



**MATEMATICKO-FYZIKÁLNÍ  
FAKULTA**  
Univerzita Karlova

## **DIPLOMOVÁ PRÁCA**

Bc. Paula Tekulová

# **Stochastické viacstupňové úlohy distribúcie liekov**

Katedra pravděpodobnosti a matematické statistiky

Vedoucí diplomové práce: doc. RNDr. Ing. Miloš Kopa, Ph.D.

Studijní program: Finanční a pojistná matematika

Studijní obor: Finanční a pojistná matematika

Praha 2023

Prohlašuji, že jsem tuto diplomovou práci vypracoval(a) samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů. Tato práce nebyla využita k získání jiného nebo stejného titulu.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V ..... dne .....

Podpis autora

Ďakujem doc. RNDr. Ing. Milošovi Kopovi Ph.D. za vedenie tejto dimplomovej práce, cenné rady. Ďalej ďakujem spoločnosti IQVIA za poskytnutie dát a rodine za podporu.

Název práce: Stochastické viacstupňové úlohy distribúcie liekov

Autor: Bc. Paula Tekulová

Katedra: Katedra pravdepodobnosti a matematické statistiky

Vedoucí diplomové práce: doc. RNDr. Ing. Miloš Kopa, Ph.D., Katedra pravdepodobnosti a matematické statistiky

Abstrakt: Táto práca sa zaoberá stochastickými viacstupňovými úlohami distribúcie liekov. V prvej kapitole si predstavíme úlohy dvojstupňového stochastického programovania, ktoré následne zovšeobecníme na viacstupňové úlohy stochastického programovania. V druhej kapitole môžeme nájsť detailne popísanú konštrukciu scenárového stromu a dve metódy generovania scenárov - momentová metóda a metódy založené na trajektóriách. Tretia kapitola v úvode popisuje dopravný problém. Následne si predstavíme formuláciu dvojstupňovej a viacstupňovej stochastickej úlohy, ktorých cieľom je maximalizovať zisk lekární. Viacstupňovú úlohu rozšírime pridaním pravdepodobnostných obmedzení. Nasleduje praktická časť, kde sa bližšie pozrieme na reálne dáta, ktoré je potrebné očistiť od sezónnosti. Následne vygenerujeme trajektórie a skonštruujeme scenárové stromy. Na záver si predstavíme výsledky jednotlivých modelov.

Klíčová slova: stochastické programovanie, dynamické úlohy, distribuční problémy

Title: Stochastic multistage problems for drug transportation

Author: Bc. Paula Tekulová

Department: Department of Probability and Mathematical Statistics

Supervisor: doc. RNDr. Ing. Miloš Kopa, Ph.D., Department of Probability and Mathematical Statistics

Abstract: This thesis deals with the multi-stage stochastic problems for drug distribution. In the first chapter, we introduced two-stage and multi-stage stochastic programming problems. In the second chapter, we constructed a scenario tree and introduced two methods on how to generate scenarios - the moment method and paths-based methods. In the third chapter, the transportation problem is formulated. Next, we introduced two-stage and multi-stage models, which aim to maximize the profit of pharmacies. The multi-stage models are then extended with probability constraints. In the practical part, we analyzed the historical data and eliminated the seasonality component. Then, we generated scenarios and constructed scenario trees. Lastly, we analyzed the results of the models.

Keywords: stochastic programming, dynamic models, transportation problems

# Obsah

Úvod	3
<b>1 Viacstupňové stochastické programovanie</b>	<b>5</b>
1.1 Lineárna dvojstupňová úloha stochastického programovania . . . . .	6
1.1.1 Dvojstupňová úloha stochastického programovania s diskrétnym rozdelením . . . . .	6
1.1.2 Pravdepodobnostné obmedzenia . . . . .	8
1.2 Viacstupňová úloha stochastického programovania . . . . .	9
1.2.1 Všeobecná viacstupňová úloha stochastického programovania	10
1.2.2 Lineárna viacstupňová úloha stochastického programovania	11
1.3 Viacstupňová úloha s diskrétnym rozdelením . . . . .	13
1.3.1 Scenárový strom . . . . .	13
1.3.2 Lineárna viacstupňová úloha stochastického programovania založená na scenároch . . . . .	16
<b>2 Metódy tvorby scenárových stromov</b>	<b>18</b>
2.1 Momentová metóda . . . . .	18
2.2 Scenárový strom založený na simulovaných trajektóriách . . . . .	21
2.3 Klastrovací algoritmus . . . . .	24
2.4 Dekompozičné metódy . . . . .	25
2.4.1 Regresný prístup . . . . .	26
2.4.2 Holt-Wintersová metóda . . . . .	26
2.4.3 Algoritmus na predpovedanie viacrozmerných časových rád pomocou Holt-Wintersovej metódy . . . . .	29
<b>3 Optimalizácia rozvozu liekov do lekární</b>	<b>31</b>
3.1 Dopravný problém . . . . .	31
3.2 Dvojstupňová formulácia úlohy založená na scenároch . . . . .	32
3.3 T-stupňová formulácia úlohy . . . . .	33
3.3.1 Model pre jednu lekárňu a jeden typ lieku . . . . .	34
3.3.2 Rozšírenie pre L lekární a I liekov . . . . .	35
3.3.3 Model pre L lekární a I liekov + pravdepodobnostné obmedzenie . . . . .	35
<b>4 Praktická časť</b>	<b>37</b>
4.1 Vstupné dáta . . . . .	37
4.2 Generovanie scenárových stromov . . . . .	38
4.2.1 Odstránenie sezónnosti z dát . . . . .	38
4.2.2 Intervalová predikcia a generovanie trajektórií . . . . .	40
4.2.3 Tvorba scenárových stromov . . . . .	41
4.3 Vstupy do modelov . . . . .	41
4.4 Výsledky modelov . . . . .	42
4.4.1 T-stupňová úloha - základný model . . . . .	42
4.4.2 T-stupňová úloha - základný model + pravdepodobnostné obmedzenia . . . . .	47

<b>Záver</b>	<b>55</b>
<b>Zoznam použitej literatury</b>	<b>56</b>
<b>Zoznam obrázkov</b>	<b>58</b>
<b>Zoznam tabuliek</b>	<b>59</b>
<b>A Prílohy</b>	<b>60</b>
A.1 Korelačné matice - typy liekov . . . . .	60
A.2 Korelačné matice - lekárne . . . . .	61

# Úvod

Stochastické programovanie, známe tiež pod názvom stochastická optimalizácia Birge a Lauveaux (1997), predstavuje matematickú štruktúru, ktorá slúži na modelovanie rozhodnutí za neistoty. Stochastické programovanie je relatívne nová oblasť matematiky, ktorá sa datuje do 50. rokov 20. storočia, keď George B. Dantzig, uznávaný ako otec simplexového algoritmu pre lineárne programovanie, napísal priekopnícku prácu „Linear Programming under Uncertainty“ Dantzig (1955). Vo svojej práci George B. Dantzig priblížil motiváciu vytvorenia modelu stochastického programovania, kde je potrebné sa rozhodovať na základe neistoty. Konkrétne išlo o úlohu stochastického programovania, kde bolo potrebné zahrnúť neisté požiadavky pri riešení optimálneho rozdelenia flotily dopravcov na trasy leteckých spoločností, aby sa splnilo očakávané rozloženie dopytu. Jednou z ďalších prvých prác je napríklad možné nájsť aj v Beale (1955). Stochastickému programovaniu sa odvtedy dostalo veľmi veľa pozornosti a vyvinulo sa do jednej z hlavných oblastí výskumu a bodu záujmu pre matematickú komunitu. Jednou z najznámejších kníh je Birge a Lauveaux (1997) alebo A. Shapiro a Ruszczyński (2009) kde môžeme nájsť značné množstvo teoretických poznatkov, ako aj rôzne algoritmické metódy. S postupným vývojom rôznych algoritmov a výpočtových metód sa stochastické programovanie začalo aplikovať na široké spektrum problémov, ako je napríklad energetické odvetvie - výroba elektriny, ďalej finančné plánovanie, riadenie a dodávanie v rámci obchodných reťazcov a mnoho ďalších odvetví. Rôzne aplikácie stochastického programovania môžeme nájsť v Wallace a Ziemba (2005).

Cielom tejto práce je spracovanie teoretických poznatkov o dvojstupňových stochastických úlohách a následne zovšeobecnenie na T-stupňové stochastické úlohy. V oboch prípadoch sa bližšie pozrieme na prevedenie týchto úloh za predpokladu diskrétného rozdelenia. Pri spracovaní poznatkov budeme vychádzať predovšetkým s vyššie spomínaných kníh Birge a Lauveaux (1997), A. Shapiro a Ruszczyński (2009) a Shapiro a Ruszczyński (2003). Následne si predstavíme scenárový strom a jeho konštrukciu. Teoretickú časť tejto práce zakončíme podrobným popisom metódy tvorby scenárových stromov, konkrétne sa pozrieme na momentovú metódu a metódy založené na trajektóriách. V tom prípade budeme čerpať základné poznatky z článku Høyland a Wallace (2003) a Dupačová a Štěpán, 2002.

V praktickej časti tejto práci sa zameriame už na konkrétne viacstupňové stochastické modely. Ako prvé si predstavíme známy dopravný problém, ktorého podrobné spracovanie spolu s rôznymi metódami riešenia môžeme nájsť popísané v knihe Dupačová a Lachout (2011). Ako prvé sa zameriame na konštrukciu dvojstupňového modelu. Jednoduchšiu verziu tohto modelu, ako aj jeho rozšírenie, môžeme nájsť v bakalárskej práci Tekulová (2020). Následne si predstavíme viacstupňový stochastický model, čo predstavuje hlavný prínos tejto práce. Úlohou týchto modelov bude maximalizovať zisk lekární s tým, že dopyt po liekoch je neznámy a teda predstavuje spomínanú neistotu. Na záver si základný model rozšírime o pravdepodobnostné obmedzenie.

V prípade tejto práce budeme konštruovať modely pre reálny problém a aplikovať na historické dáta. Konkrétne sa pozrieme na optimalizáciu rozvozu liekov

do lekární v meste Praha. Zameriame sa na centrum Prahy a to na lekárne, ktoré sa nachádzajú v časti Praha 1 a Praha 2. Okruh lekární zúžime na komerčnú sieť lekární. Čo sa týka liekov tak v modeloch sa obmedzíme na viacero typov balení jedného lieku.

V poslednej časti tejto práci sa pozrieme na historické dáta, ktoré boli poskytnuté pre účel tejto práce spoločnosťou IQVIA. Na základe dát, ktoré očistíme o sezónnu zložku, vygenerujeme scenárové stromy za pomoci software Python. Pre výpočet jednotlivých modelov budeme používať software Gurobi. Po získaní všetkých výsledkov sa bližšie pozrieme na jednotlivé optimálne riešenia.



# 1. Viacstupňové stochastické programovanie

Stochastické lineárne programovanie (SLP) je veľmi rýchlo rastúce odvetvie optimalizácie. V posledných rokoch bolo publikovaných mnoho akademických výskumov a článkov práve na túto tému. SLP sa zaoberá riešením lineárnych optimalizačných úloh, v ktorých sa vyskytujú náhodné veličiny.

Táto oblasť optimalizácie je špecificky zaujímavá, pretože sa zaoberá problémami a scenármi, s ktorými sa bežne stretávame v našom každodennom živote. Náhoda a neistota sa vyskytujú aj v rôznych sektoroch a odvetviach, ako je napríklad finančný sektor, poisťovníctvo, energetika a mnoho iných. Oplyvnený náhodou a neistotou je napríklad aj majiteľ potravinového obchodu, ktorý pri objednávaní tovaru nevie s určitosťou povedať koľko kusov, ktorého tovaru sa mu podarí prediť. Do úvahy je potrebné brať dva najdôležitejšie faktory a to je aby bol požadovaný dopyt zákazníkov splnený a zároveň aby na druhú stranu nebolo tovaru príliš veľa čo v prípade nízkeho dopytu môže znamenať stratu, keďže by tovar musel byť vyhodnený. Stochastické modely lineárneho programovania rozdeľujú rozhodnutia, ktoré sa vytvárajú, do dvoch kategórií.

1. Rozhodnutia prvého stupňa - sú to rozhodnutia, ktoré sa musia uskutočniť predtým, ako sa náhodné udalosti stanú skutočnosťou. Inými slovami, pred tým, ako dôjde k realizácii náhodných javov.
2. Rozhodnutia druhého stupňa - tieto rozhodnutia sú prijímané až po tom, čo sa dozvieme výsledky náhodných udalostí.

Pre lepšie porozumenie sa môžeme bližšie pozrieť na príklad majiteľa obchodu. Na začiatku každého týždňa musí spraviť rozhodnutie, koľko spotrebného tovaru určitého druhu objedná. Avšak dopredu nepozná počet kusov tovaru koľko sa mu podarí v ten týždeň prediť. Keďže to je spotrebný tovar tak má stanovený dátum spotreby a teda je potrebné brať ohľad aj na tento aspekt. Teda rozhodnutie koľko kusov spotrebného tovaru nakúpi je rozhodnutie prvého stupňa. Následne počas celého týždňa predáva svoj tovar, ktorý určuje jeho zisk. Množstvo predaného tovaru je založené na náhode a teda je to rozhodnutie druhého stupňa.

V nasledujúcom texte si predstavíme všeobecný dvojstupňový model stochastického programovania a potom sa zameriame na úlohu s diskretným rozdelením. Pojmy a označenia použité v texte sme prevzali z knihy Birge a Louveaux (1997), doplnené o informácie z knihy A. Shapiro a Ruszczyński (2009). Formuláciu dvojstupňového stochastického lineárneho modelu môžeme tiež nájsť spracovanú v bakalárskej práci Tekulová (2020). Následne si stručne predstavíme pravdepodobnostné obmedzenia. Ďalej sa už zameriame na viacstupňové stochastické programovanie a rôzne formulácie úlohy.

## 1.1 Lineárna dvojstupňová úloha stochastického programovania

V tejto časti sa budeme zaoberať tzv. dvojstupňovým optimalizačným problémom. Pomenovanie dvojstupňová úloha plynie priamo zo samotného priebehu problému. Ako prvé je totiž potrebné vykonať rozhodnutie  $\mathbf{x}$ , ktoré označuje rozhodnutie prvého stupňa. Potom nasleduje realizácia náhodnej udalosti  $\omega \in \Omega$ , ktorý je definovaný na pravdepodobnostnom priestore  $(\Omega, \mathcal{F}, P)$ , čím zaistíme dáta pre rozhodnutie druhého stupňa. Reálny náhodný vektor  $\boldsymbol{\xi}(\omega)$  vyjadruje závislosť na náhode, kde element  $\omega$  vyjadruje skutočnosť, že sa jedná o náhodnú veličinu. Na záver nasleduje rozhodnutie druhého stupňa  $\mathbf{y}(\omega) \in \mathbb{R}^{n_2}$ , ktoré je závislé na elemente  $\omega$ . Úloha druhého stupňa alebo dvojstupňová úloha potom minimalizuje vektor  $\mathbf{y}(\omega)$ . Všeobecne formulujeme dvojstupňovú lineárnu stochastickú úlohu nasledovne

$$\min_{\mathbf{x} \in M} \quad \mathbf{c}^T \mathbf{x} + \mathbb{E}[Q(\mathbf{x}, \boldsymbol{\xi}(\omega))] \quad (1.1)$$

$$\text{s.t.} \quad \mathbf{A}\mathbf{x} = \mathbf{b} \quad (1.2)$$

$$\mathbf{x} \geq 0,$$

kde deterministicky určené sú reálne vektory  $\mathbf{b} \in \mathbb{R}^{m_1}$ ,  $\mathbf{c} \in \mathbb{R}^{n_1}$  a matica  $\mathbf{A} \in \mathbb{R}^{m_1 \times n_1}$ . Vektor  $\boldsymbol{\xi}(\omega)$  sa skladá z vektorov  $\mathbf{q}(\omega) \in \mathbb{R}^{n_2}$  a  $\mathbf{h}(\omega) \in \mathbb{R}^{m_2}$ , a zo stĺpcov matíc  $\mathbf{T}(\omega)$ ,  $\mathbf{W}(\omega) \in \mathbb{R}^{m_2 \times n_2}$ :

$$\boldsymbol{\xi}(\omega) = (\mathbf{q}(\omega)^T, \mathbf{h}(\omega)^T, \mathbf{T}(\omega)^T, \mathbf{W}(\omega)^T)^T. \quad (1.3)$$

Maticu  $\mathbf{W}(\omega)$  nazývame rekurzívna matica. V prípade, že  $\mathbf{W}(\omega)$  nezávisí na  $\omega$ , hovoríme o úlohe s fixnou rekúziou.

Táto úloha závisí nielen na rozhodnutiach prvého stupňa  $\mathbf{x}$ , ale aj na realizácii vektoru  $\boldsymbol{\xi}(\omega_s) = \boldsymbol{\xi}_s$ . Funkciu  $Q(\mathbf{x}, \boldsymbol{\xi}(\omega))$  považujeme za účelovú funkciu druhého stupňa. Optimum pre túto funkciu nájdeme pomocou lineárneho programovania pre konkrétne zvolené  $\boldsymbol{\xi}(\omega) \in \Omega$ . Úlohu môžeme zapísať nasledovne

$$\min_{\mathbf{y} \in \mathbb{R}^{n_2}} \quad \mathbf{q}(\omega)^T \mathbf{y}(\omega) \quad (1.4)$$

$$\text{s.t.} \quad \mathbf{T}(\omega)\mathbf{x} + \mathbf{W}(\omega)\mathbf{y}(\omega) = \mathbf{h}(\omega)$$

$$\mathbf{y}(\omega) \geq 0.$$

Ak existuje také  $\mathbf{x}$  a  $\omega \in \Omega$ , pre ktoré (1.4) nemá žiadne prípustné riešenie, potom hodnotu účelovej funkcie druhého stupňa definujeme ako  $Q(\mathbf{x}, \boldsymbol{\xi}(\omega)) = +\infty$ . V prípade, že je úloha neohraničená zdola tak  $Q(\mathbf{x}, \boldsymbol{\xi}(\omega)) = -\infty$ .

### 1.1.1 Dvojstupňová úloha stochastického programovania s diskretným rozdelením

Dvojstupňovú úlohu SLP môžeme tiež vnímať ako úlohu s diskretným rozdelením. Predpokladáme, že náhodný vektor  $\boldsymbol{\xi}(\omega)$  má diskretné rozdelenie s konečným

počtom možných realizácií. Potom vieme dvojstupňový model SLP preformulovať ako deterministickú úlohu lineárneho programovania.

Nech  $\boldsymbol{\xi}(\omega)$  je náhodný vektor s diskrétnym rozdelením. Označme množinu realizácií ako  $S \subset N_0$ , pričom uvažujeme, že je konečná. Potom jednotlivé realizácie môžeme vyjadriť ako

$$\boldsymbol{\xi}_s = \boldsymbol{\xi}(\omega_s) \quad \forall s = 1, \dots, S. \quad (1.5)$$

Hodnotu náhodnej veličiny  $\boldsymbol{\xi}(\omega)$  pri  $s$ -tej realizácii formulujeme ako

$$\boldsymbol{\xi}(\omega_s) = (\mathbf{q}(\omega_s)^T, \mathbf{h}(\omega_s)^T, \mathbf{T}(\omega_s), \mathbf{W}_s)^T, \quad (1.6)$$

budeme ju nazývať scenárom  $s$ . Ďalej  $p_s$  označíme ako pravdepodobnosť  $s$ -tého scenára. Potom platí

$$P(\boldsymbol{\xi}(\omega) = \boldsymbol{\xi}(\omega_s)) = p_s. \quad (1.7)$$

Strednú hodnotu účelovej funkcie druhého stupňa  $Q(\mathbf{x}, \boldsymbol{\xi}(\omega))$  vieme explicitne vyjadriť ako

$$\mathbb{E}[Q(\mathbf{x}, \boldsymbol{\xi}(\omega))] = \sum_{s=1}^S p_s Q(\mathbf{x}, \boldsymbol{\xi}(\omega_s)), \quad (1.8)$$

kde

$$Q(\mathbf{x}, \boldsymbol{\xi}(\omega_s)) = \inf\{\mathbf{q}(\omega_s)^T \mathbf{y}_s : \mathbf{T}(\omega_s) \mathbf{x} + \mathbf{W}_s \mathbf{y}_s = \mathbf{h}(\omega_s), \mathbf{y}_s \geq 0\}. \quad (1.9)$$

Môžeme vidieť, že každý scenár  $s$  má svoju vlastnú pravdepodobnosť  $p_s$ . Z toho vyplýva, že viac pravdepodobné scenáre budú do úlohy vstupovať s väčšou váhou ako tie s menšou pravdepodobnosťou.

Pre dané  $\mathbf{x}$  sa očakávaná hodnota  $\mathbb{E}[Q(\mathbf{x}, \boldsymbol{\xi}(\omega))]$  rovná optimálnej hodnote úlohy lineárneho programovania

$$\min_{\mathbf{y}^1, \dots, \mathbf{y}^S} \sum_{s=1}^S p_s (\mathbf{q}_s^T \mathbf{y}_s) \quad (1.10)$$

$$\text{s.t. } \mathbf{T}_s \mathbf{x} + \mathbf{W}_s \mathbf{y}_s = \mathbf{h}_s \quad \forall s = 1, \dots, S, \quad (1.11)$$

$$\mathbf{y}_s \geq 0, \quad \forall s = 1, \dots, S. \quad (1.12)$$

Následne môžeme zapísať úlohu prvého a druhého stupňa spolu. Dvojstupňovú lineárnu úlohu (1.1) formulujeme pre diskrétno rozdelenie veličín ako

$$\min_{\mathbf{x}, \mathbf{y}^1, \dots, \mathbf{y}^S} \mathbf{c}^T \mathbf{x} + \sum_{s=1}^S p_s (\mathbf{q}_s^T \mathbf{y}_s) \quad (1.13)$$

$$\text{s.t. } \mathbf{T}_s \mathbf{x} + \mathbf{W}_s \mathbf{y}_s = \mathbf{h}_s, \quad \forall s = 1, \dots, S, \quad (1.14)$$

$$\mathbf{A} \mathbf{x} = \mathbf{b},$$

$$\mathbf{x} \geq 0,$$

$$\mathbf{y}_s \geq 0, \quad \forall s = 1, \dots, S.$$

Na základe takéhoto formulovania úlohy budeme minimalizovať cez argument zložený z vektora  $\mathbf{x}$  a  $\mathbf{y}_1, \dots, \mathbf{y}_S$ . Vektory odpovedajú jednotlivým scenárom náhodného vektora  $\boldsymbol{\xi}(\omega_s) = \boldsymbol{\xi}_s$ . Ako môžeme vidieť, tak po jednotlivých úpravách sme dostali štandardnú úlohu lineárneho programovania.

### 1.1.2 Pravdepodobnostné obmedzenia

Všeobecne môžeme formulovať pravdepodobnostné obmedzenie v nasledujúcej forme

$$Pr\{G(\mathbf{x}, \boldsymbol{\xi}) \leq 0\} \geq \alpha. \quad (1.15)$$

Hodnotu funkcie rozhodnutia  $\mathbf{x}$  a náhodného vektoru  $\boldsymbol{\xi}$  označujeme ako  $G(\mathbf{x}, \boldsymbol{\xi})$ . Pravdepodobnosť obmedzujeme parametrom  $\alpha \in [0,1]$ , ktorý je pevne daný.

Úlohy s pravdepodobnostnými obmedzeniami rozlišujeme na dva typy a to na úlohy s individuálnymi pravdepodobnostnými obmedzeniami a úlohy so združeným pravdepodobnostným obmedzením.

#### Združené pravdepodobnostné obmedzenia

Ako prvý typ pravdepodobnostného obmedzenia si predstavíme združené pravdepodobnostné obmedzenie. V tomto prípade požadujeme aby boli všetky náhodné obmedzenia súhrnne splnené s vopred stanovenou pravdepodobnosťou  $\alpha$ . Úlohu formulujeme nasledovne

$$\begin{aligned} \min_{\mathbf{x}} \quad & \mathbb{E}(f(\mathbf{x}, \boldsymbol{\xi})) \\ \text{s.t.} \quad & Pr\{G_i(\mathbf{x}, \boldsymbol{\xi}) \leq 0, i = 1, \dots, I\} \geq \alpha, \end{aligned} \quad (1.16)$$

pričom  $f(\mathbf{x}, \boldsymbol{\xi})$  a  $G_i(\mathbf{x}, \boldsymbol{\xi}), i = 1, \dots, I$  sú lineárne funkcie a  $\alpha \in [0,1]$ . Výhoda tejto formulácie je, že pri každom prípustnom riešení máme garantované splnenie všetkých podmienok súčasne s pravdepodobnosťou aspoň  $\alpha$ .

#### Individuálne pravdepodobnostné obmedzenia

Ako druhý typ úloh s pravdepodobnostnými obmedzeniami si predstavíme individuálne pravdepodobnostné obmedzenia. Na rozdiel od združeného pravdepodobnostného obmedzenia v individuálnom platí, že každé náhodné obmedzenie  $g_i$  musí byť splnené aspoň s pravdepodobnosťou  $\alpha_i, i = 1, \dots, I$ , ktorá je vopred daná, t.j.

$$\begin{aligned} \max_{\mathbf{x}} \quad & \mathbb{E}(f(\mathbf{x}, \boldsymbol{\xi})) \\ \text{s.t.} \quad & Pr\{G_i(\mathbf{x}, \boldsymbol{\xi}) \leq 0\} \geq \alpha_i, i = 1, \dots, I, \end{aligned} \quad (1.17)$$

pričom  $f(\mathbf{x}, \boldsymbol{\xi})$  a  $G_i(\mathbf{x}, \boldsymbol{\xi}), i = 1, \dots, I$  sú lineárne funkcie a  $\alpha_i \in [0,1], i = 1, \dots, I$ . Ako môžeme vidieť tak táto formulácia je oproti združenej jednoduchšia. Špeciálnym prípadom individuálnych pravdepodobnostných obmedzení je tzv. separabilný model. Znamená to, že funkcie  $G_i(\mathbf{x}, \boldsymbol{\xi}), i = 1, \dots, I$  sú separovateľné v  $\mathbf{x}$  a  $\boldsymbol{\xi}$ . Zapisujeme ako

$$G_i(\mathbf{x}, \boldsymbol{\xi}) = G_i(\mathbf{x}) - \boldsymbol{\xi}_i, i = 1, \dots, I. \quad (1.18)$$

Potom môžeme obmedzenie prepísať ako

$$Pr\{G_i(\mathbf{x}) \leq \xi_i\} \geq \alpha_i, i = 1, \dots, I. \quad (1.19)$$

Nech  $F_i$  je marginálna distribučná funkcia  $\mathbf{x}i_i, i = 1, \dots, I$ . Potom platí

$$F_i(G_i(\mathbf{x})) \leq 1 - \alpha_i, i = 1, \dots, I \quad (1.20)$$

čo ekvivalentne, za pomoci kvantilu rozdelenia  $F_i$  náhodných vektorov  $\boldsymbol{\xi}, i = 1, \dots, I$  môžeme zapísať ako

$$G_i(\mathbf{x}) \leq F_i^{-1}(1 - \alpha_i), i = 1, \dots, I. \quad (1.21)$$

## 1.2 Viacstupňová úloha stochastického programovania

V úlohe dvojstupňového stochastického programovania, predpokladáme, že náhoda je realizovaná až po prijatí rozhodnutia prvého stupňa. Avšak, väčšina praktických úloh, ktoré je potrebné riešiť zahŕňa postupnosť rozhodnutí, ktoré reagujú na výsledky, ktoré sa vyvíjajú postupne v čase. Zovšeobecnením tejto úlohy na postupnosť realizácií náhody, dostaneme úlohu viacstupňového stochastického programovania. Časový horizont je rozdelený na „stupne“, kde každý stupeň obsahuje realizáciu náhody v tej danej fáze problému. Uvedené značenie a formulácie budeme čerpať z knihy A. Shapiro a Ruszczyński (2009), Shapiro a Ruszczyński (2003) doplnené o poznatky z knihy Dupačová a Štěpán (2002).

Všeobecne, pri formulácii viacstupňovej stochastickej optimalizačnej úlohy je dôležité si vymedziť vyššie spomenutý časový horizont a určiť postupnosť časov, v ktorých budeme vykonávať rozhodnutia. Predpokladáme, že naše rozhodnutia budú vykonané v čase  $t \in \{1, \dots, T\}$ . Ďalej budeme uvažovať postupnosť náhodných dát

$$\boldsymbol{\xi} = (\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_T), \quad (1.22)$$

ktoré predstavujú stochastický proces. História procesu náhodných dát do času  $t$  budeme zapisovať ako

$$\boldsymbol{\xi}_{[t]} := (\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_t), \quad (1.23)$$

a postupnosť rozhodnutí

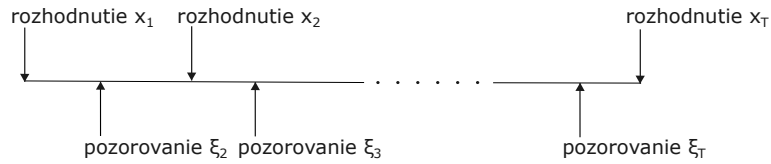
$$\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_T). \quad (1.24)$$

Rozhodovacia premenná  $\mathbf{x}_1$  je deterministický reálny vektor a  $\mathbf{x}_t, t \in \{1, \dots, T\}$  sú reálne náhodné vektory. Náhodné premenné obsiahnuté v procese  $\boldsymbol{\xi}$  môžu mať pomerne všeobecný charakter, zvyčajne sa však uvažujú reálne náhodné vektory. Ako už bolo spomenuté v predchádzajúcej časti, realizácie tohto náhodného procesu sa nazývajú trajektórie resp. scenáre.

Hodnota  $\xi_1$  a rozhodnutie  $\mathbf{x}_1$  sú známe na začiatku riešenia problému. Kľúčovým predpokladom je princíp neanticipativity, čo znamená, že rozhodnutie v čase  $t \in 1, \dots, T$  je možné urobiť len na základe informácií dostupných do času  $t$ , a nezávisí na budúcich náhodných dátach a rozhodnutiach. Postupnosť rozhodnutí je stochastický proces, pretože  $\mathbf{x}_t$  závisí na  $\xi_t$ . Hodnotu  $\xi_1$  budeme považovať za konštantnú a známu pred vykonaním rozhodnutia prvého stupňa v každom rozhodovacom procese. Vektor  $\mathbf{x}_1$  reprezentuje rozhodnutie v prvom stupni, ktoré musí byť vykonané predtým, ako budú k dispozícii ďalšie informácie. Vektor  $\mathbf{x}_2(\mathbf{x}_1; \xi_2)$  zobrazuje rozhodnutie v druhom stupni, ktoré závisí na rozhodnutí v prvom stupni  $\mathbf{x}_1$  a náhodnom vektore  $\xi_2$ . To znamená, že sa už môže prispôbiť informáciám dostupným z predchádzajúcich rozhodnutí. Analogicky budeme pokračovať až do posledného stupňa  $T$ . Teda vektor

$$\mathbf{x}_T(\mathbf{x}_1, \dots, \mathbf{x}_{T-1}; \xi_2, \dots, \xi_T) \quad (1.25)$$

vyjadruje posledné rozhodnutie v stupni  $T$ . Toto rozhodnutie závisí na všetkých predchádzajúcich stupňoch a na celej histórii náhodného procesu  $\xi_{[T]}$ . Na obrázku 1.1 môžeme vidieť znázornený tento rozhodovací proces, ktorý sme popisovali.



Obr. 1.1: Rozhodovací proces.

Z uvedeného postupu je zřejmé, že optimálne riešenie  $\mathbf{x}_t$  v jednotlivých stupňoch závisí iba na histórii náhodného procesu do času  $t$ , čím je zabezpečený princíp neanticipativity.

### 1.2.1 Všeobecná viacstupňová úloha stochastického programovania

Stochastickú optimalizačnú úlohu s viacerými stupňami je možné formulovať rôznymi spôsobmi. V tejto časti si predstavíme všeobecnú formuláciu a potom sa pozrieme na zápis pomocou systému rekurzívnych rovníc. Pri tomto budeme vychádzať z formulácií, ktoré sú dostupné v Dupačová a Štěpán (2002), A. Shapiro a Ruszczyński (2009) alebo Birge a Louveaux (1997).

Všeobecný zápis, je formulovaný nasledovne:

$$\min_{\mathbf{x}_1 \in \mathcal{X}_1} f_1(\mathbf{x}_1) + \mathbb{E}[\inf_{(\mathbf{x}_2 \in \mathcal{X}_2(\mathbf{x}_1, \xi_2))} + \mathbb{E}[\dots + \mathbb{E}[\inf_{\mathbf{x}_T \in \mathcal{X}_T(\mathbf{x}_{T-1}, \xi_T)} f_T(\mathbf{x}_T, \xi_T)]]], \quad (1.26)$$

$\mathbf{x}_t \in \mathbb{R}^{n_t}$  pre  $t = 1, \dots, T$  sú rozhodujúce premenné, funkcie  $f_t : \mathbb{R}^{n_t} \times \mathbb{R}^{d_t} \rightarrow \mathbb{R}$  sú spojité funkcie a  $\mathcal{X}_t$  je množina merateľných funkcií, ktoré pre  $\mathbf{x}_{t-1}$  a  $\xi_t$  obmedzujú hodnoty, ktoré nadobúda rozhodujúca premenná  $\mathbf{x}_t$ . Ako sme už uviedli, vektor  $\xi_1$  je známy na začiatku riešenia problému, a teda aj funkcia  $f_1 : \mathbb{R}^{n_1} \rightarrow \mathbb{R}$ . Rozhodnutie  $\mathbf{x}_1$  je deterministicky určené a z toho vyplýva, že množina  $\mathcal{X}_1$  je tiež deterministicky určená.

Takto formulovanú úlohu môžeme zaznamenať rôznymi ekvivalentnými spôsobmi. V tejto časti sa pozrieme na formuláciu, v ktorej úlohu vyjadríme ako tzv. rozšírenie dvojstupňového stochastického programovania. Budeme uvažovať úlohu tzv. "posledného stupňa", kde budeme úlohu formulovať pomocou rekurzívnych rovníc. V poslednom stupni  $T$  formulujeme úlohu ako

$$\min_{\mathbf{x}_T \in \mathcal{X}_T(\mathbf{x}_{T-1}, \boldsymbol{\xi}_T)} f_T(\mathbf{x}_T, \boldsymbol{\xi}_T). \quad (1.27)$$

Optimálna hodnota tohto problému závisí na rozhodovacom vektore  $\mathbf{x}_{T-1}$  a náhodných dátach  $\boldsymbol{\xi}_T$ , ktorú označujeme ako

$$Q_T(\mathbf{x}_{T-1}, \boldsymbol{\xi}_T). \quad (1.28)$$

Následne, v stupňoch  $t \in \{2, \dots, T-1\}$  môžeme úlohu formulovať nasledovne

$$\min_{\mathbf{x}_t} f_t(\mathbf{x}_t, \boldsymbol{\xi}_t) + \mathbb{E}\{Q_{t+1}(\mathbf{x}_t, \boldsymbol{\xi}_{[t+1]}) | \boldsymbol{\xi}_{[t]}\} \quad (1.29)$$

$$\text{s.t. } \mathbf{x}_t \in \mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t), \quad (1.30)$$

kde  $\mathbb{E}[\cdot | \boldsymbol{\xi}_{[t]}]$  predstavuje podmienenú strednú hodnotu. Optimálna hodnota závisí na rozhodnutí  $\mathbf{x}_{t-1}$ , ktoré bolo urobené v predchádzajúcom stupni úlohy, a na realizácii procesu náhodných dát  $\boldsymbol{\xi}_{[t]}$ . Označíme ako  $Q_T(\mathbf{x}_{t-1}, \boldsymbol{\xi}_{[t]})$ . Preto môžeme problém pre prvý stupeň formulovať ako minimalizačnú úlohu s rekurzívnou štruktúrou v tvare

$$\min_{\mathbf{x}_1 \in \mathcal{X}_1} f_1(\mathbf{x}_1) + \mathbb{E}[Q_2(\mathbf{x}_1, \boldsymbol{\xi}_2)]. \quad (1.31)$$

## 1.2.2 Lineárna viacstupňová úloha stochastického programovania

Špeciálnym prípadom viacstupňovej úlohy stochastického programovania je lineárne prevedenie. Túto úlohu považujeme za lineárnu, pretože účelová funkcia a obmedzenia, ktoré definujú množiny prípustných riešení, sú lineárne.

Predpokladajme, že  $\mathbf{x}_1, \dots, \mathbf{x}_T$  sú vektory rozhodnutí, potom optimalizačnú úlohu viacstupňového lineárneho programovania môžeme formulovať nasledovne

$$\min_{\mathbf{x}_1, \dots, \mathbf{x}_T} c^T 1x_1 + c^T 2x_2 + \dots + c^T T x_T \quad (1.32)$$

$$\text{s.t. } A_1 x_1 = b_1$$

$$B_2 x_1 + A_2 x_2 = b_2 \quad (1.33)$$

$$B_3 x_2 + A_3 x_3 = b_3 \quad (1.34)$$

$$\vdots \quad (1.35)$$

$$B_T x_{T-1} + A_T x_T = b_T \quad (1.36)$$

$$\mathbf{x}_t \geq 0, \forall t = 1, \dots, T. \quad (1.37)$$

Ako môžeme vidieť zo zápisu, tak úlohu vieme zovšeobecniť ako úlohu viac-  
stupňového stochastického programovania. Konkrétne, v tomto prípade bude pre  
náhodný proces  $\boldsymbol{\xi}$  platiť, že jeho realizácia v prvom stupni je známa. Teda  $\boldsymbol{\xi}_1 =$   
 $(\mathbf{c}_1, \mathbf{A}_1, \mathbf{b}_1)$  sú nenáhodné. Avšak pre  $\boldsymbol{\xi}_t = (\mathbf{c}_t, \mathbf{A}_t, \mathbf{B}_t, \mathbf{b}_t)$ ,  $t \in \{2, \dots, T\}$  platí, že  
môžu mať náhodný charakter. Aby sme mohli úlohu zovšeobecniť je potrebné  
minimalizovať strednú hodnotu účelovej funkcie. Zovšeobecnená úloha teda bude  
vyzerať nasledovne

$$\begin{aligned} \min_{\mathbf{x}_1, \dots, \mathbf{x}_T} \quad & \mathbb{E}[c_1^T x_1 + c_2^T x_2 + c_3^T x_3 + \dots + c_T^T x_T] \quad (1.38) \\ \text{s.t.} \quad & A_1 x_1 = b_1 \\ & B_2 x_1 + A_2 x_2 = b_2 \\ & B_3 x_2 + A_3 x_3 = b_3 \\ & B_T x_{T-1} + A_T x_T = b_T \\ & \mathbf{x}_t \geq 0, \forall t = 1, \dots, T. \end{aligned}$$

Je potrebné dodať, že aj v tomto prípade platí neanticipativnosť rozhodnutí.  
Znamená to, že rozhodnutie vykonané v ľubovoľnom stupni úlohy závisí len na  
už známych rozhodnutiach a realizáciách, t.j. nesmie závisieť na budúcnosti.

Na úlohu sa môžeme opäť pozrieť aj ako na vnorenú, z pohľadu posledného  
stupňa  $T$ . V takomto prípade riešime jednoduchú úlohu lineárneho programova-  
nia.

$$\begin{aligned} \min_{\mathbf{x}_T} \quad & \mathbf{c}_T^T \mathbf{x}_T \quad (1.39) \\ \text{s.t.} \quad & \mathbf{B}_T \mathbf{x}_{T-1} + \mathbf{A}_T \mathbf{x}_T = \mathbf{b}_T, \\ & \mathbf{x}_T \geq 0 \end{aligned}$$

Optimálna hodnota závisí na predchádzajúcom rozhodovacom vektore  $\mathbf{x}_{T-1}$  a  
dátach  $\boldsymbol{\xi}_T = (\mathbf{c}_T, \mathbf{B}_T, \mathbf{A}_T, \mathbf{b}_T)$ . V stupni  $T - 1$  poznáme hodnoty  $\mathbf{x}_{T-2}$  a  $\boldsymbol{\xi}_{[T-1]}$ . Ana-  
logicky môžeme pokračovať až po stupeň  $t = 2$ . Účelová funkciu s obmedzením  
pre  $t = 2, \dots, T$  môžeme vyjadriť nasledovne

$$f_t(\mathbf{x}_t, \boldsymbol{\xi}_t) := \mathbf{c}_t^T \mathbf{x}_t \quad (1.40)$$

$$\mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t) := \{\mathbf{x}_t : \mathbf{B}_t \mathbf{x}_{t-1} + \mathbf{A}_t \mathbf{x}_t = \mathbf{b}_t, \mathbf{x}_t \geq 0\}, t = 2, \dots, T, \quad (1.41)$$

Všeobecne môžeme úlohu stochasticky formulovať ako

$$\begin{aligned} \min_{\mathbf{x}_T} \quad & \mathbf{c}_T^T \mathbf{x}_T + \mathbb{E}[\mathcal{Q}_{t+1}(\mathbf{x}_t, \boldsymbol{\xi}_{t+1}) | \boldsymbol{\xi}_{[t]}] \quad (1.42) \\ \text{s.t.} \quad & \mathbf{B}_t \mathbf{x}_{t-1} + \mathbf{A}_t \mathbf{x}_t = \mathbf{b}_t, \\ & \mathbf{x}_t \geq 0, \end{aligned}$$

Avšak, ešte nám zostáva prípad kedy  $t = 1$ , v tomto prípade musíme obme-  
dzenie upraviť nasledovne



$$\mathcal{X}_1 := \{\mathbf{x}_1 : \mathbf{A}_1 \mathbf{x}_1 = \mathbf{b}_1, \mathbf{x}_1 \geq 0\}. \quad (1.43)$$

Úlohu opäť stochasticky formulujeme ako

$$\begin{aligned} \min_{\mathbf{x}_1} \quad & \mathbf{c}_1^T \mathbf{x}_1 + \mathbb{E}[\mathcal{Q}_2(x, \xi_2)] \\ \text{s.t.} \quad & \mathbf{A}_1 \mathbf{x}_1 = \mathbf{b}_1, \\ & \mathbf{x}_1 \geq 0. \end{aligned} \quad (1.44)$$

Zo zápisu vidíme, že všetky predchádzajúce stupne  $t = 2, \dots, T$  sú obsiahnuté vo vnútri funkcie  $\mathcal{Q}_2(x, \xi_2)$  cez korešpondujúce stredné hodnoty.

Na záver môžeme teda úlohu zapísať pomocou vnorenej formulácie nasledovne

$$\min_{\substack{\mathbf{A}_1 \mathbf{x}_1 = \mathbf{b}_1 \\ \mathbf{x}_1 \geq 0}} \mathbf{c}_1^T \mathbf{x}_1 + \mathbb{E}[\mathbf{c}_2^T \mathbf{x}_2 + \mathbb{E}[\dots + \mathbb{E}[\min_{\substack{\mathbf{A}_T \mathbf{x}_{T-1} + \mathbf{A}_T \mathbf{x}_T = \mathbf{b}_T \\ \mathbf{x}_T \geq 0}} \mathbf{c}_T^T \mathbf{x}_T]]]]. \quad (1.45)$$

## 1.3 Viacstupňová úloha s diskretným rozdelením

V tejto kapitole sa bližšie pozrieme na špeciálny prípad úlohy viacstupňového stochastického programovania. Uvedené pojmy a značenie budeme čerpať z Dupačová a Wallace (2000), doplnené o poznatky z Dupačová a Štěpán (2002).

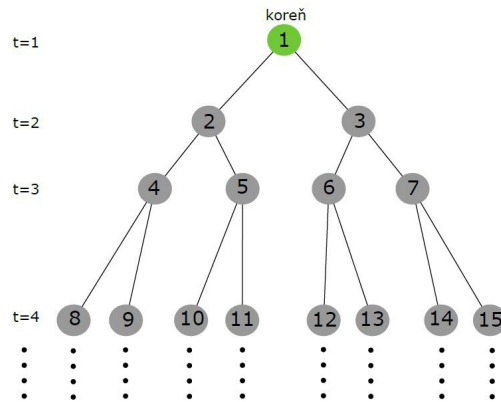
V celej nasledujúcej kapitole budeme predpokladať, že náhodné dáta procesu  $\xi_t, t \in \{1, \dots, T\}$  majú diskretné rozdelenie, s konečným počtom realizácií, ktoré budeme nazývať *scenáre*. Jednotlivé scenáre označíme ako  $s \in \{1, \dots, S\}$ . Informáciu dostupnú v čase  $t$  v rámci konkrétneho scenára  $s$  vieme zaznamenať ako  $\xi_t^s$ . Pre  $s \in 1, \dots, S$  teda bude realizácia náhodného procesu  $\xi^s = \xi_1^s, \dots, \xi_T^s$  označovaná ako scenár náhodného procesu.

### 1.3.1 Scenárový strom

Náhodný proces  $\xi^s = \{\xi_1^s, \dots, \xi_T^s\}$  s konečným diskretným rozdelením je možné reprezentovať ako *scenárový strom*. Na základe predpokladov, ktoré sme uviedli na začiatku tejto časti, môžeme konštatovať, že scenárový strom je explicitnou reprezentáciou procesu vetvenia spôsobeného postupným pozorovaním  $\xi_1^s, \dots, \xi_T^s$ , za predpokladu diskretného rozdelenia náhodných premenných. Uzly scenárového stromu sú usporiadané v úrovniach, ktoré zodpovedajú stupňom  $1, \dots, T$ . Scenárový strom môžeme zostaviť nasledovne.

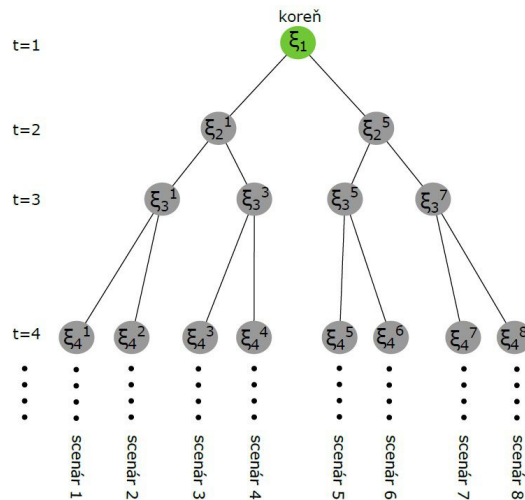
V úvodnom stupni  $t = 1$  sa nachádza iba jeden uzol, známy ako *koreňový uzol (koreň)*. Tento uzol je spojený s hodnotou náhodného vektora  $\xi_1^s$ , ktorá je známa na začiatku prvého stupňa úlohy. Koreňový uzol je rovnaký pre všetky scenáre  $s \in \{1, \dots, S\}$ . V druhom stupni, teda  $t = 2$ , je toľko uzlov, koľko je možných rozličných realizácií náhodného vektora  $\xi_2^s$  v rámci náhodného procesu  $\{\xi_1^s, \dots, \xi_T^s\}$ . Pre každý uzol v druhom stupni,  $t = 2$ , vytvoríme v treťom stupni  $t = 3$  toľko uzlov, koľko je možných rozličných náhodných realizácií náhodného

vektora  $\xi_3^s$  vzhľadom na  $\xi_2^s$ . Proces vytvárania stromu pokračuje až do posledného stupňa  $T$ . Jednotlivý scenár tak predstavuje unikátnu cestu od koreňa po konečný uzol v stupni  $T$ , ktorý sa nazýva *list*.



Obr. 1.2: Reprezentácia scenárového stromu.

Všeobecne platí, že uzly v stupni  $t$  zodpovedajú možným realizáciám náhodného vektora  $\xi_t^s$ , ktoré môžu nastať. Každý z týchto uzlov je spojený práve s jedným uzlom v stupni  $t - 1$ , tento uzol sa nazýva *rodič* alebo *predchodca uzlov* v stupni  $t - 1$ . Uzly v stupni  $t$  sú taktiež spojené aj s uzlami v stupni  $t + 1$ , tieto uzly nazývame *deti* alebo *nasledovníci*. Treba zdôrazniť, že niektoré hodnoty náhodných realizácií vektora  $\xi_t^s$  môžu byť rovnaké, avšak sú reprezentované rôznymi uzlami. Dôvodom je, že aj keď sa zhodujú, môžu mať odlišnú históriu procesu.



Obr. 1.3: Scenárový strom s príslušnými realizáciami.

Množinu všetkých uzlov v stupni  $t \in \{1, \dots, T\}$  označíme ako  $\mathcal{N}_t$ . Pre lepšie porozumenie, množina  $\mathcal{N}_1$  obsahuje jedinečný koreň scenárového stromu. Množina  $\mathcal{N}_2$  obsahuje toľko elementov, koľko je rôznych realizácií náhodného vektora  $\xi_2^s$ . Analogicky pokračujeme až po posledný stupeň  $T$ .

Uzol scenárového stromu budeme označovať ako  $n$ . Potom pre každý uzol  $n \in \mathcal{N}_t$ , označíme množinu všetkých detí alebo bezprostredných nasledovníkov uzla  $n$  ako

$$\mathcal{D}_n \subset \mathcal{N}_{t+1}, t \in \{1, \dots, T-1\}. \quad (1.46)$$

Množinu všetkých rodičov uzla  $n$  označíme ako

$$r(n) \in \mathcal{N}_{t-1}, t \in \{2, \dots, T\}. \quad (1.47)$$

Následne platí, že  $\mathcal{N}_{t+1} = \cup_{n \in \mathcal{N}_t} \mathcal{D}_n$  a množiny  $\mathcal{D}_n$  sú disjunktné. Teda pre  $\forall n, m \in \mathcal{D}_n$  také, že  $n \neq m$  platí  $\mathcal{D}_n \cap \mathcal{D}_m = \emptyset$ . Na základe konštrukcie scenárového stromu môžeme povedať, že každý scenár korešponduje s príslušnou množinou  $\mathcal{N}_T$  a teda platí, že  $S = |\mathcal{N}_T|$ .

Do úvahy môžeme brať aj pravdepodobnostné rozdelenie náhodných premenných. Váhy pravdepodobností sú spojené s uzlami stromu scenára. Koreňový uzol má pravdepodobnosť 1, zatiaľ čo deti sú vážené pravdepodobnosťami, ktoré predstavujú pravdepodobnosť hodnoty, ku ktorej sú priradené, podmienené hodnotou spojenou s ich nadradeným uzlom. Vynásobením pravdepodobnosti uzlov cesty od koreňa k listu získame pravdepodobnosť scenára.

Ak berieme do úvahy vnorenú podobu problému, tak potrebujeme špecifikovať podmienenú distribúciu  $\xi_{t+1}^s$  za daného vektora  $\xi_{[t]}^s, t = \{1, \dots, T-1\}$ . Konkrétne to znamená, že potrebujeme zistiť pravdepodobnosť posunu z uzla  $n \in \mathcal{N}_t$ , v ktorom sa aktuálne nachádzame do uzla  $d \in \mathcal{D}_n$ . Pravdepodobnosť tohto prechodu označíme ako  $q_{nd}$ . Platí, že

$$\sum_{d \in \mathcal{D}_n} q_{nd} = 1, \quad (1.48)$$

$$q_{nd} \geq 0. \quad (1.49)$$

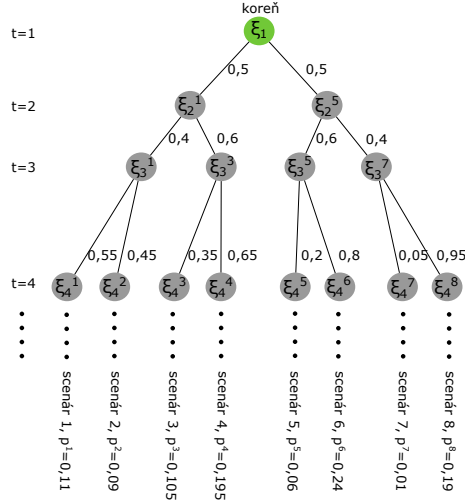
Pravdepodobnosti  $q_{nd}, d \in \mathcal{D}_n$  reprezentujú podmienené rozdelenie  $\xi_{t+1}^s$ , za podmienky, že cesta procesu  $\xi_1^s, \dots, \xi_t^s$  skončila v uzle  $n$ .

Ako sme spomínali vyššie, každý scenár je definovaný uzlami  $n_1, \dots, n_T$ . Pravdepodobnosť takéhoto scenára môžeme vyjadriť vynásobením všetkých pravdepodobností od daného koreňa až po list, ako  $q_{n_1 n_2} \times q_{n_2 n_3} \times \dots \times q_{n_{T-1} n_T}$ . Teda, množina podmienených pravdepodobností definuje pravdepodobnostné rozdelenie množiny scenárov. Obrátene, môžeme vyjadriť podmienené pravdepodobnosti z pravdepodobností konkrétnych scenárov  $p_s, s \in \{1, \dots, S\}$ .

Množinu scenárov prechádzajúcich cez uzol  $n$  v stupni  $t$  scenárového stromu označíme ako  $\mathcal{S}^{(n)}$ . Nech  $p^{(n)} := Pr[\mathcal{S}^{(n)}]$  je suma pravdepodobností všetkých scenárov prechádzajúcich cez uzol  $n$ . Nech  $n_1, n_2, \dots, n_t$  označuje históriu procesu do uzla  $n$ , kde  $n_1$  označuje koreň a  $n_t = n$ . Potom pravdepodobnosť  $p^{(n)}$  je daná nasledovne

$$p^{(n)} = q_{n_1 n_2} \times q_{n_2 n_3} \times \dots \times q_{n_{t-1} n_t}. \quad (1.50)$$

Tento zápis môžeme vyjadriť aj rekurzívne z  $p^{(n)} = q_{an} p^{(a)}$ , kde  $a = a(n)$  je predchodca uzlu  $n$ . Táto rovnica definuje podmienenú pravdepodobnosť  $q_{an}$  z pravdepodobností  $p^{(n)}$  a  $p^{(a)}$ . Ak  $a = a(n)$  je predchodca uzlu  $n$  tak potom platí, že  $\mathcal{S}^{(n)} \subset \mathcal{S}^{(a)}$  a teda platí, že  $p^{(n)} \leq p^{(a)}$ . Následne platí, že  $p^{(a)} > 0$  a teda  $q_{an} = \frac{p^{(n)}}{p^{(a)}}$ . V inom prípade je  $\mathcal{S}^{(a)}$  prázdna a teda žiaden scenár neprechádza uzlom  $n$ .



Obr. 1.4: Scenárový strom s príslušnými pravdepodobnosťami.

### 1.3.2 Lineárna viacstupňová úloha stochastického programovania založená na scenároch

Nech náhodný proces dát  $\xi$  má diskrétno rozdelenie, pričom množina jeho realizácií  $S \subset N_0$  je konečná. Túto množinu realizácií budeme nazývať *scenáre*. Pre každý scenár  $s \in \{1, \dots, S\}$  označíme realizácie náhodných dát nasledovne

$$\xi^s = (\xi_1^s, \dots, \xi_T^s). \quad (1.51)$$

Realizáciu scenára  $s$  až po časový okamih  $t$  budeme označovať takto

$$\xi_{[t]}^s = (\xi_1^s, \dots, \xi_t^s), \quad (1.52)$$

pravdepodobnosti jednotlivých scenárov budeme značiť ako  $p^s$  pre  $s \in \{1, \dots, S\}$ , pričom bude platiť

$$\sum_{s \in \{1, \dots, S\}} p^s = 1. \quad (1.53)$$

Lineárny problém potom môžeme formulovať nasledovne

$$\begin{aligned} \min_{\substack{x_1^s, \dots, x_T^s \\ s \in \{1, \dots, S\}}} & \sum_{s=1}^S p^s [\mathbf{c}_1^T x_1^s + (\mathbf{c}_2^s)^T x_2^s + (\mathbf{c}_3^s)^T x_3^s + \dots + (\mathbf{c}_T^s)^T x_T^s] \\ \text{s.t.} & \mathbf{A}_1 x_1 = \mathbf{b}_1 \\ & \mathbf{B}_2^s x_1 + \mathbf{A}_2^s x_2 = \mathbf{b}_2^s \\ & \mathbf{B}_3^s x_2 + \mathbf{A}_3^s x_3 = \mathbf{b}_3^s \\ & \mathbf{B}_T^s x_{T-1} + \mathbf{A}_T^s x_T = \mathbf{b}_T^s \\ & x_t^s \geq 0, \forall t = 1, \dots, T, \forall s = 1, \dots, S. \end{aligned}$$

Ako vo všetkých, zatiaľ predstavených formuláciach, musí byť aj v tomto prípade dodržaný princíp neanticipativity. Keď sa bližšie pozrieme na úlohu formulovanú vyššie môžeme vidieť, že rozhodnutie v čase  $t$  závisí ako na minulosti tak aj na budúcich rozhodnutiach a realizáciách. Pridaním jednoduchého obmedzenia množiny prípustných riešení, tak aby dva rozhodovacie náhodné procesy, ktoré majú rovnakú realizáciu náhodných dát do času  $t$  mali tak isto aj rovnaké rozhodnutia do času  $t$ , obmedzíme závislosť rozhodnutí na budúcnosti. Podmienku formulujeme nasledovne

$$\mathbf{x}_t^s = \mathbf{x}_t^q, \forall s, q \in \{1, \dots, S\}, \quad (1.54)$$

$$\boldsymbol{\xi}_{[t]}^s = \boldsymbol{\xi}_{[t]}^q, \forall t \in \{1, \dots, T\}. \quad (1.55)$$

## 2. Metódy tvorby scenárových stromov

Existuje mnoho metód, ktoré slúžia na generovanie scenárov. V tejto časti si predstavíme jednu z týchto metód, kde užívateľ konkrétne špecifikuje marginálne rozdelenie a taktiež korelačné matice. Uvedené poznatky a značenie budeme čerpať z Høyland a Wallace (2003). V druhej časti sa pozrieme na metódy založené na trajektóriách.

### 2.1 Momentová metóda

V tejto časti sa bližšie pozrieme na základný princíp fungovania momentovej metódy. Na začiatok je potrebné vygenerovať  $n$  diskretných jednorozmerných náhodných veličín, z ktorých každá splňa dopredu špecifikované podmienky pre prvé štyri momenty. Následne ich transformuje tak, že výsledný vektor je konzistentný s danou korelačnou maticou. Avšak transformácia vygenerovaných náhodných veličín spôsobí to, že marginálne momenty vyššie ako druhého rádu sú skreslené. Narazili sme teda na prvý problém tejto metódy, ktorý sa dá avšak jednoducho vyriešiť a to tak, že bude potrebné aby sme začali s vyššou sadou momentov. Potom by tento postup viedol k presným požadovaným hodnotám pre korelácie a marginálne hodnoty momentov, ak by generované jednorozmerné náhodné premenné boli nezávislé. Avšak tu nastáva ďalší problém pretože toto platí iba ak počet výsledkov ide do nekonečna a zároveň všetky scenáre sú rovnako pravdepodobné. V prípade, že máme obmedzený počet výsledkov a možnosť, že pravdepodobnosti sú rôzne, marginálne momenty a korelácie nebudú úplne zodpovedať predpokladom. Z tohto dôvodu sa teraz bližšie pozrieme na dve rôzne možné transformácie a algoritmu, ktorý zabezpečí, že v takom to prípade je chyba stále v rámci dopredu špecifikovaných predpokladov.

#### Predpoklady korelačnej matice

Nech  $R$  je korelačná matica. Predpokladáme všeobecné vlastnosti korelačnej matice a to, že matica  $R$  je symetrická, kladná a semidefinitná s 1 na hlavnej diagonále. V prípade, že  $R$  je nekladná semidefinitná tak to môže znamenať, že údaje, ktoré máme k dispozícii sú nekonzistentné. Ďalej predpokladáme, že náhodné premenné matice  $R$  nie sú kolineárne, takže  $R$  je nesingulárna – teda pozitívna definitívna matica. Znamená to, že aspoň jednu z premenných je možné po vygenerovaní vypočítať z ostatných. Môžeme tak znížiť rozmer problému.

#### Kubicá transformácia

Cieľom kubickej transformácie je ako už bolo spomenuté vyššie transformovať jednorozmernú náhodnú veličinu, ktorú označíme ako  $\hat{Y}_i$ . Táto náhodná veličina má predom stanovené hodnoty prvých štyroch momentov, ktoré budeme označovať ako  $\mathbb{E}\hat{Y}_i^k$  kde  $k = 1,2,3,4$ . Ako prvé je teda potrebné vygenerovať  $n$  vzájomne nezávislých náhodných veličín z ľubovoľného rozdelenia. Avšak je potrebné poznať

prvých 12 momentov tohto rozdelenia. Teda dostaneme vektor  $\mathbf{X} = (X_1, \dots, X_n)$ . V ďalšom kroku je potrebné aplikovať už spomínanú kubickú transformáciu na všetky zložky vektoru  $\mathbf{X} = (X_1, \dots, X_n)$

$$\hat{Y}_i = a_i + b_i X_i + c_i X_i^2 + d_i X_i^3, i = 1, \dots, n. \quad (2.1)$$

Ako môžeme vidieť tak sme získali vektor  $\hat{\mathbf{Y}} = (\hat{Y}_1, \dots, \hat{Y}_n)$ . Pre tento vektor platí, že každá jeho zložka má prvé štyri momenty totožné s tými, ktoré sme na začiatku požadovali.

Je potrebné taktiež nájsť koeficienty  $a_i, b_i, c_i, d_i$ . Ich hodnoty získame tak, že vyjadríme prvé štyri momenty transformovanej veličiny  $\hat{Y}_i$  pomocou prvých štyroch momentov náhodnej veličiny  $X_i$  kde  $i = 1, \dots, n$ . Hľadané parametre potom získame riešením sústavy štyroch rovníc. K ich vyriešeniu je možné použiť metódu najmenších štvorcov. Táto metóda minimalizuje odchýlku prvých štyroch momentov transformácie náhodnej veličiny  $\hat{Y}_i$  od prvých štyroch momentov pôvodnej veličiny  $X_i$ . Postup riešenia rovníc nám zaručí, že aj v prípade, že sústava riešenie nemá tak získané hodnoty pre dané koeficienty nám zaručia, že prvý štyri momenty budú čo najbližšie k požadovaným momentom.

Poznámka: pri rozpísaní prvých štyroch momentov zistíme, že je potrebné poznať prvých 12 momentov veličiny  $X_i, i = 1, \dots, n$ .

## Maticová transformácia

Ako druhú si predstavíme maticovú transformáciu náhodnej premennej  $\hat{\mathbf{X}}$ , ktorá predstavuje hlavný nástroj pre nižšie uvedený algoritmus

$$\hat{\mathbf{Y}} = L\hat{\mathbf{X}}. \quad (2.2)$$

Matica  $L$  je dolná trojuholníková matica, ktorá pochádza z Cholského dekompozície korelačnej matice  $R$  a teda  $LL^T = R$ . Táto metóda nám slúži k nájdeniu náhodného vektoru  $\hat{\mathbf{Y}}$ , pre ktorý platí, že jeho stredná hodnota je rovná nule, rozptyl nadobúda hodnotu, tretí a štvrtý moment sú špecifikované a korelačná matica je v tvare  $R = LL^T$ . Na záver tejto časti sa ešte bližšie pozrieme na vlastnosti vektoru  $\hat{\mathbf{Y}}$  a  $\hat{\mathbf{X}}$ .

**Věta 1.** [Høyland a Wallace (2003)] *Nech  $\hat{\mathbf{X}}$  je  $n$ -rozmerná náhodná veličina, ktorá spĺňa nasledujúce vlastnosti:*

- $\mathbb{E}\hat{\mathbf{X}}_k$  existuje pre  $k = 1, \dots, 4$
- $\mathbb{E}\hat{\mathbf{X}}_1 = 0$  a zároveň  $\mathbb{E}\hat{\mathbf{X}}_2 = 1$
- jednotlivé zložky vektorov  $\hat{X}_i$  a  $\hat{X}_j$  sú vzájomne nezávislé pre všetky  $i \neq j$

*Nech  $L$  je dolná trojuholníková matica typu  $n \times n$  taká, že  $R$  je korelačná matica. Teda  $R$  je symetrická, pozitívne-definitná matica s jednotkami na diagonále a platí, že  $R = LL^T$ . Následne definujeme  $\hat{\mathbf{Y}}$ , pre ktorú platí, že  $\hat{\mathbf{Y}} = L\hat{\mathbf{X}}$ . Potom platia nasledujúce vlastnosti:*

- $\mathbb{E}\hat{\mathbf{Y}}_k$  existuje pre  $k = 1, \dots, 4$

- $\mathbb{E}\hat{\mathbf{Y}}_1 = 0$  a zároveň  $\mathbb{E}\hat{\mathbf{Y}}_2 = 1$
- $\hat{\mathbf{Y}}$  má korelačnú maticu  $R = LL^T$
- $\mathbb{E}\hat{\mathbf{Y}}_i^3 = \sum_{j=1}^i L_{i,j}^3 \mathbb{E}\hat{\mathbf{X}}_j^3$
- $\mathbb{E}\hat{\mathbf{Y}}_i^4 = \sum_{j=1}^i L_{i,j}^4 (\mathbb{E}\hat{\mathbf{X}}_j^4 - 3) + 3$

Ak berieme do úvahy predpoklady 1 a navyše predpokladáme, že matica  $R$  je regulárna, môžeme vyjadriť tretí a štvrtý moment  $\hat{\mathbf{X}}$  nasledovne:

- $\mathbb{E}\hat{\mathbf{X}}_i^3 = \frac{1}{L_{i,i}^3} (\mathbb{E}\hat{\mathbf{Y}}_i^3 - \sum_{k=1}^{i-1} L_{i,k}^3 \mathbb{E}\hat{\mathbf{X}}_k^3)$
- $\mathbb{E}\hat{\mathbf{X}}_i^4 = \frac{1}{L_{i,i}^4} [\mathbb{E}\hat{\mathbf{Y}}_i^4 - 3 - \sum_{k=1}^{i-1} L_{i,k}^4 (\mathbb{E}\hat{\mathbf{X}}_k^4 - 3)] + 3$

### Algoritmus

Na záver si predstavíme samotný algoritmus momentovej metódy. Základná myšlienka tohto algoritmu je, že predpokladáme rovnaké rozdelenie vygenerovaných scenárov a zadaných dát ako aj totožnosť ich prvých štyroch momentov a korelačných matic. Zadané - známe dáta budeme označovať ako  $\mathbf{Z} = (Z_1, \dots, Z_n)$ .

**KROK 1:** Ako prvé je potrebné získať prvé štyri momenty z dát  $\mathbf{Z} = (Z_1, \dots, Z_n)$ , ktoré budeme označovať ako  $\mathbb{E}\mathbf{Z}^k, k = 1, \dots, 4$ . Ďalej potrebujeme zistiť korelačnú maticu  $R$  typu  $n \times n$ .

**KROK 2:** Ako bolo uvedené v popise maticovej transformácie tak požadujeme nulovú strednú hodnotu a jednotkový rozptyl pre náhodný vektor  $\hat{\mathbf{Y}}$ . Teda  $\mathbb{E}\hat{\mathbf{Y}} = 0$  a  $\mathbb{E}\hat{\mathbf{Y}}^2 = 1$ . Potom môžeme vektor  $\mathbf{Z}$  vyjadriť nasledovne

$$\mathbf{Z} = \boldsymbol{\alpha}\hat{\mathbf{Y}} + \boldsymbol{\beta}, \quad (2.3)$$

pričom platia nasledujúce vzťahy

$$\boldsymbol{\alpha} = (\mathbb{E}\mathbf{Z}^2)^{1/2}, \boldsymbol{\beta} = \mathbb{E}\mathbf{Z}, \quad (2.4)$$

na záver je ešte potrebné vyjadriť si tretí a štvrtý moment

$$\hat{\mathbf{Y}}^3 = \frac{\mathbb{E}\mathbf{Z}^3}{\alpha^3}, \mathbb{E}\hat{\mathbf{Y}}^4 = \frac{\mathbb{E}\mathbf{Z}^3}{\alpha^4}. \quad (2.5)$$

**KROK 3:** V ďalšom kroku je potrebné aby sme pomocou Cholského rozkladu spočítali dolnú trojuholníkovú maticu  $L$ , t.j.  $R = LL^T$ .

**KROK 4:** Je potrebné aby sme určili prvé štyri momenty nezávislých jednorozmerných náhodných veličín  $\hat{\mathbf{X}}_i$ , tak aby platilo, že  $\hat{\mathbf{Y}}$ , ktoré dostaneme maticou transformáciou  $\hat{\mathbf{Y}} = L\hat{\mathbf{X}}$  malo predom určené prvé štyri momenty. Vďaka prevedenej transformácii dosiahneme, že vektor  $\hat{\mathbf{Y}}$  má požadovanú korelačnú maticu  $R$ .



$$\mathbb{E}\hat{X}_i^3 = \frac{1}{L_{i,i}^3}(\mathbb{E}\hat{Y}_i^3 - \sum_{k=1}^{i-1} L_{i,k}^3 \mathbb{E}\hat{X}_k^3), \quad (2.6)$$

$$\mathbb{E}\hat{X}_i^4 = \frac{1}{L_{i,i}^4}[\mathbb{E}\hat{Y}_i^4 - 3 - \sum_{k=1}^{i-1} L_{i,k}^4(\mathbb{E}\hat{X}_k^4 - 3)] + 3, \quad (2.7)$$

Na záver je ešte potrebné aby sme dopočítali hodnoty koeficientov  $a_i, b_i, c_i, d_i$ . Výpočet týchto koeficientov je podrobnejšie popísaný v sekcii kde sa zaoberáme kubickou transformáciou.

**KROK 5:** Vygenerujeme  $n$  vzájomne nezávislých náhodných veličín  $X_i$  z ľubovoľného rozdelenia, avšak je potrebné aby sme poznali prvých 12 momentov.

**KROK 6:** Následne prevedieme kubickú transformáciu a získame náhodné veličiny  $\hat{X}_i$ .

**KROK 7:** V poslednom kroku transformujeme vektor  $\hat{Y}$  pomocou vzťahu  $Z = \alpha\hat{Y} + \beta$ , čím získame náhodný vektor  $Z$ , ktorý má požadované prvé štyri momenty a korelačnú maticu  $R$ .

## 2.2 Scenárový strom založený na simulovaných trajektóriach

Scenárový strom je taktiež možné vytvoriť pomocou vygenerovania kompletnej trajektórie stochastického procesu  $\xi_t$ . Avšak výsledný strom nie je scenárový strom v pravom slova zmysle ako sme ho zadefinovali v časti 1.3.1, ale predstavuje sadu jednotlivých ciest, nazývaných *vejár*. Aplikáciu tejto metódy môžeme najčastejšie nájsť vo finančnom sektore. Používa sa napríklad pri simulácii krátkodobých aj dlhodobých vývojev výnosov aktív či cien týchto aktív. Pri generovaní finančných scenárov sa zväčša používa Monte Carlo metóda, kde je každá cesta generovaná nezávisle na druhej. Je potrebné poznamenať, že je dôležité poznať rozdelenie dát, ktoré chceme simulovať. Ak sa jedná o simuláciu scenárov pre dve alebo viac prvkov (napr. aktív) je taktiež potrebné poznať korelačnú štruktúru a príhodne ju zahrnúť. Predstavíme si niekoľko možných metód pre generovanie ciest. Pojmy a značenie budeme čerpať z Dupačová a Štěpán (2002), kde v prípade záujmu môžeme nájsť aj ďalšie metódy.

Metódy na generovanie ciest môžeme rozdeliť do troch hlavných skupín

1. modely náhodných prechádzok s rôznymi formami driftu
2. binominálne alebo trinomiálne modely
3. autoregresné modely

Môžeme si uviesť Vašíčkov model, ktorý bol predstavený v Vašíček (1977) a využíva sa napríklad na generovanie scenárov pre vývoj úrokových sadziieb

$$d\omega(t) = b(k - \omega(t))dt + \sigma dW(t), \quad t \leq 0, \quad (2.8)$$

kde  $b$  je kladná konštanta a  $W(t)$  je Wienerov proces.

Jedná sa o tzv. Ornstein-Uhlenbeckov proces, ktorý je definovaný ako Markovský proces s normálne rozdelenými prírastkami. Tento proces má na rozdiel od Wienerovho procesu stacionárne rozdelenie. Okamžitý drift  $b(k - \omega(t))$  smeruje proces k jeho dlhodobému priemeru  $k$ . Konštanta rozptylu  $\sigma^2$  zabezpečuje, že proces sa náhodne pohybuje okolo hodnoty  $k$ . Ak chceme aplikovať tento model na generovanie scenárov pre úrokové sadzby, je dôležité odhadnúť parametre a vybrať vhodnú časovú diskretizáciu  $\delta < 1/b$ . Potom platí vzťah:

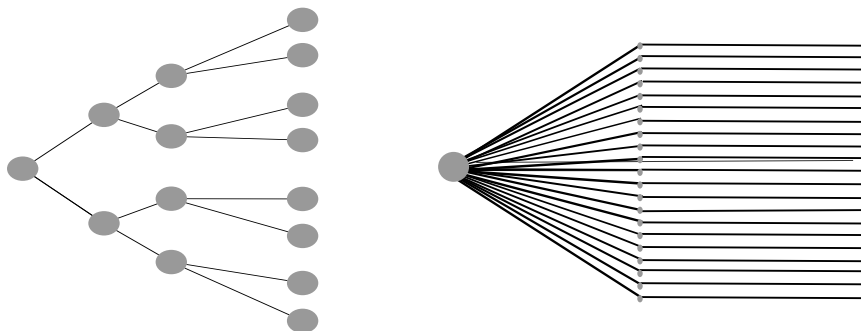
$$\omega_{(n+1)\delta} = bk\delta + \omega_{n\delta}(1 - b\delta) + \sigma\sqrt{\delta}\epsilon, \quad n = 0, 1, \dots, \quad (2.9)$$

kde  $\omega_0$  je určený ľubovoľne a nezávisle na  $\epsilon$ ,  $N(0,1)$ .

Cox-Ingersoll-Rossov model (CIR) predstavuje vylepšenie Vašíčkovho modelu. Tento model môžeme nájsť dôkladne popísaný napríklad v Cox (1985). Kľúčovým rozdielom je, že CIR model rieši otázku možnosti dosiahnutia záporných úrokových mier na bezrizikovej aktíve, ktoré je prítomné vo Vašíčkovom modeli. CIR model môžeme opísať nasledujúcou stochastickou diferenciálnou rovnicou

$$dr_t = \kappa(\alpha - r_t)dt + \sigma\sqrt{r_t}dW_t, \quad t > 0, \quad r_0 = r, \quad r, \alpha, \kappa, \sigma > 0. \quad (2.10)$$

Pokiaľ ide o generovanie trajektórií vývoja výnosov bezrizikových aktív v CIR modeli, simuláciu je možné vykonať pomocou metódy, ktorú môžeme nájsť v článku Kahl (2006), založenej na Taylorovom rozvoji druhého stupňa procesu  $(r_t)_{t=0}^T$ . Táto metóda zabezpečuje rýchlu konvergenciu diskretizovaných trajektórií k skutočnej trajektórii procesu s klesajúcou veľkosťou delenia časového intervalu  $[0, T] \Delta t$ . Ako sme už spomenuli vyššie, po na simulovaní trajektórií pomocou stochastického procesu máme k dispozícii množstvo jednotlivých scenárov, pozri obrázok 2.1.



Obr. 2.1: Scenárový strom vs scenárový vejár.

Scenárový strom s vejárovou štruktúrou avšak vedie k dvojstupňovej optimalizačnej úlohe. Na rozdiel od scenárového stromu tým pádom strácame možnosť korigovať rozhodnutia vo vyšších časových stupňoch. S týmto problém si vieme šikovne poradiť a to tak, že vytvoríme scenárový strom zo scenárového vejáru zoskupením jednotlivých scenárov do stromovej štruktúry. Je zrejmé, že je potrebné určiť štruktúru tohto stromu vopred. Konkrétne potrebujeme stanoviť počet vetiev po každom uzle v scenárovom strome. Na samotné zoskupenie scenárov do scenárového stromu sa v súčasnej dobe používa prístup založený na klastrovej analýze, ktorý je podrobne popísaný v Dupačová a Wallace (2000).

Scenárový strom s vejárovou štruktúrou však vedie k dvojstupňovej optimalizačnej úlohe. Na rozdiel od scenárového stromu preto strácame schopnosť upravovať rozhodnutia v vyšších časových úrovniach. S týmto problémom si vieme efektívne poradiť tak, že vytvoríme scenárový strom zo scenárového vejáru zoskupovaním jednotlivých scenárov do stromovej štruktúry. Je zjavné, že je potrebné určiť štruktúru tohto stromu vopred a to tak, že stanovíme počet vetiev po každom uzly v scenárovom strome. Na samotné zoskupovanie scenárov do scenárového stromu sa v súčasnosti používa metóda založená na klastrovej analýze, ktorá je detailne vysvetlená v Dupačová a Wallace (2000). Je dôležité poznamenať, že zoskupenie je prevádzané na základe neanticipativnosti rozhodnutí. To znamená, že rozhodnutie, ktoré boli vykonané v akejkolvek fáze procesu nezávisia na žiadnych budúcich realizáciách náhodných dát a ani na budúcich rozhodnutiach. Je možné využívať iba informácie, ktoré poznáme z minulosti a poznatky o pravdepodobnostnom rozdelení dát.

Scenárový strom s konštrukciou podobnou vejáru však vedie k dvojfázovej optimalizačnej úlohe. V porovnaní so scenárovým stromom preto strácame schopnosť upravovať rozhodnutia vo vyšších časových úrovniach. Tento problém vieme pomerne jednoducho vyriešiť a to tak, že vytvoríme scenárový strom zo scenárového vejáru zoskupovaním jednotlivých scenárov do stromovej štruktúry. Je zrejmé, že je potrebné určiť štruktúru tohto stromu vopred. Konkrétne potrebujeme stanoviť počet vetiev pre každý uzol v scenárovom strome. Pre zoskupovanie scenárov do scenárového stromu sa v súčasnosti používa metóda založená na klastrovej analýze, ktorá je podrobne opísaná v Dupačová a Wallace (2000). Dôležité je poznamenať, že zoskupovanie sa uskutočňuje s ohľadom na neanticipativitu rozhodnutí. To znamená, že rozhodnutia prijaté v akejkolvek etape procesu nezávisia na žiadnych budúcich realizáciách náhodných dát ani budúcich rozhodnutiach. Je možné využívať iba informácie známe z minulosti a poznatky o pravdepodobnostnom rozdelení dát.

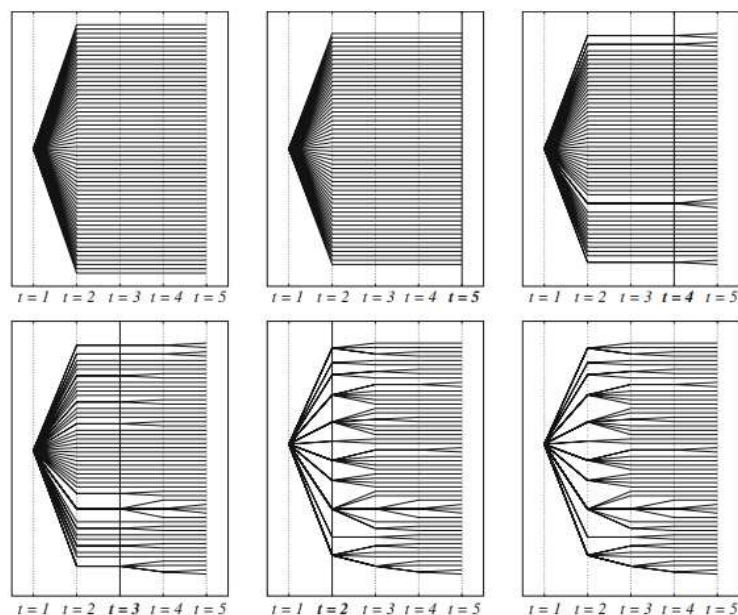
Pre lepšie porozumenie sa na to pozrieme bližšie. Ako prvé sa pozrieme na rozhodnutie  $\mathbf{x}_1$  v prvom stupni. Na základe princípu konštrukcie scenárového stromu požadujeme aby bolo toto rozhodnutie pre všetky scenáre  $s = 1, \dots, S$  rovnaké

$$\mathbf{x}_1^s = \mathbf{x}_1^r, \quad 1, \leq s < r \leq S. \quad (2.11)$$

Analogicky je túto podmienku potrebné aplikovať na rozhodnutia v každom stupni. Uvažujeme dva scenáre  $s$  a  $r$ . V prípade, že hodnoty  $\xi_{[t]}^s$  a  $\xi_{[t]}^r$  sú totožné potom chceme aby aj rozhodnutia  $\mathbf{x}_{[t]}^s = \mathbf{x}_{[t]}^r$  v stupni  $t$  boli zhodné. Podmienku neanticipativnosti môžeme obecné zapísať nasledovne

$$\mathbf{x}_{[t]}^s = \mathbf{x}_{[t]}^r, \quad \forall s, r \quad \text{pre ktoré platí, že} \quad \xi_{[t]}^s = \xi_{[t]}^r, \quad t = 1, \dots, T. \quad (2.12)$$

Na obrázku 2.2 (prevzaté z Heitsch (2009)) môžeme vidieť konštrukciu scenárového stromu z pôvodného scenárového vejáru na základe podmienok neanticipativnosti.



Obr. 2.2: Prevedenie scenárového vejáru na scenárový strom.

## 2.3 Klastrovací algoritmus

Ako sme už naznačili vyššie, na prevedenie scenárového vejáru na scenárový strom je potrebné použiť takzvaný *klastrovací algoritmus*. Existuje množstvo rozličných typov klastrovacích algoritmov. Všeobecný prehľad rôznych typov môžeme nájsť napríklad v Han a Kamber (2011) alebo Jain a Dubes (1998). Jedným z prvých takýchto algoritmov je tradičný K-means algoritmus, ktorý bol úvodne predstavený v článku Macqueen (1967). K-means algoritmus je založený na minimalizácii súčtu štvorcov L2 vzdialeností v rámci jednotlivých klastrov. Centroid je určený priemerným scenárom v rámci klastru; to znamená, že nemusí byť súčasťou pôvodnej množiny. Táto metóda vektorovej kvantizácie je obzvlášť vhodná pri redukcii scenárov. Avšak K-means algoritmus si vyžaduje veľké množstvo simulovaných trajektórií. Napriek tomu môže v pokročilejších fázach nastať problém s nedostatočným množstvom trajektórií v porovnaní s počtom očakávaných klastrov. Toto je spôsobené tým, že klastre nemajú žiadnu teoretickú garanciu veľkosti. Samozrejme, tento problém by bolo možné riešiť zväčšovaním počtu simulovaných trajektórií, ale tým by zároveň rástla aj zložitosť scenárového stromu a rovnako aj výpočtová náročnosť, vzhľadom na nelineárnu zložitosť algoritmu. Z tohto dôvodu nie je aplikácia tejto metódy ideálna a preto sme sa rozhodli bližšie preskúmať upravenú verziu, tzv. *obmedzený klastrovací algoritmus*. Túto metódu môžeme nájsť podrobne opísanú v článku Bradley (2000). Keďže metóda bude použitá v praktickej časti práce, stručne si tento algoritmus predstavíme. Vďaka tomuto algoritmu sme schopní riešiť nelineárny zmiešaný celočíselný optimalizačný problém.

$$\min_{\mathbf{C} \in \mathbb{R}^{K \times M}, \mathbf{Z} \in \{0,1\}^{N \times K}} \frac{1}{2} \sum_{i=1}^N \sum_{k=1}^K z_{i,k} \|x_i - c_k\|_2^2 \quad (2.13)$$

$$\text{s.t.} \quad \sum_{i=1}^N z_{i,k} \geq n_k, \quad k \in \{1, \dots, K\}, \quad (2.14)$$

$$\sum_{k=1}^K z_{i,k} = 1, \quad i \in \{1, \dots, N\}, \quad (2.15)$$

$$(2.16)$$

kde  $x_i$  pre  $i \in \{1, \dots, N\}$  predstavujú riadkové vektory matice dát  $\mathbf{X} = (x_{i,j}^{N,M})_{i,j=1}$ . Následne matica  $\mathbf{Z} = (z_{i,k}^{N,K})_{i,k=1}$  predstavuje zaradenie vektoru do jedného z  $K$  klastrov. Konkrétne, príslušnosť vektora  $i$  do klastru  $k$  kódujeme  $i$ -tým riadkom matice  $Z$  ako  $z_{i,k} = 1$ ,  $z_{i,k'} = 0$ ,  $k' \neq k$ . Premenné  $c_k, k \in \{1, \dots, K\}$  sú riadkové vektory matice centroidov  $\mathbf{C} = (c_{k,j})_{k,j=1}^{K,M}$  a  $n_k, k \in \{1, \dots, K\}$  sú minimálne veľkosti jednotlivých klastrov.

## 2.4 Dekompozičné metódy

Časové rady môžeme rozložiť na niekoľko zložiek, menovite

- $Tr_t$  - trendová zložka, ktorá popisuje dlhodobé správanie časových rád. Konkrétne nás zaujíma jej dlhodobý rast alebo pokles s tým, že krátkodobé fluktuácie pri popise trendu neberieme do úvahy.
- $Sz_t$  - sezónna zložka, popisuje periodické zmeny v časovej rade, ktoré sa odohrávajú behom jedného kalendárneho roku prípadne dlhšieho časového obdobia a pravidelne sa opakujú
- $C_t$  - cyklická zložka, ktorá popisuje fluktuácie okolo trendu, v ktorých sa striedajú fázy rastu s fázami poklesu
- $\epsilon_t$  - reziduálna zložka, ktorá má náhodný charakter, pričom má vlastnosti bieleho šumu a v niektorých prípadoch sa vyskytuje v podobe normálneho bieleho šumu

Cieľom dekompozície časových rád je eliminácia jednej alebo viacerých dekompozičných zložiek. V tejto práci nás bude zaujímať eliminácia sezónnej zložky. V praxi a odbornej literatúre to môžeme nájsť aj pod pojmom *sezónne očistenie*. Pojmy a prístupy na odstránenie sezónnosti budeme v tejto časti čerpať z knihy Cipra (2009).

### Sezónnosť

Na odstránenie sezónnej zložky existuje mnoho metód a rôznych modelov. Hlavným cieľom všetkých metód, je separácia sezónnej zložky, tzv. *sezónne faktory*  $I_1, I_2, \dots, I_s$ , kde  $s$  označuje dĺžku sezóny. Sezónne faktory potom slúžia na modelovanie sezónnej zložky  $Sz_t$  v jednotlivých sezónach. Sezónne faktory môžeme získať dvomi spôsobmi dekompozícií a to

1. aditívna dekompozícia -  $y_t = Tr_t + Sz_s + \epsilon_t$ , kde  $I_t$  sa meria v rovnakých jednotkách ako príslušná časová rada  $y_t$
2. multiplikatívna dekompozícia -  $y_t = Tr_t * Sz_s * \epsilon_t$ , kde  $I_t$  je v tomto prípade bezrozmerná veličina.

V tejto práci sa primárne zameriame na Holt-Wintersovu metódu a regresný prístup k sezónnosti, ktorý je využívaný pre voľbu počiatočných hodnôt vo Holt-Wintersovej metóde.

### 2.4.1 Regresný prístup

Uvažujeme časovú radu, ktorá obsahuje sezónnu zložku. V prípade aditívnej sezónnosti sa sezónnosť modeluje ako kvalitatívna premenná v lineárnom modeli s použitím  $s - 1$  dummy premenných ( $s$  označuje dĺžku sezóny). Všeobecne potom uvažujeme regresný model v tvare

$$y_t = \beta_0 + \beta_1 t + \alpha_2 x_{t2} + \dots + \alpha_s x_{ts} + \epsilon_t, \quad (2.17)$$

kde  $x_{t2}, \dots, x_{ts}$  sú dummy premenné, ktoré definujeme ako

$$x_{tj} = \begin{cases} 1 & \text{ak } t = j, \\ 0 & \text{inak.} \end{cases} \quad (2.18)$$

Sezónna zložka je vyjadrená ako

$$Sz_t = \alpha_2 x_{t2} + \dots + \alpha_s x_{ts}. \quad (2.19)$$

Odhady parametrov  $b_0, b_1, a_2, a_3, \dots, a_s$  je možné získať metódou najmenších štvorcov. Potom môžeme sezónnu zložku vyjadriť ako

$$Sz_1 = 0 \quad (2.20)$$

$$Sz_2 = a_2, \dots, Sz_s = a_s, \quad (2.21)$$

na záver trendovú zložku vyjadríme ako

$$Tr_t = b_0 + b_1 t. \quad (2.22)$$

### 2.4.2 Holt-Wintersová metóda

Holt-Wintersová metódu prvýkrát predstavil Američan Charles C. Holt v roku 1957. Novšie spracovanie môžeme nájsť v Holt (2004). Popisovaná metóda ako aj jej aplikácia je popísaná taktiež v Winters (1960). Metóda funguje na princípe exponenciálneho vyrovnávania a dokáže si poradiť aj s časovými radami, ktoré obsahujú sezónnu zložku. Rozlišujeme dve varianty a to aditívnu a multiplikatívnu. V oboch prípadoch je potrebné zaviesť tri vyrovnávacie konštanty

1.  $\alpha$  pre úroveň  $L_t$ , pričom  $0 < \alpha < 1$ ,
2.  $\gamma$  pre smernicu trendu  $T_t$ , pričom  $0 < \gamma < 1$ ,
3.  $\delta$  pre sezónnu zložku  $Sz_t$ , pričom  $0 < \delta < 1$ .

## Aditívna Holt-Wintersová metóda

Model popíšeme pomocou rekurentných vzťahov. Predpokladáme, že trend je lokálne lineárny. Nech  $s$  je počet pozorovaní v jednej sezónne. V prípade, že uvažujeme napríklad štvrtročnú sezónu tak  $s = 4$ , mesačnú sezónu  $s = 12$ .

Pre úroveň rady  $L_t$ , ktorá je rovná konvexnej kombinácii aktuálnej pozorovanej hodnoty  $y_t$  očistenej o najaktuálnejšie odhadnutú zložku  $Sz_{t-s}$  z predchádzajúcej sezóny a súčtu úrovne rady  $L_{t-1}$  v čase  $t - 1$  a odhadu  $T_{t-1}$  smernice trendu v čase  $t - 1$  platí

$$L_t = \alpha(y_t - Sz_{t-s}) + (1 - \alpha)(L_{t-1} + T_{t-1}). \quad (2.23)$$

Trend  $T_t$  získame ako

$$T_t = \gamma(L_t - L_{t-1}) + (1 - \gamma)T_{t-1}, \quad (2.24)$$

kde  $L_t - L_{t-1}$  je odhad smernice trendu v čase  $t$  a  $T_{t-1}$  je odhad trendu v čase  $t - 1$ . Sezónnu zložku vyjadríme ako

$$Sz_t = \delta(y_t - L_t) + (1 - \delta)Sz_{t-s}, \quad (2.25)$$

kde  $y_t - L_t$  je odhad aktuálnej sezónnej zložky a  $Sz_{t-s}$  je odhad sezónnej zložky v čase  $t - s$ . Vyrovnaná hodnota  $\hat{y}_t$  je rovná súčtu úrovne a sezónnej zložke v čase  $t$

$$\hat{y}_t = L_t + Sz_t. \quad (2.26)$$

Čo sa týka predpovedi, je potrebné rozlíšiť o koľko sezón dopredu predpovedáme. Je dôležité počítať s tým, že predpovede, ktoré budú viac vzdialené do budúcnosti už nemusia byť úplne presné. Pre predpoveď v čase  $t + \tau$ , kde  $\tau \in \{1, \dots, s\}$  platí

$$\hat{y}_{t+\tau} = L_t + T_t * \tau + Sz_{t+\tau-s}. \quad (2.27)$$

Pre predpoveď v čase  $t + \tau$ , kde  $\tau \in \{s + 1, \dots, 2s\}$  platí

$$\hat{y}_{t+\tau} = L_t + T_t * \tau + Sz_{t+\tau-2s}, \quad (2.28)$$

Analogicky pre  $\tau > 2s$ .

Pri voľbe vyrovnávajúcich konštánt sa používa na základe literatúry fixná voľba

$$\alpha = \delta = 0.4 \quad (2.29)$$

$$\gamma = 0.1. \quad (2.30)$$

Realizácia rekurentných vzorcov ďalej vyžaduje voľbu počiatočných hodnôt  $L_0, T_0, Sz_{-s+1}, \dots, Sz_0$  na čo použijeme regresný prístup k sezónnosti popísaný v 2.4.1. Na základe odhadu regresného modelu  $b_0, b_1, a_2, a_3, \dots, a_s$  získame nasledujúce inicializácie

$$L_0 = b_0, \quad (2.31)$$

$$T_0 = b_1, \quad (2.32)$$

$$Sz_{-s+1} = 0, \quad Sz_{-s+2} = a_2, \dots, Sz_0 = a_s. \quad (2.33)$$

## Multiplikatívna Holt-Wintersová metóda

Vzťahy pre multiplikatívny model sú analógiou aditívnej verzie. Vzhľadom na to, že predpokladáme časovú radu s multiplikatívnou sezónnou zložkou tak súčty sú nahradené súčinnami a rozdiely sú nahradené podielmi.

Rekurentné vzťahy pre úroveň, smernicu trendu, sezónnosť a predpoveď sú dané nasledovne

$$L_t = \alpha \frac{y_t}{Sz_{t-s}} + (1 - \alpha)(L_{t-1} + T_{t-1}), \quad (2.34)$$

$$T_t = \gamma(L_t - L_{t-1}) + (1 - \gamma)T_{t-1}, \quad (2.35)$$

$$Sz_t = \delta \frac{y_t}{L_t} + (1 - \delta)Sz_{t-s}, \quad (2.36)$$

$$\hat{y}_t = L_t * Sz_s, \quad (2.37)$$

$$\hat{y}_{t+\tau} = (L_t + T_t * \tau) * Sz_{t+\tau-s}, \quad \tau = 1, \dots, s \quad (2.38)$$

$$\hat{y}_{t+\tau} = (L_t + T_t * \tau) * Sz_{t+\tau-2s}, \quad \tau = s + 1, \dots, 2s. \quad (2.39)$$

Analogicky získame predpovede pre ďalšie sezóny,  $\tau > 2s$ .

Opäť je nutné pre realizácie zvoliť vyrovnávacie konštanty a počiatkové hodnoty. Voľba vyrovnávacích konštánt je zhodná ako pri aditívnej metóde. Avšak pri voľbe počiatkových hodnôt nie je v tomto prípade možné použiť regresný prístup k sezónnosti a teda odhadneme počiatkové hodnoty nasledujúcimi vzťahmi

$$T_0 = \bar{y}_m - \frac{\bar{y}_1}{(m-1)s}, \quad (2.40)$$

$$L_0 = \bar{y}_1 - \frac{s+1}{2}T_0, \quad (2.41)$$

$$Sz_{j-s} = \frac{1}{m} \sum_{i=0}^{m-1} \frac{y_{j+s*i}}{\bar{y}_{i+1} - \left(\frac{s+1}{2} - j\right)T_0}, \quad j = 1, \dots, s. \quad (2.42)$$



### 2.4.3 Algoritmus na predpovedanie viacrozmerých časových rád pomocou Holt-Wintersovej metódy

V tejto časti si predstavíme algoritmus, ktorý slúži na predikovanie budúcich pozorovaní vo viacrozmerých časových radoch s korelovanými párami premenná-subpremenná. Algoritmus využíva Holt-Wintersovu metódu na elimináciu sezónnej zložky a následné generovanie intervalových predpovedí. Okrem toho zachováme korelácie počas procesu generovania trajektórií a používame klastrovací algoritmus na vytvorenie stromovej štruktúry, ktorá zachytáva hierarchické vzťahy medzi párami premenných a podpremenných.

#### Štruktúra vstupných dát

Vstupné dáta je viacrozmerná časová rada s  $X$  premennými a  $Z$  podpremennými pre každú premennú, čo vedie k  $X * Z$  párom premenná-podpremenná. Každý pár premenná-podpremenná je korelovaný. Časový rad je reprezentovaný maticou  $T$  s rozmermi  $M \times (X * Z)$ , kde  $M$  je počet časových bodov (pozorovaní),  $X$  je počet premenných a  $Z$  je počet podpremenných pre každú premennú.

Použitím tohto zápisu  $T(i, j * Z + k)$  predstavuje  $k$ -tu podpremennú  $j$ -tej premennej v časovom bode  $i$ , kde  $0 \leq k < Z$  a  $0 \leq j < X$ .

#### Algoritmus

Algoritmus možno načrtnúť takto:

1. **Eliminácia sezónnosti:** Aplikujeme Holt-Wintersovu metódu na vybraný stĺpec historických dát (premenná-podpremenná), ktoré predstavujú časový rad do času  $t$ .
2. **Bodová predikcia:** Za pomoci Holt-Wintersovej metódy, ktorú použijeme na očistené dáta do času  $t$  vypočítame predikciu do bodu  $t + 1$ .
3. **Intervalová predikcia:** Na základe bodovej predikcie vygenerujeme interval predikcie pre bodovú predikciu v čase  $t + 1$ .
4. **Generovanie trajektórií:**
  - Vygenerujeme nový bod  $x_1$  v rámci tohto intervalu pridaním alebo odčítaním percentuálnej veľkosti intervalu, túto hodnotu označíme ako  $\epsilon$ , od bodového odhadu vykonaného v kroku 2. Hodnotu  $\epsilon$  je potrebné si zapamätať.
  - K historickým dátam pridáme novo vytvorený bod  $x_1$  v čase  $t + 1$  a aplikujeme opäť Holt-Wintersovu metódu na vytvorenie bodového odhadu do času  $t + 2$ . Vytvoríme predikčný interval pre tento bod a zopakujeme krok 4, aby sme získali nový bod  $x_2$ , pričom si zapamätáme zodpovedajúcu hodnotu  $\epsilon$ .

- Pridáme k historickým dátam s bodom  $x_1$  aj bod  $x_2$  a aplikujeme Holt-Wintersovu metódu na vytvorenie bodového odhadu do času  $t+3$ . Opäť si skonštruujeme predikčný interval pre tento bod a zopakujeme krok 4, aby sme získali nový bod  $x_3$ , pričom si zapamätáme opäť zodpovedajúcu hodnotu  $\epsilon$ .
  - Pokračujeme v tomto procese analogicky až pokiaľ sa dostaneme do času  $t + n$ . Kde  $n$  je počet krokov v čase pre ktorý chceme generovať trajektórie. Získame tak jednu trajektóriu.
  - Následne celý tento postup opakujeme až pokiaľ nedosiahneme požadovaný počet trajektórií.
5. Tento postup je potrebné zopakovať pre každý stĺpec dát (dvojica premenná-podpremenná).
  6. **Zachovanie korelácie:** Na zachovanie korelácií medzi párami premenná-subpremenná je potrebné zabezpečiť, že hodnoty  $\epsilon$  sú pre každý pár identické počas celého procesu generovania trajektórií. Na konci tohto procesu dostaneme požadované množstvo trajektórií medzi, ktorými sú zachované požadované korelácie.
  7. **Klastrovanie a konštrukcia stromov:** Na konštrukciu scenárových stromov z trajektórií použijeme klastrovací algoritmus, ako napríklad k-means, na zoskupenie vygenerovaných trajektórií na základe ich podobností. Tento krok znižuje zložitosť údajov pri zachovaní základných vzťahov medzi párami premenná-podpremenná. K-means algoritmus je možné previesť na každej dvojici premenných a získať tak  $X * Z$  stromov alebo previesť K-means algoritmus na všetkých trajektóriách pre dvojice premennej a subpremennej na raz a získať tak  $X * Z$  dimenzionálny strom.

Tento algoritmus, použitý na viacrozmerný časový rad s korelovanými párami premenná-subpremenné, poskytuje efektívny prístup na predpovedanie budúcich hodnôt pri zachovaní základných vzťahov medzi párami. Použitie Holt-Wintersovej metódy na elimináciu sezónnosti a predikciu intervalov zabezpečuje, že algoritmus je schopný zachytiť zložité závislosti v dátach. Okrem toho, použitie techník klastrovania a konštrukcie stromovej štruktúry umožňuje robustnú reprezentáciu hierarchických vzťahov medzi párami premenných a podpremenných.

# 3. Optimalizácia rozvozu liekov do lekární

V tejto kapitole si predstavíme dopravný problém. Ďalej sa budeme zaoberať zostavením optimalizačného modelu distribúcie rôznych typov balení jedného lieku do lekární po meste Praha. Ako prvé sa pozrieme na dvojstupňovú formuláciu a následne tento model zovšeobecníme na viacstupňovú úlohu. Na záver sa pozrieme na dve rozšírenia viacstupňovej úlohy.

## 3.1 Dopravný problém

Táto úloha predstavuje základnú a zároveň najjednoduchšiu verziu optimalizačného problému, ktorým sa v tejto práci budeme zaoberať. V tejto časti budeme vychádzať z knihy Dupačová a Lachout (2011).

Na účely tejto práce použijeme model dopravného problému priamo na optimalizáciu rozvozu liekov do lekární. Konkrétne sformulujeme model distribúcie jedného druhu liekov zo  $J$  skladov do  $I$  lekární. Predpokladáme, že každý lekárnik pozná presné množstvo liekov, ktoré je potrebné objednať na daný mesiac. Teda dopyt každej lekárne  $d_i$  pre  $i = 1, \dots, I$  je vopred známy a pracujeme s konkrétnymi hodnotami. Cieľom problému je optimalizovať koľko balení liekov a z akého skladu objednať, aby sa minimalizovali náklady na dopravu zo skladov do lekární. Objektívna funkcia sa preto javí ako súčin nákladov na prepravu lieku zo skladu  $j$  do lekárne  $l$  a počtu prepravených balení lieku. Účelová funkcia teda vyzerá ako

$$f(\mathbf{x}) = \sum_{i=1}^I \sum_{j=1}^J c_{i,j} x_{i,j}, \quad (3.1)$$

kde  $c_{i,j}$  predstavuje cenu prepravy lieku zo skladu  $j$  do lekárne  $l$  a  $x_{i,j}$  je počet balení lieku prepravených zo skladu  $j$  do lekárne  $i$ .

Pre tento problém je ešte potrebné zahrnúť obmedzenia. Každý sklad môže dodať maximálne toľko liekov, koľko má na sklade, čo označíme ako  $b_j$  pre  $j = 1, \dots, J$ . Toto obmedzenie môžeme vyjadriť ako (3.3). Ďalej sa vyžaduje, aby bol uspokojený dopyt každej lekárne, čo znamená, že každá lekárňa musí dostať aspoň také množstvo liekov, aké si objednala, čo je vyjadříme (3.4). Model možno formulovať nasledovne

$$\min \sum_{i=1}^I \sum_{j=1}^J c_{i,j} x_{i,j} \quad (3.2)$$

$$\text{s.t.} \quad \sum_{i=1}^I x_{i,j} \leq b_j, \quad \forall j = 1, \dots, J, \quad (3.3)$$

$$\sum_{j=1}^J x_{i,j} \geq d_i, \quad \forall i = 1, \dots, I, \quad (3.4)$$

$$x_{i,j} \geq 0, \quad \forall i = 1, \dots, I, \quad \forall j = 1, \dots, J.$$

## 3.2 Dvojstupňová formulácia úlohy založená na scenároch

Model budeme formulovať pre distribúciu rôznych typov balení lieku z jedného skladu do lekární v rámci mesta Praha. Objednávanie a dodávanie liekov prebieha na začiatku každého mesiaca a model sa zameriava na jednomesačné časové obdobie. Predpokladáme, že dopyt vo všetkých lekárňach je náhodný, čo nás vedie na dvojstupňový optimalizačný problém. Každá lekáreň nie je obmedzená na nákup iba jedného druhu balenia, ale môže si objednať rôzne množstvá rôznych typov balení. Na základe tohto rozhodnutia optimalizujeme doručovanie liekov do lekární z vybraného skladu. Počas nasledujúceho mesiaca zákazníci navštevujú lekárne a nakupujú rôzne množstvá rôznych druhov balení liekov. Na záver spočítame celkovú hodnotu, ktorá predstavuje rozdiel predaja a nákupu všetkých typov liekov. Cieľom úlohy je maximalizovať zisk lekární. V rámci tohto modelu budeme predpokladať, že dátum spotreby liekov nie je obmedzený.

Úlohy v tejto a nasledujúcej sekcii budeme formulovať v závislosti na nasledujúcom značení:

- $I$  ... celkový počet lekární
- $L$  ... celkový počet liekov
- $c_{i,l}$  ... cena, za ktorú lekáreň  $i$  kúpi liek  $l$
- $v_{i,l}$  ... cena, za ktorú lekáreň  $i$  predáva liek  $l$

Dopyt lekárne  $i$  po lieku  $l$  považujeme za náhodný a vyjadríme ho ako  $\xi(\omega) = d(\omega)$ . Závislosť na náhode vyjadruje náhodný vektor  $\xi(\omega)$ , kde element  $\omega$  predstavuje náhodu. Predpokladáme, že je diskrétno rozdelený a jeho jednotlivé realizácie

$$\boldsymbol{\xi}^s = \boldsymbol{\xi}(\omega^s) = \mathbf{d}(\omega^s) = \mathbf{d}^s, \quad s = 1, \dots, S \quad (3.5)$$

budeme nazývať scenáre. Za platnosti  $s$ -tého scenáru značíme realizáciu ako  $\mathbf{d}^s = \mathbf{d}(\omega^s) = (d_{1,1}^s, \dots, d_{I,L}^s)$ , kde  $d_{i,l}^s$  vyjadruje dopyt zákazníkov po lieku  $l$  v lekární  $i$  za platnosti scenára  $s$ . Počet balení lieku  $l$  kúpených do lekárne  $i$  označíme ako  $\mathbf{x}(\omega)$ . Rozhodnutie  $\mathbf{x}(\omega)$  za platnosti scenára  $\boldsymbol{\xi}^s$  budeme označovať ako  $\mathbf{x}^s = (x_{1,1}^s, \dots, x_{I,L}^s)$ , kde  $x_{i,l}^s$  vyjadruje počet kúpených balení lieku  $l$  do lekárne  $i$  za platnosti scenára  $s$ . Následne počet predaných balení lieku označíme premennou  $\mathbf{y}(\omega)$ . Rozhodnutie  $\mathbf{y}(\omega)$  za platnosti scenára  $\boldsymbol{\xi}^s$  budeme označovať ako  $\mathbf{y}^s = (y_{1,1}^s, \dots, y_{I,L}^s)$ , kde  $y_{i,l}^s$  vyjadruje počet predaného lieku  $l$  v lekární  $i$  za platnosti scenára  $s$ . Na záver pravdepodobnosť scenára  $s$  označíme ako  $p^s$  pre  $s = 1, \dots, S$ .

Ako sme spomínali vyššie, tak cieľom je maximalizovať zisk lekární a teda budeme úlohu riešiť ako úlohu stochastického programovania. Účelová funkcia vyjadruje rozdiel celkových nákladov lekární a zisku z predaných balení liekov, pričom tržba z predaných balení liekov je vyjadrená v tvare účelovej funkcie druhého stupňa. Účelovú funkciu formulujeme nasledovne

$$f(\mathbf{x}, \mathbf{y}) = \sum_{s=1}^S \sum_{i=1}^I \sum_{l=1}^L p^s (-c_{i,l} x_{i,l}^s + v_{i,l} y_{i,l}^s). \quad (3.6)$$

V nasledujúcej časti sa pozrieme na obmedzenia, ktoré musia byť splnené. Ako prvé formulujeme obmedzenie, ktoré nám hovorí, že za platnosti  $s$ -tého scenáru počet predaných balení rôznych typov balení lieku v každej lekárni nesmie byť väčší ako množstvo balení, ktoré bolo každou lekárňou nakúpené. Zapišeme ako

$$y_{i,l}^s \leq x_{i,l}^s, \quad \forall s = 1, \dots, S, \forall l = 1, \dots, L, \forall i = 1, \dots, I \quad (3.7)$$

Druhé obmedzenie zabezpečuje, že množstvo balenia lieku  $l$ , ktoré sa predá je menšie alebo rovné, ako dopyt po lieku  $l$  v lekárni  $i$  v scenári  $s$ . Je dôležité uviesť, že sa jedná o rôzne veľkosti balení toho istého lieku. Predpokladáme, že v prípade ak nebude dostupné balenie, ktoré požaduje zákazník tak s pravdepodobnosťou  $q$  si zakúpi inú veľkosť balenia tohto lieku. Ak by nastala situácia, že zákazník si chce zakúpiť iba konkrétnu veľkosť balenia tak opäť s určitou pravdepodobnosťou  $r$  si pôjde toto balenie kúpiť do inej lekárne. Konkrétne môžeme rozpísať nasledovne,

$$q_k = \begin{cases} 1 & \text{ak } k = l, \\ q & \text{ak } k \neq l, \end{cases} \quad (3.8)$$

$$r_j = \begin{cases} 1 & \text{ak } j = i, \\ r & \text{ak } j \neq i. \end{cases} \quad (3.9)$$

Kde prípad kedy  $k = l$  alebo  $j = i$  vyjadruje prípad kedy sa zákazník nachádza v lekárni v ktorej majú k dispozícii daný typ balenia a teda pravdepodobnosť, že zákazník pôjde do tejto lekárne je 1. Spolu môžeme toto obmedzenie zapísať ako

$$y_{i,l}^s \leq \sum_{j=1}^I \sum_{k=1}^L d_{i,l}^s q_k r_j, \quad \forall s = 1, \dots, S, \forall l = 1, \dots, L, \forall i = 1, \dots, I. \quad (3.10)$$

Teda dvojstupňový stochastický model pre  $L$  lekárni a  $I$  rôznych balení lieku formulujeme ako:

$$\max_{x^1, y^1, \dots, x^S, y^S} \sum_{s=1}^S \sum_{i=1}^I \sum_{l=1}^L p^s (-c_{i,l} x_{i,l}^s + v_{i,l} y_{i,l}^s) \quad (3.11)$$

$$\text{s.t. } y_{i,l}^s \leq x_{i,l}^s, \forall s, i, l \quad (3.12)$$

$$y_{i,l}^s \leq \sum_{j=1}^I \sum_{k=1}^L d_{i,l}^s q_k r_j, \forall s, i, l, \quad (3.13)$$

$$y_{i,l}^s \geq 0, \forall s, i, l, \quad (3.14)$$

$$x_{i,l}^s \geq 0, \forall s, i, l. \quad (3.15)$$

### 3.3 T-stupňová formulácia úlohy

V tejto časti sa zameriame na zostavenie modelu pre T-stupňovú formuláciu úlohy. Účelová funkcia ako aj obmedzenia budú vychádzať z dvojstupňového modelu 3.11, ktorý sme skonštruovali v predchádzajúcej časti. Úlohu budeme formulovať najprv pre jednu lekárňu a jeden druh balenia lieku. Následne úlohu rozšírime pre  $L$  lekárni a  $I$  liekov.

### 3.3.1 Model pre jednu lekárňu a jeden typ lieku

Začneme s formuláciou modelu, kde budeme predpokladať jednu lekárňu a jeden typ balenia lieku. V T-stupňovej formulácii sa už nebudeme obmedzovať len na rozpätie jedného mesiaca ale budeme brať do úvahy  $T$  po sebe idúcich mesiacov. Ďalej, budeme predpokladať, že dané liečivo má presne stanovený dátum expirácie, ktorý budeme taktiež uvažovať v mesiacoch. Označíme ako  $N$ . Je potrebné poznamenať, že vo všetkých modeloch prezentovaných v tejto práci budeme predpokladať, že  $N < T$ . Je potrebné si zaviesť novú premennú  $y_{n,t}^s$ , ktorá predstavuje počet predaných liekov v čase  $n$ , zakúpených v čase  $t$  za platnosti scenára  $s$ . Je dôležité poznamenať, že čo sa týka počtu objednaných balení každou lekárňou za platnosti daného scenára  $x_{i,l,t}^s$  tak hodnota v stupni  $t = 1$  je deterministicky daná a rovnaká pre každý scenár.

Účelová funkcia bude vyjadrovať rozdiel celkových nákladov lekárne a zisku z predaných liekov. Na záver je však potrebné ešte pripočítať množstvo nepredaných liekov, ktoré je možné vrátiť avšak nie za plnú cenu. Zapišeme nasledovne

$$f(\mathbf{x}, \mathbf{y}) = \sum_{s=1}^S \sum_{t=1}^T p^s (-cx_t^s + v \sum_{n=1}^T y_{n,t}^s + \lambda cz_t^s) \quad (3.16)$$

Ako môžeme vidieť, tak posledný člen nám vyjadruje hodnotu nepredaných liekov, ktoré sú na konci obdobia vrátené dodávateľovi, ktorý nám vráti zlomkovú cenu z pôvodnej. Túto cenu vypočítame tak, že zoberieme pôvodnú cenu jedného balenia lieku  $c$  a vynásobíme ju konštantou  $\lambda < 1$ . Premennú  $z_t^s$  môžeme rozpísať ako

$$0 \leq n - t < N : z_t^s = \left[ \sum_{t=1}^T x_t^s - \sum_{n=1}^T \sum_{t=1}^T y_{n,t}^s \right] \forall s = 1, \dots, S. \quad (3.17)$$

Teda T-stupňový model pre jednu lekárňu a jeden typ balenia lieku zapišeme ako

$$\max_{\mathbf{x}^1, \mathbf{y}^1, \dots, \mathbf{x}^S, \mathbf{y}^S} \sum_{s=1}^S \sum_{t=1}^T p^s (-cx_t^s + v \sum_{n=1}^T y_{n,t}^s + \lambda cz_t^s) \quad (3.18)$$

$$\text{s.t. } 0 \leq n - t < N : \sum_{n=1}^T y_{n,t}^s \leq x_t^s, \forall s, t, \quad (3.19)$$

$$0 \leq n - t < N : z_t^s = x_t^s - \sum_{n=1}^T y_{n,t}^s, \forall s, t, \quad (3.20)$$

$$0 \leq n - t < N : \sum_{t=1}^t y_{n,t}^s \leq d_n^s, \forall s, n, t, \quad (3.21)$$

$$0 \leq n - t < N : y_{n,t}^s \geq 0, \forall s, t, n, \quad (3.22)$$

$$n - t \geq N : y_{n,t}^s = 0, \forall s, t, n, \quad (3.23)$$

$$x_t^s \geq 0, \forall s, t, \quad (3.24)$$

$$z_t^s \geq 0, \forall s, t, \quad (3.25)$$

kde  $\lambda < 1$ .

### 3.3.2 Rozšírenie pre L lekární a I liekov

Úlohu (3.16), môžeme ďalej rozšíriť pre  $L$  lekární a  $I$  liekov. Pri zovšeobecnení úlohy je však potrebné upraviť obmedzenie (3.21), ktoré zabezpečuje, že množstvo balenia lieku  $i$ , ktoré sa predá je menšie alebo rovné, ako dopyt po lieku  $i$  v lekárni  $l$  v scenári  $s$ . Ako už bolo vysvetlené pri formulácii dvojstupňovej úlohy v modeloch sa jedná o rôzne veľkosti balení toho istého lieku a teda predpokladáme, že v prípade ak nebude dostupné balenie, ktoré požaduje zákazník tak s pravdepodobnosťou  $q$  si zakúpi inú veľkosť balenia tohto lieku. Ak by nastala situácia, že zákazník si chce zakúpiť iba konkrétnu veľkosť balenia tak opäť s určitou pravdepodobnosťou  $r$  si pôjde toto balenie kúpiť do inej lekárne. Opäť platia vzťahy 3.8 a 3.9.

Model formulujeme nasledovne

$$\max_{x^1, y^1, \dots, x^S, y^S} \sum_{s=1}^S \sum_{i=1}^I \sum_{l=1}^L \sum_{t=1}^T p^s (-c_{i,l} x_{i,l,t}^s + \sum_{n=1}^T v_{i,l} y_{i,l,n,t}^s + \lambda c_{i,l} z_{i,l,t}^s) \quad (3.26)$$

$$\text{s.t. } 0 \leq n - t < N : \sum_{n=1}^T y_{i,l,n,t}^s \leq x_{i,l,t}^s, \forall i, l, t, s, \quad (3.27)$$

$$0 \leq n - t < N : z_{i,l,t}^s = x_{i,l,t}^s - \sum_{n=1}^T y_{i,l,n,t}^s \forall i, l, s, t, \quad (3.28)$$

$$0 \leq n - t < N : \sum_{t=1}^t y_{i,l,n,t}^s \leq \sum_{j=1}^I \sum_{k=1}^L d_{i,l}^s q_k r_j, \forall i, l, s, t, n, \quad (3.29)$$

$$x_{i,l,t}^s \geq 0, \forall i, l, t, \quad (3.30)$$

$$0 \leq n - t < N : y_{i,l,n,t}^s \geq 0, \forall i, l, s, t, n, \quad (3.31)$$

$$n - t \geq N : y_{i,l,n,t}^s = 0, \forall i, l, s, t, n, \quad (3.32)$$

$$z_{i,l,t}^s \geq 0, \forall i, l, s, t, \quad (3.33)$$

kde  $\lambda < 1$ .

Je potrebné dodať, že pre každý typ balenia v tomto prípade budeme uvažovať rôzne ceny, za ktoré sieť lekární nakupuje a predáva tieto lieky.

### 3.3.3 Model pre L lekární a I liekov + pravdepodobnostné obmedzenie

V sekcii 1.1.2 sme si predstavili pravdepodobnostné obmedzenia v úlohách stochastické programovania. Tieto poznatky v tejto časti aplikujeme na vytvorený model a doplníme požadované pravdepodobnostné obmedzenia.

Ako prvé sa pozrieme na pravdepodobnostné obmedzenie, ktorý zručí, že pravdepodobnosť, že konkrétna lekárňa do ktorej príde zákazník bude mať dostatočné zásoby aspoň jedného typu balenia lieku, ktorý si zákazník praje, bude rovná alebo väčšia ako je dopredu zadaná hodnota  $\alpha_1$ . Toto obmedzenie zapíšeme pomocou dvoch dodatočných obmedzení.

Prvé obmedzenie zabezpečuje, že celkový dopyt pre každú lekárňu  $i$ , v stupni  $t$  a scenári  $s$  je menší alebo rovný celkovému množstvu zakúpených liekov. Zavedieme si novú binárnu premennú  $\omega_{i,t}^s$ , ktorá bude nadobúdať dve hodnoty a to 0 a 1. Ak  $\omega_{i,t}^s = 1$  znamená to, že nerovnosť nebola splnená a teda je potrebné

navýšiť počet kúpených liekov v lekárni  $i$  a to tak, že pripočítame dostatočne veľkú konštantu  $K$ . Ak  $\omega_{i,t}^s = 0$ , znamená to, že nerovnosť bola splnená a teda nie sú potrebné ďalšie kroky.

$$\begin{aligned} \sum_{l=1}^L d_{i,l,t}^s &\leq \sum_{l=1}^L x_{i,l,t}^s + K\omega_{i,t}^s, \quad \forall i,t,s \\ \omega_{i,t}^s &\in [0,1] \end{aligned} \quad (3.34)$$

Druhé obmedzenie zaručuje, že pravdepodobnosť, že lekárň  $i$  v čase  $t$  nebude mať k dispozícii žiaden druh lieku, je menšia alebo rovná ako  $1 - \alpha_1$ , pričom budeme uvažovať veľké hodnoty parametru  $\alpha_1$ .

$$\sum_{s=1}^S p^s \omega_{i,t}^s \leq 1 - \alpha_1, \quad \forall i,t \quad (3.35)$$

Toto obmedzenie môžeme ďalej rozšíriť a to tak, že budeme požadovať aby v lekárni  $i$  bol s určitou pravdepodobnosťou  $\alpha_2$  dostupný konkrétny typ lieku. Teda konkrétne, uvažujme viacstupňový stochastický model, kde chceme zabezpečiť, aby dopyt po konkrétnom type lieku  $l$  v každej lekárni  $i$  a stupni  $t$  bol menší alebo rovný objednaným liekom  $x_{i,l,t}^s$ . Opäť zavedieme binárnu premennú plus  $\phi_{i,l,t}^s$  a dostatočne veľkú konštantu  $K$ .

Obmedzenie možno zapísať ako

$$\begin{aligned} d_{i,l,t}^s &\leq x_{i,l,t}^s + K\phi_{i,l,t}^s, \quad \forall i,l,t,s \\ \phi_{i,l,t}^s &\in [0,1] \end{aligned} \quad (3.36)$$

Aby sme sa uistili, že obmedzenie je splnené s určitou pravdepodobnosťou  $\alpha_2$ , zavedieme ďalšie obmedzenie:

$$\sum_{s=1}^S p^s \phi_{i,l,t}^s \leq 1 - \alpha_2, \quad \forall i,l,t \quad (3.37)$$

$\phi_{i,l,t}^s$  je teda binárna premenná označujúca či je obmedzenie aktívne pre lekárň  $i$ , typ lieku  $l$ , stupeň  $t$  a scenár  $s$  a  $p^s$  je pravdepodobnosť scenára  $s$ .

Tento súbor obmedzení zabezpečuje, že dopyt po konkrétnom type lieku  $l$  v každej lekárni a časovom období je uspokojený s určitou pravdepodobnosťou  $\alpha_2$ , čo z neho robí náhodné obmedzenie pre konkrétny typ lieku.

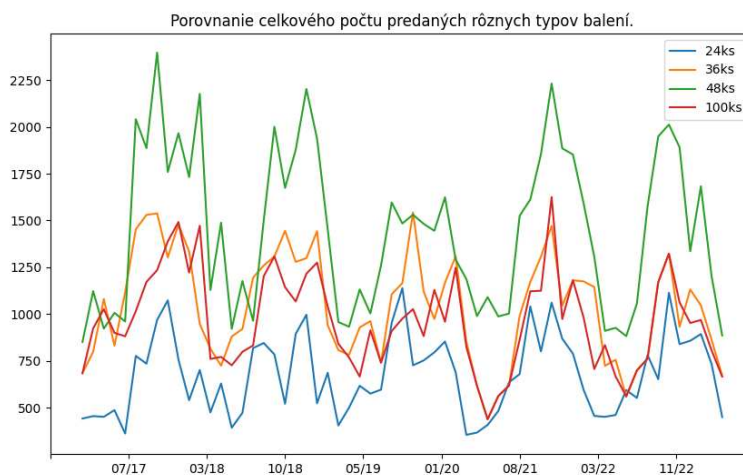


# 4. Praktická časť

## 4.1 Vstupné dáta

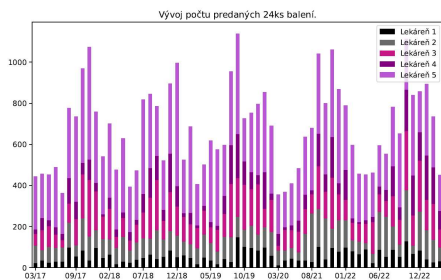
Pri riešení úloh vychádzame z historických dát, ktoré boli poskytnuté spoločnosťou IQVIA. Keďže mesto Praha je kategorizované ako veľkomesto a súčasne je to aj hlavné mesto Českej republiky je pochopiteľné, že sa tu nachádza niekoľko rôznych komerčných sietí lekární ako aj veľké množstvo nezávislých lekární. Pre účely tejto práce sme vybrali jednu nemenovanú komerčnú sieť lekární. Aj napriek tomu, že sme zúžili okruh lekární na jednu konkrétnu sieť tak počet lekární je stále veľký a teda je ešte potrebné bližšie špecifikovať mestské časti Prahy, na ktoré sa zameriame. Vzhľadom na konzistentnosť dát boli vybrané mestské časti Praha 1 a 2, kde sa nachádza 5 pobočiek. Ako konkrétny liek sme si pre účel tejto práce zvolili populárne analgetikum. Tento liek je možné zakúpiť v baleniach s rôznym počtom tabliet a to 24, 36, 48 a 100 kusov.

Dáta, ktoré máme k dispozícii sú za obdobie od marca 2017 až po marec 2023 a zaznamenávajú počet predaných kusov balení v každej z piatich lekární za mesiac. Po preskúmaní dát sme avšak zistili, že nie je možné použiť kompletnú sadu dát ale je potrebné odstrániť dáta od apríla 2020 až po marec 2021 a to z dôvodu výpadku dát. Ako si iste všetci spomíname, tak v danom období prebiehala pandémia COVID-19, ktorá bohužiaľ spôsobila výpadok dát. Tento výpadok je teda pravdepodobne spojený s dočasnými obmedzeniami, ktoré zasiahli aj lekárne. Na obrázku nižšie môžeme vidieť celkový počet predaných 4 rôznych balení počas spomínanej doby - marec 2017 až marec 2023 (bez odstráneného obdobia).

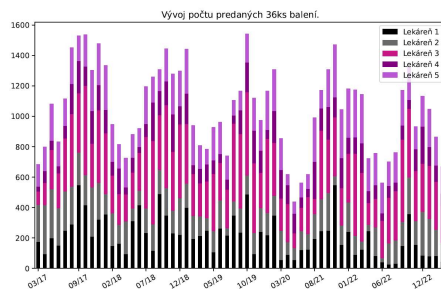


Obr. 4.1: Scenárový strom s očíslovaným uzlov.

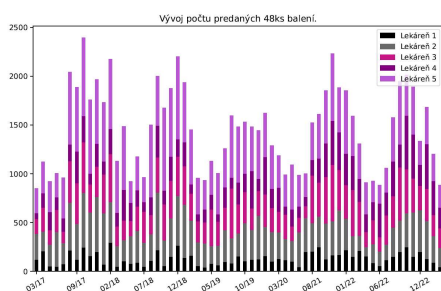
Ako môžeme vidieť na 4.1 tak najpredávanejším balením počas celého časového horizontu je 48 kusové balenie a 24 kusové balenie je najmenej predávané. Ďalej môžeme pozorovať, že sa v dátach vyskytuje sezónnosť. Pre lepšie zobrazenie a overenie sezónnosti sa pozrieme na jednotlivé lekárne a počty predaných typov balení.



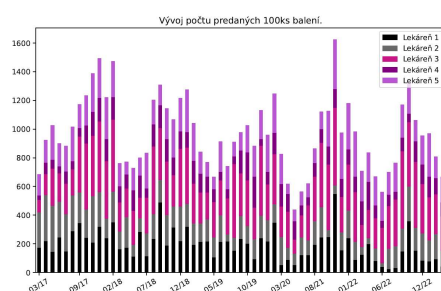
(a) Mesačný vývoj počtu predaných 24ks balení.



(b) Mesačný vývoj počtu predaných 36ks balení.



(c) Mesačný vývoj počtu predaných 48ks balení.



(d) Mesačný vývoj počtu predaných 100ks balení.

Obr. 4.2: Počet predaných balení v jednotlivých lekárňach.

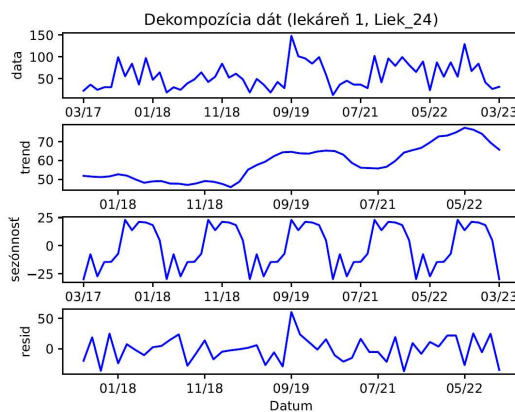
Ako vidíme tak výskyt sezónnosti sa nám potvrdil a sezónna zložka sa vyskytuje v každej z piatich lekárni. Vzhľadom na to, že na základe týchto dát budeme ďalej generovať scenáre pre modely je potrebné odstrániť *sezónnu zložku*.

## 4.2 Generovanie scenárových stromov

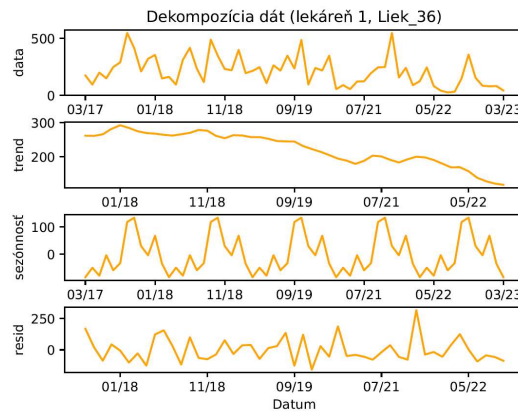
### 4.2.1 Odstránenie sezónnosti z dát

Pri prvotnej analýze historických dát v 4.1 sme zmienili podozrenie, že dáta obsahujú sezónnu zložku. Poďme sa teda bližšie pozrieť na jednotlivé zložky dát. Vzhľadom na to, že pracujeme so 4 rôznymi baleniami a 5 lekárňami tak si uvedieme len niekoľko vybraných dekompozícií na ukážku.

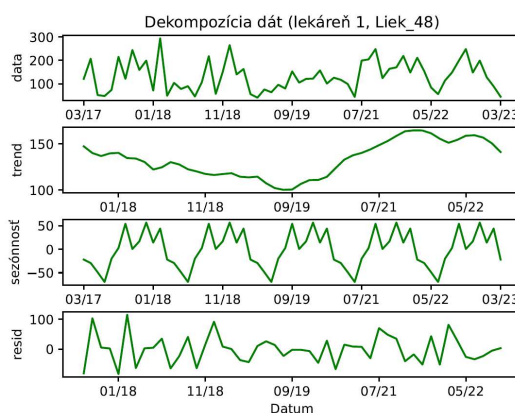
Ako môžeme vidieť na 4.3, dáta naozaj obsahujú sezónnu zložku s periódou opakovania 12 mesiacov. Ako je zrejme z rozkladu dát tak sezónny nárast nastáva koncom leta v mesiacoch august prípadne september a pretrváva cez celé zimné obdobie až do mesiaca január prípadne v niektorých prípadoch do mesiaca február. Pre presnejšie prevedenie predikcie a nasledovné generovanie scenárov je potrebné odstrániť tento sezónny výkyv. Odstránenie sezónnej zložky prevedieme za pomoci Holt-Wintersovu metódy predstavenej v časti 2.4.2. Na výpočtovú časť použijeme software Python 3.10, kde použijeme balíček ETSTModel.



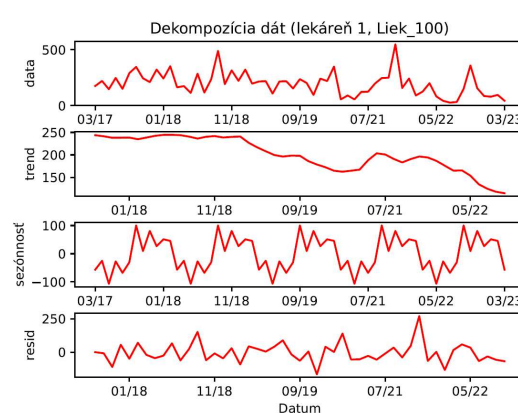
(a) Dekompozícia dát pre 24ks balenie v lekárni 1.



(b) Dekompozícia dát pre 36ks balenie v lekárni 1.



(c) Dekompozícia dát pre 48ks balenie v lekárni 1.

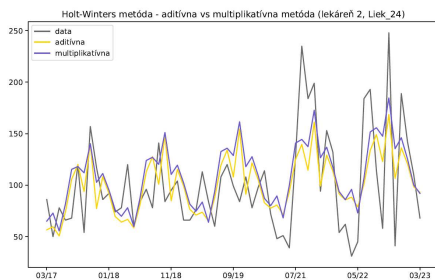


(d) Dekompozícia dát pre 100ks balenie v lekárni 1.

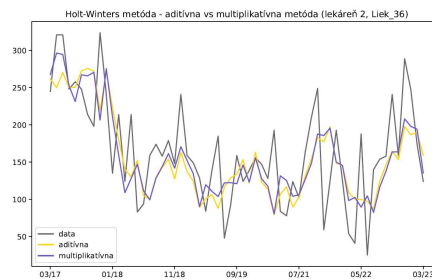
Obr. 4.3: Dekompozícia dát na trendovú a sezónnu zložku.

V časti 2.4.2 sme si predstavili dve varianty Holt-Wintersovej metódy, ktoré sa líšia na základe toho či uvažujeme aditívnu alebo multiplikatívnu dekompozíciu. Tieto dve varianty sa z praktického hľadiska líšia v interpretácii sezónneho faktoru  $I_s$ .

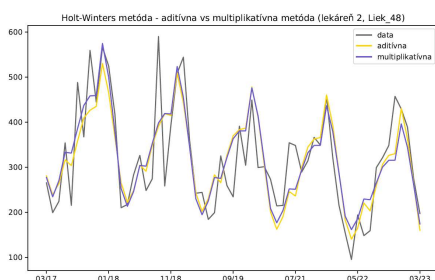
- aditívna dekompozícia -  $I_t$  sa meria v rovnakých jednotkách ako príslušná časová rada  $y_t$ , čo znamená, že ak dvanásť sezónny faktor  $I_{12}$  predaja maloobchodu je o veľkosti 50000 CZK, tak to znamená, že pre dvanásť obdobie sa sezónnosť prejavuje nárastom predaja o 50000 CZK nad celoročným priemerom.
- multiplikatívna dekompozícia -  $I_t$  je v tomto prípade bez rozmerná veličina, teda v prípade, že dvanásť sezónny faktor  $I_{12}$  predaja maloobchodu je 0.8 predaja maloobchodu tak sa sezónnosť prejavuje v dvanástom období poklesom o 20% pod celoročný priemer.



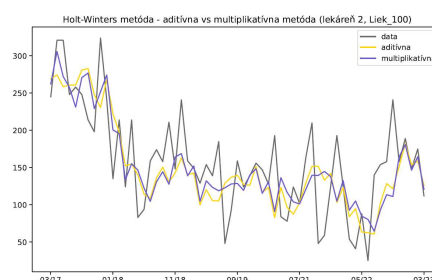
(a) Holt-Wintersová metóda pre 24ks balenie v lekárni 2.



(b) Holt-Wintersová metóda pre 36ks balenie v lekárni 2.



(c) Holt-Wintersová metóda pre 48ks balenie v lekárni 2.



(d) Holt-Wintersová metóda pre 100ks balenie v lekárni 2.

Obr. 4.4: Holt-Wintersová metóda - aditívna vs multiplikatívna varianta.

Na obrázkoch 4.4 môžeme vidieť ukážku prevedenia Holt-Wintersovej aditívnej ako aj multiplikačnej metódy. Na základe porovnania výsledkov korelačných koeficientov a strednej kvadratickej chyby sme sa rozhodli pokračovať s aditívnou verziou modelu. Jednotlivé hodnoty a štatistické testy môžeme nájsť ako výstup kódu.

#### 4.2.2 Intervalová predikcia a generovanie trajektórií

Za pomoci Holt-Wintersovej metódy vytvoríme bodovú predikciu pre každý pár lekáreň:liek a následne zostrojíme intervalovú predikciu. Budeme uvažovať predikcie na 6 mesiacov dopredu a teda od apríla 2023 až do septembra 2023. Na generovanie trajektórií použijeme algoritmus popísaný v sekcii 2.4.3. Teda každý pár lekáreň:liek obsahuje bodovú ako aj intervalovú predikciu na 6 mesiacov dopredu. Táto prvotná bodová predikcia je zároveň našou prvou vygenerovanou trajektóriou. Pre vygenerovanie druhej trajektórie si zoberieme prvú bodovú predikciu a náhodne k nej pripočítame alebo odpočítame percentuálnu časť predikčného intervalu. Označíme ako  $\epsilon$ . Získali sme nový bod - začiatok druhej trajektórie. Pre tento bod si vytvoríme novú bodovú predikciu do kroku  $t + 2$  - na základe historických dát očistených od sezónnosti a tohto novo vzniknutého odhadu. Opäť dopočítame aj predikčný interval. Pokračujeme analogicky až kým nemáme celú trajektóriu. Následne zopakujeme pre každý par lekáreň:liek avšak hodnota  $\epsilon$  už nebude určená náhodne ale v každom kroku bude presne taká istá ako bola pre prvý pár lekáreň:liek.

### 4.2.3 Tvorba scenárových stromov

Pre samotnú tvorbu scenárových stromov je dôležité si ako prvé určiť s akou štruktúrou scenárových stromov budeme pracovať. Budeme požadovať aby scenárový strom mal 6 stupňov, ktoré budú postupne reprezentovať počet predaných kusov jednotlivých liekov. Je dôležité poznamenať, že pre každú lekárňu a daný typ lieku budeme scenárový strom generovať zvlášť a teda budeme mať 20 scenárových stromov. Ďalej, si vopred stanovíme počet nasledovníkov v každom stupni. Celkový počet scenárov je teda 2880.

Stupeň	Počet uzlov
1	8
2	5
3	4
4	3
5	3
6	2

Následne už môžeme použiť k-means algoritmus na vygenerované scenárové vejáre. Počet trajektórií v týchto vejároch (8000) sa môže zdať priveľký avšak je potrebné mať dostatočné množstvo trajektórií pri použití klastrovacej metódy k-means. Veľký počet trajektórií nám zabezpečí aj lepšie zachytenie náhodnosti. Po aplikovaní k-means algoritmu budeme mať 2880 scenárov v každom scenárovom strome. V popise metódy 2.4.3 uvádzame, že algoritmus je postavený tak aby zachoval jednotlivé korelácie medzi lekárňami ako aj medzi jednotlivými liekmi v rámci jednej lekárne. Korelačné matice môžeme nájsť v prílohe v sekciiach A.1 a A.2.

## 4.3 Vstupy do modelov

### Stanovenie predajnej a kúpnej ceny

V tejto práci budeme pracovať so štyrmi rôznymi baleniami toho istého lieku. Konkrétne budeme uvažovať 24, 36, 48 a 100 kusové balenie. Všeobecne vieme, že ceny produktov sú premenlivé a zvyčajne nie sú pevne stanovené. Existujú rôzne vplyvy na ceny produktov ako je napríklad sezónnosť, zľavy plynúce s rôznych vernostných programov, akcie na daný druh produktu a mnoho iného. Z toho plynie, že ceny produktov sa naprieč dlhším obdobím môžu výrazne líšiť. To isté platí aj pre uvažovaný produkt avšak keďže sa jedná o široko dostupný liek, kolísanie cien je v tomto prípade obmedzené. Z tohto dôvodu stanovíme cenu, za ktorú budú lekárne predávať všetky typy balení ako odhadovanú priemernú cenu. Ako bolo definované v sekcii 3.2 túto cenu budeme označovať ako  $v_l$  pričom  $l = 1,2,3,4$ . Teda v prípade 24 kusového balenia lieku  $v_1 = 69$ . Cena 36 kusového balenia je stanovená na  $v_2 = 105$ . Analogicky  $v_3 = 109$  a  $v_4 = 145$ . Je potrebné poznamenať, že všetky uvažované ceny sú uvedené v Českej korune (CZK).

Ďalej sa pozrieme na cenu, za ktorú lekárne nakupujú lieky. Podľa dostupných informácií všetky lieky putujú od výrobcu k dodávateľom a až následne sú prepravené do predajní. Kúpna cena je teda zložená z výrobných nákladov na daný

liek, prepravy k dodávateľom, následnému rozdistribúvanú do lekárni a samozrejme je potrebné uvažovať aj ďalšie čiastkové náklady. Na základe dostupných informácií, ktoré boli na účely tejto práce poskytnuté spoločnosťou IQVIA stanovíme ceny  $c_l$  kde  $l = 1, 2, 3, 4$  nasledovne. V prípade 24 kusového balenia lieku  $c_1 = 30$ . Analogicky  $c_2 = 38$ ,  $c_3 = 49$  a  $c_4 = 80$ .

### Pravdepodobnosť medzi lekárňami a typmi balení

Dáta, ktoré máme k dispozícii neobsahujú dostatočne veľa informácií aby bolo možné vypočítať jednotlivé pravdepodobnosti medzi lekárňami a typmi balení uvedené v obmedzení 3.29. Z toho dôvodu budeme pracovať s pravdepodobnosťami  $q_k, k = 1, \dots, K$  a  $r_j, j = 1, \dots, J$  ako s parametrami.

Parameter  $q_k$  označuje pravdepodobnosť, že v prípade nedostupnosti určitého balenia požadovaného zákazníkom si zákazník namiesto toho kúpi iné balenie v tej istej lekárni. Predpokladáme, že všetky pravdepodobnosti  $q_k$  sú rovnaké a nadobúdajú hodnotu 0.5 (50%). Rovnako budeme uvažovať pri parametroch  $r_j$ , ktoré označujú pravdepodobnosť, že v situácii, kedy nebude dostupné požadované balenie pôjde zákazník do inej lekárne. Opäť predpokladáme, že tieto pravdepodobnosti sú rovnaké a teda  $r = 0.25$  (25%).

## 4.4 Výsledky modelov

Ako už bolo zmienené vyššie budeme pracovať so 4 rôznymi baleniami toho istého lieku, ktorý sa predáva v 5 lekárňach. Uvažované obdobie, na ktoré sa budeme pozerať je 6 mesiacov dopredu, teda  $T = 6$ . Počet mesiacov  $N$ , určuje trvanlivosť liekov a stanovili sme  $N = 4$ . Po vygenerovaní trajektórií a zostrojení scenárových stromov sme dostali pre každý pár lekáreň:liek 2880 scenárov. Pravdepodobnosť jedného scenáru budeme uvažovať  $1/2880$ . Hodnotu konštanty  $\lambda$  stanovíme na 0.3.

### 4.4.1 T-stupňová úloha - základný model

Základný model tejto práce predstavuje 6-stupňovú úlohu, ktorého všeobecnú formuláciu môžeme nájsť v časti 3.3.2. Pre konkrétne parametre úlohu formulujeme ako

$$\max_{\mathbf{x}^1, \mathbf{y}^1, \dots, \mathbf{x}^s, \mathbf{y}^s} \sum_{s=1}^{2880} \sum_{i=1}^5 \sum_{l=1}^4 \sum_{t=1}^6 p^s (-c_{i,l} x_{i,l,t}^s + v_{i,l} \sum_{n=1}^6 y_{i,l,n,t}^s + 0.3c_{i,l} z_{i,l,t}^s) \quad (4.1)$$

$$\text{s.t. } 0 \leq n - t < 4 : \sum_{n=1}^6 y_{i,l,n,t}^s \leq x_{i,l,t}, \forall i, l, t, s, \quad (4.2)$$

$$0 \leq n - t < 4 : z_{i,l,t}^s = x_{i,l,t}^s - \sum_{n=1}^6 y_{i,l,n,t}^s \forall i, l, s, t, \quad (4.3)$$

$$0 \leq n - t < 4 : \sum_{t=1}^t y_{i,l,n,t}^s \leq \sum_{j=1}^4 \sum_{k=1}^5 d_{i,l,t}^s q_k r_j, \forall i, l, s, t, n, \quad (4.4)$$

$$0 \leq n - t < 4 : y_{i,l,n,t}^s \geq 0, \forall i, l, s, t, n, \quad (4.5)$$

$$n - t \geq 4 : y_{i,l,n,t}^s = 0, \forall i, l, s, t, n, \quad (4.6)$$

$$x_{i,l,t}^s \geq 0, \forall i, l, t, \quad (4.7)$$

$$z_{i,l,t}^s \geq 0, \forall i, l, s, t, \quad (4.8)$$

+ podmienky neanticipativity.

Vo viacstupňových stochastických úlohách zabezpečujú obmedzenia neanticipativity, že rozhodnutia prijaté v každom stupni sú založené iba na informáciách dostupných do tohto času, bez poznania budúcich náhodných výsledkov. Inými slovami, tieto obmedzenia zabraňujú tomu, aby sa pri rozhodovaní využívali budúce informácie, ktoré ešte neboli odhalené. V danom optimalizačnom probléme máme rozhodovacie premenné  $\mathbf{x}^s$  a  $\mathbf{y}^s$  pre každý scenár  $s$ . Podmienky neanticipativity možno formulovať nasledovne

1. stupeň - požadujeme, aby pre všetky scenáre rozhodnutia v prvom stupni boli totožné a teda je potrebné pridať obmedzenie v tvare

$$x_{i,l,1}^s = x_{i,l,1}^{s+1}, \quad 1 \leq s \leq 2880, \forall i, l, \quad (4.9)$$

2. - 6. stupeň - požadujeme aby v prípade každej dvojice scenárov  $s$  a  $s'$  platilo, že ak hodnota  $\xi_{[t]}^s$  je rovná  $\xi_{[t]}^{s'}$  tak následne aj rozhodnutia v stupni  $t$  sú totožné  $x_{i,l,t}^s = x_{i,l,t}^{s'}$ . Z tohto dôvodu je potrebné pridať obecné obmedzenie neanticipativity, ktoré má tvar

$$x_{i,l,t}^s = x_{i,l,t}^{s'}, \quad t = 2, \dots, 6, \forall i, l, \quad (4.10)$$

pre každé  $s$  a  $s'$  také, že platí

$$\xi_{[t]}^s = \xi_{[t]}^{s'}, \quad t = 2, \dots, 6 \forall i, l, \quad (4.11)$$

kde  $s$  a  $s'$  sú dva rôzne scenáre. Ďalej sa pozrieme na rozhodovaciu premennú  $\mathbf{y}^s$ . Aj na tieto premenné sa vzťahuje podmienka, že nesmú závisieť na budúcich realizáciách. Formulujeme analogicky

$$y_{i,l,n,1}^s = y_{i,l,n,1}^{s+1}, \quad 1 \leq s \leq 2880, \forall i, l, \quad (4.12)$$

$$y_{i,l,n,t}^s = y_{i,l,n,t}^{s'}, \quad t = 2, \dots, 6, \forall i, l, \quad (4.13)$$

pre každé  $s$  a  $s'$  také, že platí

$$\xi_{[t]}^s = \xi_{[t]}^{s'}, \quad t = 2, \dots, 6 \forall i, l. \quad (4.14)$$

Čo sa týka premennej  $z^s$ , ktorá je počítaná na základe rozhodovacích premenných  $x^s$  a  $y^s$ , nemusíme pridávať ďalšie neanticipatívne obmedzenia. Je to preto, že obmedzenia neanticipativity pre  $x^s$  a  $y^s$  už zaisťujú, že rozhodnutia prijaté v každom stupni sú založené iba na informáciách dostupných dovtedy. Preto aj výpočet  $z^s$  z  $x^s$  a  $y^s$  vychádza len z dostupných informácií a neporušuje neanticipativnosť.

## Riešenie úlohy

?? V tejto časti si uvedieme výsledky riešenia rozsiahleho modelu lineárneho stochastického programovania pomocou Gurobi Optimizer 10.0.1 na CPU AMD Ryzen 5 5500U s 12 logickými procesormi, ktorý bol použitý za pomoci softwaru Python 3.10 a knižnice *gurobipy*. Model má 4 492 800 obmedzení, 2 419 320 premenných a 20 736 000 nenulových hodnôt. Vzhľadom na veľkosť problému je potrebné predbežné spracovanie celého modelu s podmienkami pred samotným riešením problému. Hlavným cieľom je zjednodušiť a zmenšiť veľkosť pôvodného problému, čím sa zjednoduší a zrýchli jeho vyriešenie. To sa dosiahne aplikáciou rôznych techník a transformácií na identifikáciu a odstránenie nadbytočných alebo irelevantných obmedzení, premenných a iných problémových komponentov. Zahŕňa to napríklad:

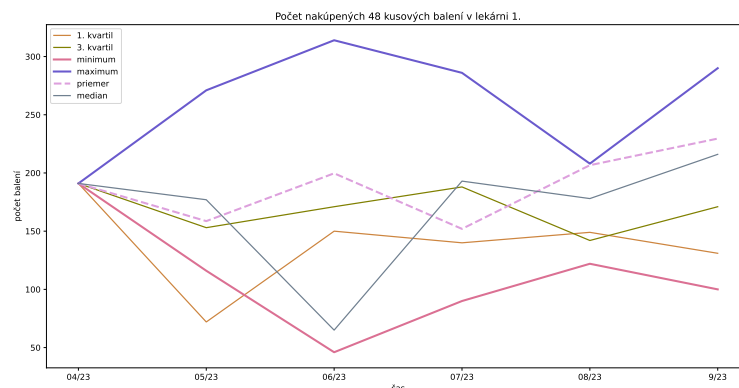
- Odstránenie nadbytočných obmedzení - ak je obmedzenie lineárnou kombináciou iných obmedzení alebo neovplyvňuje realizovateľnú oblasť, možno ho odstrániť bez zmeny riešenia problému,
- Odstránenie pevných premenných - premenné, ktoré majú rovnakú hodnotu v každom realizovateľnom riešení, môžu byť odstránené a nahradené ich konštantnými hodnotami v obmedzeniach a cieľovej funkcii,
- Sprísnenie hraníc - analýzou obmedzení môže riešiteľ niekedy určiť užšie hranice premenných, čo môže pomôcť zlepšiť výkon algoritmov riešenia
- Agregovanie obmedzení - ak dve alebo viac obmedzení zdieľajú rovnaké premenné s podobnými koeficientmi, môžu sa niekedy spojiť do jedného obmedzenia, čím sa zníži počet obmedzení v probléme,
- Detekcia neuskutočniteľnosti alebo neohraničenosti - v niektorých prípadoch môže proces predbežného riešenia určiť, že problém je neuskutočniteľný (neexistujú žiadne realizovateľné riešenia) alebo neohraničený (cieľová funkcia môže byť ľubovoľne veľká alebo malá), takže nie je potrebné pokračovať s hlavným algoritmom riešenia,



- Škálovanie - riešiteľ môže zmeniť mierku koeficientov problému, čím sú číselné hodnoty jednotnejšie, čo môže zlepšiť stabilitu a výkon algoritmov riešenia,

Účinnosť predbežného spracovania sa môže líšiť v závislosti od štruktúry a charakteristík problému. V niektorých prípadoch môže predriešenie výrazne znížiť veľkosť problému, čo vedie k podstatnému zlepšeniu výkonu, zatiaľ čo v iných prípadoch môže byť vplyv minimálny. Vo všeobecnosti sa však považuje za nevyhnutný krok v moderných optimalizačných riešeniach na zvýšenie celkovej efektívnosti a zníženie robustnosti procesu riešenia. Počas fáz spracovania a predbežného riešenia optimalizátor odstránil 3 801 600 riadkov a 1 728 000 stĺpcov, výsledkom čoho bol zmenšený model, ktorý má 691 200 obmedzení a 691 320 premenných. Čas predbežného spracovania bol 6.99 sekundy. Následne bolo nájdené optimálne riešenie modelu za 10.43 sekundy.

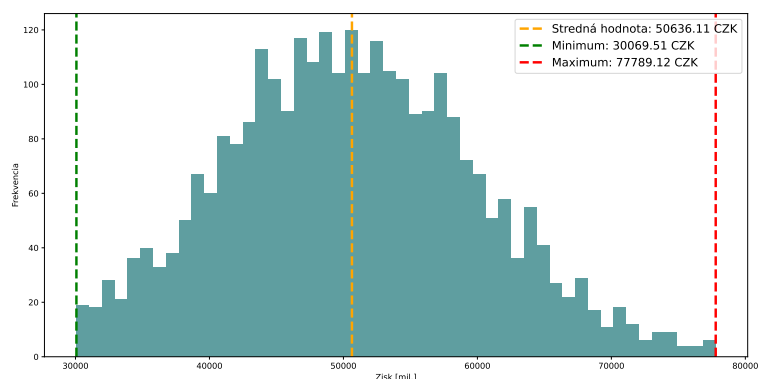
Na obrázku 4.5 môžeme vidieť ukážku počtu objednaných 48 kusových balení do lekárne 1. Vzhľadom na veľký počet možných scenárov a teda aj rôznych riešení nebudeme uvádzať všetky získané výsledky ale pozrieme sa na hlavné charakteristiky ako je priemerný počet objednaných balení v každom stupni  $t$ , medián a taktiež prvý a tretí kvantil. Ďalej môžeme ešte vidieť trajektóriu, s najväčším a s najmenším počtom objednaných balení.



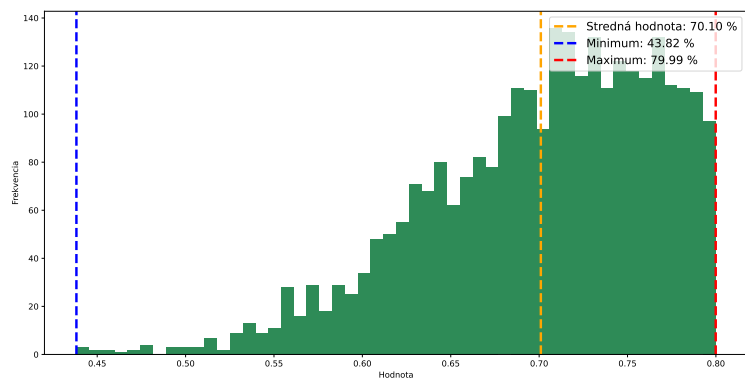
Obr. 4.5: Počet objednaných 48 kusových balení v lekárni 1.

Čo sa týka jednotlivých ziskov lekárne tak pre prehľadnosť sa pozrieme v prípade 48 kusového balenia na rozdelenie ziskov. Môžeme vidieť taktiež zobrazenú strednú hodnotu ako aj minimálny a maximálny zisk, ktorý sme po vypočítaní modelu získali. Na obrázku 4.6 môžeme vidieť vyznačenú strednú hodnotu, ktorá predstavuje **50 636.11 CZK**. Ako môžeme ďalej vidieť tak minimálny zisk je 30 069.51 CZK a maximálny predstavuje 77 789.12 CZK.

Vzhľadom na jednoduchšiu interpretáciu a predstavu môžeme zobrať postupne jednotlivé zisky pre každý scenár a vydeliť ich príslušnými nákladmi pre daný scenár, čím získame percentuálne vyjadrenie zisku voči nákladom. Na obrázku 4.7 môžeme vidieť rozdelenie príslušných dopočítaných hodnôt. V priemere pre lekárňu 1 a 48 kusové balenie predstavuje hodnota zisku 70.10% z nákladov. Môžeme vidieť, že rozdelenie je asymetrické a výrazne skosené zľava.

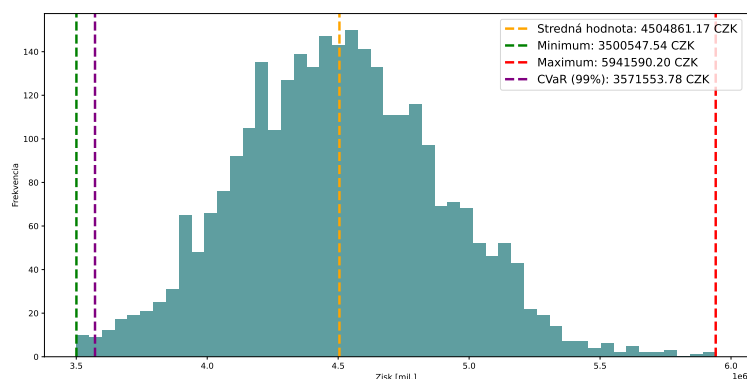


Obr. 4.6: Rozdelenie ziskov pre lekárň 1 - 48 ks balenie.



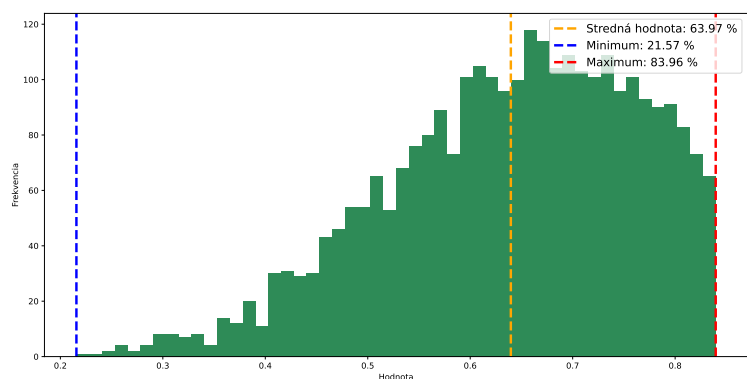
Obr. 4.7: Rozdelenie ziskov [%] pre lekárň 1 - 48 ks balenie.

Na záver sa bližšie pozrieme na rozdelenie zisku všetkých lekární a typov balení zobrazené pomocou histogramu, kde môžeme vidieť strednú hodnotu, minimálnu a maximálnu hodnotu. Ako môžeme vidieť na obrázku 4.8, stredná hodnota zisku je **4 504 861.17 CZK**. Dodatočne sa pozrieme na CVaR (Conditional Value at Risk), tiež známy ako očakávaný deficit. CVaR je miera rizika, ktorá zohľadňuje nielen očakávanú hodnotu rozdelenia, ale aj koncové riziko presahujúce určitú hranicu. V tomto prípade hranicu stanovíme na 0.99, čo znamená, že CVaR meria priemernú stratu, pri ktorej možno očakávať, že presiahne 99. percentil rozdelenia zisku. V tomto prípade môžeme teda povedať, že hodnota  $CVaR_{0.99}$  rovná 3 571 553.78 CZK predstavuje pre základný model očakávanú stratu presahujúcu 99. percentil rozdelenia zisku.



Obr. 4.8: Rozdelenie celkových ziskov.

Opäť sa pozrieme aj na pomer zisku voči nákladom. Na obrázku 4.9 môžeme vidieť rozdelenie príslušných dopočítaných hodnôt pre všetky lekárne a typy balení. V priemere predstavuje hodnota zisku približne 63.97% z nákladov.



Obr. 4.9: Rozdelenie ziskov [%].

#### 4.4.2 T-stupňová úloha - základný model + pravdepodobnostné obmedzenia

K základnému modelu pridáme pravdepodobnostné obmedzenia predstavené v časti 3.3.3. Pre pripomenutie nám prvé pravdepodobnostné obmedzenie zaručí, že pravdepodobnosť, že konkrétna lekáreň do ktorej príde zákazník bude mať dostatočné zásoby lieku, ktorý si zákazník praje, bude rovná alebo väčšia ako je dopredu zadaná hodnota  $\alpha_1$ . Toto obmedzenie môžeme ďalej rozšíriť a to tak, že budeme požadovať, aby v lekární  $i$  bol s určitou pravdepodobnosťou  $\alpha_2$  dostupný konkrétny typ lieku. Tieto pravdepodobnostné obmedzenia budeme aplikovať spolu v jednom modeli s tým, že budeme uvažovať rôzne kombinácie hodnôt parametrov  $\alpha_1$  a  $\alpha_2$ . Daný model s pridanými obmedzeniami vyzerá nasledovne

$$\max_{x^1, y^1, \dots, x^S, y^S} \sum_{s=1}^{2880} \sum_{i=1}^5 \sum_{l=1}^4 \sum_{t=1}^6 p^s (-c_{i,l} x_{i,l,t}^s + \sum_{n=1}^6 v_{i,l} y_{i,l,n,t}^s + 0.3c_{i,l} z_{i,l,t}^s) \quad (4.15)$$

$$\text{s.t. } 0 \leq n - t < 4 : \sum_{n=1}^6 y_{i,l,n,t}^s \leq x_{i,l,t}^s, \forall i, l, t, s \quad (4.16)$$

$$0 \leq n - t < 4 : z_{i,l,t}^s = x_{i,l,t}^s - \sum_{n=1}^6 y_{i,l,n,t}^s, \forall i, l, s, t \quad (4.17)$$

$$0 \leq n - t < 4 : \sum_{t=1}^t y_{i,l,n,t}^s \leq \sum_{j=1}^4 \sum_{k=1}^5 d_{i,l,t}^s q_k r_j, \forall i, l, s, t \quad (4.18)$$

$$\sum_{l=1}^4 d_{i,l,t}^s \leq \sum_{l=1}^4 x_{i,l,t}^s + K\omega_{i,t}^s, \quad \forall i, t, s \quad (4.19)$$

$$\sum_{s=1}^{2880} p^s \omega_{i,t}^s \leq 1 - \alpha_1, \quad \forall i, t, \quad (4.20)$$

$$d_{i,l,t}^s \leq x_{i,l,t}^s + K\phi_{i,l,t}^s, \quad \forall i, l, t, s \quad (4.21)$$

$$\sum_{s=1}^{2880} p^s \phi_{i,l,t}^s \leq 1 - \alpha_2, \quad \forall i, l, t, \quad (4.22)$$

$$x_{i,l,t}^s \geq 0, \forall i, l, t \quad (4.23)$$

$$0 \leq n - t < 4 : y_{i,l,n,t}^s \geq 0, \forall i, l, s, t, n \quad (4.24)$$

$$n - t \geq 4 : y_{i,l,n,t}^s = 0, \forall i, l, s, t, n \quad (4.25)$$

$$z_{i,l,t}^s \geq 0, \forall i, l, s, t, \quad (4.26)$$

$$\omega_{i,t}^s \in [0, 1], \forall i, t, s, \quad (4.27)$$

$$\phi_{i,l,t}^s \in [0, 1], \forall i, l, s, t, \quad (4.28)$$

+ podmienky neanticipativity.

## Riešenie úlohy

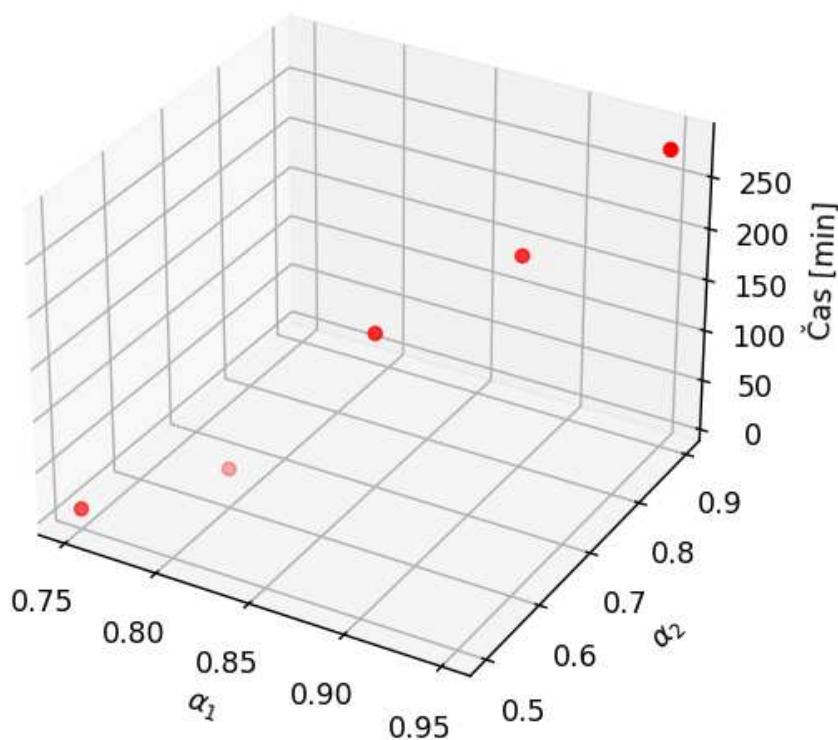
Tento model budeme riešiť pre rôzne kombinácie  $\alpha_1$  a  $\alpha_2$ , konkrétne

	varianta 1	varianta 2	varianta 3	varianta 4	varianta 5
$\alpha_1$	0.75	0.80	0.85	0.9	0.95
$\alpha_2$	0.5	0.6	0.7	0.8	0.9

Tabuľka 4.1: Kombinácie parametrov  $\alpha_1$  a  $\alpha_2$ .

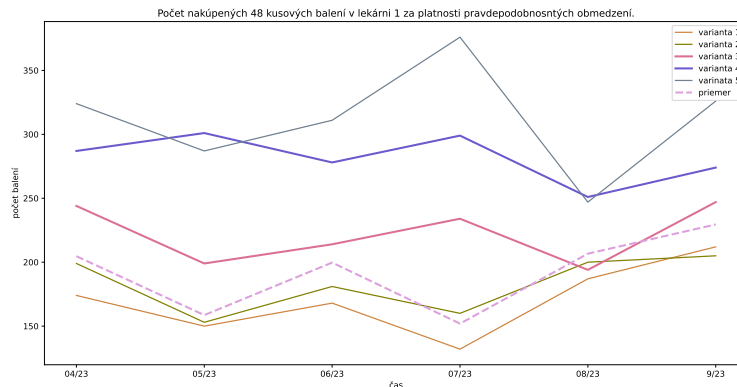
Pre všetky kombinácie platí, že model má 5 270 550 obmedzení, 2 851 320 premenných z čoho 432 000 je binárnych a 22 636 800 nenulových hodnôt. Počas fáz predspracovania a predbežného riešenia software odstránil 4 694 550 obmedzení a 2 275 320 premenných, výsledkom čoho bol zmenšený model, ktorý má 576 000 obmedzení a 576 000 premenných. Čas predbežného spracovania je pre rôzne kombinácie v rozmedzí 5.23 – 13.47 minút.

Na obrázku 4.10 môžeme vidieť zobrazený vzťah medzi časom výpočtu a hodnotami  $\alpha_1$  a  $\alpha_2$ . Ako vidíme tak čas výpočtu sa navyšuje s navyšovaním hodnôt  $\alpha_1$  a  $\alpha_2$ .



Obr. 4.10: Vzťah medzi časom výpočtu a hodnotami  $\alpha_1$  a  $\alpha_2$ .

Vzhľadom na rozsiahlosť modelu sa opäť podrobnejšie pozrieme na jednu vybranú lekárňu a daný typ balenia. Na obrázku 4.11 môžeme vidieť znázornený počet objednaných 48 kusových balení v lekární 1 pre všetkých 5 variant parametrov  $\alpha_1$  a  $\alpha_2$ . Jednotlivé trajektórie sú vypočítané ako priemerné hodnoty zo všetkých scenárov. Môžeme pozorovať, že čím väčšie hodnoty nadobúdajú parametre  $\alpha_1$  a  $\alpha_2$  tým väčší počet balení je v priemere lekáreň nútená objednať. Pre zaujímavosť si môžeme všimnúť trajektóriu, ktorá predstavuje strednú hodnotu počtu objednaných balení z hlavného modelu 4.1, kde neboli ešte zakomponované pravdepodobnostné obmedzenia. Pre nízke hodnoty  $\alpha_1$  a  $\alpha_2$  môžeme vidieť, že počet objednaných balení je nižší ako priemerný počet v modeli 4.1. Konkrétne ide o variantu 1 a 2. Pri variante 3 môžeme pozorovať mierny nárast oproti priemeru. Ako môžeme vidieť tak v prípade varianty 4 a 5 kedy parametre  $\alpha_1$  a  $\alpha_2$  nadobúdajú najväčšie hodnoty je nárast naozaj viditeľný.

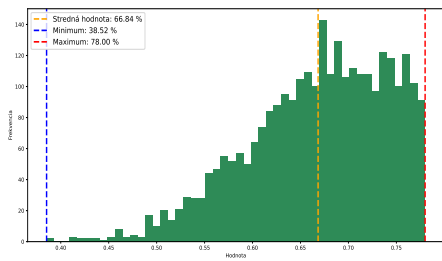


Obr. 4.11: Priemerný počet objednaných 48 kusových balení v lekárni 1 za použitia pravdepodobnostných obmedzení.

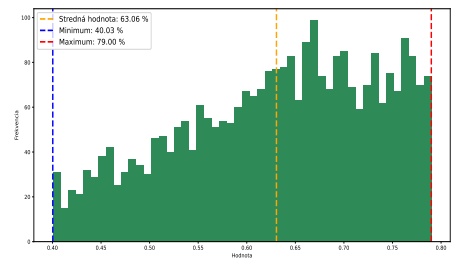
V tabuľke 4.2 môžeme vidieť priemerný zisk lekárne 1 naprieč všetkými scenármi v prípade 48 kusových balení pre všetkých 5 variant. V prvej variante kde  $\alpha_1 = 0.75$ ,  $\alpha_2 = 0.5$  je priemerný zisk približne rovnaký avšak nižší o 2248.21 CZK ako v prípade modelu bez pravdepodobnostných obmedzení kedy je priemerný zisk pre lekárne 1 liek 48 - 50 636.11 CZK. V druhej variante kedy  $\alpha_1 = 0.8$ ,  $\alpha_2 = 0.6$  je priemerný zisk nižší už o 4508.31 CZK opäť oproti základnému modelu. Keď si pozrieme percentuálne rozdelenie ziskov týchto variant na obrázku 4.12 a 4.13 tak vidíme, že v priemere zisk dosahuje 67% a 63%, čo sa pohybuje približne okolo hodnoty hlavného modelu bez pravdepodobnostných obmedzení - 64%. V prípade varianty 3 vidíme výrazný pokles priemerného zisku na 34 147.20 CZK a v priemere zisk za varianty 3 predstavuje už len 58% z nákladov. Ďalej v prípade štvrtej varianty pozorujeme výrazný pokles priemerného zisku na 29 229.00 CZK. Na obrázku 4.15 môžeme vidieť, že v priemere zisk predstavuje 46%, čo je výrazne menej ako v predchádzajúcich prípadoch. Ako vidíme vo variante 5 nastal najväčší pokles priemerného zisku. V porovnaní so základným modelom priemerný zisk klesol o 28 954.41 CZK. V priemere zisk predstavuje 49% čo je viac ako pri variante 4 avšak keď sa pozrieme na obrázok 4.16 tak môžeme vidieť, že rozdelenie je výrazne asymetrické a skosené doprava, pričom väčšina hodnôt spadá medzi 0.3 a 0.6.

	varianta 1	varianta 2	varianta 3	varianta 4	varianta 5
$\alpha_1$	0.75	0.80	0.85	0.9	0.95
$\alpha_2$	0.5	0.6	0.7	0.8	0.9
priemerný zisk [CZK]	48 387.90	46 127.80	34 147.20	29 229.00	21 681.70

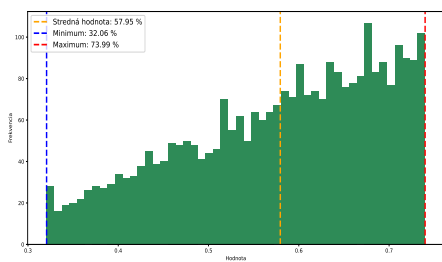
Tabuľka 4.2: Priemerný zisk lekárne 1 naprieč všetkými scenármi v prípade 48 kusových balení.



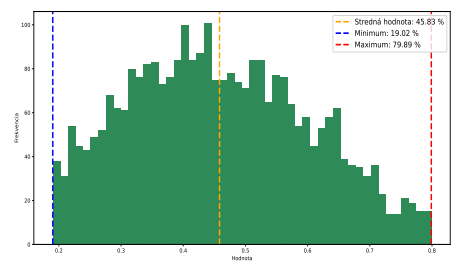
Obr. 4.12: Rozdelenie ziskov [%] - 48ks balenie, varianta 1.



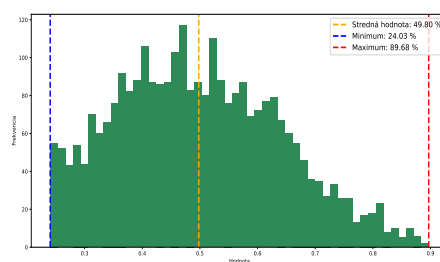
Obr. 4.13: Rozdelenie ziskov [%] - 48ks balenie, varianta 2.



Obr. 4.14: Rozdelenie ziskov [%] - 48ks balenie, varianta 3.

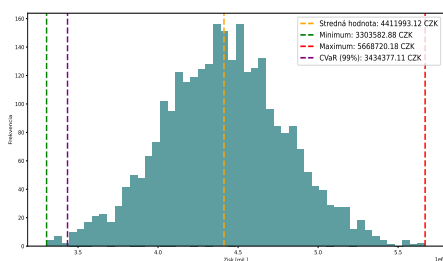


Obr. 4.15: Rozdelenie ziskov [%] - 48ks balenie, varianta 4.

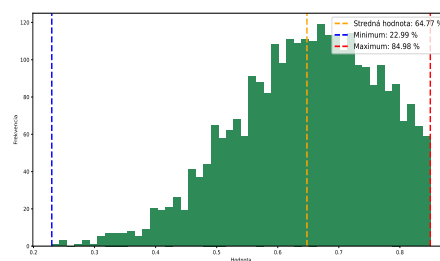


Obr. 4.16: Rozdelenie ziskov [%] - 48ks balenie, varianta 5.

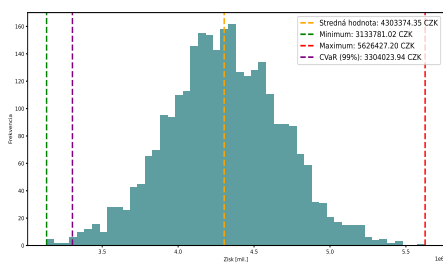
V nasledujúcej časti sa pozrieme na rozdelenia celkového zisku všetkých lekární a typov balení pre všetky varianty parametrov  $\alpha_1$  a  $\alpha_2$ . Pre lepšie porozumenie výsledkov sa zároveň pozrieme na rozdelenia absolútnych ziskov. Absolútny zisk vypočítame ako pomer nákladov a zisku. Ako môžeme vidieť, s meniacimi sa parametrami  $\alpha_1$  a  $\alpha_2$  sa menia aj rozdelenia a príslušné stredné hodnoty zisku. V porovnaní so **základným modelom** bez pravdepodobnostných obmedzení 4.1 kde stredná hodnota zisku predstavuje **4 504 861.147 CZK** je **varianta 1** najbližšie so strednou hodnotou rovnou **4 411 993.12 CZK**. Ako môžeme pozorovať tak už v prípade kedy  $\alpha_1 = 0.75$ ,  $\alpha_2 = 0.5$  je rozdiel v strednej hodnote ziskov 92 868.05 CZK. Čo sa týka hodnoty  $CVaR_{0,99}$  tak v tomto prípade je rovná 3 434 377.11 CZK. Keď sa pozrieme na rozdelenie absolútnych ziskov tak stredná hodnota je rovná 64.77%. Rozdelenie je asymetrické a skosené zľava. V prípade **varianty 2** je stredná hodnota zisku **4 303 374.38 CZK** opäť približne rovnaká ako v prípade základného modelu. Hodnota  $CVaR_{0,99}$  je rovná 3 304 023.94 CZK. V prípade rozdelenie absolútnych ziskov je stredná hodnota rovná 59.18%.



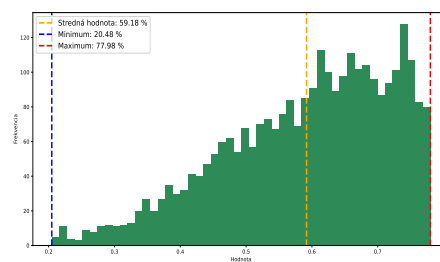
Obr. 4.17: Rozdelenie ziskov - varianta 1.



Obr. 4.18: Rozdelenie ziskov [%] - varianta 1.



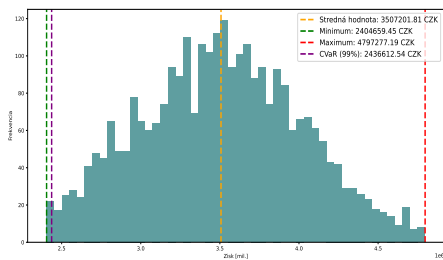
Obr. 4.19: Rozdelenie ziskov - varianta 2.



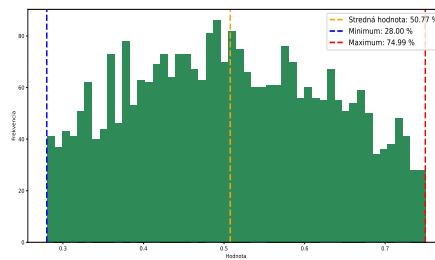
Obr. 4.20: Rozdelenie ziskov [%] - varianta 2.

V prípade **varianty 3** s  $\alpha_1 = 0.85$  a  $\alpha_2 = 0.7$  sa stredná hodnota zníži **3 507 201.81 CZK**. V porovnaní so základným modelom to predstavuje zníženie približne o 22.1%. V prípade **varianty 4**, kde  $\alpha_1 = 0.9$  a  $\alpha_2 = 0.8$ , stredná hodnota ďalej klesá na **2 718 197.15 CZK**. To predstavuje pokles o cca 39.6% v porovnaní so základným modelom a pokles o cca 22.5% v porovnaní s variantou 3. Čo sa týka hodnoty  $CVaR_{0,99}$  tak v prípade varianty 3 je rovná 2 436 612.54 CZK a pre variantu 4 je rovná 1 926 146.71 CZK. Zaujímavé sú aj absolútne hodnoty ziskov. Pri variante 3, znázornenej na obrázku 4.22 pozorujeme symetrické rozdelenie so strednou hodnotou 50.77%. V prípade varianty 4, na obrázku 4.24 pozorujeme mierne asymetrické rozdelenie skosené sprava so strednou hodnotou 41.48%.

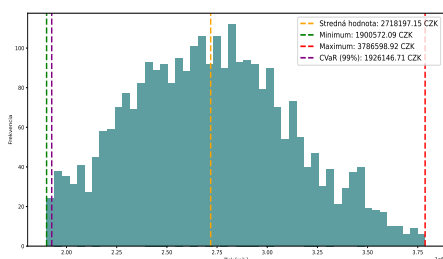




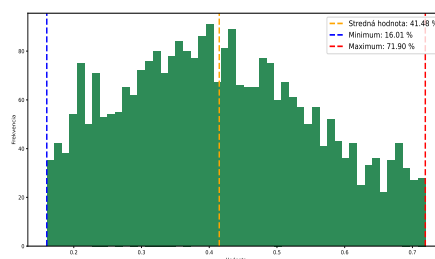
Obr. 4.21: Rozdelenie ziskov - varianta 3.



Obr. 4.22: Rozdelenie ziskov [%] - varianta 3.

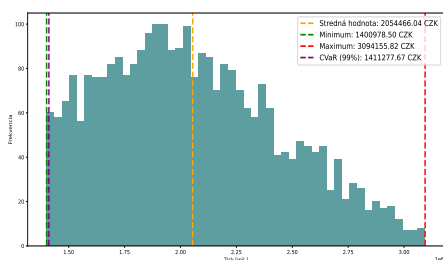


Obr. 4.23: Rozdelenie ziskov - varianta 4.

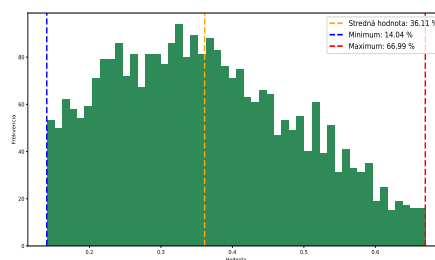


Obr. 4.24: Rozdelenie ziskov [%] - varianta 4.

V prípade **varianty 5**, kde požadujeme najvyššie hodnoty  $\alpha_1 = 0.9$  a  $\alpha_2 = 0.95$  môžeme vidieť, že rozdelenie ziskov je asymetrické a skosené sprava. Prísnejšie pravdepodobnostné obmedzenia vo variante 5 vedú k najväčšiemu poklesu strednej hodnoty zisku v porovnaní s hlavným modelom. Stredná hodnota zisku klesla na **2 054 466.04 CZK**. V porovnaní s hlavným modelom to predstavuje zníženie strednej hodnoty zisku o približne 54.4%.  $CVaR_{0.99}$  je rovné 1 411 277.67 CZK. Pri variante 5, znázornenej na obrázku 4.26 pozorujeme asymetrické rozdelenie skosené sprava so strednou hodnotou 36.11%. Absolútna hodnota zisku je najnižšia zo všetkých piatich variánt. Môžeme predpokladať, že vo väčšine prípadov je potrebné neúmerne veľké množstvo prostriedkov na náklady v porovnaní s finálnym ziskom.



Obr. 4.25: Rozdelenie ziskov - varianta 5.



Obr. 4.26: Rozdelenie ziskov [%] - varianta 5.

V tabulke môžeme vidieť zhrnutie výsledkov pre model s pridanými pravdepodobnostnými obmedzeniami.

	varianta 1	varianta 2	varianta 3	varianta 4	varianta 5
$\alpha_1$	0.75	0.80	0.85	0.9	0.95
$\alpha_2$	0.5	0.6	0.7	0.8	0.9
minimum [CZK]	3 303 582.88	3 133 781.02	2 404 659.45	2 1 900 572.09	1 400 978.50
priemerný zisk [CZK]	4 411 993.12	4 303 374.35	3 507 201.81	2 718 197.15	2 054 466.04
maximum [CZK]	5 668 720.18	5 626 427.20	4 797 277.19	3 786 598.92	3 094 155.82
CVaR <sub>0.99</sub>	3 434 377.11	4 303 374.38	2 436 612.54	1 926 146.71	1 411 277.67

Tabuľka 4.3: Súhrn výsledkov pre rozšírenie modelu s pravdepodobnostnými obmedzeniami.

# Záver

V tejto práci sme sa zaoberali problémom distribúcie liekov pomocou viacstupňového stochastického programovania. V prvej kapitole sme predstavili základné pojmy dvojstupňového a viacstupňového stochastického programovania, pričom v druhej kapitole sme navrhli dve metódy na generovanie scenárov - momentovú metódu a metódy založené na trajektóriách. Tretia kapitola formulovala dopravný problém a predstavila dvojstupňové a viacstupňové modely maximalizácie ziskov lekární, ktoré boli ďalej rozšírené o obmedzenia pravdepodobnosti. Praktická časť práce zahŕňala analýzu historických údajov, odstránenie sezónnosti, generovanie scenárov a konštrukciu stromov scenárov.

Ďalej táto práca navrhla novú metódu generovania trajektórií pomocou Holt-Wintersovej metódy na predpovedanie viacrozmerných časových radov, ktorú možno považovať za významný prínos v tejto oblasti. Modely vyvinuté v tejto práci boli aplikované na dáta z reálneho sveta. Pozreli sme sa na základný model a následne model rozšírený o pravdepodobnostné obmedzenia. Tieto pravdepodobnostné obmedzenia zabezpečovali, aby s určitou pravdepodobnosťou bol vždy v konkrétnej lekárni dostupný aspoň nejaký liek a zároveň aby bol s inou prípadne rovnakou pravdepodobnosťou dostupný konkrétny liek. Pri realizácii rôznych kombinácií pravdepodobností sme potvrdili očakávaný výsledok a to taký, že čím viac zväčšujeme obe pravdepodobnosti tým viac klesá zisk lekární. Na záver sme sa pozreli na hodnoty CVaR.

Pokiaľ ide o budúce smerovanie, táto práca otvára príležitosti na ďalšie rozšírenie navrhovaných modelov, ako je začlenenie komplexnejších obmedzení alebo zváženie ďalších faktorov, ktoré by mohli ovplyvniť distribúciu liekov. Napríklad môže ísť o výpočet konkrétnych pravdepodobností medzi lekárňami a typmi balení ak zákazník nenájde požadovaný typ balenia. Konkrétne ide o situáciu keď zákazník príde do lekárne a nemajú požadovanú veľkosť balenia tak s nejakou pravdepodobnosťou si kúpi iné balenie a s nejakou pravdepodobnosťou pôjde do inej lekárne. Avšak na takýto výpočet bohužiaľ nie sú k dispozícii priame dáta a teda by takáto analýza a následný výpočet musel byť vytvorený prieskumom priamo v lekárňach. Okrem toho je možné vykonať ďalší výskum s cieľom preskúmať alternatívne prístupy ku generovaniu scenárov alebo testovaniu navrhovaných modelov na väčších súboroch údajov. Celkovo táto práca prispieva do oblasti stochastického programovania a poskytuje pohľad na optimalizáciu distribúcie liekov v praktickom prostredí.

# Zoznam použitej literatury

- A. SHAPIRO, D. D. a RUSZCZYSŃKI, A. (2009). *Lectures on stochastic programming: Modeling and Theory*. Second Edition, SIAM. Philadelphia.
- BEALE, E. M. L. (1955). On minimizing a convex function subject to linear inequalities. *Royal Statistical Society*, **17**.
- BIRGE, J. R. a LAUVEAUX, F. (1997). *Introduction to Stochastic Programming*. Second edition. Springer. ISBN 0-387-98217-5.
- BRADLEY, P., B. K. A. D. A. (2000). Constrained kmeans clustering. URL <https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/tr-2000-65.pdf>.
- CIPRA, T. (2009). *Time Series in Economics and Finance*. Springer. ISBN 978-3-030-46346-5.
- COX, J., I. J. A. R. S. (1985). A theory of the term structure of interest rates. *Econometrica*, **53**. doi: 10.2307/1911242.
- DANTZIG, G. B. (1955). Linear programming under uncertainty. *Management Science*.
- DUPAČOVÁ, J., C. G. a WALLACE, S. (2000). *Scenarios for multistage stochastic programs*, volume 100.
- DUPAČOVÁ, J., H. J. a ŠTĚPÁN, J. (2002). *Stochastic modeling in economics and finance*. Kluwer Academic Publishers, Dordrecht. ISBN 1-4020-0840-6.
- DUPAČOVÁ, J. a LACHOUT, P. (2011). *Úvod do optimalizace*. Matfyzpress, Praha. ISBN 978-80-7378-176-7.
- HAN, J., P. J. a KAMBER, M. (2011). *Data mining: concepts and techniques*. Elsevier.
- HEITSCH, HOLGER A ROEMISCH, W. (2009). Scenario tree modeling for multistage stochastic programs. *Math. Program.*, **118**. doi: 10.1007/s10107-007-0197-2.
- HOLT, C. C. (2004). Forecasting seasonals and trends by exponentially weighted moving averages. *International Journal of Forecasting*, **20**.
- HØYLAND, K., K. M. a WALLACE, S. (2003). A heuristic for momentmatching scenario generation. *Computational Optimization and Applications*, **24**.
- JAIN, A. K. a DUBES, R. C. (1998). *Algorithms for clustering data*. Prentice-Hall, Upper Saddle River, USA.
- KAHL, C. A JACKEL, P. (2006). Fast strong approximation monte carlo schemes for stochastic volatility models. *Quantitative Finance*, **6**.

- MACQUEEN, J. (1967). Some methods for classification and analysis of multivariate observations. *In 5-th Berkeley Symposium on Mathematical Statistics and Probability.*
- SHAPIRO, A. a RUSZCZYŃSKI, A. (2003). *Stochastic programming.* Second Edition. Boston. ISBN 0444508546.
- TEKULOVÁ, P. (2020). *Optimalizácia rozvozu liekov do lekárni.* Bakalárska práca. Praha.
- V AŠÍČEK, O. (1977). An equilibrium characterization of the term structure. *Journal of Financial Economics*, **5**.
- WALLACE, S. W. a ZIEMBA, W. T. (2005). *Applications of stochastic programming.* SIAM, Philadelphia.
- WINTERS, P. R. (1960). Forecasting sales by exponentially weighted moving averages. *Management Science*, **6**.

# Zoznam obrázkov

1.1	Rozhodovací proces. . . . .	10
1.2	Reprezentácia scenárového stromu. . . . .	14
1.3	Scenárový strom s príslušnými realizáciami. . . . .	14
1.4	Scenárový strom s príslušnými pravdepodobnosťami. . . . .	16
2.1	Scenárový strom vs scenárový vejár. . . . .	22
2.2	Prevedenie scenárového vejáru na scenárový strom. . . . .	24
4.1	Scenárový strom s očíslovaným uzlov. . . . .	37
4.2	Počet predaných balení v jednotlivých lekárňach. . . . .	38
4.3	Dekompozícia dát na trendovú a sezónnu zložku. . . . .	39
4.4	Holt-Wintersová metóda - aditívna vs multiplikatívna varianta. . . . .	40
4.5	Počet objednaných 48 kusových balení v lekárni 1. . . . .	45
4.6	Rozdelenie ziskov pre lekárne 1 - 48 ks balenie. . . . .	46
4.7	Rozdelenie ziskov [%] pre lekárne 1 - 48 ks balenie. . . . .	46
4.8	Rozdelenie celkových ziskov. . . . .	47
4.9	Rozdelenie ziskov [%]. . . . .	47
4.10	Vzťah medzi časom výpočtu a hodnotami $\alpha_1$ a $\alpha_2$ . . . . .	49
4.11	Priemerný počet objednaných 48 kusových balení v lekárni 1 za použitia pravdepodobnostných obmedzení. . . . .	50
4.12	Rozdelenie ziskov [%] - 48ks balenie, varianta 1. . . . .	51
4.13	Rozdelenie ziskov [%] - 48ks balenie, varianta 2. . . . .	51
4.14	Rozdelenie ziskov [%] - 48ks balenie, varianta 3. . . . .	51
4.15	Rozdelenie ziskov [%] - 48ks balenie, varianta 4. . . . .	51
4.16	Rozdelenie ziskov [%] - 48ks balenie, varianta 5. . . . .	51
4.17	Rozdelenie ziskov - varianta 1. . . . .	52
4.18	Rozdelenie ziskov [%] - varianta 1. . . . .	52
4.19	Rozdelenie ziskov - varianta 2. . . . .	52
4.20	Rozdelenie ziskov [%] - varianta 2. . . . .	52
4.21	Rozdelenie ziskov - varianta 3. . . . .	53
4.22	Rozdelenie ziskov [%] - varianta 3. . . . .	53
4.23	Rozdelenie ziskov - varianta 4. . . . .	53
4.24	Rozdelenie ziskov [%] - varianta 4. . . . .	53
4.25	Rozdelenie ziskov - varianta 5. . . . .	53
4.26	Rozdelenie ziskov [%] - varianta 5. . . . .	53
A.1	Korelačné matice pre jednotlivé typy liekov v rámci jednotlivých lekárňach. . . . .	60
A.2	Korelačné matice pre jednotlivé lekárne v rámci jednotlivých typov liekov. . . . .	61

# Zoznam tabuliek

4.1	Kombinácie parametrov $\alpha_1$ a $\alpha_2$ . . . . .	48
4.2	Priemerný zisk lekárne 1 naprieč všetkými scenármi v prípade 48 kusových balení. . . . .	50
4.3	Súhrn výsledkov pre rozšírenie modelu s pravdepodobnostnými obmedzeniami. . . . .	54

# A. Prílohy

## A.1 Korelačné matice - typy liekov

Korelačná matica pre lekára 1					Korelačná matica pre lekára 1				
	Liek_24	Liek_36	Liek_48	Liek_100		Liek_24	Liek_36	Liek_48	Liek_100
Liek_24	1.000000	0.232845	0.490879	0.299739	Liek_24	1.000000	0.220929	0.416966	0.200862
Liek_36	0.232845	1.000000	0.120030	0.782933	Liek_36	0.220929	1.000000	0.120961	0.724292
Liek_48	0.490879	0.120030	1.000000	0.374040	Liek_48	0.416966	0.010961	1.000000	0.417235
Liek_100	0.299739	0.782933	0.374040	1.000000	Liek_100	0.300862	0.724292	0.317235	1.000000
-----					-----				
Korelačná matica pre lekára 2					Korelačná matica pre lekára 2				
	Liek_24	Liek_36	Liek_48	Liek_100		Liek_24	Liek_36	Liek_48	Liek_100
Liek_24	1.000000	0.170217	0.190218	0.040134	Liek_24	1.000000	0.198344	0.264734	0.088507
Liek_36	0.170217	1.000000	0.305382	0.871592	Liek_36	0.188344	1.000000	0.352435	0.873285
Liek_48	0.190218	0.305382	1.000000	0.269468	Liek_48	0.164734	0.252435	1.000000	0.287423
Liek_100	0.040134	0.871592	0.269468	1.000000	Liek_100	0.088507	0.873285	0.317423	1.000000
-----					-----				
Korelačná matica pre lekára 3					Korelačná matica pre lekára 3				
	Liek_24	Liek_36	Liek_48	Liek_100		Liek_24	Liek_36	Liek_48	Liek_100
Liek_24	1.000000	0.418906	0.385365	0.644733	Liek_24	1.000000	0.394914	0.365092	0.690233
Liek_36	0.418906	1.000000	0.562047	0.560055	Liek_36	0.394914	1.000000	0.551746	0.536753
Liek_48	0.385365	0.562047	1.000000	0.660651	Liek_48	0.365092	0.551746	1.000000	0.669644
Liek_100	0.644733	0.560055	0.660651	1.000000	Liek_100	0.590233	0.536753	0.659644	1.000000
-----					-----				
Korelačná matica pre lekára 4					Korelačná matica pre lekára 4				
	Liek_24	Liek_36	Liek_48	Liek_100		Liek_24	Liek_36	Liek_48	Liek_100
Liek_24	1.000000	0.143442	0.545275	0.025354	Liek_24	1.000000	0.275609	0.557081	0.022480
Liek_36	0.143442	1.000000	0.120446	0.493508	Liek_36	0.275609	1.000000	0.132182	0.423909
Liek_48	0.545275	0.120446	1.000000	0.135382	Liek_48	0.517081	0.132182	1.000000	0.114689
Liek_100	0.025354	0.493508	0.135382	1.000000	Liek_100	0.052480	0.423909	0.114689	1.000000
-----					-----				
Korelačná matica pre lekára 5					Korelačná matica pre lekára 5				
	Liek_24	Liek_36	Liek_48	Liek_100		Liek_24	Liek_36	Liek_48	Liek_100
Liek_24	1.000000	0.352971	0.601341	0.335498	Liek_24	1.000000	0.399283	0.661151	0.336739
Liek_36	0.352971	1.000000	0.322947	0.860302	Liek_36	0.399283	1.000000	0.362639	0.878552
Liek_48	0.601341	0.322947	1.000000	0.353346	Liek_48	0.581151	0.262639	1.000000	0.365250
Liek_100	0.335498	0.860302	0.353346	1.000000	Liek_100	0.306739	0.808552	0.365250	1.000000
-----					-----				

(a) Korelačné matice - historické dáta.

(b) Korelačné matice - v šiestom stupni vygenerovaných stromov.

Obr. A.1: Korelačné matice pre jednotlivé typy liekov v rámci jednotlivých lekární.



## A.2 Korelačné matice - lekárne

Korelačná matica pre Liek_24					
	1	2	3	4	5
1	1.000000	0.287222	0.474519	0.488454	0.270570
2	0.287222	1.000000	0.365745	0.317665	0.081391
3	0.474519	0.365745	1.000000	0.586196	0.346622
4	0.488454	0.317665	0.586196	1.000000	0.335295
5	0.270570	0.081391	0.346622	0.335295	1.000000

---

Korelačná matica pre Liek_36					
	1	2	3	4	5
1	1.000000	-0.031375	0.380325	0.193771	0.144220
2	-0.031375	1.000000	0.060132	0.126633	0.144489
3	0.380325	0.060132	1.000000	0.289752	0.277930
4	0.193771	0.126633	0.289752	1.000000	0.318711
5	0.144220	0.144489	0.277930	0.318711	1.000000

---

Korelačná matica pre Liek_48					
	1	2	3	4	5
1	1.000000	0.461522	0.517972	0.346040	0.402767
2	0.461522	1.000000	0.432666	0.194583	0.524989
3	0.517972	0.432666	1.000000	0.387105	0.574866
4	0.346040	0.194583	0.387105	1.000000	0.113079
5	0.402767	0.524989	0.574866	0.113079	1.000000

---

Korelačná matica pre Liek_100					
	1	2	3	4	5
1	1.000000	0.181395	0.425097	0.138984	0.288333
2	0.181395	1.000000	0.102392	-0.175294	0.104021
3	0.425097	0.102392	1.000000	0.123130	0.348999
4	0.138984	-0.175294	0.123130	1.000000	0.218213
5	0.288333	0.104021	0.348999	0.218213	1.000000

(a) Korelačné matice - historické dáta.

Korelačná matica pre Liek_24					
	1	2	3	4	5
1	1.000000	0.270629	0.473402	0.461976	0.263523
2	0.270629	1.000000	0.362891	0.343878	0.108982
3	0.473402	0.362891	1.000000	0.517909	0.265965
4	0.461976	0.343878	0.517909	1.000000	0.341984
5	0.263523	0.108982	0.265965	0.341984	1.000000

---

Korelačná matica pre Liek_36					
	1	2	3	4	5
1	1.000000	0.013919	0.303992	0.188902	0.130636
2	0.013919	1.000000	0.056685	0.112210	0.144203
3	0.303992	0.056685	1.000000	0.264535	0.260652
4	0.188902	0.112210	0.264535	1.000000	0.305139
5	0.130636	0.144203	0.260652	0.305139	1.000000

---

Korelačná matica pre Liek_48					
	1	2	3	4	5
1	1.000000	0.374029	0.512952	0.357014	0.396346
2	0.374029	1.000000	0.431347	0.205420	0.484727
3	0.512952	0.431347	1.000000	0.301136	0.557406
4	0.357014	0.205420	0.301136	1.000000	0.118770
5	0.396346	0.484727	0.557406	0.118770	1.000000

---

Korelačná matica pre Liek_100					
	1	2	3	4	5
1	1.000000	0.185563	0.465470	0.123750	0.221872
2	0.185563	1.000000	0.166108	0.129136	0.104610
3	0.465470	0.166108	1.000000	0.115135	0.341567
4	0.123750	0.129136	0.115135	1.000000	0.210894
5	0.221872	0.104610	0.341567	0.210894	1.000000

(b) Korelačné matice - v šiestom stupni vygenerovaných stromov.

Obr. A.2: Korelačné matice pre jednotlivé lekárne v rámci jednotlivých typov liekov.