat optimální nastavení.

3.2 Personalizovaná cenotvorba

Příbuzným termínem dynamické cenotvorby je cenotvorba personalizovaná. Protože jsou občas tyto termíny v některých publikacích a článcích zaměňovány, ačkoliv nejde o totéž, tato podkapitola vysvětluje jejich rozdíly.

Zatímco v případě dynamické cenotvorby se bavíme o upravování ceny na základě proměnných, které nesouvisejí se zákazníkem, jako je denní doba, venkovní teplota, nabídka a poptávka či ceny u konkurence a ve stejný čas vidí všichni zákazníci stejnou cenu, cenotvorba personalizovaná je přesný opak, tedy cena se v tomto případě určuje podle charakteristik a chování zákazníka. Může se tak stát, že loajální zákazník, který pro prodejce představuje vysokou hodnotu, může dostat výrazně jinou cenu než úplně nový zákazník, který právě navštívil daný e-shop poprvé (Retail Systems Research, 2017).

Častá záměna pak pravděpodobně přichází z důvodu podobností mezi oběma přístupy v některých charakteristikách. Oba přístupy k cenotvorbě totiž vyžadují velké množství real-time dat, mohou často používat velmi podobné algoritmy a lišit se tak jen v povaze dat, kterou do nich posíláme (Retail Systems Research, 2017).

Personalizace obsahu na webech není ničím novým a mnoho firem ji již dlouhou dobu používá pro vylepšení uživatelského zážitku při návštěvě svých stránek a s tím obvykle spojeném zvýšení svých zisků. Díky personalizovaným doporučením například na stránkách jako je YouTube či Netflix uživatelé tráví mnohem více času, kdy v případě relevantních doporučení klikají na stále další obsah. Právě Netflix byl v jistém smyslu velkým katalyzátorem rozvoje algoritmů pro personalizaci obsahu, a to vyhlášením „Netflix price“ v roce 2006. Tehdy nabídl jeden milion dolarů tomu, kdo by dokázal vylepšit tehdejší algoritmus, který byl v Netflixu využíván, alespoň o 10 %. Co se může zdát jako poměrně jednoduchý úkol, nakonec trvalo téměř tři roky a vyústilo v mnoho nových algoritmů a postupů, na kterých mnohdy personalizační řešení staví dodnes (Jackson, 2017).

Na rozdíl od personalizace obsahu však personalizace cen stále mimo oblast prodeje letenek či hotelů tak hojně rozšířená není, a to nejspíš především z důvodu strachu z reakce zákazníků při použití mimo tato odvětví, ve kterých se jedná už o poměrně tradiční přístup (Haucap a další, 2018).

Ani v případě cenotvorby personalizované však nutně nemusí být využito žádných osobních či citlivých údajů zákazníka, stačí jen párovat jeho návštěvy pomocí anonymizovaných údajů, jako jsou například cookies, a vytvořit tak vzorce jeho chování a pomocí těch následně odhadnout jeho hodnotu, případně cenu, kterou by byl ochoten zaplatit. Jak již však bylo zmíněno výše, s personalizovanou cenotvorbou jsou spojeny etické problémy či negativní náhled zákazníků ještě více než s cenotvorbou dynamickou, pokud není využita s dostatečnou opatrností (Walker, 2017).

Mezi příklady přinejmenším kontroverzního využití personalizované cenotvorby lze zmínit například agregátor cen Orbitz, u kterého se v roce 2012 přišlo na to, že upravuje ceny pro uživatele Apple počítačů Mac, protože byli podle jejich odhadu ochotni zaplatit až o 30 % více za hotelový pokoj (Walker, 2017). Ve stejném roce pak například vyšlo najevo, že obchod s kancelářskými potřebami Staples nastavoval na webu ceny podle blízkosti uživatele ke konkurenčnímu obchodu na základě ZIP kódu, kdy zákazníkům, kteří konkurenci měli blíže, nabízel výhodnější ceny, aby je od konkurence přetáhl k sobě (Walker, 2017).

Hrozbou provinění minimálně vůči etice pak může být, když při detailním nastavení personalizovaného přeceňování podle různých demografických údajů, které má obchodník k dispozici, dojde jako vedlejší efekt k diskriminaci podle jedné ze zakázaných charakteristik, jako je například rasa. Podle průzkumu ProPublica se tak například stalo, že cenová strategie

Princeton Review vedla k tomu, že u asijských zákazníků byla dvojnásobná pravděpodobnost naúčtování vyšší ceny (Mohammed, 2017). Ačkoliv to původně jistě nebyl záměr, v důsledku velmi detailní segmentace na základě například geografické dimenze, se může stát, že algoritmus vyhodnotí, že obyvatelé konkrétní čtvrti jsou ochotni zaplatit za produkt více. Pokud je pak tato čtvrť výrazně více obývaná příslušníky některé národnostní či náboženské skupiny, dojde neúmyslně k cenové diskriminaci i na základě zakázaných faktorů zmíněných výše v předchozí podkapitole. Ačkoliv pravděpodobně nebude snadné napadnout takovou skutečnost soudně, může způsobit negativní publicitu či vnímaní ze strany zákazníků a mít tak záporné finanční implikace i tak.

Personalizovaná cenotvorba je ve své podstatě založena na asymetrii informací, kdy nakupující neví, co o něm obchodník má všechno za informace a ten se snaží vytvořit takové prostředí, aby nakupující věřil, že to, co vidí, je tržní cena. Na druhou stranu je nutné počítat také s tím, že spotřebitelé jsou dnes více informovaní než kdykoliv v historii, takže jakákoliv taktika, která je založena na jejich oklamávání, pravděpodobně dlouhodobě úspěšná nebude (Walker, 2017). Navíc jak ve své práci zmiňuje Katharina Kusch (2017), důvěra zákazníků se postupně přesouvá od značek k jednotlivcům, především pak v poslední době často skloňovaným „influencerům“, tedy vlivným lidem s početným zástupem sledujících na svých profilech na sociálních sítích. V dnešní době velmi jednoduchého šíření jakýchkoliv negativních zkušeností spojených s nákupem skrze sociální sítě je tedy opatrnost v tomto směru pro dlouhodobý úspěch naprosto kritická.

3.3 Shrnutí kapitoly

Tato kapitola se věnovala termínu dynamické cenotvorby jako takovému. Na začátku termín definovala jako součást revenue managementu, při které je cena upravována na základě vývoje zákaznické poptávky pro dosažení maximálních tržeb. Popsala též jeho historii, která je úzce spojena s odvětvími dočasných produktů, jako jsou letenky či ubytování, ze kterých v současnosti silně proniká i do prostředí e-commerce.

Identifikovala také hlavní potenciální problémy, které jsou s využitím dynamické cenotvorby spojeny, tedy například problémy s datovou kvalitou, špatným nastavením a neočekáváným chováním algoritmu či legalitou a etikou. Důležitou částí je nakonec odlišení termínů dynamická cenotvorba a personalizovaná cenotvorba, které bývají často zaměňo-

vány, ale ve skutečnosti se nejedná o synonyma. Ačkoliv mohou stát na podobném technologickém základu a obě řešení vyžadují množství dat, v případě dynamické cenotvorby všichni zákazníci v jeden čas vidí stejnou cenu a k jejímu výpočtu, v kontrastu s personalizovanou cenotvorbou, je využito proměnných, které nejsou vázané na konkrétního zákazníka.

Z této kapitoly si čtenář může odnést především lepší pochopení obecné logiky a přínosu řešení, které je v praktické části práce představeno a zároveň nad jakými problémy je nutné se zamyslet před jeho implementací.

4 Využité metody analýzy

Následující kapitola se věnuje hlavním metodám analýzy, které jsou v praktické části využity, a to jak při hledání komplementů pro rozšíření stávajících modelů v podkapitole o analýze nákupního košíku, tak v případě implementace konkrétních modelů a měření jejich kvality v podkapitole regresní analýzy. Seznamuje tak čtenáře s teoretickými základy nezbytnými pro pochopení technické části implementace dynamické cenotvorby v praktické části práce.

4.1 Analýza nákupního košíku

Důležitou data mining technikou, která byla využita pro naplnění dílčího cíle této práce, tedy rozšíření existujících modelů o vliv komplementů na cenovou elasticitu, je analýza nákupního košíku neboli market basket analysis. Právě pomocí této data mining metody byly v praktické části identifikovány jednotlivé komplementy produktů, které jsou poté do modelů začleněny, následující kapitola tedy tuto techniku popisuje na teoretické úrovni.

Analýza nákupního košíku pomocí asociačních pravidel je klíčovou metodou, kterou dnes většina větších obchodů využívá pro vydolování asociací mezi prodávanými produkty a identifikaci příbuzných či společně nakupovaných předmětů. Samotný termín asociační pravidla byl zpopularizován počátkem 90. let Agrawalem, a to právě v souvislosti s analýzou nákupního košíku (Berka, 2003).

Analýza nákupního košíku a hledání asociačních pravidel pro identifikaci produktů kupovaných zároveň tedy skutečně není žádnou novinkou. Dobrým příkladem právě ze začátku 90. let je historka o pokusech analyzovat data z věrnostních kartiček zákazníků a pokladen obchodních domů, aby bylo možno je zkombinovat a sledovat, co který zákazník nakupoval. Některé kombinace, které z analýzy vyplynuly, byly relativně triviální (jako například častý nákup ginu společně s tonikem a citrony), ale vyplynula i jedna korelace, která na první pohled vypadala překvapivě, a která se stala základem této poměrně často skloňované historky z analýzy nákupního košíku. Mladí otcové často v pátek odpoledne nakupovali společně pivo a pleny (Whitehorn, 2006).

Celá historka, která pravděpodobně zahrnuje i domyšlené části, se však zakládá, alespoň zčásti, na pravdě – Thomas Blischok, který byl v roce 1992 manažerem v Teradata vydoloval z dat sítě lékáren Osco Drug, že mezi 17 a 19 hodinou si zákazníci často nakupovali společně pivo a pleny. Dodatečná korelace s věkem či pohlavím pravděpodobně zkoumána

vůbec nebyla, ale přesto tato analýza mohla posloužit ke snadnému zvýšení prodejů piva přesunutím regálů blíže k plenám, aby na ně při nakupování zákazníci narazili a nechali se snadněji zlákat (Whitehorn, 2006).

Tento dnes již dost starý příklad dobře ilustruje důležitost analýzy nákupního košíku zákazníků pro navýšení prodejů v kamenných obchodech a zároveň fakt, že se již dávno nejedná o zrovna přelomovou myšlenku. Dnes řetězce téměř vždy nabízejí věrnostní či jiné zákaznické karty, a to obvykle právě proto, že slouží pro sběr dat do následných analýz prodejů, optimální umístění zboží v prodejně a plánování slevových akcí či zásob.

Nezanedbatelný přínos má nicméně tato analýza i v prostředí e-commerce. Právě v online prostředí je shromažďování dat o mnoho jednodušší než v případě klasického kamenného obchodu a stejně tak je obvykle snadnější i následná implementace zlepšení, která z analýzy vyplynou. Využití analýzy nákupního košíku je nejčastěji ve své podstatě podobné, jako v případě kamenných obchodů – pokud známe, které produkty zákazníci nakupují současně, můžeme je na e-shopu návštěvníkům při prohlížení zobrazovat u sebe, či je doporučovat společně. Nejčastěji se tak děje prostřednictvím výběru několika souvisejících produktů při přihazování zboží do košíku, či přímo v košíku před dokončením objednávky, u kterých je napsáno, že zákazníci, kteří zakoupili tyto produkty, také zakoupili níže zobrazené. Tak lze zvýšit šanci na cross-sell, neboli křížový prodej, tedy že do košíku přihodí zákazník i nabídnutý související produkt. Především pokud je doporučený produkt skutečně k některé z položek v košíku relevantní, lze docílit poměrně snadného zvýšení prodejů. Zároveň lze tato zjištění využít v reklamních kampaních, například zlevňovat produkty společně, či slevu podmínit zakoupením též některého ze souvisejících produktů.

Některé e-shopy pak v tomto nabízení jdou ještě dále a zákazníkům zboží do košíku přihazují automaticky rovnou. Tato praktika se nicméně často s nadšením u nakupujících nesetkává a v předvánočním období 2018 dokonce proběhla médii kauza, ve které Alza.cz byla za případy takového chování z jara 2018 pokutována, jelikož se jednalo o přihazování nevyžádaného zboží. Zákazníci si často těsně před zaplacením přidané položky v košíku nevšimli, navzdory jejímu zelenému zvýraznění. Ačkoliv krajský inspektorát ČOI vyhodnotil toto chování za porušení zákona, Alza se ohradila a s praktikou přesto v době předvánočních prodejů pokračovala (Wolf, 2018).

Analýza dat metodou market basket analysis tedy již dnes velmi výrazně nakupování na internetu ovlivňuje.

4.2 Asociační pravidla

Následující podkapitola se podrobněji věnuje teorii asociačních pravidel, na jejichž principu analýza nákupního košíku obvykle funguje.

Jak již bylo zmíněno v kapitole výše, asociační pravidla jsou užitečná k hledání vzájemných vazeb v datech, jsou tedy velice užitečnou technikou při hledání vztahů mezi produkty při nakupování. Berka (2003) tvar pravidla popisuje následovně:

$$\text{Ant} \Rightarrow \text{Suc},$$

kde Ant na levé straně vzorce výše představuje antecedent, tedy předpoklad, a Suc na straně pravé sukcedent, tedy závěr, který z předpokladu vyplývá.

Tabulka 2: *Kontingenční tabulka asociačních pravidel (Zdroj: Berka, 2003), upr. autorem*

	Suc	**¬Suc**
Ant	a	b
¬Ant	c	d

Tabulka 2 výše poskytuje základ pro výpočet několika důležitých charakteristik, kterými můžeme jednotlivá pravidla ohodnotit. Těmi základními, které jsou následně využity pro výběr vhodných pravidel v praktické části této práce, je *podpora (support)* a *spolehlivost (confidence)* (Berka, 2003), jsou tedy podrobněji popsány níže.

4.2.1 Podpora

Podpora je relativní počet objektů, které splňují předpoklad i závěr (Berka, 2003), tedy:

$$\mathrm{P}(\mathrm{Ant} \wedge \mathrm{Suc}) = \frac{\mathrm{a}}{\mathrm{a} + \mathrm{b} + \mathrm{c} + \mathrm{d}}.$$

Prakticky lze výpočet podpory názorně demonstrovat pomocí tabulky níže z článku Laie a Cerpy (2001).

Tabulka 3: *Ukázková data pro demonstraci výpočtu podpory a spolehlivosti (Zdroj: Lai a Cerpa, 2001), upr. autorem*

ID	Položky
1	ABC
2	ABD
3	BC
4	AC
5	BCD

Tabulka 3 obsahuje celkově 5 transakcí. Pro lepší ilustraci v kontextu této diplomové práce si lze každý řádek představit jako jeden nákupní košík a každé písmeno ve sloupci položky jako jeden zakoupený produkt. Poté je podpora pravidla $A \Rightarrow B$, tedy že nákup produktu A vede nákupu produktu B, 2 / 5, tedy 40 %, protože se kombinace AB vyskytuje na prvních dvou transakcích z celkových pěti. Podpora pravidla $B \Rightarrow C$ by analogicky byla 3 / 5, tedy 60 %, protože se kombinace BC vyskytuje v první, třetí a poslední transakci.

Při následném generování pravidel jsou typicky v úvahu vzata pouze ta, jež splňují uživatelem zadanou hodnotu minimální podpory. Ta je v praktické části této práce vzhledem k rozsahu katalogu produktů, na kterém je analýza nákupního košíku prováděna, nastavena řádově níž, na jednotky až desetiny procenta, jelikož v rozsáhlejším produktovém katalogu budou zřídkakdy existovat produkty, které by byly součástí například nadpolovičního počtu objednávek, jako v ilustračním příkladu výše. Toto nastavení je důležité, protože pravidla s velice nízkou podporou mohou být pouze náhody či výjimky, které nevypovídají o skutečném chování spotřebitele, případně po jejich identifikaci stejně nevedou k žádnému přínosu, jelikož je možné je uplatnit pouze velmi výjimečně.

4.2.2 Spolehlivost

Druhou důležitou charakteristikou vydolovaných pravidel je spolehlivost. Tu lze definovat jako podmíněnou pravděpodobnost závěrů, pokud platí předpoklad (Berka, 2003) tedy:

$$P(\text{Suc} \mid \text{Ant}) = \frac{a}{a + b}.$$

K ilustraci výpočtu spolehlivosti lze využít ukázkových dat transakcí z tabulky 2. Například spolehlivost pravidla A ⇒ B je 2 / 3, tedy zhruba 67 %, jelikož se produkt A vyskytuje v transakci 1, 2 a 4 a z toho dvakrát (v prvních dvou transakcích) v kombinaci s produktem B. Analogicky spolehlivost pravidla B ⇒ C je 3 / 4, tedy 75 %, jelikož produkt B se vyskytuje ve všech transakcích kromě jedné a z toho ve třech v kombinaci s produktem C.

I v případě spolehlivosti uživatel při generování nastavuje zvolenou hodnotu minimální spolehlivosti a v úvahu jsou vzata pouze ta pravidla, která ji mají vyšší než tuto stanovenou mez. Ta se v případě analýzy nákupního košíku obvykle pohybuje zhruba kolem 80 %, což reflektuje i defaultní nastavení balíčku využitého v praktické části práce, její optimální nastavení je však silně závislé na povaze vstupních dat.

4.2.3 Lift

Kromě dvou základních charakteristik, které jsou v práci využity při nastavování hledání asociačních pravidel, se v praktické části práce ve výstupech analýzy nákupního košíku též poměrně často skloňuje třetí charakteristika, kterou je lift. Ten v české literatuře bývá obvykle označován jako zdvih či zajímavost. Jedná se o relativní zvýšení pravděpodobnosti, že platí závěr, pokud platí předpoklad (Berka, 2003), tedy:

$$\mathrm{Lift}(\mathrm{Ant} \rightarrow \mathrm{Suc}) = \frac{\mathrm{P}(\mathrm{Ant} \wedge \mathrm{Suc})}{\mathrm{P}(\mathrm{Ant}) * \mathrm{P}(\mathrm{Suc})}.$$

To znamená, že pro hodnoty menší než 1 je v případě výskytu předpokladu následný výskyt závěru nepravděpodobný, hodnota 1 znamená, že jsou na sobě předpoklad a závěr zcela nezávislé a hodnoty vyšší než jedna, že výskyt předpokladu znamená vyšší pravděpodobnost výskytu i závěru.

Pokud má tedy nějaké konkrétní pravidlo lift například 10, existence předpokladu v košíku zákazníka nám napovídá, že je desetinásobná šance, že zákazník nakoupí též produkt, který je v takovém pravidle závěrem. Naopak hodnoty nižší než 1, jak již bylo zmíněno v předchozím odstavci, znamenají nižší pravděpodobnost výskytu závěru, což může například ukazovat na to, že předpoklad a závěr mohou být substituty, tedy zákazníkovi produkt z pravé strany pravidla v tomto případě vůbec nenabízet.

4.3 Algoritmus apriori

Jako nejznámější algoritmus pro hledání asociačních pravidel v datech zmiňuje Berka (2003) algoritmus apriori, který byl navržen v souvislosti s analýzou nákupního košíku R. Agrawalem. Ačkoliv jej Agrawal navrhl už v devadesátých letech minulého století, jeho různé implementace včetně té, která je využita v praktické části této práce staví dodnes na stejných principech.

Základem algoritmu je hledání často se opakujících množin neboli v originále frequent itemsets (Berka, 2003). Jelikož i poměrně malá data mohou obsahovat velký počet kombinací (v případě ukázkových dat v tabulce 2 obsahující pouze čtyři produkty je to například 11 teoreticky možných kombinací – šest kombinací po 2 produktech, čtyři po 3 produktech a jednu obsahující všechny čtyři), pro získání těch relevantních, splňujících podmínku minimální podpory i spolehlivosti, je nutné optimalizovat toto hledání a eliminovat některé možnosti, jelikož při větším objemu dat by výpočetní náročnost velmi rychle překročila hranici proveditelnosti či minimálně praktičnosti.

Algoritmus apriori vychází ze skutečnosti, že četnost nadkombinace je nejvýše rovna četnosti kombinace (Berka, 2003), tedy:

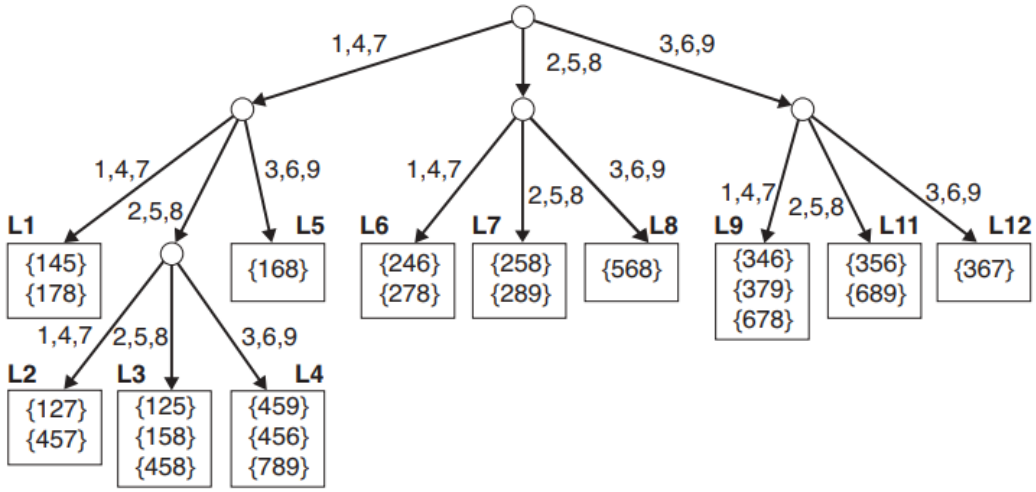
$$\mathrm{n}(\mathrm{Comb}_1) \geq \mathrm{n}(\mathrm{Comb}_2) \quad \text{pro} \quad \mathrm{Comb}_1 \succ \mathrm{Comb}_2 \,,$$

neboli pokud se produkt vyskytuje v transakcích často, budou se relativně často vyskytovat i jeho kombinace. Naproti tomu kombinace produktů, které se v transakcích téměř neobjevují, nebudou mít ani pravidla s dostatečnou podporou, a mohou tedy být rovnou vyřazena.

Algoritmus tedy postupuje tak, že nejprve udělá kombinace tvořené pouze jednou položkou a z těch vybere pouze ty, které dosáhly požadované spolehlivosti, protože pouze takové mohou obsahovat kombinace s dostatečnou podporou. Následně přidá další antecedent, a tak pokračuje až do dosažení maximální uživatelem nastavené délky, vyčerpání vyhovujících pravidel, která by šla dále rozšířit, či celkového počtu položek (Berka, 2003). Díky tomuto postupu dokáže rychle eliminovat slepé uličky a výrazně tak optimalizovat časovou i výkonovou náročnost této analýzy.

Ani při samotném hledání jednotlivých položek v transakcích algoritmus nevyužívá naivní metody, ale data si roztřídí do struktury hash tree, čímž znovu optimalizuje čas a výkon, nutný k vydolování relevantních pravidel, díky mnohem snadnějšímu vyhledávání mezi transakcemi (Aggarwal, 2015). Příklad takového stromu lze vidět na obrázku níže, na kterém

čísla uvnitř složených závorek v listech stromu představují jednotlivé transakce a čísla vedle každé šipky znázorňují, jak pravidla třídit podle čísel na odpovídající pozici tohoto pravidla (Tan a další, 2006). Logiku tohoto stromu lze demonstrovat na příkladu – pravidlo {145} v listu L1 začíná číslicí 1, je tedy zařazeno do odpovídající větve nalevo, která sdružuje pravidla s první číslicí 1, 4 nebo 7. Druhá číslice je 4, pravidlo je tedy opět zařazeno do větve nejvíce vlevo. Analogicky by bylo možné postupovat a větvit strom dále, nicméně jako příklad je tento malý strom dostačující.



Obrázek 2: *Příklad hash tree struktury (zdroj: Tan a další, 2006)*

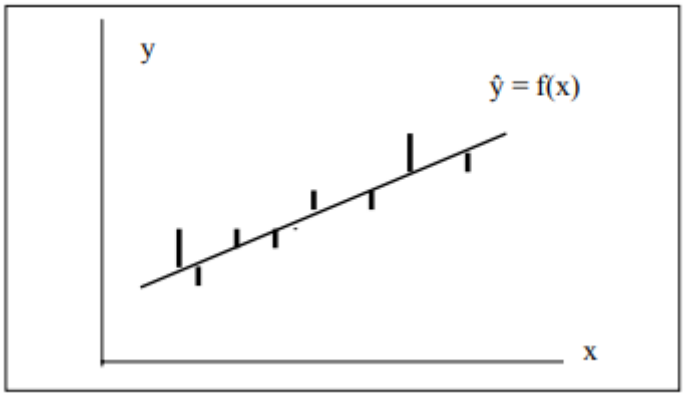
Že se v případě algoritmu apriori dodnes jedná o velmi relevantní postup v dolování asociačních pravidel, dokládá například i již citovaný Aggarwal ve své mnohem čerstvější publikaci Data Mining z roku 2015, kde velmi dopodrobna popisuje, jak tento algoritmus funguje. Pokud má tedy čtenář hlubší zájem o tuto problematiku a celkový princip, na kterém algoritmus apriori funguje a asociační pravidla doluje, jedná se o velmi přínosný zdroj.

4.4 Regresní analýza

Důležitým termínem pro bližší popsání provedené implementace v praktické části této práce je také regresní analýza. Zde na rozdíl od korelační analýzy, kde je zkoumáno pouze to, zda mezi dvěma numerickými veličinami existuje lineární závislost, hledáme parametry této závislosti (Berka, 2003). V kontextu této práce jsou zkoumanými veličinami pro sestrojení poptávkových křivek právě poptávka po konkrétním produktu a veškeré různé parametry, které mohou na ni mít vliv a cílem je vliv těchto parametrů kvantifikovat.

Lineární regresní modely jsou jednou z nejpoužívanějších metod statistické analýzy vícerozměrných dat. Vyjadřují vztah mezi vysvětlovanou proměnnou (také označovanou jako cílová proměnná či regresand), která je v případě lineární regrese numerická, a vysvětlujícími proměnnými (též prediktory či regresory). V kontextu této práce je nadále používáno termínů prediktory a cílová proměnná. Proces výpočtu parametrů této lineární rovnice se nazývá regresní modelování (Aggarwal, 2015). Při něm se parametry odhadují metodou nejmenších čtverců, která minimalizuje rozdíly mezi skutečnou pozorovanou hodnotou a hodnotou očekávanou (Berka, 2003).

V dvojrozměrném prostoru lze tento proces znázornit tak, jako na obrázku níže, tedy jako prokládání přímky body takovým způsobem, aby byla minimalizována celková odchylka, která je uvažována jako druhá mocnina (proto tedy metoda nejmenších čtverců), aby bylo možné považovat záporné i kladné rozdíly za stejně závažné (Berka, 2003).



Obrázek 3: *Metoda nejmenších čtverců (Berka, 2003)*

Východiskem lineárních modelů je nicméně linearita vztahu mezi prediktory a cílovou proměnnou. Tento předpoklad však v mnoha případech nemusí platit. V takové situaci lze použít zobecněné lineární modely, kdy je předpokládána složitější funkční závislost mezi prediktory a cílovou proměnnou (Aggarwal, 2015).

Samotná lineární regrese je v podstatě také jedním ze speciálních příkladů zobecněného lineárního modelu, mezi další lze jmenovat například logistickou regresi, u níž je závislá veličina kategoriální a model počítá pravděpodobnost, že cílová proměnná nabývá konkrétní hodnoty, či poissonovu regresi, která slouží k modelování četností (Aggarwal, 2015).

Při logistické regresi se velmi často může jednat o cílové proměnné typu boolean, tedy pravda či nepravda, ale není nutné se omezovat pouze na dvě hodnoty, lze modelovat i více

různých hodnot, je však třeba výsledky normalizovat tak, aby byl součet pravděpodobností všech možných hodnot cílové proměnné roven jedné (Aggarwal, 2015).

4.5 Poissonova regrese

Dalším termínem, který je nutný pro vysvětlení implementace v praktické části této práce, je již výše zmíněná poissonova regrese. Ta je jedním ze základních stavebních kamenů současné logiky modelů, které jsou v praktické části použity k výpočtu poptávkových křivek produktů, ze kterých lze následně zjistit optimální cenu produktu.

Konkrétní implementace poissonových regresních modelů je blíže popsána a vysvětlena v praktické části práce, zatímco tato kapitola popisuje teoretické základy, na kterých je tato implementace založena.

Před představením poissonovy regrese je vhodné v krátkosti popsat poissonovo rozdělení pravděpodobnosti. To popisuje počet výskytů nezávislých jevů v určitém intervalu veličiny, kterou je v kontextu této práce čas, nicméně může se jednat například i o délku či objem. Jelikož popisuje četnosti jevů s velmi malou pravděpodobností výskytu, bývá také často nazýváno jako rozdělení řídkých jevů. Patří mezi diskrétní rozdělení a četnosti, které popisuje, jsou množinou celých nezáporných čísel. Pravděpodobnostní funkci poissonova rozdělení lze zapsat následovně (Marek, 2013):

$$P(x) = \frac{\lambda^x}{x!} e^{-\lambda}, \quad x = 0, 1, \dots, \quad \lambda > 0 .$$

Střední hodnota je pro poissonovo rozdělení rovna rozptylu a zároveň platí, že jsou obě rovny parametru rozdělení λ, tedy

$$E(X) = \lambda ,$$

$$D(X) = \lambda .$$

Poissonovo rozdělení bývá také používáno k aproximaci binomického rozdělení, a to v případě velkého počtu pokusů a velmi malé pravděpodobnosti výskytu sledovaného jevu, obvykle kdy je $n > 30$ a $p < 0{,}1$, typicky tedy pro modelování diskrétních četností ve spojitém čase (Friesl, 2014).

Poissonova regrese má využití při modelování cílových proměnných, které představují četnosti nezávislých jevů, pokud odpovídají právě poissonovo rozdělení pravděpodobnosti, respektive je lze poissonovým rozdělením aproximovat.

V kontextu této práce to tedy znamená odhadování poptávky po produktu, tedy počet prodaných kusů v určeném časovém intervalu, v závislosti na jeho ceně a mnoha dalších prediktorech, mezi které mohou patřit například sezónnost, cena konkurence, a mnohé další.

Rovnici poissonovy regrese lze zapsat následovně (Berk a MacDonald, 2008):

$$P[Y_i = y_i|x_i] = \frac{e^{-\lambda_i}\lambda_i^{y_i}}{y_i!}, \quad y_i = 0, 1, 2, \dots N. \tag{4.1}$$

V rovnici 4.1 výše pak Y_i znamená náhodnou proměnnou četnosti, y_i jednotlivou hodnotu četnosti, λ_i parametr značící očekávanou hodnotu četnosti a *i* index z *N* pozorování (Berk a MacDonald, 2008).

Jelikož se v případě poissonovy regrese, jak již bylo zmíněno, jedná o zobecněný lineární model, hodnota parametru je v tomto případě rovna transformační funkci (link function), převádějící hodnoty prediktoru na hodnoty cílové proměnné. Ta je v tomto případě tvořena přirozeným logaritmem (Berk a MacDonald, 2008), tedy:

$$\lambda_i = exp\{x_i^T \beta\}. \tag{4.2}$$

Pro reálné využití na skutečných datech je důležité zmínit, že například výše uvedená rovnice 4.2 neplatí v případech, ve kterých chybí některé prediktory, což je v kontextu této práce aktuální pro veškeré produkty, u kterých nebude možné komplementy vydolovat. Zároveň je nutné, aby byly události ze vstupních dat skutečně nezávislé (Berk a MacDonald, 2008).

4.6 Hodnocení kvality modelů

Následující podkapitola se zabývá možnými způsoby a metrikami, které lze použít k měření a porovnávání kvality modelů strojového učení pro problémy předpovídání četnosti náhodných a vzájemně nesouvisejících jevů. V kontextu této diplomové práce to znamená rozdíl mezi prodaným množstvím za určité časové období, které model při současné ceně předpovídá, a skutečně prodaným množstvím při aktuální ceně.

Při hodnocení tohoto rozdílu je však třeba adresovat některé problémy, které jej mohou ovlivnit. Typickým příkladem takového problému, který eliminují všechny níže popsané způsoby měření, je nezávislost na směru chyby. V drtivé většině případů není vhodné zprůměrovat například jeden chybný odhad o 2 nižší než skutečnou hodnotu a druhý chybný odhad o 2 vyšší jako nulovou průměrnou chybu. Tedy obyčejný průměr jednotlivých chyb, které regresní model spočítá a vypíše, není dostatečnou metrikou. Některé vhodnější metody jsou diskutovány v následujících podkapitolách.

4.6.1 Mean absolute error (MAE)

Mean absolute error bývá v česky psané literatuře obvykle nazýván střední absolutní chyba. Vyjadřuje průměrnou velikost chyby predikce od skutečnosti, nezávisle na tom, kterým směrem chyba je, tedy stejným způsobem penalizuje jak kladnou, tak zápornou odchylku od skutečné hodnoty a všechny jednotlivé hodnoty mají stejnou váhu. Vzorec MAE lze vyjádřit následovně:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - \hat{x}_i| \tag{4.3}$$

Ve vzorci 4.3 výše je n počtem hodnot, x_i je skutečná hodnota, $\hat{x}_i$ předpověděná hodnota a $|x_i - \hat{x}_i|$ je hodnota každé jednotlivé absolutní chyby.

Výhodou této metody je velmi snadná interpretovatelnost – měřítko je v shodné s daty a jedná se tedy o snadno pochopitelnou veličinu, která jednoduše představuje číslo, o kolik se průměrně odhad liší od skutečné hodnoty. Zároveň se však jedná i o hlavní nevýhodu této metody, a to v případě potřeby porovnání napříč rozdílnými měřítky (Hyndman a Koehler, 2006).

Na příkladu relevantním ke kontextu této práce si lze tento problém představit tak, že hodnota MAE, které je rovna jedné pro produkt, kterého se za určený časový interval prodá několik tisíc kusů, představuje velmi přesný model, jelikož se v odhadu prodejů lišil v relativním smyslu minimálně. Naproti tomu identická hodnota MAE pro produkt, který se za určený časový interval prodá jen jednou, již model do příliš pozitivního světla nestaví, jelikož relativně se jedná o velmi nepřesný odhad.

4.6.2 Mean square error (MSE) a root mean square error (RMSE)

Následující dvě metriky jsou spojeny do jedné podkapitoly, jelikož druhá zmíněná vychází z té první. Obě veličiny jsou velmi častou volbou při hodnocení přesnosti odhadů a druhá zmíněná, RMSE, je v mnohém podobná již výše zmíněné metrice MAE.

Vzorec MSE lze zapsat následovně:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2.$$

MSE tedy eliminuje vliv směru chyby umocněním na druhou. Jistou nevýhodou této metriky tedy je, že není ve stejném měřítku jako původní data a hůře se tedy interpretuje. Tento problém adresuje metrika druhá, tedy root mean square error, který lze, jak již může napovídat název, vypočítat odmocněním MSE, tedy:

$$RMSE = \sqrt{MSE} = \frac{1}{n}\sum_{i=1}^{n}\sqrt{(x_i - \hat{x}_i)^2}.$$

Ačkoliv se na první pohled může zdát, že v případě RMSE se bude po odmocnění jednat o stejnou hodnotu, jako v případě MAE, není to pravda. Díky umocňování hodnot tato metoda více penalizuje extrémní hodnoty a je více náchylná na zkreslení v důsledku těchto odlehlých hodnot. To může v závislosti na požadavcích konkrétní situace představovat jak problém, tak výhodu v porovnání s metodou střední absolutní chyby. Zároveň bude na identických datech vždy větší nebo rovna MAE (Willmott a Matsuura, 2005).

Navíc je s oběma těmito metodami stále spojený stejný problém, jako v případě metriky MAE, tedy obtížné porovnání napříč rozdílnými měřítky (Hyndman a Koehler, 2006).

4.6.3 Mean absolute percentage error (MAPE)

Právě porovnání napříč rozdílnými měřítky adresuje metoda MAPE, tedy mean absolute percentage error. Vzorec MAPE lze zapsat následovně:

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{x_i - \hat{x}_i}{x_i}\right|$$

I MAPE však trpí několika problémy, které jej činí nevhodnou metrikou pro použití v kontextu této práce. Právě v případě předpovídání poptávky, která často obsahuje nulové hodnoty, když se za určené časové období konkrétní produkt neprodá ani jednou, vzniká ve vzorci dělení nulou a hodnotu MAPE nelze spočítat.

Pokusy, jak tento problém odstranit, například použitím symetrického sMAPE, pak mohou znamenat dělení číslem velice blízkým nule a zavádějícími hodnotami, problém tedy stále není zcela odstraněn. Kromě toto sMAPE může nabývat i záporných hodnot, což značně komplikuje jeho interpretaci (Hyndman a Koehler, 2006).

Navíc pro příliš nízké předpovídané hodnoty nemůže MAPE přesáhnout 100 %, na rozdíl od příliš vysokých předpovědí, kde hodnoty MAPE nemají limit. V souvislosti s tím je třeba zmínit i fakt, že záporné chyby jsou více penalizovány než kladné, takže dochází k systematickému zvýhodňování těch modelů, které předpovídají nižší hodnoty (Hyndman a Koehler, 2006).

4.6.4 Mean absolute scaled error (MASE)

Další možnou metrikou, kterou lze pro měření přesnosti předpovědí modelu využít je mean absolute scaled error. Právě tato metrika je v praktické části využita jako hlavní pro porovnání přesnosti modelů a výběr toho nejlepšího.

V porovnání s výše popsanými metodami se jedná o metodu poměrně mladou, byla navržena statistikem Robem. J. Hyndmanem, který se snažil touto novou metodou adresovat některé problémy starších metod, sám ji popsal jako jako „obecně aplikovatelnou metriku přesnosti předpovědi, která postrádá problémy ostatních metod“ (Hyndman a Koehler, 2006).

Vzorec MASE lze zapsat následovně (Hyndman a Koehler, 2006):

$$MASE = \frac{MAE}{MAE_{in-sample,naive}}$$

Tedy zlomek, kde v čitateli je hodnota střední absolutní chyby odhadu a ve jmenovateli střední absolutní chyba naivního odhadu. Naivním odhadem je zde myšleno jednoduše dosazení hodnoty cílové proměnné z minulého období (Hyndman a Koehler, 2006).

MASE tak na rozdíl od výše zmíněných metod eliminuje veškeré problémy, které při použití na poptávková data vyvstávají. Jedná se o metriku splňující požadavek na škálovou invarianci, korektní chování v případě dat obsahující nulové hodnoty, symetrii, tedy stejnou váhu na negativní i pozitivní chyby a snadnou interpretovatelnost, kdy hodnoty nižší než 1 znamenají lepší předpovědi než naivní metodou a hodnoty vyšší než 1 opak. Jediný případ, kdy je výsledek MASE nekonečno nebo nedefinovaný, je v případě, kdy jsou veškerá historická pozorování identická (Hyndman a Koehler, 2006). Taková data jsou však pro strojové učení v kontextu této práce bezvýznamná, jedná se tedy o nevýznamný nedostatek v porovnání s nedostatky předchozích metod.

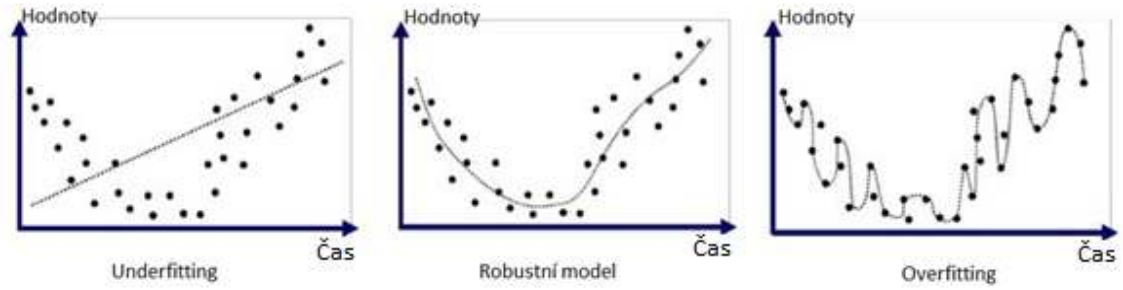
4.7 Overfitting

Následující podkapitola popisuje problém spojený s modely strojového učení, který je označovaný jako overfitting, neboli v češtině užívaný termín přeučení (Berka, 2003). Dochází k němu často především v případech, kdy je k dispozici relativně malé množství trénovacích dat případně velké množství prediktorů – obecně když natrénovaný model příliš dobře vysvětluje veškeré charakteristiky trénovacích dat. To se může zdát na první pohled jako ideální stav, v takovém případě totiž může model dosahovat velice nízkých hodnot chyby na trénovací množině, nicméně jakmile je model použit pro předpovídání na neznámých datech, jeho přesnost dramaticky klesne (Aggarwal, 2015).

Právě to je hlavním důvodem, proč je nutné validovat na datech, která model nezná, obvykle na pokud možno zcela oddělené validační či testovací množině dat, jinak budou nalezené znalosti vystihovat více náhodné charakteristiky v datech, tzv. noise. V případě, že taková možnost neexistuje, lze použít například metody křížové validace, kdy se rozdělí data typicky na 10 částí (10 fold cross-validation) tak, že desetina se oddělí pro testování a zbytek je použit pro trénování. Tento postup se zopakuje desetkrát a výsledek se zprůměruje. Dobře fungující algoritmus strojového učení bude nakonec schopný generalizovat a oddělit skutečné znalosti od náhodnosti a případných extrémních případů v datech (Berka, 2003).

S termínem overfitting se zároveň pojí příbuzný termín underfitting, který označuje přesný opak, tedy že model nevystihuje data dostatečně. V takovém případě lze o modelu říci, že má vysoký bias, neboli vychýlení či také systematickou chybu a lze jej obvykle rozpoznat poměrně snad již při trénování. Naproti tomu při overfittingu má model sice systematickou chybu nižší, má však vyšší varianci neboli rozptyl. Hledání rovnováhy mezi těmito dvěma

extrémy je často jednou z nejtěžších částí dolování znalostí z dat (Despois, 2018). Na obrázku níže lze vidět graficky znázorněné příklady overtiffingu i underfittingu a uprostřed pak ideální stav.



Obrázek 4: *Overfitting vs underfitting (zdroj: Bhande, 2018; překlad: autor)*

Jako řešení overfittingu se pak nabízí několik přístupů, z nichž některé jsou zmíněny níže.

Jedním možným přístupem je již zmíněná technika křížové validace, tedy cross-validation, která umožňuje parametry ladit již na původní trénovací množině, takže lze zcela oddělit testovací data, na kterých následně model otestovat případně je začlenit a zvětšit tak objem trénovací množiny (Moore, 2001). Tato metoda je však spíše užitečná pro odhalení, kdy k přeučení začíná docházet.

Další možností, která však ne vždy bude zcela dosažitelná, je zvětšit množinu trénovacích dat. Větší množství dat je často účinnější metodou než neustálé zdokonalování modelu, v případě problému přeučení pak navíc větší množství dat nutí model více a více generalizovat (Despois, 2018).

Alternativou k možnosti většího množství trénovacích dat je, možná na první pohled trochu neintuitivně, redukovat komplexitu modelu snížením počtu prediktorů. Menší množství prediktorů totiž obdobně jako větší množství trénovacích dat donutí model k větší generalizaci (Despois, 2018).

Mezi další metody, jak minimalizovat přeučení jsou různé úpravy dat, jako například přidání šumu, nebo regularizace, kterou lze model donutit dělat kompromisní váhy parametrů, a tedy opět k větší generalizaci (Despois, 2018).

4.8 Shrnutí kapitoly

Tato kapitola se věnovala různým analytickým metodám, které jsou v praktické části práce využité k naplnění jednotlivých cílů, po jejím přečtení tedy čtenář získal nezbytný teoretický základ pro pochopení praktické části práce, a to jak dolování komplementů, tak jejich začlenění do existujících modelů.

Jako první je v kapitole popsána analýza nákupního košíku, která je často používanou metodou pro hledání asociací mezi produkty, respektive identifikaci, co zákazníci obvykle nakupují společně, což je užitečná znalost nejen v kontextu dynamické cenotvorby.

Analýza nákupního košíku je prováděna pomocí dolování asociačních pravidel, proto následuje kapitola, která se podrobněji věnuje právě této problematice, popisuje algoritmus apriori, který je neznámější metodou, jak tato pravidla z dat dolovat a popisuje i nejdůležitější charakteristiky, tedy podporu, spolehlivost a lift, pomocí kterých jsou v praktické části vybrána vyhovující pravidla.

Následně kapitola popisuje regresní analýzu, která je klíčová pro pochopení výpočtu poptávkových křivek, jelikož ty jsou počítány pomocí poissonovy regrese. Popisuje poissonovu regresi blíže a nastiňuje nezbytné předpoklady, které musí být pro její použití naplněny, tedy především data odpovídající poissonovu rozdělení pravděpodobnosti.

Poté kapitola představuje různé metriky, pomocí kterých lze měřit kvalitu modelů, od střední absolutní chyby MAE, přes čtvercovou odchylku MSE a její odmocninu RMSE, procentuální chybu MAPE až po mean absolute scaled error, tedy MASE, která je hlavní metrikou pro hodnocení modelů v kontextu této práce, jelikož odstraňuje veškeré problémy ostatních metrik.

Nakonec kapitola též popisuje problém overfittingu, neboli přeučení. Ten nastává, když model příliš dobře vystihuje trénovací data a nedokáže generalizovat, na nových datech tedy dosahuje výrazně horších výsledků, na což je třeba při rozšiřování modelů v praktické části pamatovat.

5 Popis současného stavu

Tato kapitola je úvodní kapitolou pomyslného druhého logického celku práce, tedy praktické části. Zde jsou technologie a algoritmy popsané v předchozí teoretické části použity k naplnění cílů této práce, tedy rozšíření modelů dynamické cenotvorby o vliv komplementárních produktů. Tato kapitola popisuje v krátkosti klienta, na jehož datech bude analýza provedena, jaké nástroje jsou využity k implementaci vytyčených cílů a výchozí stav, tedy popis toho, jak je v současnosti optimální cena počítána.

V roli klienta dynamické cenotvorby, která je implementována modely rozšiřovanými v této práci, a na jehož datech bude následně porovnávána přesnost rozšířených modelů s původními, je přední český e-shop specializující se na kojenecké potřeby.

Data, která byla využita pro identifikaci komplementů, jsou blíže popsána v následující podkapitole. Veškerá data jsou uložena v databázi na MS-SQL serveru, ze kterého jsou pomocí příkazů SQL dotazována a využita k případné další analýze. Veškeré SQL příkazy či skripty jsou vytvářeny přímo v Microsoft SQL Server Management Studio. Zde zároveň dochází k většině nezbytných transformací dat z tvaru obdrženého od klienta do tvaru, který je vhodný jako vstup do modelů pro výpočet poptávkových křivek.

Všechny následné výpočty nad získanými daty, ať už se jedná o identifikaci komplementů asociačními pravidly, či samotné poptávkové křivky produktů, které jsou v práci rozšiřovány, a výpočty optimální ceny v rámci dynamické cenotvorby, jsou psány v jazyce R. Kód je v tomto případě vytvářen v programu RStudio, ve kterém je následně i spouštěn a vyhodnocen. V jazyce R zároveň dochází k některým finálním transformacím dat vytažených z datového skladu SQL dotazy, nicméně k většině potřebných transformací dochází ještě před uložením do tabulek, ze kterých se do RStudia data dostávají.

5.1 Popis výchozích dat

Následující podkapitola popisuje data, ze kterých praktická část práce vychází. Pro účely této diplomové práce bylo využito archivní databáze z třetího kvartálu 2018, tedy neprodukční verze, na které již neprobíhají v době zpracovávání praktické části diplomové práce žádné další výpočty a aktualizace dat a poptávkových křivek.

Důvodem je jednak nemožnost zasahování do produkční databáze s nejaktuálnějšími klientskými daty a zároveň snadnější porovnání přesnosti modelů před a po rozšíření o vliv komplementů. V případě provádění podobného porovnání na produkční databázi by v průběhu prací snadno mohlo dojít k aktualizaci a přepočítání poptávkových křivek a srovnání by již nemuselo vůbec dávat smysl. Zároveň by operace nad produkční databází mohla způsobit celou řadu problémů například v oblasti výkonu a využití dat z předchozího období neznamená pro vytvoření návrhu, jak modely rozšířit, žádnou překážku.

Stěžejní tabulkou pro provedení analýzy nákupního košíku je tabulka *orderitem*. Ta je zároveň i nejrozsáhlejší tabulkou ve schématu datového skladu, jelikož v sobě obsahuje údaje o veškerých provedených transakcích za poslední rok. Lze z ní tedy propojit jednotlivé produkty k objednávkám, jejich ceně i množství v každé z nich, a tedy i po vhodné transformaci získat data ve formátu pomyslného nákupního košíku – tedy ID objednávky následované seznamem produktů, které daná objednávka obsahuje. Po transformaci do této podoby lze již spustit algoritmus apriori a nechat jej spočítat jednotlivá pravidla, ze kterých lze následně identifikovat produkty, mezi kterými existují asociace.

Tabulka *orderitem* obsahuje roční historii objednaných položek z e-shopu, nejstarší transakce je z 16. září 2017 a nejnovější z 16. září 2018.

Celkový počet řádků hlavní tabulky *orderitem* je 1 633 527. Každý řádek této tabulky představuje jednu položku nákupního košíku, nicméně jak je zmíněno i v následující kapitole identifikující komplementy, obsahuje také pomocné položky jako je poštovné nebo slevové kupony, ne všechny řádky tedy představují produkty, které zákazník skutečně vložil do košíku, objednal a zaplatil. Ty je nutné pro dolování pravidel z dat následně odfiltrovat. Struktura databázové tabulky *orderitem* je popsaná v následující tabulce.

Tabulka 4: *Struktura tabulky orderitem (Zdroj: autor)*

Název sloupce	Datový typ	Popis	Příklad hodnoty
id_orderitem	nvarchar(4000)	ID konkrétní položky v transakci, unikátní ID pro každý řádek	2ef11f27-45ca-41c6-bd48-2c6cd5ce9e31
id_order_transaction	nvarchar(4000)	ID transakce, tedy nákupního košíku	1758781
date_created	datetime2(0)	Datum a čas nákupu	2018-06-15 19:48:53
obs_date	date	Datum nákupu	2018-06-15
PriceInclVAT	decimal(35,14)	Cena včetně DPH	379.00000000000000
VAT	decimal(35,14)	Částka DPH v CZK	65.79000000000000
VATRate	decimal(35,14)	Sazba DPH v procentech	21.00000000000000
PriceExclVAT	decimal(35,14)	Cena bez DPH	313.21000000000000
UnitPrice	decimal(35,14)	Jednotková cena za položku	379.00000000000000
Unit	nvarchar(4000)	Druh jednotky	ks
Quantity	decimal(35,14)	Množství zakoupené v objednávce	1
Name	nvarchar(4000)	Název a popis výrobku	PAMPERS Premium Care 1 NEWBORN 88ks

Název sloupce	Datový typ	Popis	Příklad hodnoty
date_modified	datetime2(0)	Datum změny	1900-01-01 00:00:00
State	nvarchar(4000)	Stav objednávky (objednáno, vyfakturováno, …)	Invoiced
IsDiscount	nvarchar(4000)	Identifikace, jestli jde o slevu	False
id_item	nvarchar(36)	ID produktu odkazující do produktového katalogu	82872904-5baa-4553-96cb-38468527fb07
item_code	nvarchar(4000)	Kód produktu	F123456
id_order	nvarchar(36)	ID objednávky	0000147f-f730-ce95-e92e-85db417de31e
DT_timestamp	datetime2(0)	Časová značka nahrání do datového skladu	2018-09-19 08:45:54

Další důležitou tabulkou, ve které jsou data transformovaná do formátu, ve kterém mohou lépe sloužit jako vstup pro výpočet poptávkových křivek je tabulka *pois_demand_input*. V ní jsou data pro vybrané produkty, které do optimalizace vstupují na základě různých businessových či technických definic, seskupená po požadovaných časových intervalech, podle kterých je nutné produkty přeceňovat a pro každý tento interval lze zjistit odpovídající hodnotu jednotlivých prediktorů.

Tato tabulka hraje hlavní roli po výstupu vydolovaných asociačních pravidel – do ní je třeba vložit nové prediktory, tedy údaje o jednotlivých komplementech, které budou začleněny do výpočtu poptávkových křivek a tuto rozšířenou tabulku následně použít jako vstup do poissonovy regrese s těmito novými prediktory a tím vytvořit rozšířené modely.

Data jsou do formátu této tabulky vkládána propojením několika dalších tabulek ze schématu datového skladu, včetně pomocných tabulek obsahujících kalendář či rozdělení podle stanovených časových oken, výsledný formát tabulky *pois_demand_input* lze vidět v tabulce 5 níže.

Do této tabulky je nutné připojit vydolovaná pravidla, respektive přidat několik sloupců týkajících se ceny či prodaného množství komplementárních produktů. Blíže toto popisuje kapitola 7.1.

Detailní popis struktury této tabulky níže vynechává některé sloupce, které představují další prediktory vztahující se k jednotlivým produktům, nicméně veškeré technické detaily řešení není možné v této práci zveřejnit. Jelikož ale jejich popis není pro rozšíření modelů o vliv komplementů nijak přínosný, neznamená tento fakt pro čtenáře absenci žádné klíčové informace.

Tabulka 5: *Struktura tabulky pois_demand_input (Zdroj: autor)*

Název sloupce	Datový typ	Popis	Příklad
id_item	nvarchar(4000)	ID produktu odkazující do produktového katalogu	82872904-5baa-4553-96cb-38468527fb07
week	numeric(17,0)	Číslo týdne od začátku transakční historie	100
obs_date_start	date	Datum sledovaného období (dne)	2017-09-29
obs_date_end	datetime	Datum a čas konce sledovaného dne	2018-09-29 23:59:59.000
noi	int	Množství transakcí obsahujících konkrétní produkt v aktuálním týdnu	10
q	decimal(38,14)	Prodané množství v aktuálním období	23.00000000000000
x	decimal(38,6)	Cena produktu ve sledovaném dni (v CZK)	179.000000
q1	decimal(35,14)	Prodané množství v minulém období	9.00000000000000
DT_timestamp	datetime2(0)	Časová značka nahrání do datového skladu	2018-09-19 08:45:54

5.2 Logika výpočtu optimálních cen produktů

Po popisu dat, ze kterých logika výpočtu optimálních cen vychází lze již obecně popsat, jak funguje samotný výpočet.

Zde je prvním nezbytným krokem výpočet poptávkových křivek pro vybrané produkty. Toho je docíleno především pomocí implementace poissonovy regrese v jazyce R, do které jako prediktory vstupují odpovídající data z tabulky *pois_demand_input* a cílovou proměnnou je počet prodaných kusů produktu. Právě v tomto kroku je pro splnění hlavního cíle této práce nezbytné rozšířit předpis funkce o nově identifikované komplementy, respektive jejich cenu či prodané množství, a zkoumat jejich vliv na poptávku produktů.

V momentě, kdy jsou sestrojené poptávkové křivky pro veškeré vybrané produkty ze vstupů v tabulce *pois_demand_input,* lze z nich spočítat optimální ceny produktů.

5.2.1 Účelová funkce

Hledání optimální ceny je formulována jako problém kvadratického programování, který maximalizuje výsledek následující účelové funkce:

$$J = (1 - \alpha)(GM - DC) + \alpha TU \rightarrow max, \tag{5.1}$$

$$x_{LB} \leq x \leq x_{UB}. \tag{5.2}$$

V rovnici 5.1 je GM (gross margin) hrubá marže, DC (distribution costs) distribuční náklady a TU (turnover) je obrat. Alfa je parametr agresivity, který je blíže popsán v následující podkapitole. x_{LB} a x_{UB} v rovnici 5.2 znázorňují businessová omezení ceny produktu shora a zdola, tedy lower and upper bound ceny produktu.

Další pojmy, které je tedy třeba blíže definovat je celkový obrat, tedy TU, který je definován následovně:

$$TU = q^T \frac{x}{1 + VAT}.$$

V rovnici výše představuje q^T transponovaný vektor prodaných množství jednotlivých produktů a VAT (value added tax) představuje sazbu DPH.

Dále následuje definice hrubé marže, tedy GM:

$$GM = q^T\left(\frac{x}{1+VAT} - x_{pp}\right).$$

V této rovnici pak x_{pp} představuje nákupní cenu produktu (purchase price).

Poslední nezbytnou definicí pro popsání celé účelové funkce jsou distribuční náklady, tedy DC:

$$DC = q^T C_{DC},$$

kde C_{DC} představuje cenu distribuce jednoho produktu.

5.2.2 Agresivita

Důležitým pojmem pro vysvětlení účelové funkce je již zmíněný parametr agresivity. Jedná se o parametr, která nabývá hodnot v uzavřeném intervalu od 0 do 1 a vyjadřuje poměr mezi maximalizací obratu a maximalizací hrubé obchodní marže. Klient pomocí tohoto parametru tedy může nastavit, který cíl má model při nastavování optimálních cen sledovat. Hodnota 0 znamená maximalizaci marže, tedy přístup, který vůbec nehledí na případný vývoj obratu, jelikož ten je v tomto případě vynásoben 0, jak lze vidět v účelové funkci v rovnici **Error! Reference source not found.**. Model tedy bude spíše zdražovat i za cenu snižování prodaného množství a poklesu celkového obratu. Hodnota 1 pak znamená agresivitu největší, tedy největší orientaci na maximalizaci obratu bez ohledu na případné změny v marži. V takovém případě lze očekávat více snižování cen u vybraných produktů, aby se zvýšily celkové tržby i za cenu případného snížení celkové marže.

Tento parametr tedy má na výpočet cen pochopitelně značný vliv, nicméně v kontextu této práce jej postačí neměnit z aktuální hodnoty, aby porovnání byla mezi relevantními hodnotami a žádné bližší upravování či analýza tohoto parametru nejsou nutné.

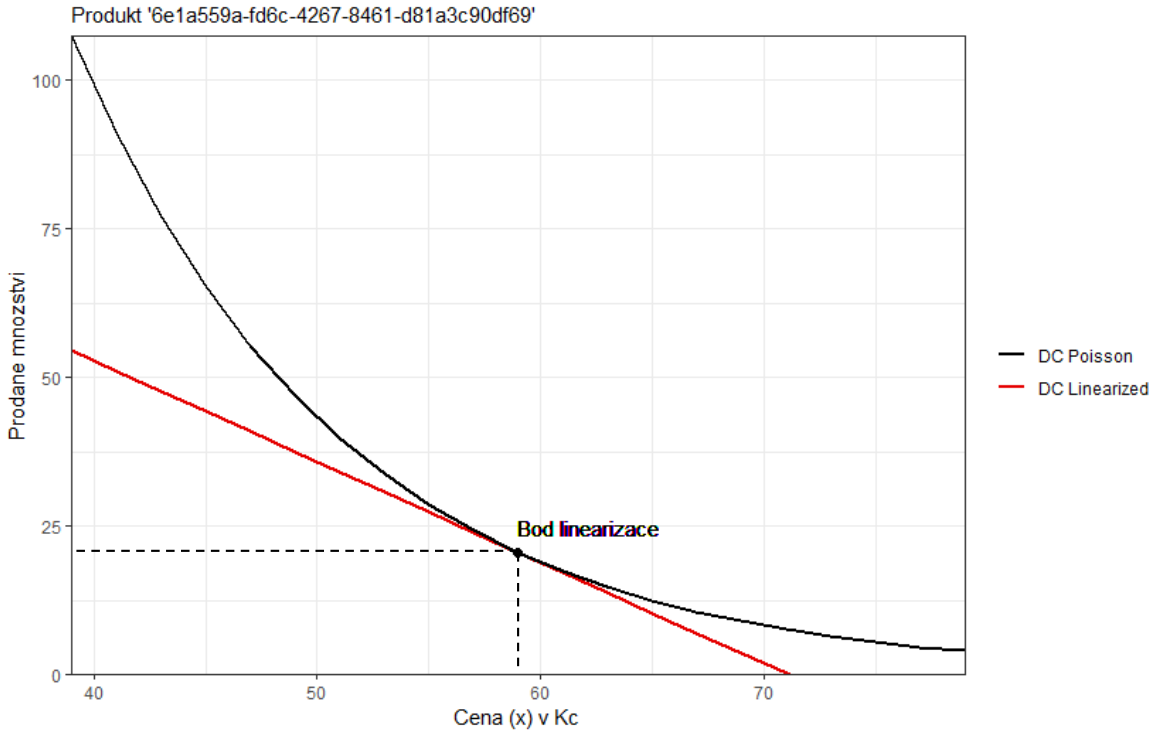
5.2.3 Linearizace funkce

Jak již bylo zmíněno v podkapitole 5.2.1, hledání optimální ceny je řešeno jako optimalizační úloha kvadratického programování a využívá funkce solveru v jazyce R z knihovny *quadprog*. Ten vyžaduje zadání jako problém kvadratické optimalizace s následujícím definovaným tvarem:

$$W = \frac{1}{2}x^T F x - f^T x + k \rightarrow min,$$

$$A^T x \geq b_0 .$$

Je tedy nezbytné úpravami získat tento obecný tvar pro solver, aby bylo možné najít optimální cenu pomocí této funkce, tedy poptávkovou křivku linearizovat. Toho je dosaženo funkcí *linearize_pois*, nadefinovanou v jazyce R, do které bude nutné odpovídající nové prediktory vztahující se ke komplementům přidat, což je popsáno v kapitole 7.2.



Obrázek 5: *Příklad linearizace poptávkové křivky (zdroj: autor)*

Na obrázku výše lze vidět grafické naznačení linearizace poptávkové křivky v bodě aktuální ceny. Cena je naznačena na ose x a na ose y prodané množství za určené období. To záleží na nastavení modelu, nicméně lze si pod ním představit například týden. V naznačeném bodě linearizace dochází k linearizaci křivky, a jak lze na grafu vidět, při pohybování po linearizované funkci příliš daleko od bodu aktuální ceny může dojít k velkým rozdílům původní a linearizované funkce v odhadech prodaného množství, a tedy nepřesnosti modelu. V případě větší změny ceny je tedy vhodné linearizovat křivku ve více bodech, případně se příliš prudkým změnám ceny vyvarovat a maximální změnu ceny omezit.

5.2.4 Posloupnost kroků algoritmu dynamické cenotvorby

V obecné rovině lze popsat postup nalezení optimálních cen pro produkty jako sled následujících kroků.

V prvním kroku je nutné zákazníkova data nahrát ETL procesy do datamartu v databázi na SQL serveru. Po připravení vstupních dat pro výpočet poptávkových křivek je ve druhém kroku nezbytné jednotlivé poptávkové křivky spočítat pomocí poissonovy regrese s cílovou proměnnou znázorňující odhadované prodané množství produktu a prediktory, které představují jeho cena, cena konkurence a mnoho dalších veličin, mezi které je v rámci této práce přidáván i vliv komplementů. Ve třetím kroku jsou vypočtené parametry jednotlivých závislostí uloženy do nové tabulky v databázi, včetně metrik jako jsou různé charakteristiky jejich přesnosti, z nichž nejdůležitější pro kontext této práce je MASE.

Tyto vypočítané poptávkové křivky následně vstupují do optimalizace. Jelikož je optimalizace řešena jako výše popsaný problém kvadratické optimalizace, je nezbytné provést odpovídající úpravy a definovat problém ve formátu vhodném pro vstup do kvadratického solveru a následně hledat optimální bod, který maximalizuje účelovou funkci podle nastavené agresivity. Výsledkem této optimalizace je buď optimální cena, pokud ji lze nalézt, nebo v případě nemožnosti optimalizace ceny pro konkrétní produkt hodnota NULL.

Důvodů pro nemožnost optimalizace je vícero, v první řadě je to především nevalidní poptávková křivka. Ty musí projít určitou kontrolou kvality, kde se kontroluje jejich sklon, shoda se skutečností či minimální hranice obrátkovosti. Z křivky tak například může vyjít nezávislost produktu na jeho ceně, což v praxi pochopitelně pravda není, nicméně model nemusel mít dostatečně průkazná data na to, aby mohl odhadnout tuto závislost správně, například pokud se v minulosti cena nikdy nezměnila. Nedostatek dat pro aproximaci poptávkové křivky může být dalším důvodem, který může vyústit v nesmyslné parametry některých závislostí, které neumožní optimální cenu nalézt.

Zároveň takové případy mohou nastat, když cena překročí některé z omezení, ať už pokles pod nákupní cenu, překročení maximální ceny nastavené ze strany klienta či mnoho z dalších volitelných omezení, jako je například maximální relativní změna optimalizované ceny proti ceně současné.

Právě z těchto důvodů zdaleka ne všechny produkty, které do optimalizace vstupují, jsou ve finále optimalizovány. Nicméně ty produkty, u nichž je nalezena validní optimální cena následně vstupují do funkce *merchant price*, která cenu zaokrouhluje na takový formát, který je již možné vložit na e-shop, jelikož cena z optimalizéru je typicky ve tvaru např. 744.505475778833, je tedy obvykle nutné ji podle odpovídajícího businessového zadání zaokrouhlit, obvykle na tzv. baťovskou cenu končící typicky číslicí 9, v příkladu výše by to

tedy mohlo být např. 749, nicméně konkrétní pravidla je pochopitelně možné upravit podle požadovaného zadání.

V případě, kdy produkt není vhodný pro optimalizaci ceny z některého z výše uvedených důvodů, využít metody tzv. mnohorukého bandity neboli v originále multi-armed bandit, který například Google využívá pro experimentování v online oblasti. Jedná se o testování několika alternativních scénářů s cílem nalézt ten nejvhodnější a ten následně použít (Scott, 2014). V kontextu tohoto řešení se tedy jedná o vyzkoušení několika různých cenových hladin a přepnutím na tu nejvýnosnější podle definovaného pravidla. V kontrastu s optimalizací pomocí poptávkových křivek lze pomocí této metody dosáhnout zlepšení i v případě, že není dostatek dat na vytvoření poptávkové křivky či křivka nesplňuje určité minimální podmínky kvality, tedy například vychází poptávková křivka rostoucí, nedefinovaná, či jsou její parametry mimo stanovené limity.

5.3 Shrnutí kapitoly

V této kapitole byl představen současný stav technického řešení, včetně struktury a objemu výchozích dat, respektive dvou stěžejních tabulek *order_item* a *pois_demand_input*. Poté je popsána současná logika a posloupnost kroků při počítání optimálních cen pro jednotlivé produkty od vstupních dat až po konečnou cenu, kterou lze nasadit na e-shop. Tato kapitola je tak nezbytným vstupem do následující, kde je nad výchozími daty provedena analýza nákupního košíku a zároveň vstupem do kapitoly 7, ve které je toto současné řešení upraveno a rozšířeno o nové prediktory.

6 Identifikace komplementů

Následující kapitola popisuje podrobně postup a komentuje skripty v jazyce R, které byly využity pro market basket analysis k analýze historie transakcí a identifikaci komplementárních produktů v produktovém katalogu klienta. Výstupy z této kapitoly jsou nezbytným vstupem pro rozšíření modelů v kapitole následující, které počítají na datech poptávkové křivky, které slouží pro nastavení optimální ceny produktu a zároveň pro splnění dílčích cílů analyzovat a popsat postupy a možnosti analýzy nákupního košíku a identifikovat v datech o transakční historii jednotlivé komplementární produkty a naplnit tak i část z přínosu této práce.

Pro identifikaci komplementů bude využito knihovny *arules*, která poskytuje infrastrukturu pro snadnou reprezentaci, manipulaci i analýzu transakčních dat a dolování frequent itemsetů a asociačních pravidel pomocí algoritmu apriori, jehož princip byl podrobněji popsán v teoretické části práce v kapitole 4.2.

Zároveň je využita i nadstavba této knihovny, a to *arulesViz,* která slouží k vizualizacím pravidel a itemsetů vydolovaných knihovnou *arules*, čímž značně usnadňuje následnou analýzu výstupů.

6.1 Vyčištění dat

Prvním krokem, který je potřeba provést, je získání relevantních dat z hlavní tabulky *orderitem*, která obsahuje informace o transakcích. Jelikož mezi transakcemi se mohou vyskytovat i data, která pro analýzu nejsou nutná, je třeba tato data vyčistit. Jedná se především o pomocné položky v transakcích, které například indikují slevovou akci či dárky k objednávce, které nijak neindikují, že zboží zákazník skutečně chtěl zakoupit společně, takže by pouze zkreslovaly výsledky analýzy.

Nejjednodušším způsobem, jak zajistit, aby v tomto případě do analýzy vstupovaly pouze ty produkty, u kterých se následně počítají poptávkové křivky, je tabulku *orderitem* připojit k tabulce *demand_pois*, která již původní napočítané poptávkové křivky obsahuje. Tyto dvě tabulky lze propojit pomocí hodnoty *id_item*. Provedený inner join z celkového počtu 1 633 527 řádku vyfiltruje pouze ty položky, které jsou pro následnou analýzu relevantní, zbyde tedy 326 145 řádků. Tyto vyčištěné řádky již neobsahují žádné pomocné položky jako je doprava, uplatnění slevových kuponů či dárky k objednávce a zároveň odstraňují i pro-

dukty, které z různých businessových i technických důvodů vůbec nevstupují do optimalizace, a tudíž nemají napočítané křivky, které by bylo možné rozšířit o vliv nalezených komplementárních produktů.

Výše popsanými filtry byla data očištěna o položky nežádoucí pro analýzu nákupního košíku a do dalšího kroku tedy vstupuje pouze zhruba pětina z celkového počtu dat, jedná se však pouze o data relevantní.

Tato základní příprava dat pro další analýzu byla implementována přímo do SQL dotazu, kterým se dotazují data z databáze, jak lze vidět na výpisu 1. Volaná funkce sql_db je definovaná v samostatném souboru sdílených často používaných funkcí. Využívá knihovny RODBC k vytvoření připojení na MS SQL server a zavolání dotazu pomocí funkce sqlQuery – z konfiguračního souboru přečte nezbytné údaje pro připojení, jako je název databáze a při jejím volání tedy stačí jako parametr pouze zadat požadovaný SQL dotaz. Zároveň, na rozdíl od původní funkce sqlQuery obsahuje také užitečné výpisy pro debugging a po jejím dokončení se již nevyužívané připojení k databázi opět uzavře pomocí příkazu *close()*.

Výpis 1: *Dotaz pro data do analýzy nákupního košíku (zdroj: autor)*

```
orderitem <- sql_db('select oi.id_item
                , id_order_transaction
                , date_created
                , obs_date
                , Deleted
                , PriceInclVAT
                , UnitPrice
                , Unit
                , Quantity
                , item_code
                , Name
                from dev_datamart.orderitem oi
                inner join development.pois_demand_input pdi
                    on oi.id_item = pdi.id_item')
```

6.2 Transformace do transakčního formátu

Poté, co jsou všechna data načtena do data framu *orderitem* v prostředí RStudio, je ve druhém kroku nutné transformovat je do formátu, který je vhodný pro dolování asociačních pravidel. K tomu je nutné seskupit jednotlivé položky po transakcích, tedy vytvořit si jed-

notlivé nákupní košíky, aby bylo možné zkoumat, jaké produkty jsou často nakupovány společně. Toho bylo docíleno pomocí knihovny *dplyr,* která je poměrně standardní knihovnou používanou pro různé transformace dat v jazyce R. Díky použití operátoru „%>%" lze přehledně přesměrovat výsledek každého kroku do kroku dalšího, tedy i složité transformace o mnoha krocích zapsat poměrně přehledně a po částech. Kód ve výpisu níže transformuje data z původního tvaru do tvaru, kde bude pro každé id transakce jeden řádek a následně seznam produktových kódů zakoupených položek oddělených čárkou.

Výpis 2: *Seskupení položek podle jednotlivých transakcí (zdroj: autor)*

```
# Group data into baskets
grouped_data <- orderitem %>%
  select(id_order_transaction, id_item) %>%
  group_by(id_order_transaction) %>%
  summarise (id_item = paste(id_item, collapse = ",")) %>%
  select(id_item)

# Save baskets into csv file
path_csv <- "transactions.csv"
write.csv(grouped_data, path_csv, quote = FALSE, row.names = FALSE)

# Load data into transaction object format
transaction_data <- read.transactions(path_csv, format = 'basket', sep=',')
```

Samotné ID transakce nicméně pro následnou analýzu nemá žádnou informační hodnotu (pro vydolování asociačních pravidel je důležité, které produkty jsou kupovány společně, ale ne v kterých konkrétních košících tomu tak bylo). Vzhledem k tomu, že každý řádek znamená jednu transakci, je jej tudíž možné odstranit bez jakékoliv informační ztráty a vybrat příkazem *select* pouze proměnnou *id_item.*

Takto přetransformovaná data jsou uložena do csv souboru. V případě definice cesty jako v příkladu výše bude soubor uložen do aktuálního working directory nastaveného v R Studio, lze ale definovat i absolutní cestu, kam soubor uložit. V případě, že soubor v cílovém místě již existuje, bude tímto příkazem nahrazen. Další vhodnou alternativou k absolutní definici cesty je využití funkce *sprintf* a parametru „%s" přímo v řetězci, kde je definována cesta, následovaným odkazem na proměnnou, ve které je uložena společná část cesty (typicky vedoucí do projektového adresáře), čímž lze zajistit znovupoužitelnost kódu a snadnou modifikaci cesty v případě její změny pomocí její definice například v konfiguračním souboru.

Poté jsou data načtena funkcí *read.transactions* z knihovny *arules* do proměnné *transaction_data*. V tomto formátu lze s daty již poměrně snadno pracovat funkcemi z knihovny *arulesViz* a data prozkoumat či vizualizovat.

Uložení do finální proměnné přes soubor csv zde má dva hlavní důvody – za prvé je to uložení mezivýsledku a s tím spojené odstranění nutnosti se pro data stále dotazovat do databáze v případě ladění kódu či testování různých parametrů a za druhé krkolomné načítání funkcí *read.transactions* přímo z data framu.

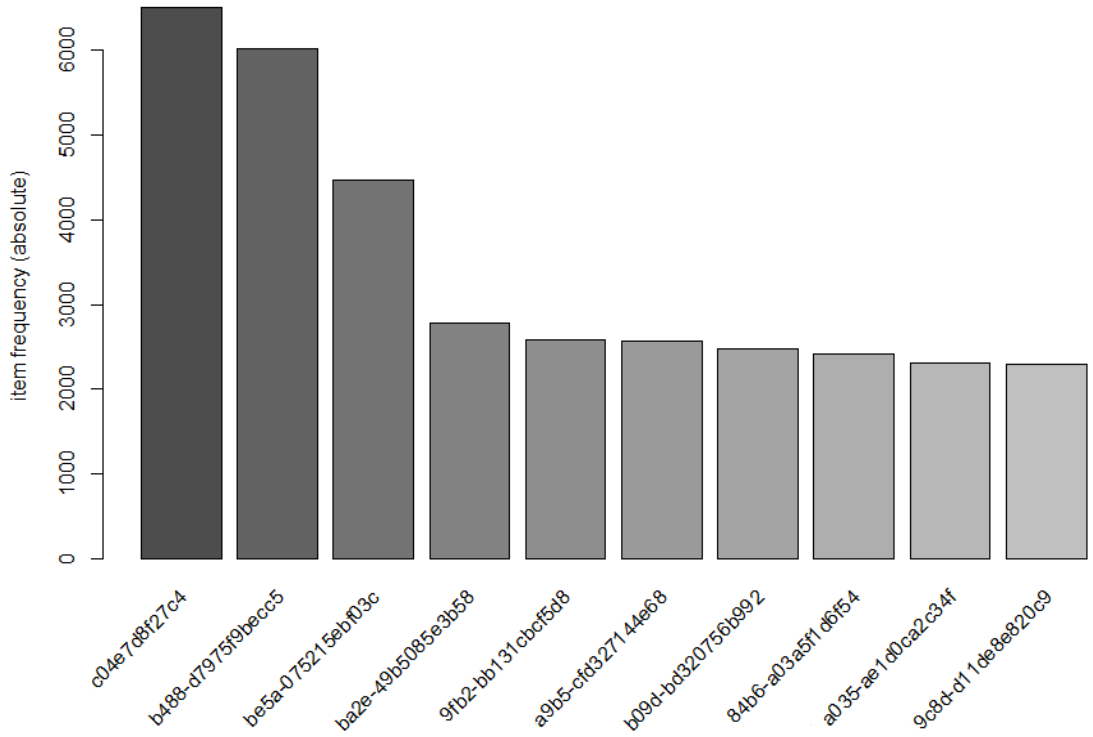
V transakčním formátu lze nad daty také sestrojit některé základní vizualizace, například počty výskytů v transakcích pro jednotlivé produkty, na obrázku níže lze vidět příklad pro 10 nejčastěji nakupovaných produktů, přičemž jejich ID bylo z důvodu velké délky, a tedy nutné velikosti celého grafu pro účely vizualizace v této práci zkráceno, skutečné řešení však obsahuje ID celé. Nejčastěji nakupované produkty tedy mají vzhledem k celkovému počtu transakcí, který je 128 365, podporu blížící se 5 %. Zároveň však zhruba po prvních třech nejčastěji nakupovaných produktech podpora klesá a už na konci první desítky se pohybuje okolo 2 %. Při hledání optimální hodnoty minimální podpory při samotném dolování pravidel v další podkapitole je tedy nutné vzít tato čísla v potaz a pro získání dostatečného množství pravidel uvažovat ještě řádově nižší hodnoty.

Sestrojit tento graf a ulož jej do souboru ve formátu png pak lze příkazy, které lze vidět na výpisu níže. Je zde pro ukázku využito již zmiňované funkce *sprintf* v kombinaci s parametrem „%s“, který za první část cesty dosadí hodnotu uloženou v proměnné *outdata*.

Třetí řádek tohoto výpisu slouží k nastavení odsazení, právě kvůli dlouhým popiskům osy x. Poslední řádek slouží k zavření vykreslovací plochy a uložení souboru.

Výpis 3: *Vygenerování grafu nejfrekventovanějších produktů (zdroj: autor)*

```
png(filename = sprintf("%s/absolute_item_freq.png", outdata), width = 1000,
height = 700)
par(mar = c(7,7,4,1) + .1)
itemFrequencyPlot(transaction_data, topN=10, type="absolute",
col=grey.colors(10), main="Graf absolutni frekvence produktu")
dev.off()
```



Obrázek 6 : *Graf nejfrekventovanějších produktů (zdroj: autor)*

6.3 Vydolování asociačních pravidel algoritmem apriori

Třetím krokem je již samotné dolování asociačních pravidel. V knihovně *arules* toho lze docílit velmi jednoduše, a to zavoláním funkce *apriori*. Ačkoliv samotná technická implementace je velice jednoduchá, jak lze i vidět na výpisu 4 níže, je na uživateli, aby určil hodnoty minimální podpory a spolehlivosti. Nalézt jejich optimální hodnoty je pak nejsložitější částí tohoto kroku, jelikož neexistuje zcela jasně dané pravidlo, podle kterého by se dalo v nastavování těchto parametrů řídit.

Nastavení optimálních hodnot těchto parametrů je totiž silně závislé na výchozích datech a na každém datasetu tak mohou dávat smysl zcela odlišné hodnoty. V příkladu v teoretické části práce například nebylo výjimkou, že podpora dosahovala hodnot v desítkách procent. Pokud nicméně v datech v tomto praktickém případu budeme hledat produkty, které by se vyskytovaly řádově v desítkách procent všech transakcí, nebudeme pochopitelně mít takové štěstí, pokud z dat odstraníme například poštovné, jak bylo provedeno v předchozím kroku, které by bylo pravděpodobně jediným takovým kandidátem.

V případě podpory se tedy musíme spokojit s mnohem menším číslem. K hodnotám, které dávají smysl, se lze propracovat nejsnáze zkoušením a dosazováním různých hodnot a následné analýze výsledného počtu a parametrů vydolovaných pravidel. Při dolování na datech v této práci byly postupně nastavovány vybrané hodnoty od 1 % (tedy dosazení hodnoty 0,01) až do hodnoty 0,01 % (tedy hodnoty 0,0001) do parametru *supp*.

I druhý nastavovaný parametr spolehlivosti se však může na různých datech poměrně dramaticky lišit, jelikož ne ve všech datech bude možné skutečně najít dostatek pravidel s vysokou hodnotou spolehlivosti kolem 80 %, jak to bylo v ilustračním příkladu v teoretické části práce. I zde tedy docházelo k postupnému zkoušení hodnot do parametru *conf*, a to mezi 80 % a 50 %.

Posledním parametrem, který je možné nastavit, je maximální délka vydolovaného pravidla. Pokud algoritmus vydoluje pravidla do této délky, nebude pokračovat v hledání delších pravidel, ani pokud by to jinak stále hodnoty minimální podpory a spolehlivosti stále umožnily. Pro následně využití v začlenění do modelů se v této práci soustředíme maximálně na tři nalezené komplementy, nicméně tento filtr je po srovnání pravidel proveden v následujícím kroku po vydolování pravidel, tedy tento parametr zde není využit. Pokud by však čtenář tento parametr nastavit chtěl, lze tak udělat pomocí vložení požadované maximální hodnoty do parametru s názvem *maxlen*.

Obdobně lze také nastavit minimální délku pravidla, pokud je v daném kontextu vhodná, a to pomocí parametru *minlen*. V případě nastavování obou parametrů musí být *minlen* pro správné fungování samozřejmě nižší než *maxlen*.

Po vyzkoušení několika kombinací hodnot podpory a spolehlivosti byly zvoleny hodnoty 0,1 % jako minimální podpora a 50 % jako minimální spolehlivost, jelikož při tomto nastavení bylo vydolováno dostatečné množství pravidel, která však zároveň stále byla dostatečně častá a silná, aby mohla vnést informační hodnotu do modelování poptávkových křivek.

Výpis 4: *Vydolování pravidel (zdroj: autor)*

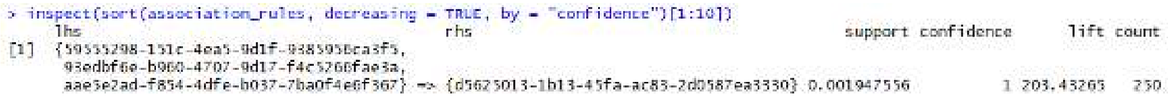
```
association_rules <- apriori(transactionData, parameter = list(supp = 0.001,
conf = 0.5))
```

Tento krok provedl analýzu transakčních dat pomocí algoritmu apriori a vydolovaná data a jejich parametry uložil do objektu typu Large rules s názvem *association_rules*. Na tyto výsledky se lze poměrně intuitivně dívat například skrze metodu *inspect*, či si vydolovaná pravidla vizualizovat některým z grafů z importované knihovny *arulesViz*.

Výpis 5: *Analýza vydolovaných pravidel metodou inspect (zdroj: autor)*

```
inspect(sort(association_rules, decreasing = TRUE, by = "confidence")[1:10])
```

Výše zavolaný řádek kódu například vypíše 10 pravidel s největší spolehlivostí. Formát výstupu lze vidět na obrázku níže, kde *lhs* představuje předpoklad a *rhs* závěr. Na příkladu níže to tedy znamená, že v každém z 250 případů, kdy se vyskytují v nákupním košíku tři produkty vypsané na levé straně, se tam také vyskytuje produkt na straně pravé.

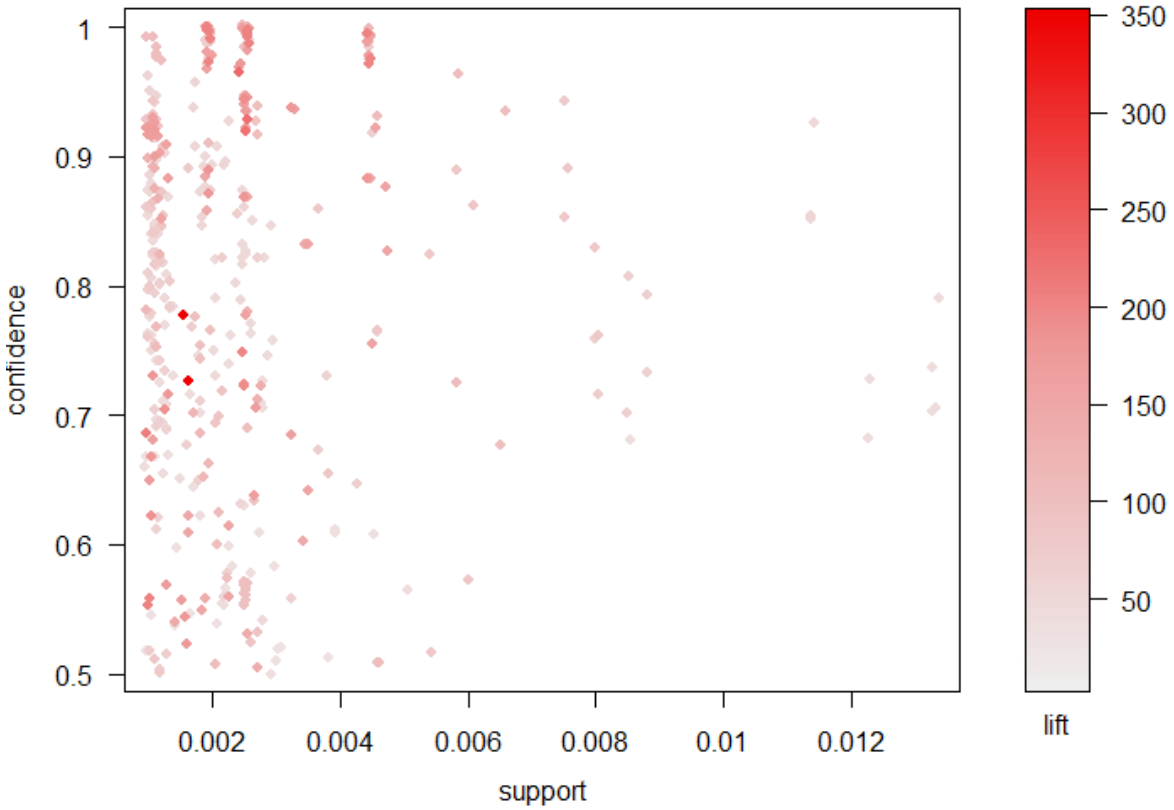


Obrázek 7: *Formát výstupu funkce inspect (zdroj: autor)*

6.4 Vizualizace pravidel pomocí knihovny arulesViz

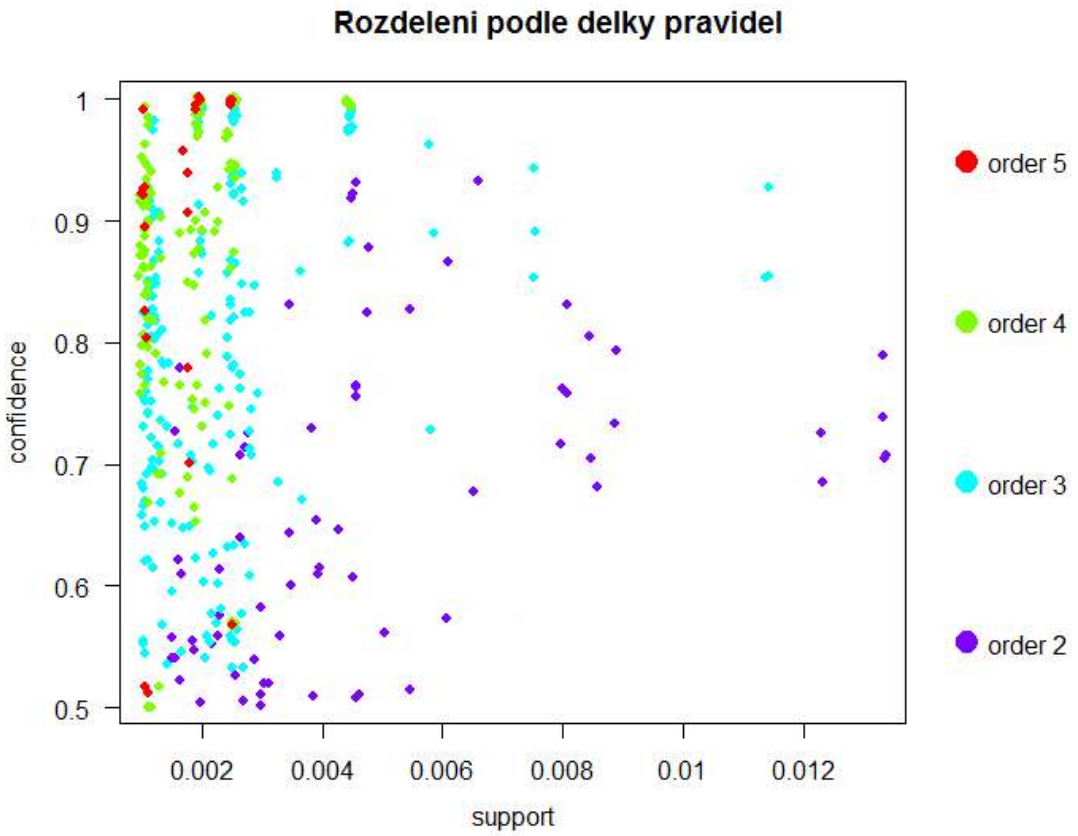
Další možností, jak lze zanalyzovat vydolovaná pravidla, je pomocí některé z vizualizací, které jsou dostupné ve zmíněné knihovně *arulesViz*. V následujících odstavcích jsou tedy vybrané typy vizualizací popsány i znázorněny na příkladech a níže lze zároveň na výpisu vidět, jak je lze v jazyce R vytvořit.

Základní pohled na vydolovaná pravidla může nejsnáze poskytnout bodový graf, tedy scatter plot. Na příkladu níže lze snadno vidět distribuci podle podpory a spolehlivosti – například že drtivá většina vydolovaných pravidel má podporu pod polovinou procenta. Body lze zároveň obarvit podle hodnoty liftu, lze tak zároveň pozorovat, že pravidla s vysokou hodnotou podpory mají lift velmi malý v porovnání s těmi s nízkou podporou a vysokou spolehlivostí.



Obrázek 8: *Graf distribuce pravidel podle podpory a spolehlivosti (zdroj: autor)*

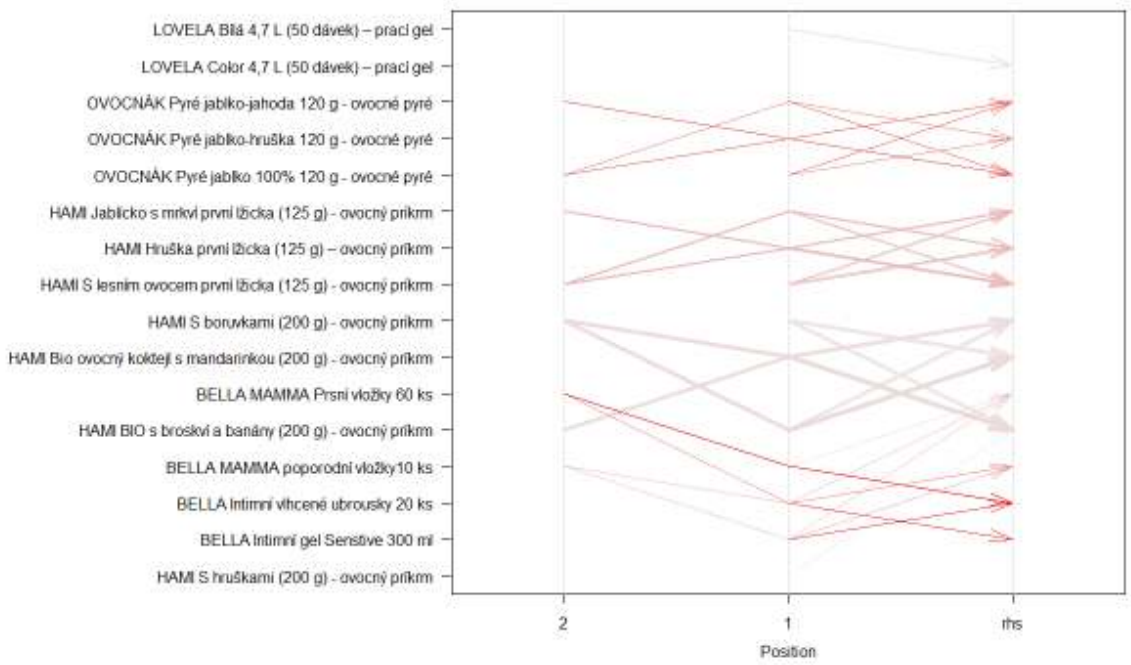
Vizualizovat jde pravidla také například podle jejich délky a sledovat, jak se v závislosti na ní mění podpora a spolehlivost vydolovaných pravidel. Příklad takového grafu lze vidět na obrázku níže. Z něj lze například vyčíst trend, že delší pravidla mají obecně nižší podporu, ale na druhou stranu obvykle nejvyšší hodnoty spolehlivosti, naproti tomu mezi pravidly s nejvyšší podporou se vyskytují téměř výhradně pravidla délky 2.



Obrázek 9: *Graf rozdělení pravidel podle jejich délky (zdroj: autor)*

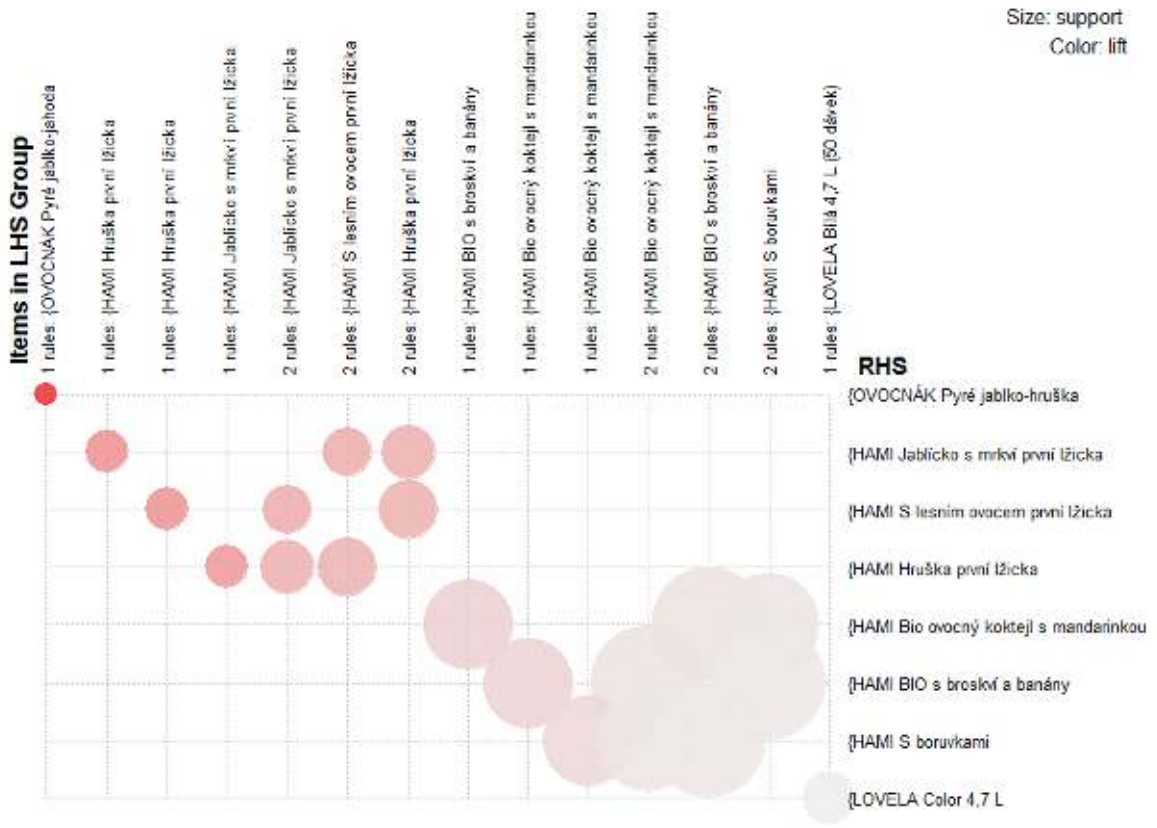
Další možností, jak vizualizovat vydolovaná pravidla je graf typu parallel coordinates, ze kterého lze dobře vidět, které produkty v košíku vedou k nákupu konkrétního produktu. Jistým problémem tohoto grafu je, že se poměrně rychle stává nepřehledným pro větší množství pravidel, nejvíce se tedy hodí pro vizualizace vybrané skupiny pravidel, například prvních 20 pravidel s největší spolehlivostí či podporou, pro získání lepší představy o některých konkrétních vzorcích, které se v datech vyskytují.

Pro vygenerování tohoto grafu byly pro názornost a přehlednost upraveny pravidla z ID jednotlivých produktů na jejich názvy. Ačkoliv pro skutečné využití vydolovaných pravidel pro začleněné do modelů je nutné mít výstup ve formátu identifikačních kódů produktů, pro vizualizace konkrétních pravidel je mnohem názornější a přínosnější vidět, co se za kódy skutečně skrývá za produkty. Pro vygenerování grafu typu parallel coordinates na příkladu níže bylo použito 40 vydolovaných pravidel s nejvyšší podporou.



Obrázek 10: *Ukázka grafu parallel coordinates (zdroj: autor)*

Z grafu výše lze tedy vidět tendenci vydolovaných pravidel přirozeně se shlukovat podle odpovídajících kategorií či výrobců, jako lze pozorovat především na grafu nahoře s pracím gelem či ve spodní části u produktů Bella Mamma, či přímo různé příchutě stejného produktu, jak lze pozorovat v prostřední části grafu. Šipka pak směřuje od jednotlivých předpokladů, v případě delších pravidel odzadu přes všechny produkty, tedy od produktů, které algoritmus apriori do pravidla přidal jako poslední, směrem k závěru na pravé straně na grafu označeném jako *rhs* (right hand side). Tloušťka šipky znázorňuje podporu (větší tloušťka znamená vyšší podporu) a barva lift pravidla (tmavší znamená vyšší hodnotu liftu), nicméně toto nastavení lze změnit pomocí parametru *shading*.



Obrázek 11: *Ukázka grafu grouped matrix (zdroj: autor)*

Jiný pohled na podobnou problematiku nabízí graf grouped matrix na obrázku výše. Ten znázorňuje levou stranu pravidla na vertikální ose a pravou stranu pravidla na horizontální a vytváří tedy tabulku, ve které v bodech, kde se nachází odpovídající pravidla, vykreslí kruhy. Jejich poloměr znázorňuje podporu pravidla a barva lift. Z tohoto grafu lze tedy například vidět, že pravidla s vysokou podporou mají obvykle menší lift a zároveň potvrdit závěry z předchozího grafu parallel coordinates o častém shlukování produktů po značkách.

V následujícím výpisu kódu lze nalézt konkrétní implementaci těchto grafů. Každý graf v kódu navíc obsahuje pro přehlednost parametr nastavující jeho nadpis pomocí nastavení parametru „main“, které byly na obrázcích v této práci vzhledem k popiskům obrázků vynechány.

Výpis 6: *Vizualizace v arulesViz (zdroj: autor)*

```
plot(association_rules_names,
     measure = c("support", "confidence"),
     shading = "lift",
     control = list(main = "Rozdeleni podle podpory a spolehlivosti"))

plot(association_rules_names,
```

```
     method = "two-key plot",
     control = list(main = "Rozdeleni podle delky pravidel"))

plot(sort(association_rules_names, decreasing = TRUE, by =
"confidence")[1:10],
     method = "graph",
     control = list(type = "items", main = "Graf 10 pravidel s nejvetsi
podporou"))

plot(head(association_rules_names, n = 40, by = "support"),
     method = "paracoord",
     control = list(reorder = TRUE, main = "Priklad grafu parallel
coordinates pro 40 pravidel s nejvetsi podporou"))

plot(head(association_rules_names, n = 20, by = 'support'),
     method="grouped",
     control = list(reorder = TRUE, main = "Priklad grafu grouped matrix pro
20 pravidel s nejvetsi podporou"))
```

V případě spouštění následujícího kódu hromadně či automaticky zavoláním z jiné metody může být vhodné výsledky vizualizací ukládat do souborů pro pozdější analýzu. Toho lze docílit poměrně snadno, například pro uložení do obrázku ve formátu png stačí zavolat funkci *png*, jako na příkladu ve výpisu 3 a nastavit cestu, název a požadovanou velikost. Ve výpisu 7 *7*níže je navíc na druhém řádku manuální nastavení odsazení z důvodu dlouhých popisků jednotlivých produktů. Po vykreslení každého z grafů do samostatného souboru je nakonec nutné zavolat funkci dev.off(), která zavře vykreslovací plochu a dojde k uložení souboru.

Výpis 7: *Uložení grafu do souboru (zdroj: autor)*

```
png(filename = 'grouped_matrix.png', width = 1000, height = 700)
par(mar = c(7,7,4,1) + .1)
plot(head(association_rules_names, n = 20, by = 'support'),
     method="grouped",
     control = list(reorder = TRUE, main = "Priklad grafu grouped matrix pro
20 pravidel s nejvetsi podporou"))
dev.off()
```

Posledním krokem po analýze pravidel a případné úpravě některých z parametrů a jejich přegenerování lze finální pravidla připravit do formátu, ve kterém mohou vstupovat do modelů, respektive ve kterém je lze vložit do vstupních dat, ze kterých modely počítají poptávkové křivky.

Tato transformace je provedena ještě v jazyce R před exportem do csv souboru a využívá opět především funkce knihovny *dplyr*. Prvním nezbytným krokem je uložení asociačních pravidel do objektu typu data frame, na kterém lze všechny další transformace provádět. Druhým je pomocí regulárních výrazů vyčistit veškeré pomocné znaky a závorky, které by dále překážely a rozdělit pravidla na levou a pravou stranu. V dalším kroku jsou všechna delší pravidla rozdělena po řádcích. Příklad pravidla vypsaného na obrázku s metodou *inspect* výše je v tomto kroku tedy rozdělen na tři řádky a v každém z nich je pouze jeden z produktů z levé strany. Dalším krokem je seřazení pravidel pro každý produkt na pravé straně podle spolehlivosti, deduplikace produktů na straně levé a poté vyfiltrování pouze 3 pravidel s nejvyšším liftem pro každý z produktů na straně pravé pomocí metody rank s parametrem -lift, kde znaménko mínus znamená seřazení sestupně a následně srovnáním podle ranku metodou arrange. Konkrétní implementaci lze vidět na výpisu níže.

Výpis 8: *Vyfiltrování a transformace pravidel do výstupního formátu (zdroj: autor)*

```
rules_df <- as(association_rules, "data.frame")

rules_clean <- regmatches(rules_df$rules, gregexpr("{([^}]*)}",
rules_df$rules, perl=T)) %>%
  lapply(., function(x) gsub("[{}]","",x)) %>%
  lapply(., function(x) cbind(left = strsplit(x[1],",")[[1]], right=x[2]))

rules_df <- cbind(rules_df[rep(1:nrow(rules_df), sapply(rules_clean,
nrow)),], do.call(rbind, rules_clean)) %>%
  select(left, right, support, confidence, lift, count) %>%
  group_by(right) %>%
  mutate(rank = rank(-lift, ties.method = "first")) %>%
  arrange(left, right, rank) %>%
  distinct(left, right, .keep_all = TRUE)  %>%
  filter(rank <= 3) %>%
  summarise (left = paste(left, collapse = ",")) %>%
  separate(left, c("comp1", "comp2", "comp3"), sep = ",")

write.csv(rules_df, 'rules.csv', quote = FALSE, row.names = FALSE)
```

Po této transformaci jsou data ve formátu, kde je na každém řádku produkt a k němu 3 jeho nalezené komplementy s největší spolehlivostí. Posledním krokem je zapsání finálního tvaru do csv souboru, případně je možné data rovnou uložit do databáze pro pozdější jednodušší připojení do tabulky se vstupními daty pro trénování modelů.

Výsledkem je tabulka obsahující 88 produktů a k nim odpovídající ID jednoho až tří komplementů každého z nich, které jsou v následující kapitole využity k rozšíření vstupních dat. Tato data jsou následně uložena do databáze, pro účely této práce je navíc vytvořen csv soubor, jelikož databáze, na které je toto řešení vyvíjeno, je již pouze pro čtení.

6.5 Shrnutí kapitoly

Tato kapitola čtenáře seznámila s praktickou implementací dolování asociačních pravidel v datech algoritmem apriori. Poskytuje komentované výpisy kódu v jazyce R, jak lze pomocí funkcí z knihovny *arules* pravidla nalézt, i jak je lze analyzovat například pomocí různých typů vizualizací z knihovny *arulesViz*.

Došlo zde tedy i k samotné identifikaci asociačních pravidel, respektive komplementů, které v rámci této práce jsou dále využity k rozšíření existujících modelů dynamické cenotvorby. Na konci kapitoly byla tedy vydolovaná pravidla transformována do požadovaného výstupního formátu, aby mohla být dále využita jako vstup do modelů.

Tato kapitola tak naplňuje první dva dílčí cíle, tedy analýzu a popis možností a postupů analýzy nákupního košíku a samotné vydolovaní asociačních pravidel z dat o transakční historii a tedy identifikaci jednotlivých komplementárních produktů v produktovém katalogu. Její výstup v podobě vydolovaných pravidel je zároveň nezbytným vstupem do kapitoly následující, ve které jsou tyto komplementy využity k rozšíření modelů.

7 Rozšíření stávajícího řešení a porovnání

V následující kapitole jsou nově vydolovaná asociační pravidla přidána do vstupních dat, následně do současné logiky modelů, poptávkové křivky jsou přepočítány s novými prediktory, a nakonec porovnány s původními.

7.1 Rozšíření vstupních dat

Prvním krokem pro rozšíření stávajícího řešení je opět příprava dat. V tomto případě je tedy nutné vrátit se k již popisované tabulce *pois_demand_input* a do ní vložit nové sloupce obsahující potřebné vlastnosti jednotlivých identifikovaných komplementů. Poté je nutné upravit kód pro výpočet poptávkových křivek a přidat do něj nové prediktory odkazující na proměnné, které byly do dat nově přidány. Následuje již pouze přepočítání křivek s novými daty, jejich export a analýza a porovnání jejich vlastností s křivkami původními.

V předchozí kapitole byla vydolovaná pravidla uložena do csv souboru, je tedy nutné dostat je do databáze, aby bylo možné přidat je do tabulky *pois_demand_input*. V případě, že jsou rovnou uložena do tabulky ve stejné databázi, tato nutnost odpadá a tabulky lze rovnou propojit. Tabulka *pois_demand_input*, jak je blíže popsáno v kapitole 6, obsahuje historii vybraných produktů, která slouží pro namodelování poptávkových křivek, vložit do ní tedy jako nové sloupce ID jednotlivých komplementů nevnese do modelů žádné nové informace. Je nezbytné použít některou z jejich charakteristik, jako je cena či poptávka po komplementu, tedy jeho prodané množství. Toho lze docílit pomocí SQL kódu, který je přiložený na výpisu níže.

Výpis 9: *Příprava vstupních dat pro modely (zdroj: autor)*

```
with pdi_comp
as (
SELECT pdi.*
  , complements.comp1 as comp1
  , complements.comp2 as comp2
  , complements.comp3 as comp3
  FROM pois_demand_input pdi
  left join complements on pdi.id_item = complements.id_item
    )

  select pdi_comp.*
  , comp1.q1 as comp1_q1
  , comp1.x as comp1_x
```

```
  , comp2.q1 as comp2_q1
  , comp2.x as comp2_x
  , comp3.q1 as comp3_q1
  , comp3.x as comp3_x
  from pdi_comp
  left join pdi_comp comp1 on pdi_comp.comp1 = comp1.id_item and
pdi_comp.[week] = comp1.[week]
  left join pdi_comp comp2 on pdi_comp.comp2 = comp1.id_item and
pdi_comp.[week] = comp2.[week]
  left join pdi_comp comp3 on pdi_comp.comp3 = comp1.id_item and
pdi_comp.[week] = comp3.[week]
  order by 1, 2
```

Zde jsou vydolované komplementy uloženy v tabulce *complements* ve tvaru, ve kterém byly vyexportovány do csv souboru, tedy ID produktu v první sloupci a ID jeho komplementů v následujících 3 sloupcích. V následujícím kroku je tabulka *pois_demand_input,* obsahující historii, ze které modely vychází, rozšířena o data o komplementech. Toho bylo docíleno pomocí self-joinu a ke každému řádku historie produktu byl doplněn odpovídající řádek vydolovaného komplementu – tedy jeho cena v daném časovém období, respektive prodané množství v minulém časovém období. V příkladu výše je toto období stanové na týden.

Přidání nejen ceny komplementu, ale též jeho prodaného množství v minulém období je v tomto konkrétním případě snaha zvětšit přínos navzdory jistému specifiku výchozích dat.

Vydolovaná pravidla, jak bylo vidět v předchozí kapitole 6, jsou totiž poměrně často různé příchutě či druhy stejného produktu. Ty jsou však v současnosti na e-shopu přeceňovány obvykle společně, přidání ceny takových komplementů by tedy nepřineslo vůbec žádnou informační hodnotu, pouze by došlo k vložení několika prediktorů se zcela stejnými hodnotami. Toto je více přiblíženo v podkapitole 7.3. V případě dat, kde k podobnému problému nedochází, však cena komplementu může být mnohem užitečnějším prediktorem. Pro rozšíření modelů v tomto případě je tedy využito poptávky po komplementu z minulého období, kde se prodaná množství mezi jednotlivými komplementárními produkty již liší, a tedy mohou představovat pro modely informační přínos.

7.2 Úprava modelů

Po rozšíření vstupních dat o nové prediktory vydolované pomocí analýzy nákupního košíku je nutné upravit současné regresní modely a přidat tyto nové prediktory do předpisu jejich funkce.

Prvním místem, kde je nutné upravit stávající řešení je při importu dat do prostředí RStudio a tedy přidat zde jak nové sloupce do SQL příkazu select, tak odpovídající beta parametry jako sloupce výsledného data framu, aby bylo kam napočítané hodnoty uložit.

Druhým místem je již samotná funkce, ve které dochází k počítání poptávkových křivek. V této funkci je třeba rozšířit předpis regresní rovnice a vložit do něj nové prediktory. Před jejich vložením je však ještě vhodné zkontrolovat, jestli ke konkrétnímu produktu existuje vybraný komplement, respektive hodnoty jeho poptávky, které jsou do rovnice začleňovány. Pokud ne, budou hodnoty tohoto prediktoru NULL, a tedy zhoršovat přesnost modelu.

To lze udělat například podmínkou, kterou lze vidět na výpisu 10 níže. Vzhledem k nutnosti pokrytí veškerých možných kombinací je však toto řešení spíše krátkodobé. Dlouhodobější řešení je začlenění těchto nových prediktorů do funkce, která podle předem daných požadavků (například na minimální množství pozorování každého z prediktorů) vybere ty, které jsou pro daný produkt vhodné a sestrojí z nich předpis poissonovy regrese bez nutnosti větvení podmínek v každém konkrétním případě.

Výpis 10: *Příklad rozšíření o vliv komplementů (zdroj: autor)*

```
if (length(which(!is.na(raw_data$comp1_q1)))>(nrow(raw_data)/2) &&
          length(which(!is.na(raw_data$comp2_q1)))>(nrow(raw_data)/2) &&
          length(which(!is.na(raw_data$comp3_q1)))>(nrow(raw_data)/2))
      {
        reg = step(reg0, direction='both', scope=(~ x + week + q1 + lc +
comp1_q1 + comp2_q1 + comp3_q1), trace = 0)
      }
```

Na výpisu 10 výše je také hned po podmínce vidět příklad samotného předpisu, v tomto případě obsahující jako prediktory cenu produktu (x), časové rozlišení (week), prodané množství v minulém období (q1), cenu konkurence (lc) a prodané množství všech tří komplementů v minulém období (comp_q1, comp2_q1 a comp3_q1). Výsledkem jsou vypočítané beta parametry pro každý z těchto prediktorů, případně hodnoty NULL pro ty z nich, které model identifikoval jako nevýznamné.

Třetím místem, kde je nutné upravit současný kód, je při linearizaci poptávkové křivky. Zde nedochází k žádné výrazné změně logiky, je však opět nutné do ní vložit nové prediktory, aby k linearizaci došlo správně. Znovu je zde nezbytné myslet na případy, kdy tyto prediktory vycházejí NULL, tedy je model nevyhodnotil jako signifikantní. V tom případě lze využít funkci *coalesce*, která parametr v případě, kdy vychází NULL, nahradí hodnotou 0.

Výpis 11: *Rozšíření linearizace o komplementy (zdroj: autor)*

```
    mutate(
      mu_0 = coalesce(as.double(beta_0), 0),
      mu_t = coalesce(as.double(beta_t), 0) * week,
      mu_c = coalesce(as.double(beta_xc), 0) * coalesce(as.double(x_c), 0),
      mu_q1 = coalesce(as.double(beta_q1), 0) * coalesce(as.double(q1), 0),
      mu_comp1 = coalesce(as.double(beta_comp1_q1), 0) *
coalesce(as.double(comp1_q1), 0),
      mu_comp2 = coalesce(as.double(beta_comp2_q1), 0) *
coalesce(as.double(comp2_q1), 0),
      mu_comp3 = coalesce(as.double(beta_comp3_q1), 0) *
coalesce(as.double(comp3_q1), 0),

      mu = mu_0 + mu_t + mu_c + mu_q1 + mu_comp1 + mu_comp2 + mu_comp3 +
beta_x*x,
      q0 = exp(mu),
      a = q0*(1 - beta_x*x),
      b = -q0*beta_x)
```

Pak již lze získat hodnoty *a* a *b*, kterými lze sestrojit linearizovanou funkci, a to pomocí výpočtu ve výpisu 11 výše, do kterého jsou přidány parametry *mu_comp1* až *mu_comp3*. Jelikož samotná funkce slouží až k počítání optimální ceny, není pro porovnání hodnot MASE v kontextu této práce sice zcela relevantní, nicméně jedná se o část celkového řešení, na které má přidání komplementů také vliv, proto je zde zmíněna.

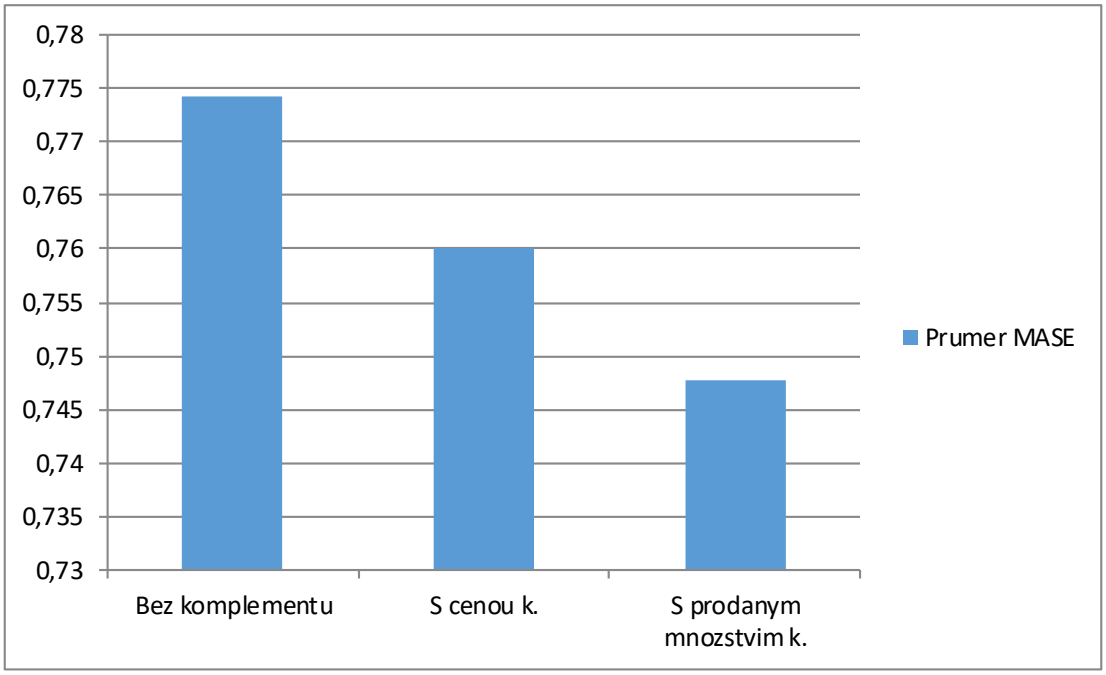
7.3 Porovnání rozšířených a původních modelů

Nově napočítané křivky je nyní třeba porovnat s křivkami, kde vliv komplementů chybí a analyzovat změny. Důležitou metrikou, která pomáhá při porovnání kvality těchto modelů, je v teoretické části práce popsaná metrika MASE, tedy mean absolute scaled error. Hypotézou je, že díky začlenění komplementů jednotlivých produktů jako nových prediktorů dojde ke snížení MASE, a tedy zlepšení přesnosti modelů, což jim umožní lépe a přesněji počítat optimální cenu a zvýšit tak přínos, který toto řešení cenotvorby nabízí.

Jak lze vidět na vizualizacích níže, původní hypotéza, že začleněním komplementů dojde ke zpřesnění původních modelů, se skutečně naplnila. Komplementy na poptávkové křivky původních produktů, a tedy na dynamickou cenotvorbu, vliv mají.

V případě využití cen komplementů však narazilo řešení na problém, který již byl zmíněn při rozšiřování vstupních dat, tedy poměrně častá kombinace komplementů, které mění svou

cenu společně, a tedy jejich začlenění nevnese do modelů žádné nové informace. Při využití cen komplementů jako nových prediktorů tak hodnoty MASE klesly jen nepatrně, jak lze vidět na následujícím grafu na obrázku 12. Když bylo do modelu místo toho začleněno prodané množství komplementů v minulém období, tedy *q1*, průměrné MASE kleslo o něco více, jak lze vidět na sloupci zcela vpravo.

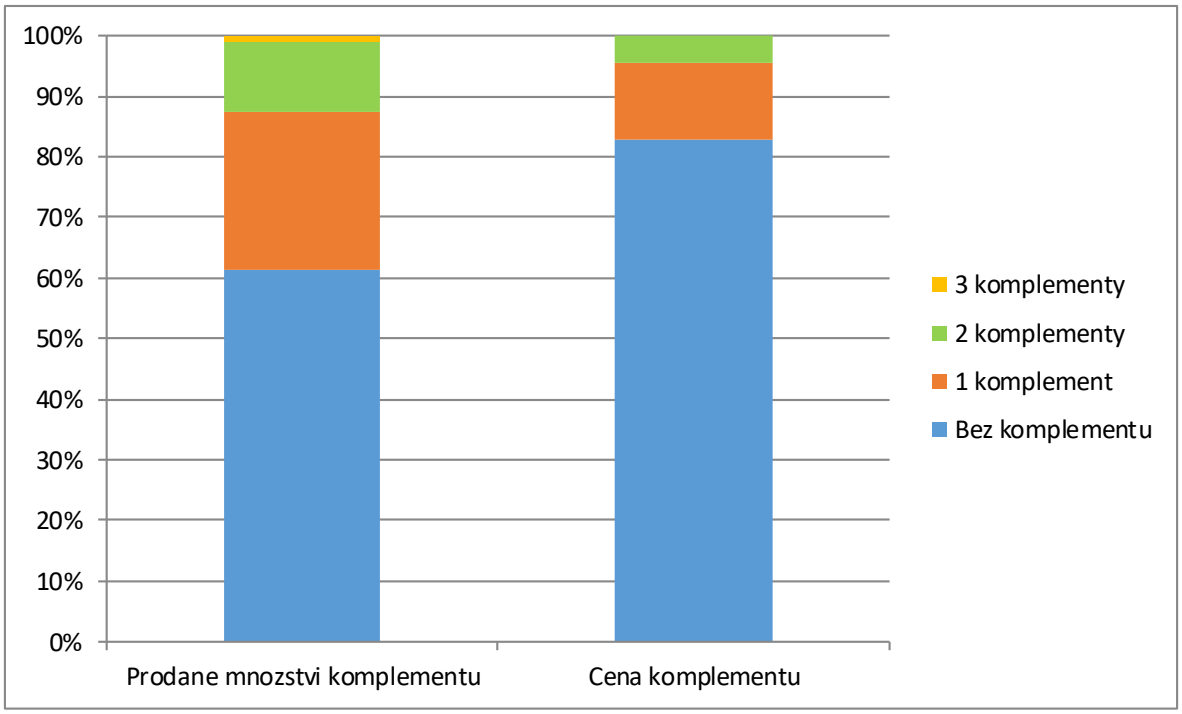


Obrázek 12: *Porovnání průměrných hodnot MASE (zdroj: autor)*

V případě využití prodaného množství se však zvýšil počet případů přeučení, tedy těch křivek, kdy došlo naopak ke zvýšení MASE. V 5 případech po přidání prediktoru prodaného množství komplementu hodnota MASE totiž naopak stoupla, oproti jedinému případu při začlenění ceny komplementu. K tomu pravděpodobně došlo v důsledku vyšší náhodnosti konkrétních hodnot prodaného množství, které se následně model snažil vystihnout. Poptávka po komplementu se nicméně alespoň v případě těchto konkrétních dat zdá být lepším prediktorem, který i navzdory více případům přeučení průměrnou chybu snižuje více.

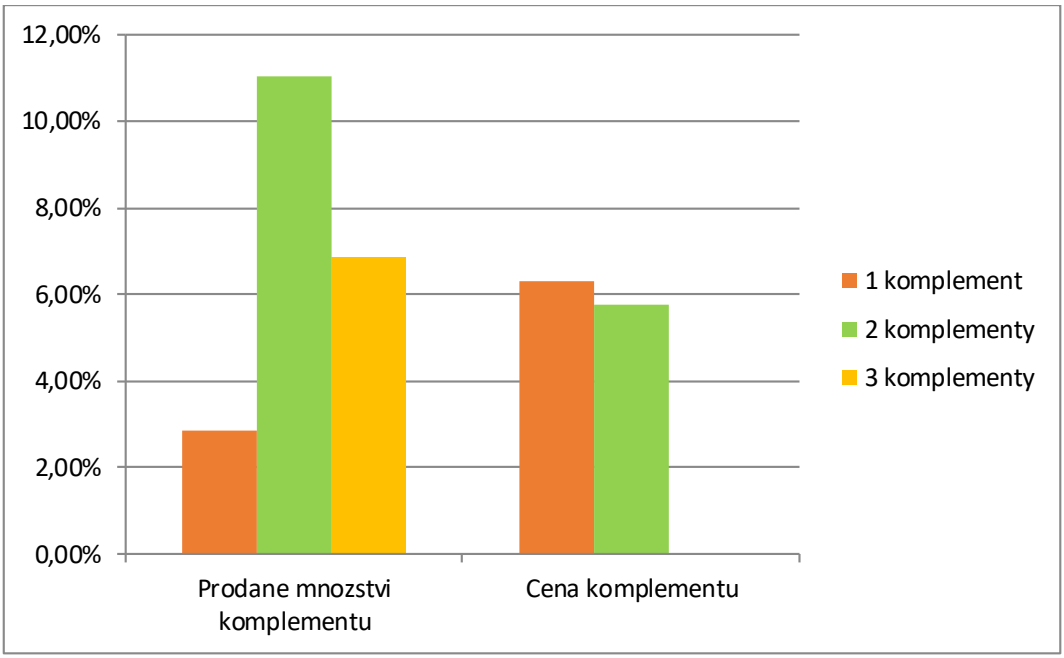
Na grafu níže lze následně vidět rozložení podle počtu komplementů, kterými byly modely rozšířeny, tedy kolik procent z 88 rozšiřovaných modelů bylo rozšířeno o odpovídající počet komplementů. Z něj lze vidět, že v případě ceny je mnohem více modelů, kde nebyl ani jeden z vydolovaných komplementů vyhodnocen jako signifikantní a do modelu začleněn, což souvisí právě se zmíněným přeceňováním určitých produktů společně, a tedy stejných hodnot ceny komplementu jako původního produktu.

V takových případech nedošlo k začlenění ceny komplementu do modelu, ale v případě prodaného množství se navzdory shodné ceně prodaná množství lišila. Právě možné náhodné výkyvy však v některých případech mohly vyústit spíše právě v přeučení modelu ve snaze tuto náhodnost vystihnout.



Obrázek 13: *Počet začleněných komplementů (zdroj: autor)*

Na třetím grafu na obrázku 14 lze vidět, jak se relativně zlepšila přesnost modelů v závislosti na počtu přidaných komplementů jako nových prediktorů. Přesnost je zde vyjádřena jako procentuální hodnota, o kolik je nižší MASE rozšířeného modelu od původního. Z grafu lze vidět, že není zcela jasná korelace mezi množstvím přidaných komplementů a nižším MASE, jelikož k největšímu rozdílu došlo u modelů se dvěma novými prediktory představujícími prodané množství dvou komplementů, kdy byl rozdíl více než 11%. U modelů, kde byl přidán i třetí komplement jako signifikantní prediktor, průměrně MASE kleslo pouze zhruba o 7 %.



Obrázek 14: *Zpřesnění podle počtu komplementů (zdroj: autor)*

Z tohoto grafu lze tedy vyvodit, že nekonečné přidávání dalších prediktorů není cestou k nejpřesnějším modelům. Navíc v kombinaci s předchozím grafem na obrázku 13 lze vidět, že rozšíření o 3 komplementy je velmi vzácné a průměrně nepřináší takové zlepšení jako v případě rozšíření o maximálně dva komplementy.

Do finálního řešení tedy v tomto případě pravděpodobně stačí přidat pouze rozšíření o první 2 komplementy místo 3, nicméně na jiném e-shopu, a tedy jiných vstupních datech je nutné provést podobnou analýzu znovu a vyhodnotit, jestli použít cenu komplementu či jeho prodané množství a kolik komplementů do modelu maximálně začlenit.

7.4 Shrnutí kapitoly

V této kapitole bylo využito výstupů z předchozí kapitoly 6, tedy seznamu komplementů k vybraným produktům, a k jejich začlenění do stávajícího řešení. Prvním nezbytným krokem bylo rozšíření tabulky *pois_demand_input,* která obsahuje vstupní data pro trénování modelů. Po rozšíření těchto vstupních dat o nové prediktory je nutné upravit definici modelů a přidat do nich tato nová data, aby bylo možno je použít k trénování. Tato úprava je blíže popsána v podkapitole 7.2.

Po úpravě těchto dat bylo nutné přepočítat křivky nad novými daty a změřit u těchto rozšířených křivek jejich MASE, tedy mean absolute scaled error, hlavní metriku, podle které jsou modely v této práci hodnoceny.

Původní hypotéza, že začleněním komplementů dojde ke zpřesnění modelů, byla potvrzena, nicméně z důvodu jistých specifik výchozích dat byl přínos začlenění ceny komplementů poměrně malý a větší zpřesnění modelů bylo dosaženo díky začlenění prodaného množství komplementů v minulém období. Jelikož však prodané množství obsahuje více náhodnosti, kterou se modely snaží vystihnout, stoupl také počet případů přeučení, kdy došlo naopak k mírnému zhoršení hodnot MASE pro vybrané produkty.

Navzdory tomu však byly modely rozšířené o prodané množství komplementů průměrně přesnější než ty, které byly rozšířené o cenu komplementů, v případě vybrané datové množiny je tedy množství prodaného komplementu lepším prediktorem, který by měl být použit do produkčního prostředí.

8 Diskuze

Následující kapitola se věnuje diskuzi nad výsledky a nad jejich možným využitím k upravení stávajícího řešení za účelem jeho zpřesnění.

Rozšířením původních modelů o vliv komplementů a porovnání takto rozšířených modelů s původními byla potvrzena původní hypotéza, že začleněním komplementů do modelů dynamické cenotvorby, které počítají poptávkové křivky produktů, lze docílit jejich zpřesnění, tedy snížení hodnot MASE.

Množství produktů, u kterých byla vstupní data rozšířena o identifikované komplementy je 88. Ačkoliv veškerých identifikovaných pravidel bylo řádově více, shlukovaly se často kolem několika skupin hodně prodávaných produktů, jak bylo vidět na vizualizacích v kapitole 6. V případě, kdy je snaha maximalizovat počet produktů, ke kterým lze vstupní data rozšířit, je nezbytné dělat větší kompromisy při ladění parametrů minimální hranice podpory a spolehlivosti algoritmu apriori.

Zde je však vhodné sledovat podporu takových pravidel i v absolutním vyjádření a nesnižovat například minimální podporu na hranici, kdy se produkty vyskytly společně jen například v jednotkách či desítkách případů, pokud je celkový počet transakcí ve statisících, jelikož rozšíření modelů o takové produkty bude pravděpodobně častěji způsobovat přeučení a MASE tak naopak zvyšovat. Kvalita přidaných prediktorů může totiž být důležitější než jejich počet, jak ukázala i analýza v této práci na vlivu třetího přidaného komplementu.

Jak již bylo zmíněno v přechozí kapitole při porovnání modelů, jako lepší prediktor pro zpřesnění modelů vlivem komplementů se ukázalo být jejich prodané množství, které oproti jejich ceně snížilo MASE více. Ačkoliv tedy původní předpoklad byl využít spíše cenu komplementů pro rozšíření modelů, v důsledku častého přeceňování identifikovaných komplementů společně byla informační hodnota těchto nových prediktorů v mnoha případech nulová, což limitovalo možnost jejich využití v modelech. Naproti tomu v případě využití prodaného množství v minulém období problém identických hodnot prediktorů nenastával, kvůli jisté míře náhodnosti prodaného množství navzdory stejným cenám.

Míra snížení MASE je tedy závislá na tom, jestli jako prediktor do modelů vstupuje cena komplementu nebo jeho prodané množství. V případě ceny kleslo MASE průměrně na dané datové množině o necelá 2 %, zatímco v případě prodaného množství kleslo MASE o zhruba

3,5 %. Tyto hodnoty berou v potaz všechny produkty, u kterých byly komplementy identifikovány, včetně těch, kde nakonec nedošlo k jejich uplatnění v modelu. Při zprůměrování napříč produkty, kde nakonec identifikované komplementy byly signifikantními prediktory, jsou zlepšení výraznější, a to především v případě začlenění prodaného množství, kde jsou výsledky MASE až zhruba o 11 % nižší v případě začlenění dvou komplementů.

Rozšíření o prodané množství je tedy variantou, která by měla být začleněna do současného produkčního řešení na daném klientovi. Zároveň byl identifikován malý přínos přidání třetího komplementu, jelikož všechny tři komplementy byly signifikantní pouze v jednom případě, rozšíření o maximálně dva komplementy je tedy v tomto případě dostačující.

Dalším krokem vyplývajícím z výsledků této práce je nutnost validace výsledků na produkčních datech a ověření, že došlo ke zpřesnění modelů nasazením rozšířených modelů na počítání skutečných cen na e-shopu a analyzovat, jaký mají nové prediktory skutečný vliv na přesnost modelů při praktické aplikaci na cenotvorbu. V rámci této práce rozšířené modely na produkčním prostředí nasazeny nebyly, především kvůli nutnosti schvalování a delší migraci na produkční prostředí a nutnost dostatečně dlouhého provozu těchto nových modelů ideálně v režimu A/B testování, na základě kterého by bylo možno vyvodit průkazné výsledky a změřit reálný přínos tohoto řešení, tedy především v oblasti zvýšené marže a obratu.

Je však třeba zmínit, že tyto výsledky, které ukazují na využití až dvou nových prediktorů představujících prodané množství identifikovaného komplementu v minulém období nemusí a pravděpodobně ani nebudou platit na odlišném e-shopu. Jestli využít cenu či prodané množství komplementu, kolik komplementů maximálně do modelů přidat jako nové prediktory, případně i jaké hodnoty podpory a spolehlivosti omezit při dolování pravidel, je nutné rozhodnout až po obdobné analýze, jaká byla provedena v této práci, jelikož jsou silně závislé na vstupních datech.

9 Závěr

Tato diplomová práce se zabývala problematikou řešení dynamické cenotvorby pomocí modelů strojového učení a jejich rozšíření o vliv komplementárních produktů. Soustředila se na rozšíření stávajícího řešení algoritmicky řešené dynamické cenotvorby na předním českém e-shopu. To bylo rozšířeno o vliv komplementů na poptávkové křivky produktů, jejichž výpočet tím byl zpřesněn, a tudíž zlepšen přínos tohoto řešení při výpočtu optimální ceny produktů, která maximalizuje tržby a marži.

Pro sestrojení praktického řešení byla nejprve provedena rešerše literatury zkoumající současný stav problematiky a následně v teoretické části práce nastíněny jak termíny dynamické cenotvorby jako takové, tak postupy data miningu nezbytné pro rozšíření stávajícího řešení vydolovanými komplementy. Potom již následoval pomyslný druhý celek práce, tedy praktická část, ve které došlo k samotnému vydolování komplementů z transakční historie a jejich začlenění do současného řešení.

Nakonec byly tyto nové křivky porovnány s původními pomocí metriky MASE (mean absolute scaled error), která byla použita pro srovnání jejich přesnosti. U rozšířených modelů skutečně došlo ke snížení MASE, a tedy zpřesnění modelů. V případě rozšíření o cenu komplementů však došlo k menšímu přínosu než v případě začlenění prodaného množství v minulém období. Důvodem je, že v práci identifikované komplementy jsou často přeceňované na e-shopu společně, jelikož se mnohdy jedná a různé druhy stejného výrobku. Začlenění prodaného množství tuto limitaci částečně eliminuje, obsahuje však více náhodnosti v datech, a tedy zvyšuje počet případů přeučení. I tak ale v průměru modely zpřesňuje až o 11 %, tedy začleněním prodaného množství lze dosáhnout přesnějších poptávkových křivek, a tedy přesnějších optimálních cen.

9.1 Splnění cílů

Hlavním cílem práce bylo rozšířit existující řešení dynamické cenotvorby o vliv komplementů jednotlivých produktů a zvýšit tak přesnost současných modelů na vybrané datové množině. Tento cíl byl naplněn především skrze jednotlivé dílčí cíle, které jsou rozepsané v tabulce 6. Splnění hlavního cíle se tedy prolíná celým textem práce, kde v kapitole 3 byla popsána dynamická cenotvorby jako taková, v kapitole 4 využité metody analýzy, kapitola 5 popsala současný stav, v kapitole 6 byly komplementy identifikovány a v kapitole

7 současné modely o jejich vliv rozšířeny. Nakonec jsou v kapitole 8 výsledky práce okomentovány.

Tabulka 6 níže shrnuje naplnění jednotlivých dílčích cílů této diplomové práce, které byly definovány v úvodní kapitole, a zároveň odkazuje na kapitoly, kde bylo konkrétního cíle dosaženo.

Tabulka 6: *Splnění cílů práce (zdroj: autor)*

Cíl diplomové práce	**Kapitoly**
Analyzovat a popsat možnosti a postupy analýzy nákupního košíku.	Na teoretické úrovni podkapitoly 4.1, 4.2 a 4.3 a konkrétní postupy v kapitole 6.
Pomocí analýzy nákupního košíku vydolovat z dat o transakční historii asociační pravidla a identifikovat jejich pomocí v katalogu jednotlivé komplementární produkty.	Popis výchozích dat v kapitole 5 a samotná identifikace v kapitole 6.
Rozšířit existující modely o vliv komplementů na elasticitu poptávky vybraných produktů.	Popis teorie dynamické cenotvorby v kapitole 3, existujících modelů v kapitole 5 a samotné rozšíření v podkapitolách 7.1 a 7.2.
Porovnat přesnost původních modelů s novými, rozšířenými modely.	Kapitola 7.

9.2 Využití výsledků práce

Výstupy této práce budou začleněny do stávajícího řešení, a tedy přispějí k jeho vylepšení a zpřesnění, které bylo prezentováno v kapitole 7.3.

Nejprve budou výsledky práce prezentovány týmu, který se vývoji produktu dynamické cenotvorby věnuje a diskutovány možné přístupy k začlenění výstupů této práce do produkčního prostředí. V něm je nezbytné tyto výsledky validovat na aktuálních datech a využít je v celém cyklu výpočtu cen až po jejich nasazení na e-shop a změřit jejich vliv na skutečný vývoj tržeb a marže. V závislosti na skutečných výsledcích je také vhodné další ladění parametrů algoritmu apriori pro vydolování optimální množiny pravidel, která budou jak dostatečně početná, tak dostatečně silná, aby byla v modelech informačním přínosem.

Následně lze řešení této práce začlenit do hotového produktu jako modul, který je možné na každém dalším klientovi spustit a ladit tak, aby bylo možné rozhodnout o jeho začlenění, respektive nezačlenění, na základě výsledků validace nad konkrétní datovou množinou. Především zde bude nezbytné ladit parametry algoritmu apriori a rozhodnout, zda využít cenu komplementu či prodané množství.

Poslední je třeba zmínit využití samotné implementace dolování asociačních pravidel, která může být využita i mimo kontext dynamické cenotvorby pro identifikaci komplementů v produktovém katalogu pro libovolného klienta jako samostatný produkt, kterým tak může klient dosáhnout přínosu i v případě, že se rozhodne řešení dynamické cenotvorby nakonec nevyužít.

9.3 Další výzkum

Ačkoliv práce splnila veškeré vytyčené cíle, vyvstává z ní zároveň mnoho námětů na další zlepšení a výzkum v této oblasti, kterým lze na ni navázat. Jednou z možných oblastí dalšího výzkumu je například rozšíření o analýzu substitutů a jejich případné začlenění do modelů jako dalších prediktorů. Nicméně vzhledem k tomu, že pro některé produkty již v případě přidání prediktorů vztahujících se ke komplementům docházelo k přeučení, jednalo by se v takovém případě nejspíše o zvýšení počtu případů, kdy k tomuto jevu dochází. V tomto případě by tedy bylo vhodné ideálně vycházet z většího množství zdrojových dat, tedy například trénovat modely na delším časovém období než jeden rok, případně sledovat výsledky těchto rozšířených modelů a využít některou z dalších technik omezování vlivu přeučení.

Analýzu substitutů je zároveň vhodné využít i mimo kontext samotné dynamické cenotvorby a spojit s případným výstupem zmíněným výše v podkapitole 9.2, pomocí kterého by bylo možné identifikovat substituty a komplementy i v případě nevyužití výstupů této analýzy pro modely dynamické cenotvorby.

Zajímavou možností další validace výsledků této práce by bylo využití na některém z e-shopů s potravinami, či na jiném e-shopu, kde by se dal očekávat vysoký počet silných komplementů, například e-shopu prodávajícím elektroniku a k ní odpovídající příslušenství. V takovém kontextu by bylo pravděpodobně vydolováno více asociačních pravidel, případně silnější pravidla, a tedy nalezeno více komplementů, bylo by tedy zajímavé sledovat přínos tohoto řešení v takovém případě, stejně tak jako přínos samotných vydolovaných komplementů.

Dalším zajímavým bodem je již zmíněné další ladění parametrů minimální podpory a spolehlivosti při dolování asociačních pravidel pro nalezení komplementů. Tyto parametry budou vždy velmi závislé na vstupních datech, tedy transakční historii každého konkrétního e-shopu. Ačkoliv v rámci této práce proběhlo několik iterací ladění těchto parametrů, pravděpodobně nebylo nalezeno optimální nastavení, které by bylo schopno identifikovat veškeré vhodné a dostatečně silné komplementy, dalším laděním těchto parametrů by tedy mohlo být nalezeno více pravidel, která ještě pro model představují informační hodnotu a zpřesnit tak křivky více produktům.

Přínosná by též mohla být bližší analýza způsobu výběru vhodných komplementů mezi asociačními pravidly pro začlenění do vstupních dat. Ta v současnosti volí 3 pravidla s největším liftem jako komplementy, nicméně implementací robustnějšího řešení by bylo možné vybrat komplementy, které by mohly mít ještě více pozitivní vliv na přesnost modelů.

Seznam literatury

AGGARWAL, Charu C. Data Mining. Cham: Springer International Publishing, 2015 [cit. 2019-02-05]. ISBN 978-3-319-14141-1.

BAKER, Tim, 2018. Dynamic pricing: an introduction. In: The Audience Agency [online]. [cit. 2018-11-26]. Dostupné z: https://www.theaudienceagency.org/insight/dynamic-pricing-an-introduction

BERK, Richard a John M. MACDONALD. Overdispersion and Poisson Regression. *Journal of Quantitative Criminology* [online]. 2008, **24**(3), 269-284 [cit. 2019-04-01]. ISSN 0748-4518. Dostupné z: http://link.springer.com/10.1007/s10940-008-9048-4

BERKA, P. Dobývání znalostí z databází. Praha: Academia, 2003. ISBN 80-200-1062-9.

COBLE, Christopher, 2015. Is Dynamic Pricing Legal? In: FindLaw [online]. [cit. 2018-11-30]. Dostupné z: https://blogs.findlaw.com/free_enterprise/2015/07/is-dynamic-pricing-legal.html

DESPOIS, Julien. Memorizing is not learning! —6 tricks to prevent overfitting in machine learning. In: *Harckernoon* [online]. 2018 [cit. 2019-04-01]. Dostupné z: https://hackernoon.com/memorizing-is-not-learning-6-tricks-to-prevent-overfitting-in-machine-learning-820b091dc42

FRANK, E. -- WITTEN, I H. Data mining : practical machine learning tools and techniques. Amsterdam: Morgan Kaufmann, 2005. ISBN 0-12-088407-0.

HALOUN, Petr. Business intelligence application in online retail. Praha, 2017. Diplomová práce. Vysoká škola ekonomická v Praze.

HAUCAP, Justus, Werner REINARTZ a Nico WIEGAND. When Customers Are — and Aren't — OK with Personalized Prices. Harvard Business Review [online]. 2018 [cit. 2019-02-18]. Dostupné z: https://hbr.org/2018/05/when-customers-are-and-arent-ok-with-personalized-prices

HIOTIS, Dimitris, 2018. The 8 Dos and Don'ts of Dynamic Pricing. In: Simon-Kucher & Partners [online]. [cit. 2019-01-15]. Dostupné z: https://www.simon-kucher.com/en/blog/8-dos-and-donts-dynamic-pricing

HORÁČEK, Filip, 2018. Bude pršet, zdražíme deštníky. O cenách zboží v e-shopech rozhodují roboti. In: Idnes.cz [online]. [cit. 2018-11-30]. Dostupné z: https://ekonomika.idnes.cz/dynamicka-cenotvorba-pricing-e-shopy-ceny-fgs-/ekonomika.aspx?c=A180713_414155_ekonomika_fih

HYNDMAN, Rob J. a Anne B. KOEHLER. Another look at measures of forecast accuracy. *International Journal of Forecas*ting [online]. 2006, **22**(4), 679-688 [cit. 2019-03-26]. ISSN 01692070. Dostupné z: https://linkinghub.elsevier.com/retrieve/pii/S0169207006000239

CHEN, Le, Alan MISLOVE a Christo WILSON. An Empirical Analysis of Algorithmic Pricing on Amazon Marketplace. In: Proceedings of the 25th International Conference on World Wide Web - WWW '16 [online]. New York, New York, USA: ACM Press, 2016, s. 1339-1349 [cit. 2018-12-4]. DOI: 10.1145/2872427.2883089.

CHRIST, Steffen. Operationalizing Dynamic Pricing Models. Wiesbaden: Gabler, 2011. ISBN 978-3-8349-2749-1

JACKSON, Dan, 2017. The Netflix Prize: How a $1 Million Contest Changed Binge-Watching Forever. Thrillist.com [online]. [cit. 2019-02-18]. Dostupné z: https://www.thrillist.com/entertainment/nation/the-netflix-prize

KUSCH, Katharina. Beyond customer perception of price discrimination: A consumer behavior analysis and its implications on aviation revenue management. Praha, 2017. Diplomová práce. Vysoká škola ekonomická v Praze.

LAI, Kenneth a Narcis CERPA. Support vs Confidence in Association Rule Algorithms. In: OPTIMA 2001 - Conference of the ICHIO (The Chilean Operations Research Society) [online]. Curicó, Chile, 2001 [cit. 2018-12-28]. Dostupné z: https://www.researchgate.net/publication/233754781_Support_vs_Confidence_in_Association_Rule_Algorithms

LAKE, Andrew, 2017. Dynamic pricing: The top 5 things to consider. In: The Delta Group [online]. [cit. 2018-11-30]. Dostupné z: http://thedeltagroup.co.uk/dynamic-pricing-top-5-things-consider/

MAREK, L.: Pravděpodobnost. 1. vyd. Praha: Professional Publishing, 2013. ISBN 978-80-7431-087-4

Prisync, 2018. The Advantages and Disadvantages of Dynamic Pricing. In: Medium.com [online]. [cit. 2018-12-10]. Dostupné z: https://medium.com/swlh/the-advantages-and-disadvantages-of-dynamic-pricing-c2d914fe644f

MOHAMMED, Rafi, 2017. How Retailers Use Personalized Prices to Test What You're Willing to Pay. In: Hardvard Business Review [online]. [cit. 2018-12-01]. Dostupné z: https://hbr.org/2017/10/how-retailers-use-personalized-prices-to-test-what-youre-willing-to-pay

MOORE, Andrew W. Cross-validation for detecting and preventing overfitting. Autonlab [online]. 2001 [cit. 2019-04-01]. Dostupné z: https://clm.utexas.edu/fietelab/Quant-Neuro/readings/crossvalidation_slides_Moore_CMU.pdf

NOVOTNÝ, Radek, 2018. Maloobchodní tržby zpomalily růst na 4,2 procenta. Chuť utrácet ještě vydrží, lidé už ale spoří na horší časy, říkají analytici. In: ihned.cz [online]. [cit. 2018-11-30]. Dostupné z: https://byznys.ihned.cz/c1-66279270-maloobchodni-trzby-zpomalily-rust-na-4-2-procenta-chut-utracet-jeste-vydrzi-lide-uz-ale-spori-na-horsi-casy-rikaji-analytici

Retail Systems Research, 2017. Dynamic vs. Personalized Pricing [online]. [cit. 2018-11-30]. Dostupné z: https://www.rsrresearch.com/research/dynamic-vs-personalized-pricing

ROTH, Aaron, Aleksandrs SLIVKINS, Jonathan ULLMAN a Zhiwei Steven WU. Multidimensional Dynamic Pricing for Welfare Maximization. In: Proceedings of the 2017 ACM

Conference on Economics and Computation - EC '17 [online]. New York, New York, USA: ACM Press, 2017, 2017, s. 519-536 [cit. 2018-10-26]. ISBN 9781450345279

SCOTT, Steven L. Multi-armed bandit experiments in the online service economy [online]. 2014 [cit. 2019-04-02]. Dostupné z: https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/42550.pdf

STAMBOR, Zak, 2015. A former e-commerce trade group executive pleads guilty to price-fixing. In: Digital Commerce 360 [online]. [cit. 2018-12-01]. Dostupné z: https://www.digitalcommerce360.com/2015/04/07/former-e-commerce-executive-pleads-guilty-price-fixing/

SUTTER, John, 2011. Amazon seller lists book at $23,698,655.93 -- plus shipping. In: CNN [online]. [cit. 2018-10-12]. Dostupné z: http://edition.cnn.com/2011/TECH/web/04/25/amazon.price.algorithm/

TAN, Pang-Ning, Michael STEINBACH a Vipin KUMAR. Introduction to data mining. Boston: Pearson Addison Wesley, c2006. ISBN 978-0321321367.

UĎAN, Miroslav, 2018. Stav e-commerce v ČR v roce 2018. E-commerce v ČR [online] [cit. 2018-11-30]. Dostupné z: https://www.ceska-ecommerce.cz/

WALKER, Tim, 2017. How much …? The rise of dynamic and personalised pricing. In: The Guardian [online]. [cit. 2018-11-30]. Dostupné z: https://www.theguardian.com/global/2017/nov/20/dynamic-personalised-pricing

WHITEHORN, Mark, 2006. The parable of the beer and diapers: Never let the facts get in the way of a good story. In The Register [online]. [cit. 2018-12-28]. Dostupné z: https://www.theregister.co.uk/2006/08/15/beer_diapers/

WILLMOTT, CJ a K. MATSUURA. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate Research* [online]. 2005, **30**, 79-82 [cit. 2019-04-01]. ISSN 0936-577X. Dostupné z: http://www.int-res.com/abstracts/cr/v30/n1/p79-82/

WOLF, Karel, 2018. Alza.cz lidem do košíku přihazuje nevyžádané zboží. In: Lupa.cz [online]. [cit 2018-12-28]. Dostupné z: https://www.lupa.cz/aktuality/alza-cz-lidem-do-kosiku-prihazuje-nevyzadane-zbozi/

Seznam obrázků, tabulek a výpisů kódu

Seznam obrázků

Seznam tabulek

Seznam výpisů kódu