UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY



POUŽITIE POMEROV V OBÁLKOVEJ ANALÝZE DÁT

DIPLOMOVÁ PRÁCA

2025 Bc. Viktória MORAVČÍKOVÁ

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

POUŽITIE POMEROV V OBÁLKOVEJ ANALÝZE DÁT

DIPLOMOVÁ PRÁCA

Študijný program:	Ekonomicko-finančná matematika a modelovanie
Študijný odbor:	Matematika
Školiace pracovisko:	Katedra aplikovanej matematiky a štatistiky
Vedúci práce:	doc. RNDr. Margaréta Halická, CSc.

Bratislava 2025 **Bc. Viktória MORAVČÍKOVÁ**





Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Bc. Viktória Moravčíková
Študijný program: ekonomicko-finančná matematika a modelovanie (Jednoodborové štúdium, magisterský II. st., denná forma)
Študijný odbor: matematika
Typ záverečnej práce: diplomová
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Použitie pomerov v obálkovej analýze dát
Use of ratios in data envelopment analysis

Anotácia: Štandardné modely obálkovej analýzy dát (DEA) boli navrhnuté na prácu so vstupmi a výstupmi vyjadrenými v objemových mierach, ktoré odrážajú veľkosť pozorovaných útvarov. V aplikáciách sa však často objavujú vstupy a výstupy reprezentované pomerovými mierami. Tieto môžu byť uvedené vo forme podielov, percent, priemerov a sadzieb získaných vydelením jednej objemovej miery druhou a často sa používajú ako kvalitatívne charakteristiky výrobného procesu. V takýchto prípadoch však štandardné modely môžu poskytovať skreslené resp. priamo nesprávne výsledky. V ostatnom období bolo publikovaných viacero prác, ktoré sa zaoberajú modelmi akceptujúcimi pomerové miery vstupov a výstupov.

Cieľ: Cieľom diplomovej práce je spracovať prehľad existujúcich prístupov pre použitie DEA na pomerové miery. Podrobne popísať a analyzovať niektoré z prístupov, možné úskalia ilustrovať príkladmi. Vybrané prístupy aplikovať na reálne dáta podľa vlastného výberu.

Literatúra: Papaioannou and Podinovski: Production technologies with ratio inputs and outputs, EJOR 310

Vedúci: doc. RNDr. Margaréta Halická, CSc.
Katedra: FMFI.KAMŠ - Katedra aplikovanej matematiky a štatistiky
Vedúci katedry: doc. Mgr. Igor Melicherčík, PhD.

Dátum zadania: 11.01.2024

Dátum schválenia: 25.01.2024

prof. RNDr. Daniel Ševčovič, DrSc.
garant študijného programu

.....
študent

.....
vedúci práce

Poďakovanie Rada by som poďakovala svojej školiteľke a vedúcej diplomovej práce doc. RNDr. Margaréte Halickej, CSc. za jej ochotu, cenné rady, podnetné pripomienky a čas, ktorý mi venovala pri konzultáciách. Zároveň vyjadrujem úprimnú vďaku svojej rodine a priateľom za ich trpezlivosť a podporu počas celého štúdia.

Abstrakt v štátnom jazyku

MORAVČÍKOVÁ, Viktória: Použitie pomerov v obálkovej analýze dát [Diplomová práca], Univerzita Komenského v Bratislave, Fakulta matematiky, fyziky a informatiky, Katedra aplikovanej matematiky a štatistiky; školiteľ: doc. RNDr. Margaréta Halická CSc., Bratislava, 2025, 85 s.

Obálková analýza dát je metóda hodnotenia efektívnosti, ktorá využíva vstupy a výstupy vyjadrené najmä v objemových mierach. V praxi sa však často stretávame s pomerovými mierami, ako sú podiely, percentá či priemery, ktoré zachytávajú kvalitatívne aspekty procesov. Štandardné DEA modely pri ich použití môžu viesť k nesprávnym záverom, čo viedlo k vývoju alternatívnych prístupov na ich integráciu. Táto práca poskytuje prehľad týchto prístupov, analyzuje ich výhody a obmedzenia a prezentuje konkrétne príklady ilustrujúce potenciálne úskalia.

Praktická časť sa venuje aplikácii vybraných metód na reálne dáta komerčných poisťovní, s cieľom analyzovať ich efektívnosť. Výskum sa zameriava na identifikáciu rozdielov medzi tradičnými a adaptovanými modelmi DEA pri práci s pomerovými mierami. Prínosom je ukázanie vhodnosti pomerových mier pre hodnotenie kvalitatívnych charakteristík poisťovní a ich výkonnosti. Výsledky môžu slúžiť ako základ pre optimalizáciu rozhodovacích procesov a ďalšie aplikácie DEA v praxi.

Kľúčové slová: obálková analýza dát, efektivita, objemové miery, pomerové miery, vstupy, výstupy, R-VRS model

Abstract

MORAVČÍKOVÁ, Viktória: Use of ratios in data envelopment analysis [Master Thesis], Comenius University in Bratislava, Faculty of Mathematics, Physics and Informatics, Department of Applied Mathematics and Statistics; Supervisor: doc. RNDr. Margaréta Halická, CSc., Bratislava, 2025, 85 p.

Data Envelopment Analysis (DEA) is a method for evaluating efficiency that typically uses inputs and outputs expressed in volume measures. In practice, ratio measures such as ratios, percentages, or averages are often encountered, reflecting qualitative aspects of processes. Standard DEA models applied to such measures can lead to incorrect conclusions, prompting the development of alternative approaches for their integration. This thesis provides an overview of these approaches, analyzes their advantages and limitations, and presents specific examples illustrating potential problems.

The practical part focuses on applying selected methods to real data from commercial insurance companies to assess their efficiency. The research aims to identify differences between traditional and adapted DEA models when working with ratio measures. The contribution lies in demonstrating the suitability of ratio measures for evaluating the qualitative characteristics and performance of insurance companies. The results can serve as a basis for optimizing decision-making processes and further applications of DEA in practice.

Keywords: data envelopment analysis, efficiency, volume measures, ratio measures, inputs, outputs, R-VRS model

Obsah

Úvod **9**

1 Základné pojmy a definície **11**

1.1 Terminológia a symbolika 11

1.2 BCC model 13

1.2.1 Vstupne orientovaný BCC model 14

1.2.2 Výstupne orientovaný BCC model 15

1.3 Superefektivita 16

2 Pomerové miery v DEA **17**

2.1 Objemové miery 17

2.2 Pomerové miery 17

2.3 Použitie pomerových mier v klasických DEA modeloch 18

2.3.1 Riešenia problémov a alternatívne prístupy 19

2.4 Množina M_{VRS}^{R} 20

3 DEA modely nad čisto pomerovými dátami: FDH model **23**

3.1 Množina M_{FDH} 23

3.2 Vstupne orientovaný FDH model 25

3.3 Výstupne orientovaný FDH model 27

4 DEA modely nad zmiešanými pomerovými aj objemovými mierami **30**

4.1 Vstupne orientovaný model R-VRS 30

4.2 Výstupne orientovaný model R-VRS 35

5 Prehľad literatúry **41**

6 Experimenty na reálnych dátach **46**

6.1 Dáta 46

6.2 Vstupne orientované modely 49

6.2.1 Prípadová štúdia 1: Použitie BCC-I modelu len na objemové dáta . 49

6.2.2 Prípadová štúdia 2: Použitie BCC-I modelu na dáta s pomerovými ukazovateľmi . . . 50
6.2.3 Prípadová štúdia 3: Použitie R-VRS-I modelu . . . 51
6.2.4 Prípadová štúdia 4: Použitie FDH-I modelu . . . 51
6.2.5 Porovnanie výsledkov vstupne orientovaných modelov . . . 52
6.3 Výstupne orientované modely . . . 54
6.3.1 Prípadová štúdia 5: Použitie BCC-O modelu len na objemové dáta 54
6.3.2 Prípadová štúdia 6: Použitie BCC-O modelu na dáta s pomerovými ukazovateľmi . . . 55
6.3.3 Prípadová štúdia 7: Použitie R-VRS-O modelu . . . 56
6.3.4 Prípadová štúdia 8: Použitie FDH-O modelu . . . 56
6.3.5 Porovnanie výsledkov výstupne orientovaných modelov . . . 57
6.4 Sumárne porovnanie výsledkov v Prípadových štúdiách 1-8 . . . 59
6.5 Sumárne porovnanie superefektivít v Prípadových štúdiách 1-8 . . . 60

Záver **62**

Zoznam použitej literatúry **64**

Príloha A **66**

A.1 Vstupne orientovaný BCC model **66**

A.2 Výstupne orientovaný BCC model **69**

A.3 Vstupne orientovaný R-VRS model **72**

A.4 Výstupne orientovaný R-VRS model **75**

A.5 Vstupne orientovaný FDH model **79**

A.6 Výstupne orientovaný FDH model **82**

Úvod

Obálková analýza dát je nástroj na hodnotenie efektívnosti rozhodovacích jednotiek, ktoré využívajú vstupy na produkciu výstupov. Táto metóda je široko používaná v oblastiach, ako je priemysel, zdravotníctvo, školstvo, či finančný sektor, pretože umožňuje porovnávať jednotky v skupine na základe ich relatívnej efektivity. DEA (Data Envelopment Analysis) je založená na matematickom programovaní a nevyžaduje špecifikáciu funkčnej podoby produkčnej funkcie, čo ju robí mimoriadne flexibilnou a prispôsobivou.

Štandardné DEA modely pracujú s objemovými mierami vstupov a výstupov, ktoré odrážajú absolútnu veľkosť sledovaných jednotiek. V praxi však často narážame na situácie, kde sú vstupy a výstupy reprezentované pomerovými mierami, ako sú podiely, percentá, priemery alebo sadzby. Tieto miery vznikajú vydelením jednej objemovej miery druhou a často sa používajú na vyjadrenie kvalitatívnych charakteristík výrobných procesov či iných ekonomických javov.

Použitie pomerových mier v štandardných DEA modeloch prináša výzvy, pretože tieto modely boli navrhnuté na prácu s absolútnymi hodnotami. V dôsledku toho môžu byť výsledky analýzy skreslené alebo dokonca nesprávne. Tieto problémy v posledných rokoch získali pozornosť odborníkov, čo viedlo k vzniku nových modelov a prístupov, ktoré berú do úvahy špecifiká pomerových mier. Olesen a kolektív v článku [5] navrhli nové modely R-VRS a R-CRS, ktoré umožňujú zahrnúť pomerové miery do DEA analýz bez potreby transformácie údajov. Tieto modely redefinujú produkčnú množinu a zohľadňujú špecifiká pomerových mier, čím zabezpečujú spoľahlivejšiu interpretáciu efektívnosti. Ďalšie rozšírenia tejto témy uviedli Papaioannou a Podinovski v článku [8], ktorí vyvinuli technológie umožňujúce pracovať s dolnými a hronými hranicami čitateľov a menovateľov pomerových mier. Tieto metodologické inovácie predstavujú dôležitý krok k riešeniu výziev spojených s využitím pomerových mier v DEA a otvárajú nové možnosti pre efektívnu analýzu v oblastiach, kde tieto miery zohrávajú kľúčovú úlohu.

Cieľom tejto diplomovej práce je spracovať prehľad existujúcich prístupov pre použitie DEA modelov na pomerové miery, detailne popísať a analyzovať vybrané prístupy, identifikovať ich možné úskalia a demonštrovať ich na konkrétnych príkladoch. Súčasťou diplomovej práce bude aj aplikácia vybraných DEA modelov na reálne dáta. Cieľom tejto časti je overiť praktickú použiteľnosť modelov v reálnych podmienkach a posúdiť ich

schopnosť poskytovať spoľahlivé a zmysluplné výsledky. Pri tejto analýze sa budú skúmať konkrétne výhody modelov, ako napríklad ich schopnosť lepšie zohľadniť kvalitatívne charakteristiky procesov, a tiež ich obmedzenia, ako sú požiadavky na kvalitu a dostupnosť dát.

Práca spája teoretický a praktický prístup, pričom sa snaží prispieť k rozšíreniu poznatkov o využití DEA modelov v situáciách, kde sú pomerové miery nevyhnutnou súčasťou analýzy. Výsledky môžu byť užitočné nielen pre akademickú komunitu, ale aj pre odborníkov v praxi, ktorí sa zaoberajú hodnotením efektívnosti v prostrediach so špecifickými typmi údajov.

1 Základné pojmy a definície

V prvej kapitole našej práce sa venujeme základnej terminológii a symbolike používanej v DEA modeloch. Definujeme matematické označenia vstupov, výstupov a rozhodovacích jednotiek. Osobitnú pozornosť venujeme aproximáciám (M_{VRS} a M_{CRS}) množiny produkčných možností ako aj pojmom efektívnosti a pseudoefektívnosti, ktoré interpretujeme pomocou hraníc týchto množín. V závere kapitoly formulujeme vstupne aj výstupne orientovaný BCC model.

1.1 Terminológia a symbolika

Základnú terminológiu a symboliku si zadefinujeme podľa skrípt [3].

Technológiu T charakterizuje množina m vstupov $I = \{I_1, I_2, ..., I_m\}$ a množina s výstupov $O = \{O_1, O_2, ..., O_s\}$. Vstupy sú technológiou spotrebúvané, pričom výstupy technológia produkuje. Vstupy môžu predstavovať materiálové, finančné, personálne, či priestorové náklady, zatiaľ čo výstupy môžu byť rôzne služby, tržby alebo produkty. Pod technológiou teda rozumieme súbor faktorov a procesov, ktoré umožňujú pretvorenie vstupov na výstupy.

Vektorom $x \in \mathbb{R}^m$ označujeme konkrétne hodnoty vstupov pre určitého producenta (útvar DMU). Každá zložka tohto vektora predstavuje hodnotu príslušného vstupu. Podobne hodnoty výstupov tvoria vektor $y \in \mathbb{R}^s$, kde každá zložka vektora reprezentuje hodnotu konkrétneho výstupu.

Dvojica vektorov (x, y) je prípustná pre danú technológiu T práve vtedy, keď je možné pomocou technológie T vyrobiť množstvo výstupov y zo vstupov x. Inými slovami, technológia T umožňuje premeniť vstupné množstvo zdrojov x na výstupy y. Množinu všetkých prípustných dvojíc (x, y), ktoré zodpovedajú technológii T, označujeme ako množinu produkčných možností G danej technológie. Táto množina obsahuje všetky hodnoty vstupov a výstupov, ktoré sú realizovateľné v rámci danej technológie.

V praxi však množinu G nepoznáme. Nevieme presne, aké hodnoty vstupov a výstupov sú prípustné v rámci celej technológie. Poznáme len niektoré vlastnosti G ako konvexnosť a voľná disponovateľnosť a hodnoty vstupov a výstupov niekoľkých producentov (napríklad poisťovní), ktorí v danej technológii produkujú. Množinu týchto známych hodnôt označíme

Z. Na základe znalosti množiny Z sa v DEA definujú aproximácie M množiny G pomocou nasledujúcich axióm:

- **(A1) Inklúzia pozorovaných bodov:** Všetky pozorované body zo Z patria do M ($Z \subset M$).
- **(A2) Konvexnosť:** M je konvexná, to znamená, že ak $(x_A, y_A) \in M$ a zároveň $(x_B, y_B) \in M$, potom pre všetky $\lambda \in [0, 1]$ platí, že $(\lambda x_A + (1 - \lambda)x_B, \lambda y_A + (1 - \lambda)y_B) \in M$.
- **(A3) Voľná disponovateľnosť:** Ak $(x_A, y_A) \in M$ potom, pre všetky $(x, y) \in \mathbb{R}_+^{m+s}$ také, že $x_A \leq x, y_A \geq y$ platí, že $(x, y) \in M$.
- **(A4) Konštantné výnosy z rozsahu (CRS):** Ak $(x_A, y_A) \in M$, potom aj $(cx_A, cy_A) \in M$, pre všetky $c > 0$.

V ďalšom budeme predpokladať, že množina Z je tvorená hodnotami vstupov a výstupov n producentov, t.j.

$$Z = \{(x_j, y_j) \in \mathbb{R}_+^{m+s} \mid j = 1, ..., n\}. \tag{1.1}$$

Tu vektory vstupov x_j a výstupov y_j j-teho producenta sú definované nasledovne

$$x_j = \begin{pmatrix} x_{1j} \\ \vdots \\ x_{ij} \\ \vdots \\ x_{mj} \end{pmatrix}, \qquad y_j = \begin{pmatrix} y_{1j} \\ \vdots \\ y_{rj} \\ \vdots \\ y_{sj} \end{pmatrix}, \quad j = 1, ..., n,$$

kde x_{ij} je hodnota i-teho vstupu pre j-teho producenta a y_{rj} je konkrétna hodnota r-teho výstupu pre j-ty útvar. Zároveň je potrebné, aby dáta n producentov z množiny Z spĺňali predpoklad z [3]:

Predpoklad 1.1. *Pre všetky $j = 1, \ldots, n$ platí $x_j \geq 0_m$, $x_j \neq 0_m$ a $y_j \geq 0_s$, $y_j \neq 0_s$.*

Na základe predchádzajúcich axióm a známych hodnôt vstupov a výstupov n producentov z množiny Z sa v DEA definujú aproximácie množiny produkčných možností pre variabilné výnosy z rozsahu (VRS) a konštantné výnosy z rozsahu (CRS). Množina

M_{VRS} je aproximáciou množiny G pre variabilné výnosy z rozsahu a je to najmenšia množina spĺňajúca axiómy inklúzie, konvexnosti a voľnej disponovateľnosti (A1, A2, A3). Formulácia množiny M_{VRS} je nasledujúca:

$$M_{VRS} = \left\{(x,y) \in \mathbb{R}^{m+s} \mid \sum_{j=1}^{n} \lambda_j x_j \leq x, \quad \sum_{j=1}^{n} \lambda_j y_j \geq y, \quad \lambda \geq 0, \quad \sum_{j=1}^{n} \lambda_j = 1\right\}. \quad (1.2)$$

Aproximácia množiny produkčných možností G pre konštantné výnosy z rozsahu M_{CRS} je najmenšou množinou, ktorá spĺňa všetky štyri vyššie uvedené axiómy. Predpis M_{CRS} je nasledujúci:

$$M_{CRS} = \left\{(x,y) \in \mathbb{R}^{m+s} \mid \sum_{j=1}^{n} \lambda_j x_j \leq x, \quad \sum_{j=1}^{n} \lambda_j y_j \geq y, \quad \lambda \geq 0\right\}. \quad (1.3)$$

Ako ďalší termín si podľa [3] definujeme hranicu efektívnosti. Hranica efektívnosti predstavuje množinu efektívnych bodov na hranici produkčných možností. Bod je efektívny, ak neexistuje iný bod, ktorý by pri rovnakých alebo menších vstupoch poskytoval rovnaké alebo vyššie výstupy. Matematicky ju možno vyjadriť ako:

$$H_E = \{(x,y) \in \partial M \mid \nexists(\bar{x},\bar{y}) \in M : (\bar{x},\bar{y}) \neq (x,y), \bar{x} \leq x, \bar{y} \geq y\}. \quad (1.4)$$

Hranica pseudoefektívnosti zahŕňa body, ktoré síce ležia na hranici produkčných možností, ale nie sú efektívne. Bod je pseudoefektívny, ak existuje iný bod s rovnakými alebo menšími vstupmi a rovnakými alebo vyššími výstupmi. Matematicky ju možno zapísať ako:

$$H_P = \{(x,y) \in \partial M \mid \exists(\bar{x},\bar{y}) \in M : (\bar{x},\bar{y}) \neq (x,y), \bar{x} \leq x, \bar{y} \geq y\}. \quad (1.5)$$

Aproximácia M následne umožňuje vyhodnocovať efektívnosť a efektivitu, pričom efektívni producenti sú tí, ktorí sa nachádzajú na hranici efektívnosti množiny M. Takto získané výsledky sú však relatívne a platné len pre konkrétnu skupinu producentov zahrnutých v analýze.

1.2 BCC model

Podľa [3], základným modelom v rámci DEA je CCR model, ktorý operuje za predpokladu konštantných výnosov z rozsahu (CRS). Ako alternatíva k tomuto prístupu sa často

využíva BCC model, ktorý predpokladá variabilné výnosy z rozsahu (VRS). Na rozdiel od CCR modelu, BCC model zavádza dodatočné obmedzenie $\sum_{j=1}^{n} \lambda_j = 1$.

BCC model je možné formulovať dvomi spôsobmi – vstupne orientovaný a výstupne orientovaný – v závislosti od toho, či je cieľom znížiť vstupy pri zachovaní úrovne výstupov, alebo zvýšiť výstupy pri zachovanej úrovni vstupov.

Vzhľadom na to, že sa v našej práci zameriavame na analýzu efektivity za predpokladu variabilných výnosov z rozsahu, model CCR ďalej nepopisujeme a venujeme sa priamo modelom založeným na predpoklade VRS.

1.2.1 Vstupne orientovaný BCC model

Vstupne orientovaný BCC model je založený na minimalizácii vstupov pri nezmenenej hodnote výstupov. Optimalizačná úloha pre (x_o, y_o) je definovaná ako:

$$(BCC-I)_o: \quad \min_{\theta,\lambda} \quad \theta \tag{1.6}$$

$$\sum_{j=1}^{n} x_j \lambda_j \leq \theta x_o, \tag{1.7}$$

$$\sum_{j=1}^{n} y_j \lambda_j \geq y_o, \tag{1.8}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \quad \lambda \geq 0_n. \tag{1.9}$$

Optimálnu hodnotu účelovej funkcie θ označíme ako θ_0^*, pričom táto hodnota bude ležať v intervale $(0, 1]$. Projekciou je $(\theta^* x_o, y_o)$. Následne pomocou druhej fázy, overíme či sa jedná o efektivitu daného útvaru alebo o jeho pseudoefektivitu. Optimalizačná úloha druhej fázy pre vstupne orientovaný BCC model je formulovaná nasledovne:

$$\max_{\lambda, s^x, s^y} \quad \sum_{i=1}^{m} s_i^x + \sum_{r=1}^{s} s_r^y \tag{1.10}$$

$$\sum_{j=1}^{n} x_j \lambda_j + s^x = \theta^* x_o, \tag{1.11}$$

$$\sum_{j=1}^{n} y_j \lambda_j - s^y = y_o, \tag{1.12}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \quad \lambda \geq 0_n, \quad s^x \geq 0, \quad s^y \geq 0, \tag{1.13}$$

kde s^x a s^y sú tzv. slacky, ktoré ukazujú o koľko by ešte projekcia mala znížiť vstupy alebo zvýšiť výstupy, aby sa stala efektívnou. Ak sa optimálna hodnota účelovej funkcie v úlohe (1.10)–(1.13) rovná nule, všetky vstupné aj výstupné slacky sú nulové, čo znamená, že θ^* vyjadruje efektivitu, lebo (x_o, y_o) bola projektovaná na hranicu efektívnosti. Naopak, ak je optimálna hodnota účelovej funkcie kladná, t.j. aspoň jedna zo zložiek slackov je nenulová, tak θ^* vyjadruje pseudoefektivitu, lebo (x_o, y_o) bola projektovaná na hranicu pseudoefektívnosti.

1.2.2 Výstupne orientovaný BCC model

Výstupne orientovaný BCC model je definovaný podobne, pričom cieľom je zvýšiť výstupy pri zachovaní úrovne vstupov. Optimalizačná úloha je v tomto prípade definovaná ako:

$$(BCC-O)_o: \quad \max_{\psi,\lambda} \quad \psi \tag{1.14}$$

$$\sum_{j=1}^{n} x_j\lambda_j \leq x_o, \tag{1.15}$$

$$\sum_{j=1}^{n} y_j\lambda_j \geq \psi y_o, \tag{1.16}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \quad \lambda \geq 0_n. \tag{1.17}$$

Optimálnu hodnotu účelovej funkcie ψ označíme ψ^*, pričom je zrejmé, že $\psi^* \in [1, \infty)$. Rovnako, ako pri vstupne orientovanom modeli, aj tu nasleduje druhá fáza, v ktorej sa pri pevnej hodnote ψ^* maximalizuje súčet slackov. Ak sú všetky slacky nulové, hodnota $\frac{1}{\psi^*}$ je efektivitou danej jednotky. Ak sú slacky nenulové, hodnota $\frac{1}{\psi^*}$ je jej pseudoefektivitou. Druhá fáza pre výstupne orientovaný model má nasledovnú formuláciu:

$$\max_{\lambda,s^x,s^y} \quad \sum_{i=1}^{m} s_i^x + \sum_{r=1}^{s} s_r^y \tag{1.18}$$

$$\sum_{j=1}^{n} x_j\lambda_j + s^x = x_o, \tag{1.19}$$

$$\sum_{j=1}^{n} y_j\lambda_j - s^y = \psi^* y_o, \tag{1.20}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \quad \lambda \geq 0_n, \quad s^x \geq 0, \quad s^y \geq 0. \tag{1.21}$$

Analogicky, ako pri druhej fáze vstupne orientovaného modelu, ak v modeli (1.18)-(1.21) dostaneme nulovú optimálnu hodnotu účelovej funkcie, tak prevrátená hodnota ψ^* je efektivitou, lebo (x_o, y_o) bola projektovaná na hranicu efektívnosti. Ak optimálna hodnota účelovej funkcie v modeli (1.18)-(1.21) je nenulová, $\frac{1}{\psi^*}$ je pseudoefektivitou, lebo dvojica (x_o, y_o) bola projektovaná na hranicu pseudoefektívnosti.

1.3 Superefektivita

Koncept superefektivity je v [3] predstavený ako rozšírenie základných DEA modelov, ktoré umožňuje diferencovane porovnávať aj medzi rozhodovacími jednotkami, ktoré boli pôvodne označené ako efektívne. Klasické modely identifikujú jednotky ležiace na efektívnej hranici ako efektívne, nedokážu medzi nimi ďalej rozlišovať. Superefektivita tento nedostatok prekonáva tým, že hodnotenú rozhodovaciu jednotku vylúči zo súboru, voči ktorému sa tvorí efektívna hranica. Následne sa táto jednotka hodnotí voči upravenej množine a jej vzdialenosť od novej hranice slúži ako miera superefektivity.

Vo vstupne orientovanom modeli sa potom superefektivita vyjadruje ako minimálna hodnota parametra θ, pričom $\theta \geq 1$. Vo výstupne orientovanom modeli je to $\frac{1}{\psi} \geq 1$. Platí však, že takéto hodnoty vznikajú výhradne pre jednotky, ktoré boli v základnom DEA modeli označené ako efektívne. Pre neefektívne jednotky nemá zmysel superefektivitu počítať, keďže ich efektivita je už menšia ako 1.

Podľa [3] umožňuje tento prístup zostaviť úplné poradie medzi efektívnymi jednotkami a zároveň poskytuje informáciu o stabilite ich efektívnosti - teda o tom, ako citlivé sú výsledky na malé zmeny v dátach. Poznamenajme však, že model je spoľahlivo definovaný pre prípad CRS. Avšak v prípade VRS môže dôjsť k neexistencii riešenia, najmä ak sa po vylúčení pozorovaného útvaru nedá zostaviť jeho efektívny vzor.

2 Pomerové miery v DEA

V obálkovej analýze dát sa v rámci analýzy efektívnosti a produktivity stretávame s rôznymi typmi vstupov a výstupov, ktoré môžeme rozdeliť na objemové a pomerové faktory. Tieto dva typy údajov zohrávajú kľúčovú úlohu pri modelovaní produkčných technológií a hodnotení efektívnosti. Zatiaľ čo objemové faktory reprezentujú priamo merateľné veličiny, pomerové faktory poskytujú relatívne miery kvalitatívnych charakteristík produkčného procesu. Rozdiely medzi objemovými a pomerovými faktormi majú zásadný vplyv na správnosť a použiteľnosť modelov obálkovej analýzy dát.

V druhej kapitole našej práce sa venujeme definícii oboch typov faktorov a analýzou problémov, ktoré vznikajú pri použití pomerových faktorov v klasických DEA modeloch.

2.1 Objemové miery

Pod objemovými faktormi v kontexte obálkovej analýzy dát rozumieme vstupy a výstupy produkčného procesu, ktoré môžu byť merané v rôznych jednotkách. Objemové vstupy predstavujú zdroje, ktoré technológia spotrebováva, ako napríklad materiálové náklady, finančné výdavky, ľudské zdroje či priestorové náklady. Na druhej strane, výstupy sú výsledkom produkčného procesu. Výstupy vyjadrené pomocou objemových faktorov môžu zahŕňať rôzne typy služieb, tržby z predaja, alebo počet vyrobených produktov.

V DEA modeloch sa tieto objemové faktory používajú na analýzu efektívnosti, pričom sa hodnotí, ako efektívne jednotlivé rozhodovacie jednotky premieňajú vstupy na výstupy. Každá jednotka je analyzovaná z pohľadu efektívnosti v rámci množiny produkčných možností, čo umožňuje určiť, či daná jednotka dosahuje maximálny výstup s danými vstupmi alebo či existuje priestor na zlepšenie v podobe zníženia vstupov pri zachovaní rovnakého výstupu [3].

2.2 Pomerové miery

Pomerové faktory sú miery založené na pomere dvoch objemových faktorov a často sa vyskytujú vo forme percent, priemerov alebo podielov. V aplikáciách obálkovej analýzy dát, sú pomerové faktory užitočné najmä tam, kde sú objemové údaje buď nedostupné alebo neposkytujú dostatočný kontext na hodnotenie produktivity a efektívnosti. Pome-

rové miery dokážu reprezentovať rôzne aspekty kontextu, v ktorom organizácie fungujú, ako napríklad kvalitu vstupov a výstupov.

V zdravotníctve, školstve či finančnom sektore sa pomerové faktory často používajú na zachytenie kvalitatívnych vlastností alebo sociálno-ekonomického prostredia, napríklad úspešnosť liečby, pomer kvalifikovaných pracovníkov, miera nezamestnanosti či podiel obyvateľstva s určitými zdravotnými rizikami. Ako naznačuje článok [5], ak by mohli byť tieto faktory použité priamo, tak by sme vedeli analyzovať efektívnosť bez potreby ich transformácie na objemové miery, čím by sa zachovala ich informačná hodnota v kontexte kvality alebo socio-ekonomického prostredia.

2.3 Použitie pomerových mier v klasických DEA modeloch

Použitie pomerových mier v klasických modeloch obálkovej analýzy dát predstavuje viaceré problémy, ktoré narúšajú štandardné predpoklady týchto modelov. Prvým významným problémom je nesúlad pomerových mier s predpokladom konvexnosti množiny produkčných možností (A2). Konvexnosť množiny produkčných možností (A2) predpokladá, že aj konvexné kombinácie vstupov a výstupov jednotlivých producentov patria do množiny produkčných možností. Olesen sa v článku [5] snažil demonštrovať tento problém na príklade nemocníc, ktorých výstupy boli počet pacientov a počet úspešných ošetrení:

Nemocnica	**Počet pacientov**	**Úspešné ošetrenia**	**Miera úspechu (%)**
A	1000	200	20
B	500	400	80
C (0.5A + 0.5B)	750	300	50 (nesprávne), 40 (správne)

Tabuľka 1: Príklad demonštrujúci problém s konvexnosťou pri použití pomerových mier

Máme dve nemocnice, A a B, ktorých miera úspechu liečby sa počíta ako pomer úspešných ošetrení k celkovému počtu pacientov. Vytvoríme priemer týchto dvoch nemocníc a označíme ho ako nemocnica C. Priemer je konvexná kombinácia s rovnakými váhami 0,5. Celkový počet pacientov v nemocnici C teda je 750 a počet úspešných ošetrení je 300. Miera úspechu, vypočítaná ako pomer úspešných ošetrení k celkovému počtu pacientov, nemocnice C by teda mala byť 40 %. Avšak tento výsledok nekorešponduje s aritmetickým priemerom úspešnosti nemocníc A a B, ktorý je 50 %. Táto nezrovnalosť vzniká, pretože nemocnice A a B majú rôzne celkové počty pacientov (menovatele v miere úspechu).

Olesen a kol. vo svojom článku opísali ako ďalší problém to, že pomerové miery nespĺňajú axiómu konštantných výnosov z rozsahu (A4), podľa ktorej môžu byť všetky vstupy a výstupy škálované proporcionálne bez zmeny efektivity. Tento predpoklad je charakteristický pre konštantné výnosy z rozsahu a pri pomerových mierach nie je splnený, pretože tieto miery sa nemenia priamo úmerne so škálovaním. Niektoré pomerové miery dokonca predstavujú exogénne premenné, ktoré nezávisia od veľkosti danej jednotky.

Posledným významným problémom, ktorý je spojený s použitím pomerových mier v klasických DEA modeloch je, podľa Olesena a kol., [5], nedostatok ich technologického významu. Aj keď efektivitu možno vyjadriť pomocou multiplikatívnych formulácií CRS a VRS modelov, v prípade pomerových mier tieto modely nevytvárajú produkčnú množinu, ktorá by verne odrážala možnosti zlepšovania vstupov a výstupov. Výsledné hodnoty efektivity tak môžu byť arbitrárne a bez jasného technologického významu.

2.3.1 Riešenia problémov a alternatívne prístupy

Po identifikovaní vyššie spomenutých problémov, ktoré vznikajú pri použití pomerových mier v štandardných DEA modeloch, bolo navrhnutých viacero alternatívnych prístupov. Avšak každý z týchto prístupov, ktoré sa pokúšajú začleniť pomerové miery do štandardných DEA modelov, má svoje obmedzenia a nevýhody. Ďalší výklad bude voľne spracovaný podľa článku [5].

Thanassoulis a kol. vo svojom článku [11] ako prví navrhli použitie objemových mier namiesto pomerových. Ako sme definovali v časti 2.2, pomerové faktory v obálkovej analýze dát vyjadrujú vzťah medzi dvomi objemovými faktormi. Tento prístup teda transformuje pomerovú mieru na dve objemové miery. Toto riešenie problému s použitím pomerových mier môže byť ilustrované na nasledujúcom príklade: ak by sme mali mieru úspešnosti liečby ako pomerový faktor, tak v tomto prístupe by sme ju nahradili dvoma objemovými mierami - počet úspešných ošetrení a celkový počet pacientov. Pri použití tohto prístupu však narážame na viacero nevýhod a problémov. Prvým problémom je, že objemové miery, pomocou ktorých je vyjadrená pomerová miera nemusia byť dostupné v dátach alebo môžu byť príliš dôverné, teda je potrebné použiť pomerové miery. Použitím objemových mier sa stráca informačná hodnota pomerových mier, ktoré často zachytávajú

kvalitatívne vlastnosti alebo špecifický kontext, čo je ďalšou nevýhodou tohoto prístupu.

Deprins a kolektív [1] použili FDH (free disposal hull) model, ako ďalší prístup. Keďže tento model eliminuje predpoklad konvexnosti produkčnej množiny, tak sa v zásade môže použiť ako pri objemových mierach, tak aj pri pomerových mierach. Nevýhodou tohoto prístupu je, že tento model nezohľadňuje informáciu o tom, či tie vstupy a výstupy, ktoré sú vyjadrené v objemových mierach môžu tvoriť konvexné kombinácie, čo vedie k zbytočne malej produkčnej množine a nadhodnoteným efektivitám.

Ako ďalší prístup bol návrh Dysona a kol. [2] prenásobiť pomerové miery vhodnou mierou veľkosti rozhodovacej jednotky. Takáto úprava by umožnila, aby výsledné meranie efektívnosti reflektovalo proporčné zmeny vo veľkosti daného DMU. Nevýhodou tohto prístupu je však subjektívnosť výberu mier, pričom jedinou objektívnou zložkou ostáva menovateľ pôvodného pomeru. Ďalšou nevýhodou môže byť, že tento spôsob stále vyžaduje určitú formu transformácie údajov, čo môže narušiť ich pôvodný význam.

Hlavným problémom pri týchto alternatívnych prístupoch opísaných v [5] je, že ani jeden neumožňuje plnohodnotné začlenenie pomerových mier do analýzy bez potreby ich transformácie. Táto nevyhnutná úprava údajov často vedie k zmene pôvodného významu pomerových mier, čo môže negatívne ovplyvniť interpretáciu výsledkov.

2.4 Množina M^R_{VRS}

Ako sme uviedli v kapitole 2.3 našej práce, pomerové miery vo všeobecnosti nespĺňajú predpoklad konvexnosti produkčnej množiny M_{VRS}. Z tohoto dôvodu bola v článku [5] navrhnutá produkčná množina M^R_{VRS}, ktorá umožňuje efektívne zapracovanie pomerových mier bez potreby ich transformácie na objemové miery.

Pred definovaním samotnej množiny M^R_{VRS} potrebujeme presne špecifikovať, ako sú vstupy a výstupy v produkčnej technológii rozdelené na objemové a pomerové miery, keďže každá z týchto kategórií má špecifické vlastnosti a vyžaduje odlišný prístup pri modelovaní.

Definície množín vstupov a výstupov boli v [5] upravené tak, aby zohľadnili aj pomerové miery. Nech $I^V \subseteq I$ ($O^V \subseteq O$) je podmnožina množiny vstupov (výstupov) obsahujúca len objemové miery. Potom komplementy k týmto podmnožinám $I^R = I \setminus I^V$ ($O^R = O \setminus O^V$) obsahujú pomerové miery. Vektory vstupov x_j a výstupov y_j j-teho

producenta sú potom definované nasledovne

$$x_j = \begin{pmatrix} x_{1j}^V \\ \vdots \\ x_{i'j}^V \\ x_{(i'+1)j}^R \\ \vdots \\ x_{mj}^R \end{pmatrix} = \begin{pmatrix} x_j^V \\ x_j^R \end{pmatrix}, \qquad y_j = \begin{pmatrix} y_{1j}^V \\ \vdots \\ y_{r'j}^V \\ y_{(r'+1)j}^R \\ \vdots \\ y_{sj}^R \end{pmatrix} = \begin{pmatrix} y_j^V \\ y_j^R \end{pmatrix}, \qquad \begin{matrix} j = 1, \dots, n \\ i \in \{1, \dots, m\}, \\ r \in \{1, \dots, s\}, \end{matrix}$$

kde x_j^V a y_j^V sú objemové vstupy a výstupy j-teho DMU, ktoré predstavujú tradičné kvantitatívne údaje a x_j^R a y_j^R sú pomerové vstupy a výstupy, ktoré zahŕňajú údaje vo forme pomerov, ako sú napríklad percentuálne miery a pomery. Pod x_j^R a y_j^R si môžeme predstaviť vektory reprezentujúce kvalitatívne charakteristiky.

Keďže máme všetky potrebné informácie, môžeme prejsť k definícii samotnej množiny M_{VRS}^R. Kvôli nesúladu pomerových mier s predpokladom konvexnosti musíme pôvodné axiómy, ktoré spĺňa štandardná množina M, upraviť. Axiómy A1 (inklúzia pozorovaných bodov) a A3 (voľná disponovateľnosť) zostávajú rovnaké. Ako sme demonštrovali v časti 2.3, problém s predpokladom konvexnosti nastáva, keď pomerové miery majú pre rôzne DMU odlišné hodnoty menovateľov. Výnimkou je situácia, kedy všetky rozhodovacie jednotky majú rovnaké hodnoty pomerových ukazovateľov, pričom nezáleží na hodnotách ich menovateľov. V takomto prípade nedochádza k problému s predpokladom konvexnosti, pretože hodnoty pomerových ukazovateľov sa nezmenia ani po ich konvexnej kombinácii.

Podinovski v článku [9] odvodil axiómu selektívnej konvexnosti. Teda množina M^R musí spĺňať nasledujúce axiómy:

- **(A1R) Inklúzia pozorovaných bodov:** Všetky pozorované body zo Z patria do M^R, t.j. $Z \subset M^R$.

- **(A2R) Selektívna konvexnosť:** M^R je konvexná vzhľadom na objemové ukazovatele, to znamená, že ak $(x_A, y_A) \in M^R$ a zároveň $(x_B, y_B) \in M^R$, pričom $x_A^R = x_B^R$ a $y_A^R = y_B^R$, potom pre všetky $\lambda \in [0, 1]$ platí, že $(\lambda x_A + (1-\lambda)x_B, \lambda y_A + (1-\lambda)y_B) \in M^R$.

- **(A3R) Voľná disponovateľnosť:** Ak $(x_A, y_A) \in M^R$, potom pre všetky $(x, y) \in \mathbb{R}_+^{m+s}$ také, že $x_A \leq x,\ y_A \geq y$ platí, že $(x, y) \in M^R$.

Množina M^R_{VRS} je definovaná ako prienik všetkých množín $M^R \subset \mathbb{R}^{m+s}_+$, ktoré spĺňajú axiómy (A1R) - (A3R). Teda množina M^R_{VRS} je navrhnutá tak, aby to bola najmenšia množina, ktorá spĺňa všetky vyššie uvedené axiómy. V prípade, že je splnený Predpoklad 1.1, tak podľa [9], Theorem 1 (citované tiež v [5] ako Theorem 1, resp. [7] ako Theorem 1) množinu M^R_{VRS} tvoria dvojice vstupov a výstupov (x, y), kde $x = (x^V, x^R)$ a $y = (y^V, y^R)$, pre ktoré existuje taký vektor λ, že sú splnené nasledujúce ohraničenia:

$$\sum_{j=1}^{n} \lambda_j x_j^V \leq x^V, \tag{2.1}$$

$$\sum_{j=1}^{n} \lambda_j y_j^V \geq y^V, \tag{2.2}$$

$$\lambda_j (x_j^R - x^R) \leq 0, \qquad j = 1, ..., n, \tag{2.3}$$

$$\lambda_j (y_j^R - y^R) \geq 0, \qquad j = 1, ..., n, \tag{2.4}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \tag{2.5}$$

$$\lambda_j \geq 0, \qquad j = 1, ..., n. \tag{2.6}$$

Môžeme si všimnúť, že ak pomerové ukazovatele nie sú prítomné (t.j. ohraničenia (2.3) a (2.4) sa odstránia) množina M^R_{VRS} je totožná s množinou M_{VRS}.

V prípade, že nie sú prítomné objemové ukazovatele, množina M^R_{VRS} sa redukuje na M_{FDH} (Free Disposal Hull), ktorým sa budeme zaoberať v nasledujúcej kapitole.

3 DEA modely nad čisto pomerovými dátami: FDH model

V tejto kapitole sa venujeme Free Disposal Hull (FDH) modelu, ktorý bol pôvodne definovaný v [1] autormi Deprins, Simar a Tulkens ako alternatíva k BCC modelu bez predpokladu konvexnosti produkčnej množiny. Model FDH bol sformulovaný na základe predpokladu voľnej disponovateľnosti vstupov a výstupov.

V ďalších podkapitolách predstavíme podľa [1] a [4] základnú štruktúru FDH modelu a zameriame sa osobitne na vstupne a výstupne orientovaný prístup.

3.1 Množina M_{FDH}

Ako sme popísali v časti 2.3.1, tento model je vhodný pre prácu s pomerovými dátami, pretože nepredpokladá konvexnosť produkčnej množiny. Podľa [4], FDH model konštruuje produkčnú množinu na základe vlastnosti voľnej disponovateľnosti pozorovaných dát. Teda s každým bodom $(x_A, y_A) \in M_{FDH}$ patria do množiny aj všetky body s vyššími vstupmi alebo nižšími výstupmi. Formálne možno množinu M_{FDH} zapísať nasledovne:

$$M_{FDH} = \left\{(x,y) \in \mathbb{R}^{m+s} \,\middle|\, \exists j \in \{1,\dots,n\} : x \geq x_j,\ y \leq y_j \right\}. \tag{3.1}$$

Alternatívne, podľa [12], možno túto množinu pre variabilné výnosy z rozsahu prepísať do tvaru:

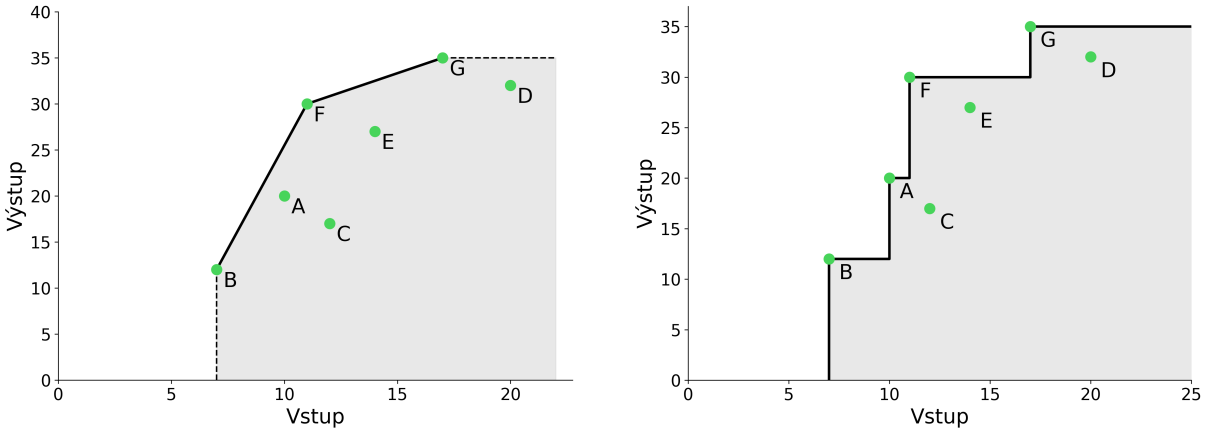
$$M_{FDH} = \left\{(x,y) \in \mathbb{R}^{m+s} \mid \sum_{j=1}^{n} \lambda_j x_j \leq x, \sum_{j=1}^{n} \lambda_j y_j \geq y, \sum_{j=1}^{n} \lambda_j = 1, \lambda_j \in \{0,1\},\ j = 1,\dots,n \right\}. \tag{3.2}$$

Porovnanie množín M_{VRS} a M_{FDH} si ilustrujeme na príklade s hypotetickými dátami:

Príklad 3.1. *Ak dáta zobrazené v tabuľke 2 odpovedajú objemovým mieram, tak uvažujeme množinu M_{VRS}. V prípade, že tieto dáta predstavujú pomerové miery, volíme množinu M_{FDH}, ktorá nevyžaduje predpoklad konvexnosti.*

DMU	A	B	C	D	E	F	G
x	10	7	12	20	14	11	17
y	20	12	17	32	27	30	35

Tabuľka 2: Hypotetické hodnoty vstupov a výstupov siedmich producentov



Obr. 1: Množiny M_{VRS} a M_{FDH}

Na obrázku 1 sú zobrazené produkčné množiny odvodené z dvoch rôznych prístupov (M_{VRS} a M_{FDH}) aplikované na tie isté údaje. Naľavo je znázornená množina M_{VRS}, ktorá pracuje s predpokladom konvexnosti. Vďaka tomu má hranica konkávny, lomený tvar. Napravo je množina M_{FDH}. V tomto prípade sa nepredpokladá konvexnosť a porovnávanie prebieha iba voči jednotlivým pozorovaniam, čo vedie k schodovitej produkčnej hranici. Pri VRS sú ako efektívne identifikované body B, F a G. V prípade FDH, kde sa efektivita posudzuje len voči existujúcim útvarom bez tvorby konvexných kombinácií, pribúda medzi efektívne aj bod A. Tento bod síce neleží na hranici konvexnej obálky, no v rámci schodovitej štruktúry FDH patrí medzi efektívne útvary.

Z predpisov množín M_{FDH} (3.2) a M_{VRS} (1.2) vieme jednoducho zdôvodniť inklúziu $M_{FDH} \subseteq M_{VRS}$. Množina M_{FDH} povoľuje len binárne váhy $\lambda_j \in \{0, 1\}, \quad j = 1, ..., n$, teda každé pozorovanie v tejto množine vzniká ako výber reálnych jednotiek bez ich kombinovania. Naopak, v množine M_{VRS} môžu byť váhy ľubovoľné nezáporné čísla so súčtom 1, teda umožňuje vytvárať konvexné kombinácie jednotiek. Každá binárna kombinácia je teda špeciálnym prípadom konvexnej kombinácie, a preto každý bod M_{FDH} patrí aj do M_{VRS}. Túto inklúziu môžeme vidieť aj na obrázku 1 - hranica FDH leží celá pod alebo na hranici VRS.

3.2 Vstupne orientovaný FDH model

Vstupne orientovaný FDH model je založený na minimalizácii vstupu pri zachovaní výstupu. Narozdiel od štandardných DEA modelov, FDH nepredpokladá konvexnosť produkčnej množiny. Každá rozhodovacia jednotka sa porovnáva len s existujúcimi jednotkami, t.j. bez interpolácie medzi nimi, čo sa dosahuje tým, že premenné λ_j sú binárne, t.j. $\lambda_j \in \{0, 1\}$.

Optimalizačná úloha pre jednotku (x_o, y_o) má nasledovnú podobu:

$$(FDH - I)_o : \quad \min_{\theta,\lambda} \quad \theta \tag{3.3}$$

$$\sum_{j=1}^{n} x_j \lambda_j \leq \theta x_o, \tag{3.4}$$

$$\sum_{j=1}^{n} y_j \lambda_j \geq y_o, \tag{3.5}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \tag{3.6}$$

$$\lambda_j \in \{0, 1\}, \qquad j = 1, .., n. \tag{3.7}$$

Binárne λ_j zaručujú, že sa každá jednotka porovnáva len s jednou konkrétnou pozorovanou jednotkou, nie ich kombináciou. Produkčná množina má v dôsledku toho schodovitý tvar.

Optimálnu hodnotu účelovej funkcie označíme θ^*. Je zrejmé, že $\theta^* \in (0, 1]$.

Druhou fázou FDH modelu dopočítame hodnoty slackov a overíme, či skúmaná jednotka je efektívna alebo hodnota θ^* je len pseudoefektivitou. V prípade, že optimálna hodnota účelovej funkcie (3.8) je nulová, hovoríme, že jednotka je efektívna. V opačnom prípade hovoríme o pseudoefektivite.

Formulácia druhej fázy:

$$\max_{\lambda, s^x, s^y} \quad \sum_{i=1}^{m} s_i^x + \sum_{r=1}^{s} s_r^y \tag{3.8}$$

$$\sum_{j=1}^{n} x_j \lambda_j + s^x = \theta^* x_o, \tag{3.9}$$

$$\sum_{j=1}^{n} y_j \lambda_j - s^y = y_o, \tag{3.10}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \quad s^x \geq 0, \quad s^y \geq 0 \tag{3.11}$$

$$\lambda_j \in \{0, 1\}, \qquad j = 1, .., n. \tag{3.12}$$

Príklad 3.2. *Na dáta zobrazené v tabuľke 3 sme aplikovali vstupne orientovaný FDH model. V tomto príklade budeme pracovať so vstupom x a výstupom y ako pomerovými premennými.*

DMU	A	B	C	D	E	F	G
x	10	7	12	20	14	11	17
y	20	12	17	32	27	30	35

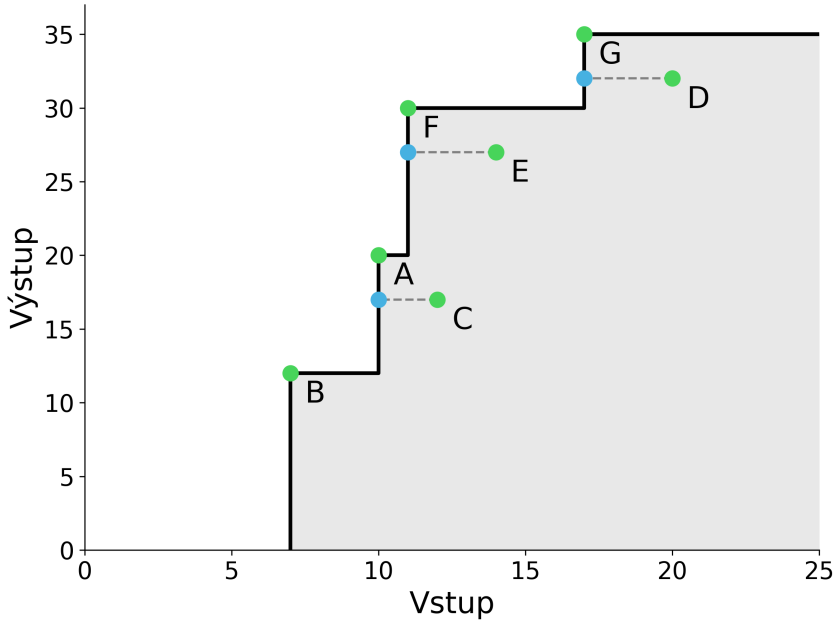
Tabuľka 3: Hypotetické hodnoty vstupov a výstupov siedmich producentov

Efektivity a výsledky po aplikovaní modelu sú zobrazené v tabuľke 4.

DMU	θ^*	λ_A	λ_B	λ_C	λ_D	λ_E	λ_F	λ_G	s^x	s^y
A	1	1	0	0	0	0	0	0	0	0
B	1	0	1	0	0	0	0	0	0	0
C	0.8333	1	0	0	0	0	0	0	0	3
D	0.85	0	0	0	0	0	0	1	0	3
E	0.7857	0	0	0	0	0	1	0	0	3
F	1	0	0	0	0	0	1	0	0	0
G	1	0	0	0	0	0	0	1	0	0

Tabuľka 4: Výsledky po aplikovaní FDH-I

Obrázok 2 zobrazuje množinu M_{FDH}, ktorá reprezentuje produkčné možnosti. Zelené body v tomto grafe predstavujú pôvodné pozorované jednotky. Môžeme pozorovať, že body C, D, E sa nachádzajú vo vnútri množiny, teda nie sú efektívne. Keďže sme na dáta aplikovali vstupne orientovaný model, ktorého cieľom je znižovať vstupy pri zachovaní výstupov, dostali sme ako výsledok horizontálne projekcie neefektívnych jednotiek (modré body). V tabuľke 4 môžeme pozorovať, že všetky neefektívne jednotky majú v tomto prípade nenulové slacky, čo znamená, že ich projekcia sa nachádza iba na hranici pseudoefektívnosti. Teda hodnoty θ^* sú pseudoefektivitami týchto útvarov.



Obr. 2: Množina M_{FDH} a projekcie neefektívnych bodov po aplikovaní FDH-I

3.3 Výstupne orientovaný FDH model

Výstupne orientovaný FDH model sa zameriava na maximalizáciu výstupu pri pevne daných vstupoch. Analogicky, ako pri vstupne orientovanom FDH modeli, aj výstupne orientovaný model pracuje s $\lambda_j \in \{0, 1\}$, čím sa zabezpečí, že každá jednotka je porovnávaná len s jednou inou jednotkou, nie s ich kombináciou.

Optimalizačná úloha pre dvojicu (x_o, y_o) má nasledujúci tvar:

$$(FDH - O)_o : \quad \max_{\psi,\lambda} \quad \psi \tag{3.13}$$

$$\sum_{j=1}^{n} x_j \lambda_j \leq x_o, \tag{3.14}$$

$$\sum_{j=1}^{n} y_j \lambda_j \geq \psi y_o, \tag{3.15}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \tag{3.16}$$

$$\lambda_j \in \{0, 1\}, \qquad j = 1, .., n. \tag{3.17}$$

Optimálnu hodnotu účelovej funkcie značíme ψ^*. Efektivita rozhodovacej jednotky sa následne interpretuje ako prevrátená hodnota $\frac{1}{\psi^*}$. Je zrejmé, že $\psi^* \in [1, \infty)$.

Aj pri výstupne orientovanom modeli overujeme pomocou druhej fázy, či sa jedná o

efektivitu alebo pseudoefektivitu daného útvaru.

$$\max_{\lambda, s^x, s^y} \quad \sum_{i=1}^{m} s_i^x + \sum_{r=1}^{s} s_r^y \tag{3.18}$$

$$\sum_{j=1}^{n} x_j \lambda_j + s^x = x_o, \tag{3.19}$$

$$\sum_{j=1}^{n} y_j \lambda_j - s^y = \psi^* y_o, \tag{3.20}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \quad s^x \geq 0, \quad s^y \geq 0 \tag{3.21}$$

$$\lambda_j \in \{0, 1\}, \qquad j = 1, .., n. \tag{3.22}$$

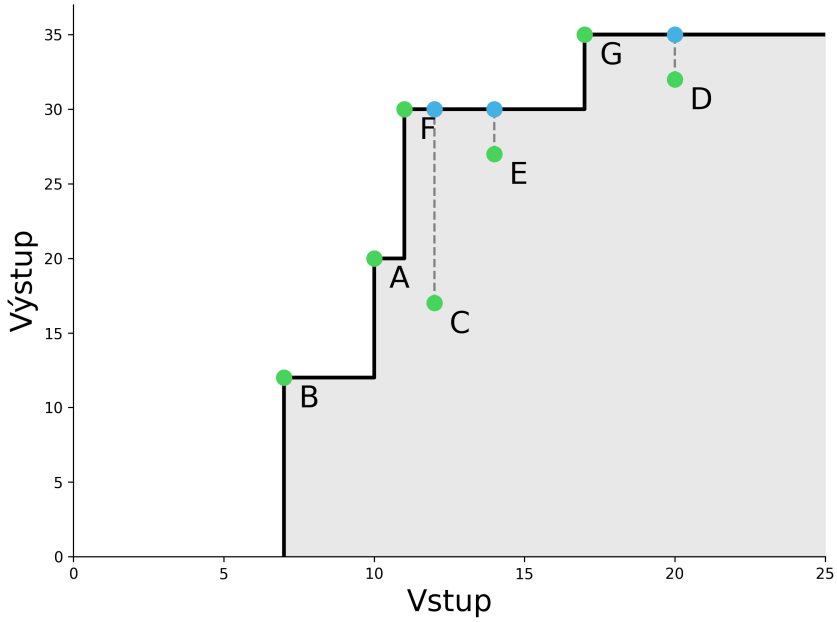
Aj v tomto prípade platí, že ak je optimálna hodnota účelovej funkcie nulová, tak pozorovaná jednotka je efektívna. V opačnom prípade, kedy je optimálna hodnota účelovej funkcie kladná, hovoríme o pseudoefektivite.

Príklad 3.3. *V tomto príklade budeme pracovať s rovnakými dátami, ako sme pracovali v príklade 3.2. Avšak tentokrát na tieto dáta aplikujeme výstupne orientovaný model. Výsledky sú zobrazené v nasledujúcej tabuľke*

DMU	θ^*	λ_A	λ_B	λ_C	λ_D	λ_E	λ_F	λ_G	s^x	s^y
A	1	1	0	0	0	0	0	0	0	0
B	1	0	1	0	0	0	0	0	0	0
C	0.5667	0	0	0	0	0	1	0	1	0
D	0.9143	0	0	0	0	0	0	1	3	0
E	0.9	0	0	0	0	0	1	0	3	0
F	1	0	0	0	0	0	1	0	0	0
G	1	0	0	0	0	0	0	1	0	0

Tabuľka 5: Výsledky po aplikovaní FDH-O

Na obrázku 3 je znázornená množina M_{FDH} spolu s projekciami neefektívnych útvarov po aplikovaní výstupne orientovaného modelu FDH-O. V množine je zobrazených sedem rozhodovacích jednotiek A-G. Zelené body predstavujú pôvodné pozície týchto jednotiek. Niektoré z nich (A, B, F, G) ležia priamo na hranici množiny M_{FDH}, teda sú efektívne. Ostatné jednotky (C, D, E) sa nachádzajú vo vnútri množiny a sú teda neefektívne. Pre každú neefektívnu jednotku je na obrázku zobrazená aj jej projekcia na hranicu množiny -



Obr. 3: Množina M_{FDH} a projekcie neefektívnych bodov po aplikovaní FDH-O

modré body. Tieto projekcie ilustrujú, ako by daná jednotka musela zvýšiť svoje výstupy pri zachovaní pôvodných vstupov, aby dosiahla efektívnosť podľa modelu FDH-O.

4 DEA modely nad zmiešanými pomerovými aj objemovými mierami

Keďže ani jeden z alternatívnych prístupov diskutovaných v kapitole 2.3 našej práce neumožňuje plnohodnotné a presné začlenenie pomerových mier do analýzy efektivity, autori Olesen a kolektív vo svojom článku [5] predstavili model, ktorý umožňuje priamu integráciu pomerových mier bez potreby ich transformácie. Tento model, označovaný ako R-VRS model, poskytuje teoretický základ na používanie pomerových mier v rámci DEA a umožňujú kombináciu objemových a pomerových mier ako natívnych typov dát.

4.1 Vstupne orientovaný model R-VRS

Ako prvý predstavíme vstupne orientovaný model R-VRS. Tento model sa zameriava na minimalizáciu vstupov pri zachovaní aktuálnych výstupov rozhodovacej jednotky $DMU_o = (x_o, y_o)$. V štandardných vstupných radiálnych modeloch sa faktor radiálneho skrátenia θ aplikuje výhradne na objemové vstupy, zatiaľ čo pomerové miery a objemové výstupy slúžia ako kontrolné premenné. Tieto pomerové miery sa nemenia, pretože často reprezentujú vonkajšie podmienky alebo kvalitatívne charakteristiky, ktoré nie sú priamo ovplyvniteľné manažmentom, ale môžu mať významný vplyv na hodnotenie efektívnosti rozhodovacích jednotiek.

Vo všeobecnosti však môžeme uvažovať aj o súčasnom skrátení pomerových mier, čo posúva tradičný význam radiálnej efektívnosti na širšiu úroveň. Faktor radiálneho skrátenia θ v tomto prípade nemusí byť obmedzený len na objemové vstupy, ale môžeme ho použiť ako univerzálnu mieru efektívnosti, ktorú aplikujeme na ľubovoľnú podmnožinu vstupov. Takto rozšírená interpretácia prináša väčšiu flexibilitu pri hodnotení efektívnosti, najmä v situácii, kde sa v analýze kombinujú objemové vstupy a výstupy. V práci [5] je definovaný vektor $x_o(\theta)$, ktorý umožňuje aplikovať faktor θ na vybranú podmnožinu vstupov $I^\theta \subseteq I$. Tento vektor má nasledujúci predpis:

$$x_{oi}(\theta) = \begin{cases} \theta x_{oi}, & \text{ak } i \in I^\theta, \\ x_{oi}, & \text{ak } i \in I \setminus I^\theta. \end{cases}$$

Takáto formulácia umožňuje flexibilne pristupovať k meraniu efektívnosti, pričom zo-

hľadňuje špecifiká jednotlivých vstupov. Výber podmnožiny I^θ môže byť motivovaný rôznymi faktormi, ako je dostupnosť údajov, význam vstupov pre konkrétnu analýzu alebo záujem o zlepšenie iba určitého typu vstupov. Vstupne orientovaný R-VRS model pre $(x_o, y_o) \in Z$ je formulovaný nasledovne:

$$(R-VRS-I)_o: \quad \min_{\theta,\lambda} \quad \theta \tag{4.1}$$

$$\sum_{j=1}^{n} x_j^V \lambda_j \leq x_o^V(\theta), \tag{4.2}$$

$$\sum_{j=1}^{n} y_j^V \lambda_j \geq y_o^V, \tag{4.3}$$

$$\lambda_j(x_j^R - x_o^R(\theta)) \leq 0, \quad j = 1, ..., n \tag{4.4}$$

$$\lambda_j(y_j^R - y_o^R) \geq 0, \quad j = 1, ..., n \tag{4.5}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \quad \lambda \geq 0_n. \tag{4.6}$$

Na základe definície $x_{oi}(\theta)$, môžeme pozorovať, že ak i-ty vstup je pomerová miera a zároveň $i \in I^\theta$, tak ohraničenie (4.4) je nelineárne, pretože v ňom dostávame súčin dvoch premenných λ_j a θ:

$$\lambda_j(x_j^R - \theta x_o^R) \leq 0, \quad j = 1, ..., n \tag{4.7}$$

Podľa [5], toto nelineárne ohraničenie môžeme linearizovať pomocou tzv. „buď - alebo“ podmienok. Aby (4.7) platila musí byť pre každé $j = 1, ..., n$ splnená minimálne jedna z nasledujúcich podmienok:

$$\lambda_j = 0 \qquad alebo \qquad x_j^R - x_o^R(\theta) \leq 0. \tag{4.8}$$

Toto vieme dosiahnuť pomocou zavedenia novej binárnej premennej $z_j \in \{0, 1\}$ a dostatočne veľkej konštanty M. Takto nahradíme n nelineárnych ohraničení (4.7) pomocou $2n$ lineárnych a n binárnych:

$$\lambda_j \leq z_j, \qquad j = 1, ..., n \tag{4.9}$$

$$x_j^R - x_o^R(\theta) \leq M(1 - z_j), \qquad j = 1, ..., n \tag{4.10}$$

$$z_j \in \{0, 1\}, \qquad j = 1, ..., n. \tag{4.11}$$

Konštantu M zvolíme tak, aby bola čo najmenšia možná pričom by v prípade $z_j = 0$ zaručovala, že ohraničenie (4.10) je vždy splnené. Ak je $z_j = 0$ ohraničenie (4.9) je aktívne (dáva $\lambda_j = 0$) a ohraničenie (4.10) je vždy pravdivé, pretože konštantu M sme zvolili dostatočne veľkú. V tomto prípade nám ohraničenie (4.9) spolu s podmienkou (4.6) implikuje $\lambda_j = 0$. V prípade, že je $z_j = 1$, nerovnosť (4.9) podľa podmienok (4.6) platí vždy a aktívna je nerovnosť (4.10).

Zavedením nových lineárnych ohraničení namiesto nelineárneho ohraničenia (4.4) dostaneme upravenú formuláciu $R - VRS - I$ modelu:

$$(R-VRS-I)_o: \quad \min_{\theta,\lambda,z} \quad \theta \tag{4.12}$$

$$\sum_{j=1}^{n} x_j^V \lambda_j \leq x_o^V(\theta), \tag{4.13}$$

$$\sum_{j=1}^{n} y_j^V \lambda_j \geq y_o^V, \tag{4.14}$$

$$\lambda_j \leq z_j, \quad j = 1, ..., n \tag{4.15}$$

$$(x_j^R - x_o^R(\theta)) \leq M(1 - z_j), \quad j = 1, ..., n \tag{4.16}$$

$$\lambda_j (y_j^R - y_o^R) \geq 0, \quad j = 1, ..., n \tag{4.17}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \quad \lambda \geq 0_n \tag{4.18}$$

$$z_j \in \{0, 1\}, \quad j = 1, ..., n. \tag{4.19}$$

Takto naformulovaný model, ktorý obsahuje binárnu premennú z_j, je úlohou zmiešaného celočíselného programovania.

Je zrejmé, že optimálna hodnota $\theta^* \leq 1$. Táto hodnota predstavuje mieru radiálnej (pseudo)efektivity útvaru. To, či ide o efektivitu alebo pseudoefektivitu možno zistiť aplikovaním druhej fázy, v ktorej maximalizujeme súčet zložiek slackov s_m^x (pre vstupy) a s_s^y (pre výstupy). V rámci nášho problému si tieto vektory rozdelíme ešte na zložky zodpovedajúce objemovým a pomerovým premenným: $s^x = (s^{xV}, s^{xR})$, $s^y = (s^{yV}, s^{yR})$.

Podľa článku [6] má druhá fáza pre vstupne orientovaný model nasledujúcu formuláciu:

$$\max_{\lambda,s^x,s^y} \quad \sum_{i=1}^{m} s_i^x + \sum_{r=1}^{s} s_r^y \tag{4.20}$$

$$\sum_{j=1}^{n} x_j^V \lambda_j + s^{xV} \leq x_o^V(\theta^*), \tag{4.21}$$

$$\sum_{j=1}^{n} y_j^V \lambda_j - s^{yV} \geq y_o^V, \tag{4.22}$$

$$\lambda_j(x_j^R - (x_o^R(\theta^*) - s^{xR})) \leq 0, \quad j = 1, ..., n \tag{4.23}$$

$$\lambda_j(y_j^R - (y_o^R + s^{yR})) \geq 0, \quad j = 1, ..., n \tag{4.24}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \quad \lambda \geq 0_n, \quad s^x \geq 0, \quad s^y \geq 0. \tag{4.25}$$

Podobne ako v prvej fáze, aj v druhej fáze sa objavili nelineárne ohraničenia (4.23), kde sa násobí λ_j so slackom s^{xR} a (4.24), v ktorom sa násobia dve premenné λ_j a s^{yR}. Analogicky, ako pri prvej fáze, sa podľa [6] týchto ohraničení zbavíme pomocou tzv. „buď - alebo“ podmienok a dostaneme upravenú formuláciu druhej fázy, ktorá už obsahuje len lineárne ohraničenia:

$$\max_{\lambda,s^x,s^y,z} \quad \sum_{i=1}^{m} s_i^x + \sum_{r=1}^{s} s_r^y \tag{4.26}$$

$$\sum_{j=1}^{n} x_j^V \lambda_j + s^{xV} \leq x_o^V(\theta^*), \tag{4.27}$$

$$\sum_{j=1}^{n} y_j^V \lambda_j - s^{yV} \geq y_o^V, \tag{4.28}$$

$$\lambda_j \leq z_j, \quad j = 1, ..., n \tag{4.29}$$

$$(x_j^R - (x_o^R(\theta^*) - s^{xR})) \leq M_1(1 - z_j), \quad j = 1, ..., n \tag{4.30}$$

$$(y_j^R - (y_o^R + s^{yR})) \geq -M_2(1 - z_j), \quad j = 1, ..., n \tag{4.31}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \quad \lambda \geq 0_n, \quad s^x \geq 0, \quad s^y \geq 0, \tag{4.32}$$

$$z_j \in \{0, 1\}, \quad j = 1, ..., n, \tag{4.33}$$

kde M_1, M_2 sú dostatočne veľké konštanty, aby ohraničenia (4.30), (4.31) platili vždy v prípade $z_j = 0$.

Podľa [3], ak v prvej fáze dostaneme optimálnu hodnotu $\theta^* = 1$ a v druhej fáze sú optimálne slacky nulové, tak útvar je efektívny a θ^* je jeho efektivitou. Ak je $\theta^* \in (0, 1)$ a slacky sú nulové, tak skúmaný útvar je neefektívny a θ^* je jeho efektivitou. Ak je $\theta^* = 1$ a slacky sú nenulové, tak je útvar pseudoefektívny (neefektívny so pseudoefektivitou $\theta^* = 1$). Ak je $\theta^* \in (0, 1)$ a slacky sú nenulové, tak skúmaný útvar je neefektívny a θ^* je jeho pseudoefektivitou.

Príklad 4.1. *V tomto príklade budeme pracovať s hypotetickými údajmi desiatich DMU. Tieto dáta sú zobrazené v tabuľke 6. Ku každému útvaru máme informácie o jednom objemovom vstupe, jednom objemovom výstupe a jednom pomerovom výstupe. Ná dáta následne aplikujeme R-VRS-I model.*

DMU	A	B	C	D	E	F	G	H	I	J
x^V	1	2	1	2	5	2	5	5	5	4
y^V	2.5	4	1	4	4	1	4	1	1	2.5
y^R	0.5	1.5	0.5	0.5	0.5	1.5	1.5	1.5	0.5	1

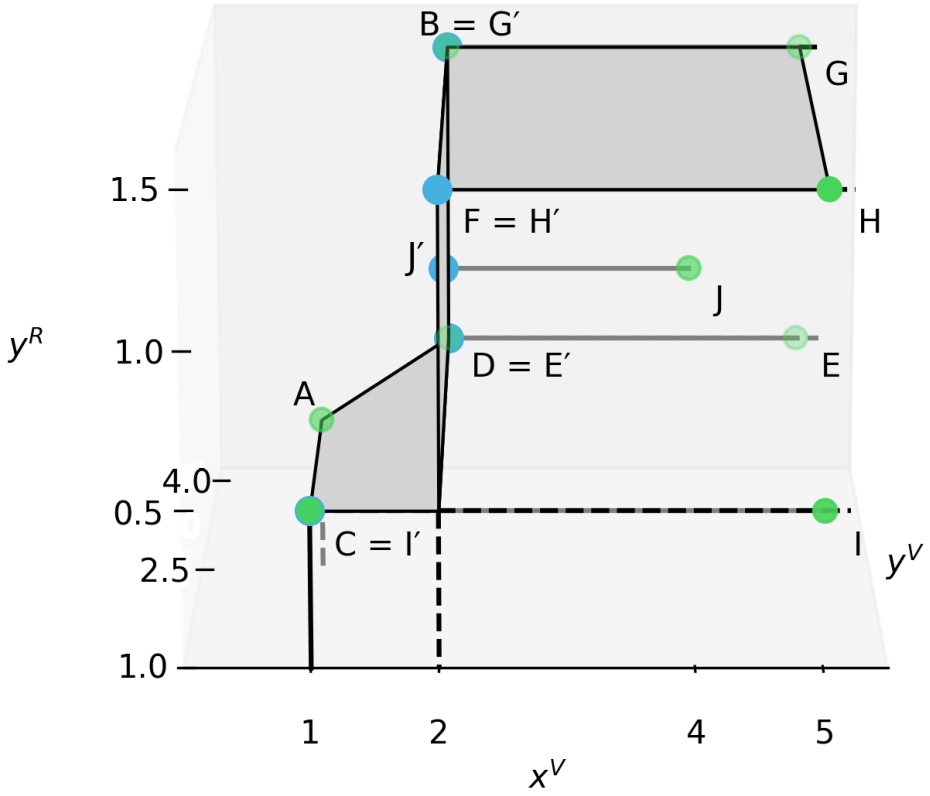
Tabuľka 6: Hypotetické hodnoty vstupov a výstupov desiatich producentov

Efektivity a výsledky po aplikovaní modelu sú zobrazené v tabuľke 8.

DMU	θ^*	λ_A	λ_B	λ_C	λ_D	λ_E	λ_F	λ_G	λ_H	λ_I	λ_J	s^x	s_1^y	s_2^y
A	1	1	0	0	0	0	0	0	0	0	0	0	0	0
B	1	0	1	0	0	0	0	0	0	0	0	0	0	0
C	1	1	0	0	0	0	0	0	0	0	0	0	1.5	0
D	1	0	1	0	0	0	0	0	0	0	0	0	0	1
E	0.4	0	1	0	0	0	0	0	0	0	0	0	0	1
F	1	0	1	0	0	0	0	0	0	0	0	0	3	0
G	0.4	0	1	0	0	0	0	0	0	0	0	0	0	0
H	0.4	0	1	0	0	0	0	0	0	0	0	0	3	0
I	0.2	1	0	0	0	0	0	0	0	0	0	0	1.5	0
J	0.5	0	1	0	0	0	0	0	0	0	0	0	1.5	0.5

Tabuľka 7: Výsledky po aplikovaní R-VRS-I

Obrázok 4 zobrazuje množinu M_{VRS}^R, ktorá reprezentuje produkčné možnosti za predpokladu variabilných výnosov z rozsahu, pričom vstupy a výstupy sú rozdelené na objemové a pomerové. V tomto 3D priestore sú jednotlivé rozhodovacie jednotky znázornené zelenými bodmi, ktoré sa nachádzajú buď na hranici množiny produkčných možností,



Obr. 4: Množina M^R_{VRS} a projekcie neefektívnych bodov po aplikovaní R-VRS-I

alebo pod touto hranicou. Pri neefektívnych jednotkách sú zobrazené aj ich projekcie na hranicu množiny. Keďže ide o vstupne orientovaný model, vstupy sa znižujú (projektovanie v smere osi x^V) , zatiaľ čo hodnoty výstupov - či už objemové y^V alebo pomerové y^R- zostávajú nezmenené.

4.2 Výstupne orientovaný model R-VRS

Analogicky ako pri vstupne orientovanom modeli R-VRS, kde sme definovali zlepšený vektor vstupov $x_o(\theta)$, si pre výstupne orientovaný model R-VRS definujeme vektor $y_o(\psi)$. Tento vektor umožňuje aplikovať faktor zlepšenia ψ na vybranú podmnožinu výstupov $O^\psi \subseteq O$, zatiaľ čo ostatné výstupy zostávajú nezmenené. Formálne je vektor $y_o(\psi)$ definovaný nasledovne:

$$y_{or}(\psi) = \begin{cases} \psi y_{or}, & \text{ak } r \in O^\psi, \\ y_{or}, & \text{ak } r \in O \setminus O^\psi. \end{cases}$$

Podobne ako pri vektore $x_o(\theta)$, aj v tomto prípade výber podmnožiny O^ψ poskytuje flexibilitu v analýze. Tento prístup je užitočný najmä pri kombinácii objemových a pomerových výstupov, kde môže byť potrebné zvýrazniť konkrétne výstupy alebo merať efektívnosť len voči ich vybranej podskupine.

V tomto prípade však efektivitu rozhodovacej jednotky vyjadrujeme prostredníctvom prevrátenej hodnoty optimálneho ψ, čo znamená, že efektivita sa rovná $\frac{1}{\psi^*}$, kde ψ^* je optimálna hodnota.

Úloha výstupne orientovaného R-VRS modelu má tvar:

$$(R-VRS-O)_o: \quad \max_{\psi,\lambda} \quad \psi \tag{4.34}$$

$$\sum_{j=1}^{n} x_j\lambda_j \leq x_o^V, \tag{4.35}$$

$$\sum_{j=1}^{n} y_j\lambda_j \geq y_o^V(\psi), \tag{4.36}$$

$$\lambda_j(x_j^R - x_o^R) \leq 0, \quad j = 1, ..., n \tag{4.37}$$

$$\lambda_j(y_j^R - y_o^R(\psi)) \geq 0, \quad j = 1, ..., n \tag{4.38}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \quad \lambda \geq 0_n. \tag{4.39}$$

V prípade, že do množiny O^ψ patria aj pomerové miery, dostávame v modeli $R-VRS-O$ nelineárne ohraničenie (4.38). Analogicky, ako pri vstupne orientovanom modeli $R-VRS-I$, toto nelineárne ohraničenie nahradíme dvomi lineárnymi zavedením novej binárnej premennej $z_j \in \{0, 1\}$ a konštanty M:

$$(R-VRS-O)_o: \quad \max_{\psi,\lambda} \quad \psi \tag{4.40}$$

$$\sum_{j=1}^{n} x_j\lambda_j \leq x_o^V, \tag{4.41}$$

$$\sum_{j=1}^{n} y_j\lambda_j \geq y_o^V(\psi), \tag{4.42}$$

$$\lambda_j(x_j^R - x_o^R) \leq 0, \quad j = 1, ..., n \tag{4.43}$$

$$\lambda_j \leq z_j, \quad j = 1, ..., n \tag{4.44}$$

$$y_j^R - y_o^R(\psi) \geq -M(1 - z_j), \quad j = 1, ..., n \tag{4.45}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \quad \lambda \geq 0_n \tag{4.46}$$

$$z_j \in \{0, 1\}, \quad j = 1, ..., n. \tag{4.47}$$

Môžeme pozorovať, že ak je $z_j = 0$, tak nutne $\lambda_j = 0$ a ohraničenie (4.45) platí vždy, pretože M je dostatočne veľké. Ak je $z_j = 1$, tak na základe podmienok (4.46) ohraničenie (4.44) platí vždy a rozdiel $y_j^R - y_o^R(\psi)$ musí byť nezáporný. Zavedením binárnej premennej z_j sme dostali úlohu zmiešaného celočíselného programovania. Je zrejmé, že optimálna hodnota $\psi^* \geq 1$. Jej prevrátená hodnota predstavuje mieru (pseudo)efektivity skúmaného útvaru. Analogicky, ako pri vstupne orientovanom modeli, zistíme, či ide o efektivitu alebo pseudoefektivitu pomocou druhej fázy. Olesen a kolektív v článku [6] formulovali úlohu druhej fázy pre výstupne orientovaný model nasledovne:

$$\max_{\lambda, s^x, s^y} \quad \sum_{i=1}^{m} s_i^x + \sum_{r=1}^{s} s_r^y \tag{4.48}$$

$$\sum_{j=1}^{n} x_j^V \lambda_j + s^{xV} \leq x_o^V, \tag{4.49}$$

$$\sum_{j=1}^{n} y_j^V \lambda_j - s^{yV} \geq y_o^V(\psi^*), \tag{4.50}$$

$$\lambda_j(x_j^R - (x_o^R - s^{xR})) \leq 0, \quad j = 1, ..., n \tag{4.51}$$

$$\lambda_j(y_j^R - (y_o^R(\psi^*) + s^{yR})) \geq 0, \quad j = 1, ..., n \tag{4.52}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \quad \lambda \geq 0_n, \quad s^x \geq 0, \quad s^y \geq 0. \tag{4.53}$$

Ako pri vstupne orientovanom modeli, tak aj pri výstupne orientovanom modeli vzniká v ohraničeniach (4.51) a (4.52) súčin dvoch premenných. Týchto nelineárnych ohraničení sa zbavíme zavedením binárnej premennej z_j a dostatočne veľkých konštánt M_1, M_2:

$$\max_{\lambda, s^x, s^y, z} \quad \sum_{i=1}^{m} s_i^x + \sum_{r=1}^{s} s_r^y \tag{4.54}$$

$$\sum_{j=1}^{n} x_j^V \lambda_j + s^{xV} \leq x_o^V, \tag{4.55}$$

$$\sum_{j=1}^{n} y_j^V \lambda_j - s^{yV} \geq y_o^V(\psi^*), \tag{4.56}$$

$$\lambda_j \leq z_j, \quad j = 1, ..., n \tag{4.57}$$

$$(x_j^R - (x_o^R - s^{xR})) \leq M_1(1 - z_j), \quad j = 1, ..., n \tag{4.58}$$

$$(y_j^R - (y_o^R(\psi^*) + s^{yR})) \geq -M_2(1 - z_j), \quad j = 1, ..., n \tag{4.59}$$

$$\sum_{j=1}^{n} \lambda_j = 1, \quad \lambda \geq 0_n, \quad s^x \geq 0, \quad s^y \geq 0 \tag{4.60}$$

$$z_j \in \{0, 1\}, \quad j = 1, ..., n. \tag{4.61}$$

Ak nám v prvej fáze modelu vyšla optimálna hodnota $\psi^* > 1$, sledovaný útvar je neefektívny a pomocou druhej fázy určíme, či je ψ^* jeho efektivitou alebo pseudoefektivitou. Ak v druhej fáze dostaneme nulové slacky, tak sa jedná o efektivitu. Ak sú slacky nenulové, tak ide o pseudoefektivitu. Pokiaľ v prvej fáze modelu dostaneme $\psi^* = 1$, tak je sledovaný útvar buď efektívny (slacky v druhej fáze sú nulové) alebo neefektívny a ψ^* je jeho pseudoefektivitou (slacky v druhej fáze sú nenulové).

Príklad 4.2. *V tomto príklade budeme pracovať s hypotetickými údajmi desiatich DMU. Tieto dáta sú zobrazené v tabuľke 6. Ku každému útvaru máme informácie o jednom objemovom vstupe, jednom objemovom výstupe a jednom pomerovom výstupe. Ná dáta následne aplikujeme R-VRS-I model.*

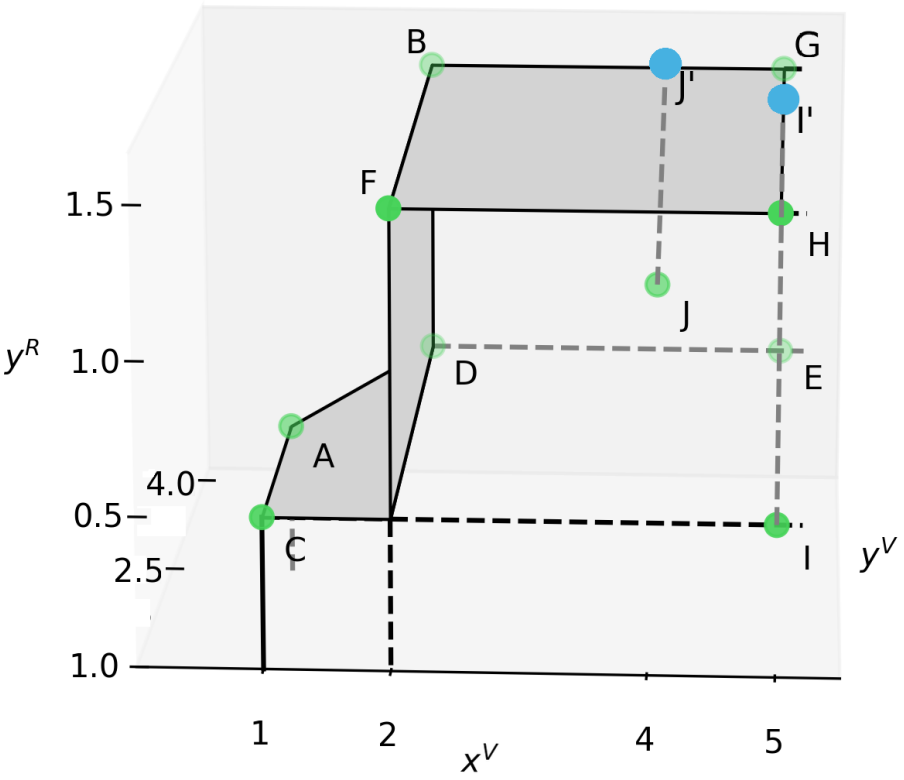
DMU	A	B	C	D	E	F	G	H	I	J
x^V	1	2	1	2	5	2	5	5	5	4
y^V	2.5	4	1	4	4	1	4	1	1	2.5
y^R	0.5	1.5	0.5	0.5	0.5	1.5	1.5	1.5	0.5	1

Tabuľka 8: Hypotetické hodnoty vstupov a výstupov desiatich producentov

Efektivity a výsledky po aplikovaní modelu sú zobrazené v tabuľke 9.

DMU	θ^*	λ_A	λ_B	λ_C	λ_D	λ_E	λ_F	λ_G	λ_H	λ_I	λ_J	s^x	s_1^y	s_2^y
A	1	1	0	0	0	0	0	0	0	0	0	0	0	0
B	1	0	1	0	0	0	0	0	0	0	0	0	0	0
C	1	1	0	0	0	0	0	0	0	0	0	0	1.5	0
D	1	0	1	0	0	0	0	0	0	0	0	0	0	1
E	1	0	1	0	0	0	0	0	0	0	0	3	0	1
F	1	0	1	0	0	0	0	0	0	0	0	0	3	0
G	1	0	1	0	0	0	0	0	0	0	0	3	0	0
H	1	0	1	0	0	0	0	0	0	0	0	3	3	0
I	0.3333	0	1	0	0	0	0	0	0	0	0	3	1	0
J	0.6667	0	1	0	0	0	0	0	0	0	0	2	0.25	0

Tabuľka 9: Výsledky po aplikovaní R-VRS-O



Obr. 5: Množina M_{VRS}^R a projekcie neefektívnych bodov po aplikovaní R-VRS-O

Obrázok 5 znázorňuje množinu M_{VRS}^R a projekcie neefektívnych jednotiek (I, J) po aplikovaní výstupne orientovaného DEA modelu R-VRS-O. Cieľom je maximalizovať výstupy pri daných vstupoch. Zelené body v grafe predstavujú pôvodné rozhodovacie jednotky, zatiaľ čo modré body predstavujú ich projekcie na hranicu množiny po optimalizácii výstupov, pričom vstupy zostali nezmenené.

V tabuľke 9 vidíme, že jednotky A, B, C, D, E, F, G, H majú efektivitu rovnú jednej. Avšak len jednotky A a B majú nulové slacky, teda len tieto dve jednotky sú efektívne a ostatné sú pseudoefektívne. Taktiež vidíme, že neefektívna jednotka J má $\theta^* = 0.6667$, čo znamená, že by musela zvýšiť svoje výstupy 1,5-násobne, aby sa dostala na hranicu množiny. Táto jednotka má však zároveň nenulové slacky, teda po predĺžení výstupov by sa stále nachádzala na hranici pseudoefektívnosti. Aby bola J efektívna musela by, okrem zvýšenia výstupov 1.5-násobne, zvýšiť vstup o 2 a objemový výstup o 0.25.

5 Prehľad literatúry

V tejto kapitole sa venujeme prehľadu relevantných prác, ktoré rozvíjajú problematiku použitia pomerových ukazovateľov v obálkovej analýze dát. Štandardné DEA modely totiž vychádzajú z predpokladu konvexnosti, ktorý nie je vždy kompatibilný s pomerovými mierami, ako sme vysvetlili v kapitole 2. Z tohto dôvodu sme do prehľadu zahrnuli najmä články, ktoré rozvíjajú metodiky na spracovanie pomerových údajov v rámci obálkovej analýzy dát, čím reflektujú ich častý výskyt v praktických aplikáciách.

Jedným z kľúčových teoretických pilierov, na ktorých staviame našu prácu je článok Olesena a kol. [5] z roku 2015, ktorý predstavuje systematický prístup k analýze efektivity tam, kde vstupy a výstupy sú dostupné v podobe pomerových ukazovateľov. Autori upozorňujú, že používanie takýchto veličín - ako sú napríklad percentuálne podiely, priemery alebo miery úspešnosti - je v štandardných modeloch problematické. Autori preto predstavujú novú technológiu Ratio-VRS (R-VRS) pre variabilné výnosy z rozsahu. Hlavnou inováciou v tomto modeli je tzv. selektívna konvexnosť, ktorá umožňuje vytvárať konvexné kombinácie iba v priestore objemových ukazovateľov, zatiaľ čo pomerové miery sa neagregujú bežným spôsobom. Tento prístup umožňuje do modelu zahrnúť aj pomerové miery bez toho, aby dochádzalo k narušeniu základných predpokladov DEA. Na rozdiel od štandardných DEA modelov, kde sa pomerové ukazovatele interpolujú lineárne, R-VRS rešpektuje to, že takéto interpolácie môžu byť matematicky alebo logicky nekorektné. V práci sú navyše predstavené ilustračné príklady a znázornenia, ktoré jasne ukazujú, ako sa R-VRS technológia líši od klasickej VRS.

Popri R-VRS modeli autori uvádzajú aj R-CRS technológiu, ktorá vychádza z podobných princípov, ale predpokladá konštantné výnosy z rozsahu a umožňuje proporcionálne škálovanie objemových údajov. Model R-CRS je obzvlášť vhodný v prípadoch, keď sú pomerové miery považované za fixné, teda sa nemenia pri zmene veľkosti výroby alebo poskytovaných služieb.

Tento článok predstavuje jeden z hlavných teoretických základov, na ktorých staviame našu prácu. Autori identifikovali problémy pri použití pomerových ukazovateľov v štandardných DEA modeloch a navrhli alternatívne technologické prístupy. V našej práci preto nadväzujeme na myšlienky Olesena a kol., pričom sa sústreďujeme najmä na analýzu R-VRS technológie a jej aplikáciu.

Na model R-VRS, ako ho sformulovali Olesen a kol. v článku [5], priamo nadväzuje štúdia [8] z roku 2023, ktorá túto technológiu ďalej rozvíja a významne rozširuje jej aplikačný potenciál. Autori sa zameriavajú na nedostatok pôvodného modelu, ktorým je predpoklad úplnej neznalosti čitateľov a menovateľov pomerových ukazovateľov. Teda predpoklad, že o objemových ukazovateľoch, z ktorých vznikli pomerové miery, nemáme k dispozícii žiadne dodatočné informácie. V praxi však často existuje aspoň čiastočné poznanie rozsahov, v ktorých sa tieto hodnoty môžu pohybovať. Autori preto zavádzajú model, ktorý umožňuje integrovať do pôvodného R-VRS modelu dolné a horné ohraničenia buď na čitatele, menovatele, alebo oboje, čím dostávajú všeobecnejšiu verziu produkčnej technológie. Tento prístup reflektuje bežné praktické situácie, kedy sú k dispozícii agregované alebo rámcové údaje - napríklad je známe, že škola má medzi 100 a 200 žiakmi, aj keď nie je k dispozícii presné číslo za daný rok. Na základe týchto informácií je potom možné určiť rozsah, v ktorom sa daný pomerový ukazovateľ môže reálne nachádzať. Na rozdiel od pôvodného R-VRS modelu, ktorý vychádzal z predpokladu úplnej neznalosti, tento model umožňuje jemnejšie odstupňovanie miery neistoty a tým zvyšuje presnosť a diskriminačnú schopnosť hodnotenia. Veľkou výhodou tohto rozšírenia je, že aj napriek jeho väčšej flexibilite a komplexnosti možno technológiu vyjadriť pomocou linearizovaných podmienok. V článku je celý prístup navyše ilustrovaný na aplikačnom príklade z oblasti hodnotenia škôl, kde sú mnohé údaje dostupné len v pomerovej forme a objemové základy ratio measures sú známe len približne.

Na teoretické modely R-VRS a R-CRS nadväzuje aj článok [6] z roku 2017, ktorého autormi sú Olesen, Petersen a Podinovski. Tento článok sa zameriava na výpočtové aspekty modelov pracujúcich s pomerovými ukazovateľmi. Kým v práci z roku 2015 autori položili axiomatické základy týchto technológií, v tejto štúdii sa venujú praktickému problému, ako efektívnosť v rámci týchto modelov vypočítať. Štandardné výpočtové postupy používané v klasických modeloch obálkovej analýzy dát totiž nie sú priamo aplikovateľné, keďže technológia R-VRS nie je konvexná a neumožňuje lineárne interpolácie pomerových veličín. Autori preto navrhujú alternatívne výpočtové prístupy v R-VRS a R-CRS technológiách.

Najskôr je potrebné vybrať skupinu porovnateľných jednotiek, ktoré môžu slúžiť ako referencia pre hodnotenú jednotku. Do tejto skupiny patria len tie jednotky, ktorých po-

merové vstupy nie sú vyššie a pomerové výstupy nie sú nižšie než u hodnotenej jednotky. Takto vytvorená skupina tvorí tzv. zúženú technológiu, nad ktorou sa potom optimalizuje iba s použitím objemových údajov. Tento postup zachováva logiku DEA, no zároveň umožňuje použiť jednoduché lineárne programovanie, keďže pomerové miery sa nemenia a berú sa ako pevné.

Ďalej autori pracujú s rôznymi spôsobmi merania efektivity. Okrem tradičného konceptu radiálnej efektívnosti, založeného na rovnomernom škálovaní všetkých vstupov alebo výstupov, analyzujú aj neradiálne prístupy, ktoré umožňujú hodnotiť zmeny jednotlivých vstupov a výstupov osobitne.

Jedným z hlavných prínosov tejto štúdie je zavedenie konceptu potencionálnej neefektívnosti na základe pomerových údajov. Tento jav vzniká vtedy, keď je jednotka v R-VRS (resp. R-CRS) hodnotená ako efektívna, avšak táto efektívnosť by mohla byť spochybnená, ak by boli známe konkrétne hodnoty čitateľov a menovateľov jej pomerových ukazovateľov. Autori preto navrhujú, ako testovať túto skrytú neefektívnosť prostredníctvom doplňujúcich výpočtov.

Dôležitou súčasťou výpočtového rámca je aj druhá fáza optimalizácie, ktorá nadväzuje na primárny výpočet efektivity. Podobne ako v štandardných modeloch obálkovej analýzy dát, kde po identifikácii projekcie na hranici technológie nasleduje druhý krok s cieľom rozlíšiť efektívne a pseudoefektívne jednotky, aj v prípade R-VRS (resp. R-CRS) autori odporúčajú realizovať druhú fázu výpočtu. Tento krok spočíva v riešení dodatočnej optimalizačnej úlohy. V článku je tento postup demonštrovaný prostredníctvom rozšírenej optimalizačnej úlohy, kde sa v druhej etape maximalizuje súčet slackov, pričom sa zachováva hodnota efektivity vypočítaná v prvej fáze.

Túto druhú fázu optimalizácie sme aplikovali aj v našej práci pri riešení vstupne a výstupne orientovaných R-VRS modelov. Po vypočítaní primárnej efektivity sme v druhej fáze maximalizovali súčet slackov pri zachovaní hodnoty efektivity z prvej fázy. Formulované úlohy sme upravili do podoby zmiešaného celočíselného programovania a následne riešili pomocou solvera Gurobi v prostredí Python.

Článok [7] sa venuje analýze škálovacích vlastností produkčnej technológie R-VRS. Táto štúdia Olesena a kol. z roku 2022 sa sústreďuje na otázku, ako možno v prostredí pomerových mier definovať a hodnotiť výnosy z rozsahu a identifikovať najproduktívnejšiu

mieru výroby. V klasických DEA modeloch sa škálovacie vlastnosti analyzujú na základe geometrie technológie, no R-VRS model je štrukturovaný inak kvôli pomerovým ukazovateľom. Preto nie je možné priamo použiť bežné postupy. Autori preto zavádzajú nový postup - tzv. čiastočné škálovanie, kde je možné meniť len objemové vstupy a výstupy, zatiaľ čo pomerové miery zostávajú fixné. Zaujímavé je, že ich metóda umožňuje zamerať sa len na vybrané vstupy a výstupy. Napríklad je možné skúmať návratnosť z rozsahu iba na základe počtu zamestnancov a výdavkov, a nebrať do úvahy kvalitatívne údaje ako napríklad percento žiakov s osobitnými potrebami. Toto je veľmi užitočné najmä v prípadoch, keď sú pomerové údaje dané alebo nemenné.

Okrem toho autori rozlišujú dva spôsoby, ako sa na výnosy z rozsahu pozerať – lokálne (teda priamo v okolí hodnotenej jednotky) a globálne (teda s ohľadom na celkový tvar technológie). V R-VRS modeli sa ukazuje, že lokálny prístup často neposkytuje dostatok informácií, preto uprednostňujú globálny pohľad, ktorý lepšie ukazuje, či by jednotka mohla zvýšiť alebo znížiť svoju mierku, aby sa priblížila k optimu.

Svoju metodológiu autori ilustrujú na príklade škôl v Anglicku. Ukazujú, ako možno zhodnotiť, či školy fungujú v správnej „veľkosti“ vzhľadom na počet zamestnancov a zdroje, pričom pomerové výstupy ako socioekonomické rozloženie žiakov zostávajú nezmenené.

Štúdia [10] autorov Podinovski, Wu, Argyris z roku 2024 sa venuje jednému z kľúčových problémov modelov R-VRS a R-CRS - konkrétne ich nedostatočnej diskriminačnej sile. Tento problém sa prejavuje najmä v prípadoch, keď je v modeli veľký počet pomerových ukazovateľov, ktoré znižujú variabilitu medzi jednotkami. V dôsledku toho sa viaceré jednotky javia ako efektívne, aj keď medzi nimi existujú významné výkonnostné rozdiely. Autori sa preto zameriavajú na posilnenie diskriminačnej sily týchto modelov prostredníctvom rozšírenia technológie o tzv. produkčné kompromisy.

Produkčné kompromisy predstavujú odborné alebo skúsenosťami podložené pravidlá, ktoré popisujú, aké zmeny vstupov a výstupov sú v reálnom svete technologicky možné. Môžu sa týkať buď objemových hodnôt (napr. počet učiteľov, financie), alebo pomerových ukazovateľov (napr. percento žiakov s dobrými výsledkami). Tieto kompromisy autori zakomponovali do dvoch existujúcich DEA modelov – R-VRS a R-CRS. Vznikli tak nové rozšírené verzie týchto modelov, ktoré nazvali R-VRS-TO a R-CRS-TO. Zavede-

nie kompromisov rozširuje možnosti modelu – umožňuje simulovať realistickejšie scenáre. V dôsledku toho niektoré jednotky, ktoré boli predtým považované za efektívne, už tak nevyzerajú. To vedie k presnejšiemu hodnoteniu výkonnosti a lepšiemu rozlíšeniu medzi jednotkami. Podobný prístup sa využíva aj v klasických DEA modeloch v rámci Assurance Region (AR) modelov [3], kde sa zavádzajú obmedzenia na váhy alebo na pomery vstupov a výstupov. Tieto produkčné kompromisy predstavujú obdobný mechanizmus.

Modely boli otestované na vzorke 891 stredných škôl v Anglicku. Tieto školy neprijímajú žiakov výberovo a poskytujú aj vyššie ročníky. Ako vstupy sa brali do úvahy počet učiteľov, výška výdavkov, podiel dobre pripravených žiakov a podiel žiakov zo sociálne silnejšieho prostredia. Ako výstupy sa hodnotil počet žiakov, úspešnosť na záverečných skúškach (GCSE) a podiel prijatých na popredné univerzity. Výsledky ukázali, že po zapracovaní kompromisov model lepšie rozlíšil medzi efektívnymi a neefektívnymi školami. Viac škôl získalo skóre nižšie ako 1, čo znamená, že ich efektívnosť bola realistickejšie vyhodnotená. Nové modely teda umožňujú presnejšie a prakticky využiteľnejšie hodnotenie, najmä v prípadoch, kde zohrávajú úlohu kvalitatívne alebo ťažko merateľné faktory, ktoré bežné DEA modely nevedia zachytiť.

V dostupnej literatúre sme nenašli aplikáciu superefektívnosti na modely pracujúce s pomerovými dátami, hoci ide o jeden z najjednoduchších spôsobov, ako zvýšiť rozlišovaciu schopnosť modelov. Superefektívnosť umožňuje lepšie odlíšiť rozhodovacie jednotky, ktoré by inak boli všetky klasifikované ako efektívne. V publikovaných prácach sa na tento účel najčastejšie využíva princíp Assurance Region, kde sa prostredníctvom dodatočných obmedzení na pomery váh zabezpečuje lepšia identifikácia efektívnych jednotiek. V našej práci sme sa rozhodli aplikovať superefektivitu, pričom jej hlavnou výhodou je jednoduchá implementácia bez potreby určovania hraníc pomerov. Nevýhodou je však skutočnosť, že v niektorých prípadoch môže dôjsť k situácii, keď optimalizačná úloha nie je dobre definovaná a výsledkom je nekonečné riešenie. Na druhej strane, prístup Assurance Region vyžaduje starostlivé nastavenie ohraničení, no zabezpečuje stabilné a dobre definované modely.

6 Experimenty na reálnych dátach

V poslednej kapitole sa venujeme praktickej aplikácii štandardných modelov a následne modelov R-VRS-I, R-VRS-O na nami získané dáta. Rozhodli sme sa pracovať s dátami slovenských a českých komerčných poisťovní, ktoré sme získali z ich výročných správ za rok 2023. Pri všetkých modeloch použijeme dvojfázovú metódu - v prvej fáze zistíme hodnotu (pseudo)efektivity a pomocou druhej fázy určíme, či sa jedná o efektivitu alebo pseudoefektivitu.

Zvyčajne sa pomerové miery používajú vtedy, keď nie je možné alebo vhodné pracovať s objemovými údajmi, pretože nie sú dostupné. V našej práci však máme k dispozícii objemové miery, z ktorých vieme vypočítať aj pomerové. Preto sa najprv venujeme analýze založenej na objemových dátach a následne sledujeme, aké zmeny vo výsledkoch efektivity nastanú pri použití pomerových mier. Takto môžeme priamo porovnať, ako sa zmena typu dát premietne do hodnotenia efektivity poisťovní.

Cieľom tejto kapitoly nie je vykonať reálnu analýzu efektivity poisťovní, ale experimentovať s reálnymi dátami. Zameriame sa na to, ako sa mení BCC efektivita pri použití dvoch typov dát: objemových a zmiešaných dát (kombinácii objemových a pomerových), ako aj zmeny efektivity nad zmiešanými dátami pri R-VRS a FDH modeloch. Porovnáme výsledky BCC modelov pri správnej aplikácii na objemové dáta a nesprávnej aplikácii na zmiešané dáta. V prípade zmiešaných dát budeme porovnávať správnu aplikáciu R-VRS modelu s nevhodným použitím BCC a FDH modelov.

Vzhľadom na veľký počet efektívnych jednotiek vo viacerých modeloch vypočítame aj superefektivitu, ktorá umožní diferencovať efektívne poisťovne medzi sebou. Vybrané modely budú aplikované na reálne dáta z komerčných poisťovní a výsledky porovnané z hľadiska efektivity. Analýza tak poukáže na rozdiely v hodnotení efektivity v závislosti od zvoleného modelu.

6.1 Dáta

Vo svojej analýze sme pracovali s homogénnou skupinou českých a slovenských poisťovní - poisťovňami, ktoré ponúkajú životné aj neživotné poistenie. Vo vzorke sa nachádzajú slovenské a české poisťovne, ktorých zoznam je zobrazený v tabuľke 11.

K objemovým ukazovateľom, s ktorými budeme ďalej pracovať sme v tabuľke 10 priradili označenia pre jednoduchšiu orientáciu.

Objemové ukazovatele	Vstup/Výstup	Označenie
Náklady na poistné plnenia	Vstup	I_1
Počet zamestnancov	Vstup	I_2
Vlastné imanie	Vstup	I_3
Ostatné náklady	Vstup	I_4
Zaslúžené predpísané poistné	Výstup	O_1
Investičné výnosy	Výstup	O_2
Ostatné výnosy	Výstup	O_3

Tabuľka 10: Označenia objemových ukazovateľov

Dáta, ktoré sme získali z výročných správ poisťovní sú zobrazené v tabuľke 11. Keďže údaje vo výročných správach českých poisťovní sú vyčíslené v českých korunách, na ich prevod na eurá sme použili priemerný kurz roku 2023 zo stránky NBS - 1 EUR = 24,0043 CZK.

		VSTUPY				VÝSTUPY		
		I_1	I_2	I_3	I_4	O_1	O_2	O_3
1.	Allianz – Slovenská poisťovňa, a.s.	335 725	836	478 571	115 031	529 247	73289	5 601
2.	BNP Paribas Cardif Poisťovňa, a.s.	39 307	43	16 199	2 479	43 347	391	126
3.	ČSOB poisťovňa a.s.	90 039	345	52 138	7 535	95 556	8 548	1 892
4.	KOMUNÁLNA poisťovňa, a.s.	87 035	380	57 799	14 185	94 649	11 490	1 003
5.	KOOPERATIVA poisťovňa, a.s.	381 659	1 265	407 689	59 617	447 803	42 696	3 880
6.	Union poisťovňa, a. s.	96 554	331	49 601	13 328	112 426	8 343	429
7.	Wüstenrot poisťovňa, a.s.	37 826	245	38 737	15 156	42 086	8 027	344
8.	Allianz pojišťovna, a. s.	428 881	661	399 854	446 760	830 518	108 550	47 011
9.	BNP Paribas Cardif Pojišťovna, a.s.	33 548	92	81 560	105 070	126 620	12 851	15 329
10.	Česká podnikatelská pojišťovna, a. s.	246 980	944	135 242	254 537	429 434	82 085	22 734
11.	ČSOB Pojišťovna, a. s.	419 274	793	319 592	110 193	553 736	95 528	15 647
12.	SV pojišťovna, a.s.	9 953	44	17 173	14 831	19 637	2 152	2 860
13.	Generali Pojišťovna a. s.	1 517 187	3 579	1 560 012	449 961	1 914 157	282 241	37 035
14.	Hasičská vzájemná pojišťovna, a. s.	14 502	122	15 057	11 134	21 671	2 496	1 764
15.	Komerční pojišťovna, a. s.	247 651	275	151 553	156 243	242 023	75 635	106 761
16.	Kooperativa pojišťovna, a.s.	1 051 636	4 083	699 784	823 923	1 656 935	297 849	85 651
17.	MAXIMA pojišťovna, a. s.	8 699	60	20 531	28 819	37 287	2 534	586
18.	Simplea pojišťovna, a.s.	4 708	46	20 282	20 413	28 836	1 052	258
19.	UNIQA pojišťovna, a. s.	358 592	64	351 655	343 265	619 363	76 146	77 554

Tabuľka 11: Objemové dáta z výročných správ vzorky 19 poisťovní z roku 2023

Na strane vstupov sme zahrnuli náklady na poistné plnenia (I_1), počet zamestnancov

(I_2), vlastné imanie (I_3) a ostatné náklady (I_4). Medzi výstupy sme zaradili zaslúžené predpísané poistné (O_1), investičné výnosy (O_2) a ostatné výnosy (O_3). Ostatné náklady predstavujú rozdiel medzi celkovými nákladmi a nákladmi na poistné plnenia. Ostatné výnosy sme získali odčítaním zaslúženého predpísaného poistného a investičných výnosov od celkových výnosov. Dostali sme tak hodnoty 4 vstupov a 3 výstupov pre 19 poisťovní. Možno povedať, že je to priveľa faktorov vzhľadom na počet poisťovní, ale pre účely nášho porovnávania s pomerovými ukazovateľmi sme tento počet zachovali. Označenia a význam jednotlivých vstupov a výstupov sme zhrnuli do tabuľky 11.

		VSTUPY			VÝSTUPY	
		I_2	I_3	I_1/O_1	O_2	O_3/I_4
1.	Allianz – Slovenská poisťovňa, a.s.	836	478 571	0.634	73 289	0.049
2.	BNP Paribas Cardif Poisťovňa, a.s.	43	16 199	0.907	391	0.051
3.	ČSOB poisťovňa a.s.	345	52 138	0.942	8 548	0.251
4.	KOMUNÁLNA poisťovňa, a.s.	380	57 799	0.920	11 490	0.071
5.	KOOPERATIVA poisťovňa, a.s.	1 265	407 689	0.852	42 696	0.065
6.	Union poisťovňa, a. s.	331	49 601	0.859	8 343	0.032
7.	Wüstenrot poisťovňa, a.s.	245	38 737	0.899	8 027	0.022
8.	Allianz pojišťovna, a. s.	661	399 854	0.516	108 550	0.105
9.	BNP Paribas Cardif Pojišťovna, a.s.	92	81 560	0.265	12 851	0.146
10.	Česká podnikatelská pojišťovna, a. s.	944	135 242	0.575	82 085	0.089
11.	ČSOB Pojišťovna, a. s.	793	319 592	0.757	95 528	0.142
12.	SV pojišťovna, a.s.	44	17 173	0.507	2 152	0.193
13.	Generali Pojišťovna a. s.	3 579	1 560 012	0.793	282 241	0.082
14.	Hasičská vzájemná pojišťovna, a. s.	122	15 057	0.669	2 496	0.158
15.	Komerční pojišťovna, a. s.	275	151 553	1.023	75 635	0.683
16.	Kooperativa pojišťovna, a.s.	4 083	699 784	0.635	297 849	0.104
17.	MAXIMA pojišťovna, a. s.	60	20 531	0.233	2 534	0.020
18.	Simplea pojišťovna, a.s.	46	20 282	0.163	1 052	0.013
19.	UNIQA pojišťovna, a. s.	64	351 655	0.579	76 146	0.226

Tabuľka 12: Objemové aj pomerové dáta vzorky 19 poisťovní

Následne sme vytvorili pomerové dáta predelením dvoch objemových ukazovateľov. Vypočítali sme podiel nákladov na poistné plnenia a zaslúženého predpísaného poistného ($\frac{I_1}{O_1}$) - tento pomerový ukazovateľ sme zaradili medzi vstupy, pretože čím je menší tento podiel, tým je to pre poisťovňu výhodnejšie. Následne sme vypočítali ešte pomer ostatných výnosov k ostatným nákladom ($\frac{O_2}{I_4}$). Tento pomerový ukazovateľ sme zaradili medzi výstupy, pretože, čím je vyšší tým je to pre poisťovňu lepšie. Hodnoty tohoto vypočítaného pomerového vstupu a vypočítaného pomerového výstupu spolu so vstupmi I_2, I_3 a

výstupom O_2 sú uvedené v tabuľke 12.

Na dáta sme následne aplikovali nami naprogramované DEA modely v jazyku Python, pričom na riešenie optimalizačných úloh sme využili solver Gurobi.

6.2 Vstupne orientované modely

V tejto časti práce analyzujeme efektivitu poisťovní pomocou vstupne orientovaných DEA modelov. Pozornosť venujeme najmä tomu, ako sa hodnotenie efektivity mení v závislosti od zvoleného modelu a rôznych kombinácií vstupov a výstupov.

6.2.1 Prípadová štúdia 1: Použitie BCC-I modelu len na objemové dáta

V tejto prípadovej štúdii aplikujeme vstupne orientovaný BCC model výlučne na objemové údaje, ktoré sú zobrazené v tabuľke 11. Cieľom tejto prípadovej štúdie je zistiť, ako BCC model vyhodnotí efektivitu poisťovní v prípade, že sú do analýzy zahrnuté výlučne objemové miery.

	DMU	E^I	s_1^x	s_2^x	s_3^x	s_4^x	s_1^y	s_2^y	s_3^y
1.	Allianz – Slovenská poisťovňa, a.s.	1	0	0	0	0	0	0	0
2.	BNP Paribas Cardif Poisťovňa, a.s.	1	0	0	0	0	0	0	0
3.	ČSOB poisťovňa a.s.	1	0	0	0	0	0	0	0
4.	KOMUNÁLNA poisťovňa, a.s.	0.9794	0	72.7582	0	0	1772.343	0	903.2694
5.	KOOPERATIVA poisťovňa, a.s.	1	0	0	0	0	0	0	0
6.	Union poisťovňa, a. s.	1	0	0	0	0	0	0	0
7.	Wüstenrot poisťovňa, a.s.	1	0	0	0	0	0	0	0
8.	Allianz pojišťovna, a. s.	1	0	0	0	0	0	0	0
9.	BNP Paribas Cardif Pojišťovna, a.s.	1	0	0	0	0	0	0	0
10.	Česká podnikatelská pojišťovna, a. s.	1	0	0	0	0	0	0	0
11.	ČSOB Pojišťovna, a. s.	1	0	0	0	0	0	0	0
12.	ERGO pojišťovna, a.s.	1	0	0	0	0	0	0	0
13.	Generali Pojišťovna a. s.	1	0	0	0	0	0	0	0
14.	Hasičská vzájemná pojišťovna, a. s.	1	0	0	0	0	0	0	0
15.	Komerční pojišťovna, a. s.	1	0	0	0	0	0	0	0
16.	Kooperativa pojišťovna, a.s.	1	0	0	0	0	0	0	0
17.	MAXIMA pojišťovna, a. s.	1	0	0	0	0	0	0	0
18.	Simplea pojišťovna, a.s.	1	0	0	0	0	0	0	0
19.	UNIQA pojišťovna, a. s.	1	0	0	0	0	0	0	0

Tabuľka 13: Hodnoty efektivít a slackov z druhej fázy v Prípadovej štúdii 1.

V tabuľke 13 sú zobrazené výsledky efektivity jednotlivých komerčných poisťovní po

aplikovaní vstupne orientovaného BCC modelu na výlučne objemové miery. Tento model vyhodnotil 18 poisťovní ako efektívne, pričom každá z efektívnych poisťovní má nulové slacky - hodnota 1 je jej efektivitou.

6.2.2 Prípadová štúdia 2: Použitie BCC-I modelu na dáta s pomerovými ukazovateľmi

V tejto časti sme niektoré objemové ukazovatele nahradili pomerovými. Teda sme pracovali s dátami uvedenými v tabuľke 12. Výsledky, ktoré sme získali aplikovaním vstupne orientovaného BCC modelu na dáta sú zobrazené v nasledujúcej tabuľke 14:

	DMU	E^I_{BCC}	s^x_1	s^x_2	s^x_3	s^y_1	s^y_2
1.	Allianz – Slovenská poisťovňa, a.s.	0.6133	0	26 593.63	0	0	0.02
2.	BNP Paribas Cardif Poisťovňa, a.s.	1	0	0	0	0	0
3.	ČSOB poisťovňa a.s.	0.6355	147.81	0	0.03	2332.49	0
4.	KOMUNÁLNA poisťovňa, a.s.	0.5319	11.49	0	0	0	0.02
5.	KOOPERATIVA poisťovňa, a.s.	0.3416	0	0	0	0	0
6.	Union poisťovňa, a. s.	0.5328	23.28	0	0	0	0.05
7.	Wüstenrot poisťovňa, a.s.	0.6365	0	0	0	0	0.11
8.	Allianz pojišťovna, a. s.	1	0	0	0	0	0
9.	BNP Paribas Cardif Pojišťovna, a.s.	1	0	0	0	0	0
10.	Česká podnikatelská pojišťovna, a. s.	1	0	0	0	0	0
11.	ČSOB Pojišťovna, a. s.	0.8574	0	0	0	0	0.09
12.	SV pojišťovna, a.s.	1	0	0	0	0	0
13.	Generali Pojišťovna a. s.	1	0	0	0	0	0
14.	Hasičská vzájemná pojišťovna, a. s.	1	0	0	0	0	0
15.	Komerční pojišťovna, a. s.	1	0	0	0	0	0
16.	Kooperativa pojišťovna, a.s.	1	0	0	0	0	0
17.	MAXIMA pojišťovna, a. s.	1	0	0	0	0	0
18.	Simplea pojišťovna, a.s.	1	0	0	0	0	0
19.	UNIQA pojišťovna, a. s.	1	0	0	0	0	0

Tabuľka 14: Hodnoty efektivít a slackov z druhej fázy v Prípadovej štúdii 2.

Môžeme pozorovať, že BCC model aplikovaný na kombináciu objemových a pomerových dát vyhodnotil, že sedem poisťovní spracúva svoje vstupy na výstupy neefektívne. Medzi tieto poisťovne patria Allianz - Slovenská poisťovňa, ČSOB poisťovňa, Komunálna poisťovňa, KOOPERATIVA poisťovňa, Union poisťovňa, Wüstenrot poisťovňa a ČSOB pojišťovna. Môžeme si všimnúť, že vstupný BCC model aplikovaný výlučne na objemové dáta vyhodnotil z týchto poisťovní ako neefektívnu len Komunálnu poisťovňu a ostatné

označil za efektívne. V tabuľke 14 môžeme vidieť, že všetky poisťovne, ktorých $E^I_{BCC} < 1$ majú nenulové slacky, čo znamená, že hodnota E^I_{BCC} je ich pseudoefektivitou.

6.2.3 Prípadová štúdia 3: Použitie R-VRS-I modelu

Na dáta z tabuľky 12 sme aplikovali vstupne orientovaný R-VRS model. Konkrétne sme použili jeho verziu, kde sme faktor skrátenia θ priradili ku každému vstupu, t.j. k všetkým objemovým ukazovateľom aj k všetkým pomerovým ukazovateľom. Cieľom tejto štúdie je ukázať, ako tento model hodnotí efektivitu v porovnaní s klasickým BCC modelom pri použití pomerových údajov.

	DMU	E^I_{R-VRS}	s^x_1	s^x_2	s^x_3	s^y_1	s^y_2
1.	Allianz – Slovenská poisťovňa, a.s.	0.8141	224.04	116 560.4	0	0	0.06
2.	BNP Paribas Cardif Poisťovňa, a.s.	1	0	0	0	0	0
3.	ČSOB poisťovňa a.s.	1	0	0	0	0	0
4.	KOMUNÁLNA poisťovňa, a.s.	0.6254	88.53	5 184.32	0	0	0.02
5.	KOOPERATIVA poisťovňa, a.s.	0.6059	487.35	84 021.32	0	0	0.04
6.	Union poisťovňa, a. s.	0.6697	107.96	6899.20	0	0	0.06
7.	Wüstenrot poisťovňa, a.s.	0.6673	53.35	0	0.02	0	0.07
8.	Allianz pojišťovna, a. s.	1	0	0	0	0	0
9.	BNP Paribas Cardif Pojišťovna, a.s.	1	0	0	0	0	0
10.	Česká podnikatelská pojišťovna, a. s.	1	0	0	0	0	0
11.	ČSOB Pojišťovna, a. s.	1	0	0	0	0	0
12.	SV pojišťovna, a.s.	1	0	0	0	0	0
13.	Generali Pojišťovna a. s.	1	0	0	0	0	0
14.	Hasičská vzájemná pojišťovna, a. s.	1	0	0	0	0	0
15.	Komerční pojišťovna, a. s.	1	0	0	0	0	0
16.	Kooperativa pojišťovna, a.s.	1	0	0	0	0	0
17.	MAXIMA pojišťovna, a. s.	1	0	0	0	0	0
18.	Simplea pojišťovna, a.s.	1	0	0	0	0	0
19.	UNIQA pojišťovna, a. s.	1	0	0	0	0	0

Tabuľka 15: Hodnoty efektivít a slackov z druhej fázy v Prípadovej štúdii 3.

6.2.4 Prípadová štúdia 4: Použitie FDH-I modelu

Zo vstupne orientovaných modelov sme sa ešte rozhodli na dáta z tabuľky 12 aplikovať vstupne orientovaný model FDH, ktorý oproti modelom BCC a R-VRS porovnáva rozhodovaciu jednotku len s jednou inou jednotkou, nie ich kombináciou. Cieľom tejto štúdie je zistiť, ako vyhodnotí efektivity jednotlivých poisťovní model FDH oproti modelom BCC

a R-VRS. Výsledky sú zobrazené v tabuľke 16. Môžeme pozorovať, že z modelov aplikovaných na mix pomerových a objemových ukazovateľov, vyhodnotil FDH model najviac poisťovní za efektívne.

	DMU	E^I_{FDH}	s^x_1	s^x_2	s^x_3	s^y_1	s^y_2
1.	Allianz – Slovenská poisťovňa, a.s.	0.8355	37.49	0	0.01	35 260.88	0.06
2.	BNP Paribas Cardif Poisťovňa, a.s.	1	0	0	0	0	0
3.	ČSOB poisťovňa a.s.	1	0	0	0	0	0
4.	KOMUNÁLNA poisťovňa, a.s.	1	0	0	0	0	0
5.	KOOPERATIVA poisťovňa, a.s.	0.7462	0	168 993.47	0.06	39 389.33	0.02
6.	Union poisťovňa, a. s.	1	0	0	0	0	0
7.	Wüstenrot poisťovňa, a.s.	1	0	0	0	0	0
8.	Allianz pojišťovna, a. s.	1	0	0	0	0	0
9.	BNP Paribas Cardif Pojišťovna, a.s.	1	0	0	0	0	0
10.	Česká podnikatelská pojišťovna, a. s.	1	0	0	0	0	0
11.	ČSOB Pojišťovna, a. s.	1	0	0	0	0	0
12.	SV pojišťovna, a.s.	1	0	0	0	0	0
13.	Generali Pojišťovna a. s.	1	0	0	0	0	0
14.	Hasičská vzájemná pojišťovna, a. s.	1	0	0	0	0	0
15.	Komerční pojišťovna, a. s.	1	0	0	0	0	0
16.	Kooperativa pojišťovna, a.s.	1	0	0	0	0	0
17.	MAXIMA pojišťovna, a. s.	1	0	0	0	0	0
18.	Simplea pojišťovna, a.s.	1	0	0	0	0	0
19.	UNIQA pojišťovna, a. s.	1	0	0	0	0	0

Tabuľka 16: Hodnoty efektivít a slackov z druhej fázy v Prípadovej štúdii 4.

6.2.5 Porovnanie výsledkov vstupne orientovaných modelov

Na obrázku 6 je zobrazené porovnanie efektivít poisťovní vypočítaných v prípadových štúdiách 1-4. Z grafu možno vidieť, že pri použití BCC modelu len na objemové ukazovatele (Prípadová štúdia 1) dosahujú všetky poisťovne okrem jednej (Komunálna poisťovňa, a.s.) efektivitu rovnú jednej. V tabuľke 13 pritom vidíme, že pri všetkých DMU, kde je efektivita rovná 1 sú všetky slacky nulové, t.j. hodnota 1 je ich efektivitou.

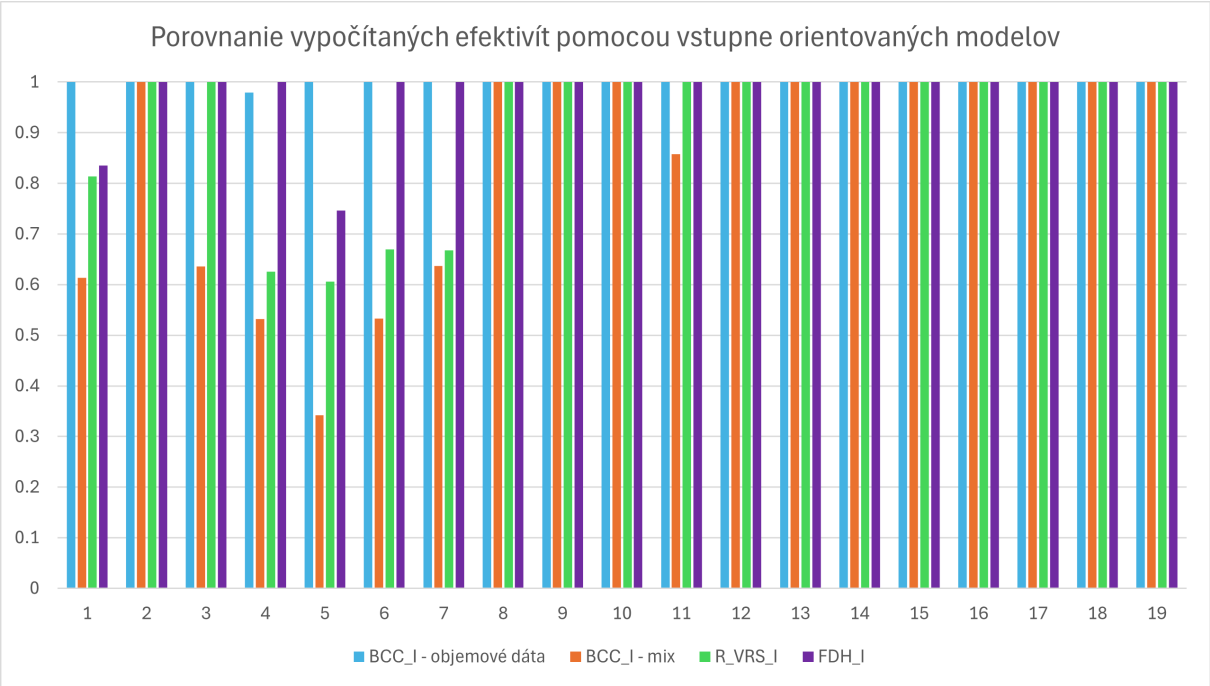
Po nahradení niektorých objemových ukazovateľov za pomerové, teda pri použití BCC modelu na kombináciu objemových a pomerových ukazovateľov (Prípadová štúdia 2), sa efektivity viacerých jednotiek znížili. Významný rozdiel nastal napríklad pri poisťovniach Allianz - Slovenská poisťovňa, ČSOB poisťovňa, KOOPERATIVA poisťovňa, Union poisťovňa, Wüstenrot poisťovňa a ČSOB pojišťovna, ktoré v Prípadovej štúdii 1 boli efektívne,

avšak po aplikovaní BCC modelu na mix ukazovateľov v Prípadovej štúdii 2 už efektívne nie sú.

R-VRS-I model, ktorý je navrhnutý tak, aby pracoval s pomerovými ukazovateľmi bez predpokladu konvexnosti, prináša podobné výsledky, ako BCC model v Prípadovej štúdii 2. Avšak v niektorých prípadoch sa výsledky líšia. Hlavným rozdielom medzi výsledkami v prípadoch 2 a 3 je, že R-VRS-I model v prípadovej štúdii 3 vyhodnotil efektívnymi poisťovne ČSOB poisťovňa a ČSOB pojišťovna, ktoré boli v Prípadovej štúdii 2 vyhodnotené ako neefektívne.

FDH model vyhodnotil ako neefektívne len dve poisťovne - Allianz - Slovenská poisťovňa a Kooperativa poisťovňa. Hodnoty efektivít ostatných poisťovní sú rovné jednej, pričom slacky v tabuľke 16 majú nulové, t.j. všetky tieto poisťovne sú efektívne. Taktiež môžeme pozorovať, že vo vnútri množiny M_{FDH}, t.j. neefektívne útvary podľa FDH, sa nachádzajú len 2 jednotky, pričom vo vnútri M^R_{VRS} ich je 5 a vo vnútri M_{VRS} ich je 7. Toto pozorovanie nám potvrdzuje, že $M_{FDH} \subseteq M^R_{VRS} \subseteq M_{VRS}$.

Z porovnania vyplýva, že zmena typu dát aj výber modelu má významný vplyv na výsledné hodnotenie efektivity.



Obr. 6: Porovnanie výsledkov vstupne orientovaných modelov

Zároveň môžeme sledovať, že vo všetkých prípadových štúdiách 1-3 boli poisťovne BNP

PAribas Cardif Poisťovňa, Allianz pojišťovna, BNP Paribas Cardif Pojišťovna, Česká podnikatelská pojišťovna, SV pojišťovna, Generali pojišťovna, Hasičská vzájemná pojišťovna, Komerční pojišťovna, Kooperativa pojišťovna, MAXIMA pojišťovna, Simplea pojišťovna a UNIQA pojišťovňa vyhodnotené ako efektívne.

6.3 Výstupne orientované modely

V tejto časti sa zameriavame na výstupne orientované DEA modely, ktoré predpokladajú fixné vstupy a skúmajú možnosti zvýšenia výstupov. Cieľom analýzy je porovnať efektivity poisťovní pri použití rôznych modelov a typov dát - objemových a kombinácie objemových a pomerových. Sledujeme, ako sa mení hodnotenie efektivity v závislosti od zvoleného modelu a štruktúry dát.

6.3.1 Prípadová štúdia 5: Použitie BCC-O modelu len na objemové dáta

V tejto prípadovej štúdii sa zameriavame na použitie výstupne orientovaného BCC modelu, pričom do analýzy vstupujú výhradne objemové ukazovatele z tabuľky 11.

	DMU	E^O	s_1^x	s_2^x	s_3^x	s_4^x	s_1^y	s_2^y	s_3^y
1.	Allianz – Slovenská poisťovňa, a.s.	1	0	0	0	0	0	0	0
2.	BNP Paribas Cardif Poisťovňa, a.s.	1	0	0	0	0	0	0	0
3.	ČSOB poisťovňa a.s.	1	0	0	0	0	0	0	0
4.	KOMUNÁLNA poisťovňa, a.s.	0.9718	0	76.29	0	0	1249.69	0	941.77
5.	KOOPERATIVA poisťovňa, a.s.	1	0	0	0	0	0	0	0
6.	Union poisťovňa, a. s.	1	0	0	0	0	0	0	0
7.	Wüstenrot poisťovňa, a.s.	1	0	0	0	0	0	0	0
8.	Allianz pojišťovna, a. s.	1	0	0	0	0	0	0	0
9.	BNP Paribas Cardif Pojišťovna, a.s.	1	0	0	0	0	0	0	0
10.	Česká podnikatelská pojišťovna, a. s.	1	0	0	0	0	0	0	0
11.	ČSOB Pojišťovna, a. s.	1	0	0	0	0	0	0	0
12.	ERGO pojišťovna, a.s.	1	0	0	0	0	0	0	0
13.	Generali Pojišťovna a. s.	1	0	0	0	0	0	0	0
14.	Hasičská vzájemná pojišťovna, a. s.	1	0	0	0	0	0	0	0
15.	Komerční pojišťovna, a. s.	1	0	0	0	0	0	0	0
16.	Kooperativa pojišťovna, a.s.	1	0	0	0	0	0	0	0
17.	MAXIMA pojišťovna, a. s.	1	0	0	0	0	0	0	0
18.	Simplea pojišťovna, a.s.	1	0	0	0	0	0	0	0
19.	UNIQA pojišťovna, a. s.	1	0	0	0	0	0	0	0

Tabuľka 17: Hodnoty efektivít a slackov z druhej fázy v Prípadovej štúdii 5.

Výsledky prípadovej štúdie 5 sú zobrazené v tabuľke 17. Výstupne orientovaný BCC model vyhodnotil, že jediná Komunálna poisťovňa spracúva svoje vstupy neefektívne. Ostatné poisťovne vyhodnotil ako efektívne, nakoľko majú efektivitu rovnú 1 a slacky sú pri všetkých nulové.

6.3.2 Prípadová štúdia 6: Použitie BCC-O modelu na dáta s pomerovými ukazovateľmi

Analogicky, ako pri vstupne orientovaných modeloch, sme aplikovali BCC model aj na dáta uvedené v tabuľke 12, t.j. na kombináciu objemových a pomerových ukazovateľov.

	DMU	E^O_{BCC}	s^x_1	s^x_2	s^x_3	s^y_1	s^y_2
1.	Allianz – Slovenská poisťovňa, a.s.	0.6131	0	0	0.03	0	0.12
2.	BNP Paribas Cardif Poisťovňa, a.s.	1	0	0	0	0	0
3.	ČSOB poisťovňa a.s.	0.7835	240.90	0	0.30	10 362.91	0
4.	KOMUNÁLNA poisťovňa, a.s.	0.3902	26.96	0	0.25	0	0
5.	KOOPERATIVA poisťovňa, a.s.	0.3056	0	0	0.16	0	0.08
6.	Union poisťovňa, a. s.	0.3356	0	0	0.26	0	0.05
7.	Wüstenrot poisťovňa, a.s.	0.4602	0	0	0.32	0	0.11
8.	Allianz pojišťovna, a. s.	1	0	0	0	0	0
9.	BNP Paribas Cardif Pojišťovna, a.s.	1	0	0	0	0	0
10.	Česká podnikatelská pojišťovna, a. s.	1	0	0	0	0	0
11.	ČSOB Pojišťovna, a. s.	0.8609	0	0	0	0	0.21
12.	SV pojišťovna, a.s.	1	0	0	0	0	0
13.	Generali Pojišťovna a. s.	1	0	0	0	0	0
14.	Hasičská vzájemná pojišťovna, a. s.	1	0	0	0	0	0
15.	Komerční pojišťovna, a. s.	1	0	0	0	0	0
16.	Kooperativa pojišťovna, a.s.	1	0	0	0	0	0
17.	MAXIMA pojišťovna, a. s.	1	0	0	0	0	0
18.	Simplea pojišťovna, a.s.	1	0	0	0	0	0
19.	UNIQA pojišťovna, a. s.	1	0	0	0	0	0

Tabuľka 18: Hodnoty efektivít a slackov z druhej fázy v Prípadovej štúdii 6.

Výsledky tejto analýzy sú zhrnuté v tabuľke 18, kde možno pozorovať, ako sa zmenili hodnoty efektivity po prechode z čisto objemových ukazovateľov na kombináciu objemových a pomerových dát. BCC model použitý len na objemové miery vyhodnotil ako neefektívnu len jednu poisťovňu (Komunálna poisťovňa), pričom BCC model použitý na mix pomerových a objemových dát k tejto poisťovni priradil ďalších šesť (Allianz - Slovenská poisťovňa, ČSOB poisťovňa, KOOPERATIVA poisťovňa, Union poisťovňa, Wüstenrot

poisťovňa a ČSOB pojišťovna), ktoré tiež vyhodnotil ako neefektívne. Všetky z neefektívnych v prípadovej štúdii 6 majú nenulové slacky, teda E^O_{BCC} je ich pseudoefektivitou.

6.3.3 Prípadová štúdia 7: Použitie R-VRS-O modelu

V prípadovej štúdii 7 sme na dáta z tabuľky 12 aplikovali výstupne orientovaný R-VRS model. Faktor predĺženia výstupov ψ sme priradili ku každému výstupu, či už bol výstup objemový ukazovateľ alebo pomerový. Cieľom tejto štúdie bolo sledovať ako výstupne orientovaný R-VRS model vyhodnotí efektivity 19 poisťovní v našej vzorke. Model $R-VRS-O$ vyhodnotil efektívnymi 14 poisťovní. Žiadna z týchto poisťovní nie je pseudoefektívna, pretože všetky majú nulové slacky.

	DMU	E^O_{R-VRS}	s^x_1	s^x_2	s^x_3	s^y_1	s^y_2
1.	Allianz – Slovenská poisťovňa, a.s.	0.6752	175	78 716.56	0.12	0	0.03
2.	BNP Paribas Cardif Poisťovňa, a.s.	1	0	0	0	0	0
3.	ČSOB poisťovňa a.s.	1	0	0	0	0	0
4.	KOMUNÁLNA poisťovňa, a.s.	0.6802	64.93	9 367.41	0.25	0	0
5.	KOOPERATIVA poisťovňa, a.s.	0.4583	489.13	95 871.34	0.10	0	0.04
6.	Union poisťovňa, a. s.	0.3604	0	3 203.49	0.64	0	0.10
7.	Wüstenrot poisťovňa, a.s.	0.4602	0	0	0.32	0	0.04
8.	Allianz pojišťovna, a. s.	1	0	0	0	0	0
9.	BNP Paribas Cardif Pojišťovna, a.s.	1	0	0	0	0	0
10.	Česká podnikatelská pojišťovna, a. s.	1	0	0	0	0	0
11.	ČSOB Pojišťovna, a. s.	1	0	0	0	0	0
12.	SV pojišťovna, a.s.	1	0	0	0	0	0
13.	Generali Pojišťovna a. s.	1	0	0	0	0	0
14.	Hasičská vzájemná pojišťovna, a. s.	1	0	0	0	0	0
15.	Komerční pojišťovna, a. s.	1	0	0	0	0	0
16.	Kooperativa pojišťovna, a.s.	1	0	0	0	0	0
17.	MAXIMA pojišťovna, a. s.	1	0	0	0	0	0
18.	Simplea pojišťovna, a.s.	1	0	0	0	0	0
19.	UNIQA pojišťovna, a. s.	1	0	0	0	0	0

Tabuľka 19: Hodnoty efektivít a slackov z druhej fázy v Prípadovej štúdii 7.

6.3.4 Prípadová štúdia 8: Použitie FDH-O modelu

Na záver sme na mix pomerových a objemových dát z tabuľky 12 použili výstupne orientovaný FDH model. Tento model vyhodnotil 17 poisťovní ako efektívne. Všetky z poisťovní, ktorých $E^O_{FDH} = 1$, majú nulové slacky, t.j. E^O_{FDH} je ich efektivitou a nie pseudoefek-

tivitou. Zvyšné dve (Allianz - Slovenská poisťovňa, Kooperativa poisťovňa) označil za neefektívne, pričom ich slacky sú nenulové, teda E^O_{FDH} je ich pseudoefektivitou. Výsledky prípadovej štúdie 8 sú zobrazené v nasledujúcej tabuľke:

	DMU	E^O_{FDH}	s^x_1	s^x_2	s^x_3	s^y_1	s^y_2
1.	Allianz – Slovenská poisťovňa, a.s.	0.6752	175	78 716.56	0.12	0	0.03
2.	BNP Paribas Cardif Poisťovňa, a.s.	1	0	0	0	0	0
3.	ČSOB poisťovňa a.s.	1	0	0	0	0	0
4.	KOMUNÁLNA poisťovňa, a.s.	1	0	0	0	0	0
5.	KOOPERATIVA poisťovňa, a.s.	0.4583	472	88 096.51	0.10	2 375.11	0
6.	Union poisťovňa, a. s.	1	0	0	0	0	0
7.	Wüstenrot poisťovňa, a.s.	1	0	0	0	0	0
8.	Allianz pojišťovna, a. s.	1	0	0	0	0	0
9.	BNP Paribas Cardif Pojišťovna, a.s.	1	0	0	0	0	0
10.	Česká podnikatelská pojišťovna, a. s.	1	0	0	0	0	0
11.	ČSOB Pojišťovna, a. s.	1	0	0	0	0	0
12.	SV pojišťovna, a.s.	1	0	0	0	0	0
13.	Generali Pojišťovna a. s.	1	0	0	0	0	0
14.	Hasičská vzájemná pojišťovna, a. s.	1	0	0	0	0	0
15.	Komerční pojišťovna, a. s.	1	0	0	0	0	0
16.	Kooperativa pojišťovna, a.s.	1	0	0	0	0	0
17.	MAXIMA pojišťovna, a. s.	1	0	0	0	0	0
18.	Simplea pojišťovna, a.s.	1	0	0	0	0	0
19.	UNIQA pojišťovna, a. s.	1	0	0	0	0	0

Tabuľka 20: Hodnoty efektivít a slackov z druhej fázy v Prípadovej štúdii 8.

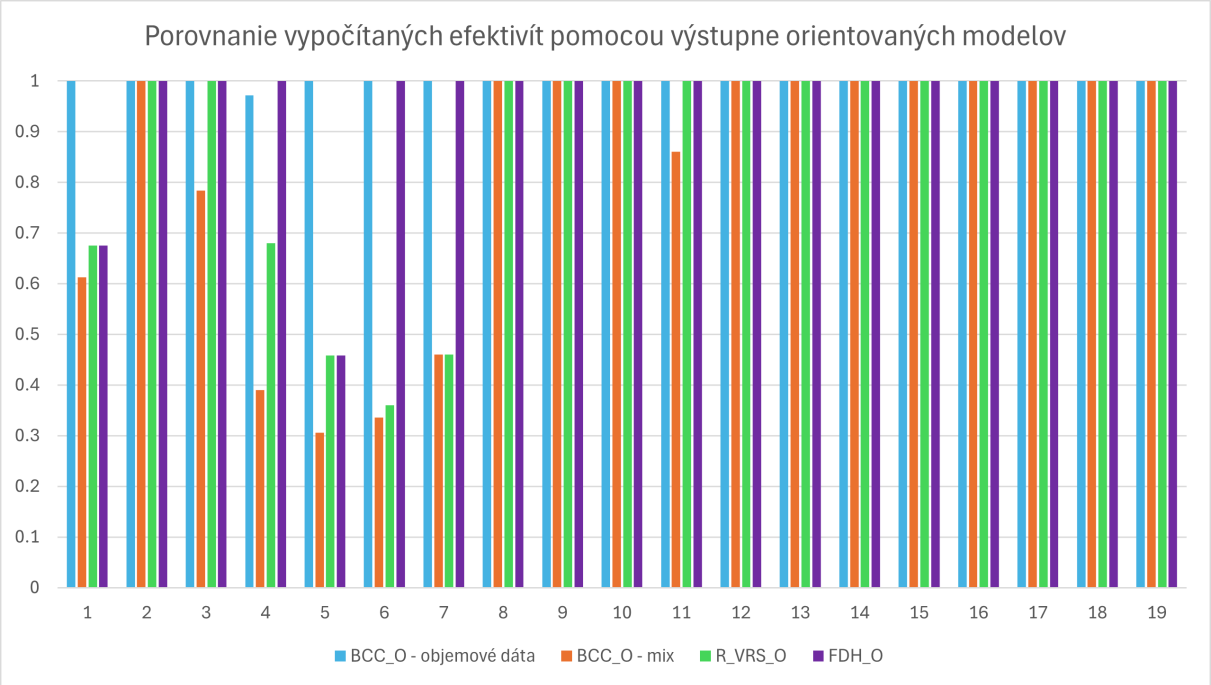
6.3.5 Porovnanie výsledkov výstupne orientovaných modelov

Na obrázku 7 je znázornené porovnanie výsledkov po aplikovaní výstupne orientovaných modelov na dáta z tabuliek 11 a 12.

Pri použití BCC modelu výlučne na objemové dáta dosahuje väčšina poisťovní efektivitu rovnú jednej. Zároveň v tabuľke 17 môžeme vidieť, že všetky tieto poisťovne majú nulové slacky, t.j. hodnota 1 je ich efektivitou a nie pseudoefektivitou. Výstupne orientovaný BCC model použitý výhradne na objemové dáta označil ako neefektívnu jedinú poisťovňu - Komunálnu poisťovňu.

Keď sme nahradili niektoré objemové ukazovatele pomerovými, a teda sme aplikovali výstupne orientovaný BCC model na dáta z tabuľky 12, tak sa efektivita niektorých jednotiek výrazne znížila. Výstupne orientovaný BCC model použitý na objemové uka-

zovatele vyhodnotil ako efektívne 18 poisťovní. Po nahradení niektorých objemových dát pomerovými vyhodnotil rovnaký model efektívnymi už len 12 poisťovní a ostatné označil za neefektívne.



Obr. 7: Porovnanie výsledkov výstupne orientovaných modelov

Po aplikovaní výstupne orientovaného R-VRS modelu na mix objemových aj pomerových dát z tabuľky 12 sme v prípadovej štúdii 7 dostali podobné výsledky ako v prípadovej štúdii 6, avšak oproti BCC modelu aplikovanému na mix dát R-VRS model vyhodnotil ako efektívne navyše aj poisťovne ČSOB poisťovňa a ČSOB pojišťovna, ktoré boli v prípadovej štúdii 6 označené ako neefektívne.

Ako posledný sme aplikovali výstupne orientovaný model FDH, ktorý vyhodnotil, že až 17 poisťovní je efektívnych a ostatné dve (Allianz - Slovenská poisťovňa, Kooperativa poisťovňa) sú neefektívne, pričom hodnoty ich efektivít sú rovnaké ako boli po aplikovaní výstupne orientovaného R-VRS modelu.

Zároveň z grafu môžeme vidieť, že poisťovne BNP PAribas Cardif Poisťovňa, Allianz pojišťovna, BNP Paribas Cardif Pojišťovna, Česká podnikatelská pojišťovna, SV pojišťovna, Generali pojišťovna, Hasičská vzájemná pojišťovna, Komerční pojišťovna, Kooperativa pojišťovna, MAXIMA pojišťovna, Simplea pojišťovna a UNIQA pojišťovňa boli vyhodnotené ako efektívne vo všetkých prípadových štúdiách 5-8.

6.4 Sumárne porovnanie výsledkov v Prípadových štúdiách 1-8

Na záver porovnávame vstupne orientované modely s výstupne orientovanými, pričom sa zameriavame na zhodu efektívnych jednotiek a rozdiely v hodnotách efektivít pri neefektívnych útvaroch. Výsledky všetkých prípadových štúdií sú zhrnuté v tabuľke 21, ktorá obsahuje hodnoty efektivít pre všetky analyzované poisťovne a modely.

	DMU	E^I	E^I_{BCC}	E^I_{R-VRS}	E^I_{FDH}	E^O	E^O_{BCC}	E^O_{R-VRS}	E^O_{FDH}
1.	Allianz – Slovenská poisťovňa, a.s.	1	0.6133	0.8141	0.8355	1	0.6131	0.6752	0.6752
2.	BNP Paribas Cardif Poisťovňa, a.s.	1	1	1	1	1	1	1	1
3.	ČSOB poisťovňa a.s.	1	0.6355	1	1	1	0.7835	1	1
4.	KOMUNÁLNA poisťovňa, a.s.	0.9794	0.5319	0.6254	1	0.9718	0.3902	0.6802	1
5.	KOOPERATIVA poisťovňa, a.s.	1	0.3416	0.6059	0.7462	1	0.3056	0.4583	0.4583
6.	Union poisťovňa, a. s.	1	0.5328	0.6697	1	1	0.3356	0.3604	1
7.	Wüstenrot poisťovňa, a.s.	1	0.6365	0.6673	1	1	0.4602	0.4602	1
8.	Allianz pojišťovna, a. s.	1	1	1	1	1	1	1	1
9.	BNP Paribas Cardif Pojišťovna, a.s.	1	1	1	1	1	1	1	1
10.	Česká podnikatelská pojišťovna, a. s.	1	1	1	1	1	1	1	1
11.	ČSOB Pojišťovna, a. s.	1	0.8574	1	1	1	0.8609	1	1
12.	SV pojišťovna, a.s.	1	1	1	1	1	1	1	1
13.	Generali Pojišťovna a. s.	1	1	1	1	1	1	1	1
14.	Hasičská vzájemná pojišťovna, a. s.	1	1	1	1	1	1	1	1
15.	Komerční pojišťovna, a. s.	1	1	1	1	1	1	1	1
16.	Kooperativa pojišťovna, a.s.	1	1	1	1	1	1	1	1
17.	MAXIMA pojišťovna, a. s.	1	1	1	1	1	1	1	1
18.	Simplea pojišťovna, a.s.	1	1	1	1	1	1	1	1
19.	UNIQA pojišťovna, a. s.	1	1	1	1	1	1	1	1

Tabuľka 21: Porovnanie výsledkov prípadových štúdii 1-8

Podľa vety 14 z [3] platí, že ak je útvar (x_o, y_o) efektívny vo vstupne orientovanom BCC modeli, je efektívny aj vo výstupne orientovanom modeli a naopak. Táto vlastnosť sa potvrdila aj v našich výsledkoch a môžeme pozorovať, že platí pre každý analyzovaný model. Všetky poisťovne, ktoré boli efektívne vo vstupnom modeli, boli efektívne aj vo výstupne orientovanej verzii daného modelu a naopak. Z výsledkov môžeme pozorovať, že v prípade neefektívnych útvarov sa hodnoty efektivít medzi vstupne orientovaným modelom a jeho výstupne orientovanou verziou môžu líšiť.

Pri porovnaní vstupných a výstupných modelov môžeme pozorovať, že pri FDH modeli (prípadové štúdie 4 a 8) dosahujú neefektívne útvary spravidla vyššiu efektivitu vo vstupne orientovanom modeli ako vo výstupnom. Naopak, pri BCC modeli (prípadové štúdie 2 a 6) a pri R-VRS modeli tento vzťah nie je jednotný - pri niektorých útvaroch je efektivita

vyššia vo vstupnom, pri iných vo výstupnom. Napr. efektivita Komunálnej poisťovne je vyššia pri vstupne orientovanom BCC modeli ako pri výstupnom BCC, avšak pri R-VRS modeli to je naopak.

Použitie BCC alebo FDH modelu na kombináciu objemových a pomerových dát môže byť problematické. Ako sme opisovali v kapitole 2, BCC model má tendenciu podhodnocovať efektivity, zatiaľ čo FDH ich môže nadhodnocovať. Z tabuľky 21 vidíme, že niektoré jednotky boli vyhodnotené ako efektívne len pri FDH modeli.

Zároveň vieme, že medzi produkčnými množinami platí vzťah $M_{VRS} \supseteq M^R_{VRS} \supseteq M_{FDH}$, z čoho vyplýva aj usporiadanie príslušných efektivít $E_{VRS} \leq E_{R-VRS} \leq E_{FDH}$. Tento vzťah sa v našich výsledkoch potvrdil - efektivity BCC modelu boli najnižšie, nasledovali ich hodnoty modelu R-VRS, pričom najvyššie efektivity mali útvary pri použití FDH modelu.

6.5 Sumárne porovnanie superefektivít v Prípadových štúdiách 1-8

Vzhľadom na to, že všetky modely v prípadových štúdiách 1-8 identifikovali veľké množstvo poisťovní ako efektívne, ukázalo sa ako žiadúce tieto jednotky ďalej diferencovať. Práve preto sme do analýzy zaradili výpočet superefektivity, ktorá umožňuje rozlíšiť mieru efektívnosti aj medzi jednotkami, ktoré v základných modeloch dosiahli efektivitu rovnú jednej. Tento prístup sa ukázal ako veľmi prínosný, pretože odhalil výrazné rozdiely medzi jednotkami, ktoré by inak boli hodnotené rovnako. Výsledky superefektivity sú uvedené v tabuľke 22.

Pri porovnaní vstupne orientovaných modelov superefektivity $SE^I_{BCC}, SE^I_{FDH}, SE^I_{R-VRS}$ pre poisťovne, ktoré boli v základných modeloch klasifikované ako efektívne, možno pozorovať výrazné rozdiely v hodnotách superefektivít. Napríklad MAXIMA pojišťovna, a. s., ktorá bola efektívna vo všetkých vstupne orientovaných modeloch, dosahuje superefektivitu $SE^I_{BCC} = 1.0111$, no superefektivita v zmysle R-VRS je vyššia ($SE^I_{R-VRS} = 2.1048$) a ešte viac stúpla v modeli $SE^I_{FDH} = 3.2766$. Podobný vývoj je možné pozorovať aj pri ostatných poisťovniach, ktoré boli vyhodnotené ako efektívne v základných modeloch, t.j. $SE^I_{BCC} \leq SE^I_{R-VRS} \leq SE^I_{FDH}$. Tieto výsledky sú v súlade s teoretickými predpokladmi, že FDH model - vďaka menšej množine produkčných možností - má tendenciu jednotky

nadhodnocovať. Model R-VRS tak poskytuje kompromis medzi konzervatívnejším BCC a nadhodnocujúcim FDH modelom.

	DMU	SE^I	SE^I_{BCC}	SE^I_{R-VRS}	SE^I_{FDH}	SE^O	SE^O_{BCC}	SE^O_{R-VRS}	SE^O_{FDH}
1.	Allianz – Slovenská poisťovňa, a.s.	1.1377	0.6133	0.8141	0.8355	1.1355	0.6131	0.6752	0.6752
2.	BNP Paribas Cardif Poisťovňa, a.s.	4.2017	1.0577	1.0577	1.0601	∞	∞	∞	∞
3.	ČSOB poisťovňa a.s.	1.6808	0.6355	2.9068	2.9068	1.7901	0.7835	1.3021	3.4241
4.	KOMUNÁLNA poisťovňa, a.s.	0.9794	0.5319	0.6254	1.4111	0.9718	0.3902	0.6802	2.1967
5.	KOOPERATIVA poisťovňa, a.s.	1.4502	0.3416	0.6059	0.7462	1.3652	0.3056	0.4583	0.4583
6.	Union poisťovňa, a. s.	1.1771	0.5328	0.6697	1.0972	1.1419	0.3356	0.3604	3.2919
7.	Wüstenrot poisťovňa, a.s.	1.0713	0.6365	0.6673	1.3510	1.1183	0.4602	0.4602	3.1672
8.	Allianz pojišťovna, a. s.	1.2042	1.0954	∞	∞	1.1462	1.1255	8.4470	8.4470
9.	BNP Paribas Cardif Pojišťovna, a.s.	1.9863	1.2612	2.1852	3.8621	1.8077	1.5872	7.1743	7.1743
10.	Česká podnikatelská pojišťovna, a. s.	1.4321	1.3345	1.4757	2.3631	1.4588	1.3981	2.3467	6.3876
11.	ČSOB Pojišťovna, a. s.	1.4411	0.8574	∞	∞	1.4426	0.8609	1.3494	7.4337
12.	SV pojišťovna, a.s.	1.4072	2.3763	6.5531	7.8409	∞	∞	∞	∞
13.	Generali Pojišťovna a. s.	∞	1.0618	1.0618	1.1408	1.7742	1.0452	1.0452	2.6001
14.	Hasičská vzájemná pojišťovna, a. s.	1.2238	1.1743	1.2439	3.4628	∞	∞	∞	∞
15.	Komerční pojišťovna, a. s.	∞	∞	∞	∞	3.2481	2.7980	3.0244	5.8857
16.	Kooperativa pojišťovna, a.s.	∞	∞	∞	∞	1.8666	1.8790	2.7439	2.7439
17.	MAXIMA pojišťovna, a. s.	1.1132	1.0411	2.1726	3.9726	1.2071	1.3393	2.4100	2.4100
18.	Simplea pojišťovna, a.s.	1.8476	1.4288	1.4288	1.4288	∞	∞	∞	∞
19.	UNIQA pojišťovna, a. s.	11.3025	4.3906	∞	∞	10.4154	8.9431	11.5199	30.0449

Tabuľka 22: Porovnanie superefektivít v prípadových štúdiách 1-8. Znak ∞ sme priradili prípadom, v ktorých príslušná úloha superefektivity bola neprípustná.

Hodnoty superefektivít vo výstupne orientovaných modeloch SE^O_{BCC}, $SE^O R-VRS$ a SE^O_{FDH} sa odlišujú rovnako ako pri vstupne orientovaných modeloch. Pri jednotkách, ktoré boli efektívne vo výstupne orientovaných štúdiách, pozorujeme systematický nárast hodnôt v smere od konzervatívneho BCC po benevolentnejší FDH model. Výnimočné sú však najmä tie prípady, kde všetky tri modely priraďujú danej jednotke nekonečnú hodnotu superefektivity. Takéto výsledky dosiahli BNP Paribas Cardif Poisťovňa, SV pojišťovna, Hasičská vzájemná pojišťovna a Simplea pojišťovna. Nekonečná hodnota znamená, že žiadna iná jednotka - ani ich kombinácia - nedosahuje podobný pomer výstupov k vstupom ako tieto DMU.

Celkovo možno konštatovať, že výpočet superefektivity priniesol dôležitý pohľad na relatívnu výkonnosť efektívnych jednotiek a umožnil vytvoriť ich jemnejšie poradie. Rozdiely medzi modelmi navyše poukazujú na citlivosť hodnotenia voči voľbe DEA prístupu. Kým BCC model ostáva najkonzervatívnejší, FDH model priraďuje najvyššie hodnoty a R-VRS model zohľadňuje pomerové premenné vyváženejším spôsobom.

Záver

Cieľom tejto diplomovej práce bolo spracovať prehľad prístupov k aplikácii DEA modelov na pomerové miery, detailne analyzovať vybrané metodiky a demonštrovať ich použitie na reálnych údajoch komerčných poisťovní.

V teoretickej časti práce sme v prvej kapitole zhrnuli základné koncepty DEA modelov, predstavili sme BCC model, ktorý vychádza z predpokladu variabilných výnosov z rozsahu a objasnili koncept superefektivity ako prostriedku na rozlíšenie medzi efektívnymi rozhodovacími jednotkami. Pri definovaní základných termínov sme vychádzali najmä z [3].

V druhej kapitole sme sa zamerali na špecifiká pomerových vstupov a výstupov, taktiež problémom vznikajúcim pri ich použití v klasických DEA modeloch, keďže tieto modely na ne nie sú priamo koncipované. Predstavili sme alternatívne prístupy, ktoré boli navrhnuté na riešenia týchto problémov a zaviedli sme predpis množiny produkčných možností M^R_{VRS}, ktorá zohľadňuje špecifiká pomerových mier.

Tretia kapitola sa venovala FDH modelu, pričom sme vychádzali zo štúdií [1] a [4]. Ukázali sme, že ak vstupy a výstupy pozostávajú výhradne z pomerových veličín, potom množina M^R_{VRS}, ktorá rešpektuje špecifiká pomerových dát, zodpovedá množine M_{FDH}. V úvode kapitoly sme v príklade 3.1 porovnali množiny M^R_{VRS} a M_{FDH} na rovnakých údajoch, aby sme demonštrovali rozdiely v ich tvare a rozsahu. Zvláštnu pozornosť sme venovali vizuálnemu porovnaniu efektívnostných hraníc. Pri vstupne aj výstupne orientovaných FDH modeloch sme znázornili projekcie neefektívnych jednotiek na hranicu efektívnosti. Tieto vizualizácie boli vytvorené v prostredí Python a slúžili ako intuitívna pomôcka pre lepšie pochopenie geometrických vlastností FDH modelov.

Vo štvrtej kapitole sme sa detailne venovali modifikovaným R-VRS modelom podľa článkov [5] a [6], ktoré umožňujú konzistentné spracovanie pomerových veličín v rámci DEA. Pracovali sme so vstupne aj výstupne orientovanými verziami týchto modelov, pričom ich základné formulácie sme čerpali priamo zo spomínaných zdrojov. Následne sme ich modifikovali pomocou binárnych premenných z_j a dostatočne veľkých konštánt M_1 a M_2, čím sme dostali úlohy zmiešaného celočíselného programovania, s ktorými sme ďalej pracovali. Venovali sme sa aj druhej fáze týchto modelov, pričom sme aj tieto úlohy upravili do formy úloh zmiešaného celočíselného programovania, ako sme to urobili pri

prvej fáze. Pri oboch variantoch, aj vstupnom aj výstupnom, sme znázornili projekcie rozhodovacích jednotiek na efektívne hranice pomocou 3D vizualizácií, ktoré sme vytvorili v prostredí Python. Tieto grafické výstupy slúžili ako dôležitý nástroj na intuitívne pochopenie správania modelov pri práci s pomerovými veličinami.

V piatej kapitole sme sa venovali viacerým odborným článkom, ktoré riešia otázku spracovania pomerových veličín v DEA. Nezostali sme pri ich jednoduchom vymenovaní, ale pri každom sme stručne opísali, aký konkrétny problém identifikuje a aké riešenie navrhuje. Vďaka tomu kapitola poskytuje ucelený prehľad prístupov z literatúry k tejto tematike.

V poslednej kapitole sme modely BCC, FDH a R-VRS aplikovali na dáta slovenských a českých komerčných poisťovní. V každej z ôsmich prípadových štúdií výsledky ukázali, že väčšina poisťovní bola označená ako efektívna, preto sme do analýzy zahrnuli aj výpočet superefektivity, aby sme medzi nimi mohli vytvoriť porovnanie. Výsledky poukázali na to, že zmena typu dát (objemové vs. pomerové) a voľba modelu majú významný vplyv na výsledné hodnotenie efektívnosti. Ukázalo sa, že niektoré modely môžu nadhodnocovať alebo podhodnocovať efektívnosť v závislosti od toho, ako pracujú s pomerovými ukazovateľmi. Všetky výpočty boli realizované v prostredí Python, pričom sme využili solver Gurobi. Ten bol nevyhnutný najmä pri výpočte R-VRS modelov, ktoré predstavujú úlohy zmiešaného celočíselného programovania. Výsledky boli opäť vizualizované pomocou grafov a diagramov, čím sme sa snažili o čo najprehľadnejšiu interpretáciu výstupov.

Za hlavné prínosy práce považujeme jednak teoretické rozšírenie poznatkov o modeloch DEA v oblasti pomerových mier, ako aj praktické demonštrovanie ich aplikovateľnosti na aktuálne údaje z komerčného sektora. Naša analýza poukázala na konkrétne výhody a obmedzenia jednotlivých metodík, čo môže slúžiť ako základ pre ďalší výskum a optimalizáciu hodnotiacich procesov v poisťovníctve.

Možným rozšírením práce by mohlo byť doplnenie analýzy o ďalšie prístupy publikované v najnovšej literatúre alebo zahrnutie väčšieho počtu poisťovní s rôznorodými typmi poistných produktov, čo by umožnilo detailnejšie hodnotenie efektívnosti v rôznych segmentoch poisťovacieho trhu.

Zoznam použitej literatúry

[1] Deprins, D., Simar, L., Tulkens, H.: *Measuring labor-efficiency in post offices*, The performance of public enterprises: Concepts and measurements (1984), 243-267

[2] Dyson, R. G., et al.: *Pitfalls and protocols in DEA*, European Journal of Operational Research 132.2 (2001), 245–259

[3] Halická, M.: *DEA modely*, učebné texty, FMFI UK Bratislava, 2015

[4] Lim, B., Lee, K., Lee, C.: *Free Disposal Hull (FDH) analysis for efficiency measurement: An update to DEA*, The Stata Journal 10.2 (2016), 1-8

[5] Olesen, O. B., Petersen, N. C., Podinovski, V. V.: *Efficiency analysis with ratio measures*, European Journal of Operational Research 245 (2015), 446-462

[6] Olesen, O. B., Petersen, N. C., Podinovski, V. V.: *Efficiency measures and computational approaches for data envelopment analysis models with ratio inputs and outputs*, European Journal of Operational Research 261 (2017), 640-655

[7] Olesen, O. B., Petersen, N. C., Podinovski, V. V.: *Scale characteristics of variable returns-to-scale production technologies with ratio inputs and outputs*, Annals of Operations Research 318 (2022), 383-423

[8] Papaioannou, G., Podinovski, V. V.: *Production technologies with ratio inputs and outputs*, European Journal of Operational Research 310 (2023), 1164-1178

[9] Podinovski, V. V.: *Selective convexity in DEA models*, European Journal of Operational Research 161 (2005), 552-563

[10] Podinovski, V. V., Wu, J., Argyris, N.: *Production trade-offs in models of data envelopment analysis with ratio inputs and outputs: An application to schools in England*, European Journal of Operational Research 313 (2024), 359-372

[11] Thanassoulis, E., Boussofiane, A., Dyson, R. G.: *Exploring output quality targets in the provision of perinatal care in England using data envelopment analysis*, European Journal of Operational Research 80 (1995), 588-607

[12] Thrall, R. M.: *What Is the Economic Meaning of FDH?*, Journal of Productivity Analysis 11(3) (1999), 243–250

Príloha A

A.1 Vstupne orientovaný BCC model

```
# Nacitanie Excel suboru
df = pd.read_excel("data.xlsx", sheet_name="Sheet1")

# vstupy (mxn)
X = df[['Naklady z poistnych sluzieb', 'Pocet zamestnancov', 'Zavazky na
    aktivum']].to_numpy().T

# vystupy (sxn)
Y = df[['Vynosy z poistnych sluzieb', 'ROE', 'ROA']].to_numpy().T

# Rozmery
m = X.shape[0]  # Pocet vstupov
s = Y.shape[0]  # Pocet výstupov

# Pocet DMUs
n = X.shape[1]

efficiency = np.zeros(n)
lambda_values = np.zeros((n, n))
input_slack = np.zeros((n, m))
output_slack = np.zeros((n, s))

for o in range(n):
    x_o = X[:, o]
    y_o = Y[:, o]

    model = gp.Model("BCC-I")

    theta = model.addVar(vtype=GRB.CONTINUOUS, name="theta")
    lambdas = model.addVars(n, vtype=GRB.CONTINUOUS, lb=0, name="lambda")

    model.setObjective(theta, GRB.MINIMIZE)

```

```
    # Obmedzenia pre vstupy
    for i in range(m):
        model.addConstr(gp.quicksum(X[i, j] * lambdas[j] for j in range(n))
    <= theta * x_o[i])

    # Obmedzenia pre výstupy
    for r in range(s):
        model.addConstr(gp.quicksum(Y[r, j] * lambdas[j] for j in range(n))
    >= y_o[r])

    model.addConstr(gp.quicksum(lambdas[j] for j in range(n)) == 1)

    model.optimize()

    if model.status == GRB.OPTIMAL:
        efficiency[o] = theta.x
        for j in range(n):
            lambda_values[o, j] = lambdas[j].x

result = pd.DataFrame()
for j in range(n):
    result[f'lambda_{j+1}'] = lambda_values[:, j]
print("Efektivity pre vsetky DMU:")
print(efficiency)
print(result)

input_slack = np.zeros((n, m))
output_slack = np.zeros((n, s))
lambda_values = np.zeros((n, n))

for o in range(n):
    x_o = X[:, o]
    y_o = Y[:, o]

    theta_opt = efficiency[o]

    model_phase2 = gp.Model("BCC-II")

```

```
    lambda_vars_phase2 = model_phase2.addVars(n, vtype=GRB.CONTINUOUS, lb=0, name="lambda")
    sx_phase2 = model_phase2.addVars(m, vtype=GRB.CONTINUOUS, lb=0, name="sx")
    sy_phase2 = model_phase2.addVars(s, vtype=GRB.CONTINUOUS, lb=0, name="sy")

    model_phase2.setObjective(
        gp.quicksum(sx_phase2[i] for i in range(m)) + gp.quicksum(sy_phase2[r] for r in range(s)),
        GRB.MAXIMIZE
    )

    for i in range(m):
        model_phase2.addConstr(gp.quicksum(X[i, j] * lambda_vars_phase2[j] for j in range(n)) + sx_phase2[i] == theta_opt * x_o[i])

    for r in range(s):
        model_phase2.addConstr(gp.quicksum(Y[r, j] * lambda_vars_phase2[j] for j in range(n)) - sy_phase2[r] == y_o[r])

    model_phase2.addConstr(gp.quicksum(lambda_vars_phase2[j] for j in range(n)) == 1)

    model_phase2.optimize()

    if model_phase2.status == GRB.OPTIMAL:
        for i in range(m):
            input_slack[o, i] = sx_phase2[i].x
        for r in range(s):
            output_slack[o, r] = sy_phase2[r].x
        for j in range(n):
            lambda_values[o, j] = lambda_vars_phase2[j].x

result = pd.DataFrame()
for j in range(n):
    result[f'lambda_{j+1}'] = lambda_values[:, j]
for i in range(m):
```

```
    result[f'input_slack_{i+1}'] = input_slack[:, i]
for r in range(s):
    result[f'output_slack_{r+1}'] = output_slack[:, r]

print("Lambda hodnoty a slacky pre vsetky DMU:")
print(result)
```

A.2 Výstupne orientovaný BCC model

```
efficiency = np.zeros(n)
psi_values = np.zeros(n)
lambda_values = np.zeros((n, n))

# Iterácia cez vsetky DMU
for o in range(n):
    x_o = x[:, o]
    y_o = y[:, o]

    model = gp.Model("BCC-O")

    psi = model.addVar(vtype=GRB.CONTINUOUS, name="psi")
    lambdas = model.addVars(n, vtype=GRB.CONTINUOUS, lb=0, name="lambda")
    sx = model.addVars(m, vtype=GRB.CONTINUOUS, lb=0, name="sx")
    sy = model.addVars(s, vtype=GRB.CONTINUOUS, lb=0, name="sy")

    model.setObjective(psi, GRB.MAXIMIZE)

    for i in range(m):
        model.addConstr(gp.quicksum(x[i, j] * lambdas[j] for j in range(n))
    <= x_o[i])

    for r in range(s):
        model.addConstr(gp.quicksum(y[r, j] * lambdas[j] for j in range(n))
    >= psi * y_o[r])

    model.addConstr(gp.quicksum(lambdas[j] for j in range(n)) == 1)

```

```
    model.optimize()

    if model.status == GRB.OPTIMAL:
        efficiency[o] = 1 / psi.x if psi.x > 0 else np.nan
        psi_values[o] = psi.x
        for j in range(n):
            lambda_values[o, j] = lambdas[j].x

result = pd.DataFrame()
for j in range(n):
    result[f'lambda_{j+1}'] = lambda_values[:, j]

print("Efektivity pre vsetky DMU:")
print(efficiency)
print(result)

input_slack = np.zeros((n, m))
output_slack = np.zeros((n, s))
lambda_values = np.zeros((n, n))

for o in range(n):
    x_o = x[:, o]
    y_o = y[:, o]

    psi_opt = psi_values[o]

    model_phase2 = gp.Model("BCC-II")

    lambda_vars_phase2 = model_phase2.addVars(n, vtype=GRB.CONTINUOUS, lb
   =0, name="lambda")
    sx_phase2 = model_phase2.addVars(m, vtype=GRB.CONTINUOUS, lb=0, name="
   sx")
    sy_phase2 = model_phase2.addVars(s, vtype=GRB.CONTINUOUS, lb=0, name="
   sy")

    model_phase2.setObjective(
        gp.quicksum(sx_phase2[i] for i in range(m)) + gp.quicksum(sy_phase2
   [r] for r in range(s)),
```

```
        GRB.MAXIMIZE
    )

    for i in range(m):
        model_phase2.addConstr(gp.quicksum(x[i, j] * lambda_vars_phase2[j]
    for j in range(n)) + sx_phase2[i] == x_o[i])

    for r in range(s):
        model_phase2.addConstr(gp.quicksum(y[r, j] * lambda_vars_phase2[j]
    for j in range(n)) - sy_phase2[r] == psi_opt * y_o[r])

    model_phase2.addConstr(gp.quicksum(lambda_vars_phase2[j] for j in range
    (n)) == 1)

    model_phase2.optimize()

    if model_phase2.status == GRB.OPTIMAL:
        for i in range(m):
            input_slack[o, i] = sx_phase2[i].x
        for r in range(s):
            output_slack[o, r] = sy_phase2[r].x
        for j in range(n):
            lambda_values[o, j] = lambda_vars_phase2[j].x

# Výsledky v ľtabuke
result = pd.DataFrame()
for j in range(n):
    result[f'lambda_{j+1}'] = lambda_values[:, j]
for i in range(m):
    result[f'input_slack_{i+1}'] = input_slack[:, i]
for r in range(s):
    result[f'output_slack_{r+1}'] = output_slack[:, r]

# Zobrazenie výsledkov
print("Lambda hodnoty a slacky pre vsetky DMU:")
print(result)
```

A.3 Vstupne orientovaný R-VRS model

```
volume_inputs = [0,1,2,3]   # Indexy objemových vstupov
ratio_inputs = [4]  # Indexy pomerových vstupov
volume_outputs = [0,1,2]  # Indexy objemových výstupov
ratio_outputs = [3,4]   # Indexy pomerových výstupov

theta_inputs = [0,1,2,3,4]  # Indexy výstupov násobených theta

# Dostatocne velka konstanta M
M1=float(np.max(x))
M2=float(np.max(y))

efficiency = np.zeros(n)
psi_opt = np.zeros(n)

for o in range(n):
    x_o = x[:, o]
    y_o = y[:, o]

    model = gp.Model("DEA_I_VRS")

    theta = model.addVar(vtype=GRB.CONTINUOUS, name="theta")
    lambda_vars = model.addVars(n, vtype=GRB.CONTINUOUS, lb=0, name="lambda
    ")
    z_vars = model.addVars(n, vtype=GRB.BINARY, name="z")  # binarna
    premenna

    model.setObjective(theta, GRB.MINIMIZE)

    # Obmedzenia pre vstupy
    for i in range(len(x)):
        if i in theta_inputs and i in volume_inputs:
            model.addConstr(sum(x[i,j] * lambda_vars[j] for j in range(n))
    <= theta * x_o[i])
        elif i in theta_inputs and i in ratio_inputs:
            for j in range(n):
                model.addConstr(lambda_vars[j] <= z_vars[j])
```

```
                model.addConstr(x[i,j] - theta * x_o[i] <= M1 * (1 - z_vars
    [j]))
        elif i in volume_inputs:
            model.addConstr(sum(x[i,j] * lambda_vars[j] for j in range(n))
    <= x_o)
        elif i in ratio_inputs:
            for j in range(n):
                model.addConstr(lambda_vars[j] * (x[i,j] - x_o[i]) <= 0)

    for i in range(y.shape[0]):
        if i in volume_outputs:
            model.addConstr(sum(y[i, j] * lambda_vars[j] for j in range(n))
     >= y_o[i])
        elif i in ratio_outputs:
            for j in range(n):
                model.addConstr(lambda_vars[j] * (y[i, j] - y_o[i]) >= 0)

    model.addConstr(sum(lambda_vars[j] for j in range(n)) == 1)

    model.optimize()

    if model.status == GRB.OPTIMAL:
        efficiency[o] = theta.x

# Výpis výsledkov
print("Efektivity pre vsetky DMU:")
print(efficiency)
input_slack = np.zeros((n, len(x)))
output_slack = np.zeros((n, len(y)))
lambda_values = np.zeros((n, n))

for o in range(n):
    x_o = x[:, o]
    y_o = y[:, o]

    theta_opt = efficiency[o]

    model = gp.Model("R_VRS_I_Second_Phase")
```

```
69
70      s_x = model.addVars(len(x), vtype=GRB.CONTINUOUS, lb=0, name="s_x")
71      s_y = model.addVars(len(y), vtype=GRB.CONTINUOUS, lb=0, name="s_y")
72      lambda_vars = model.addVars(n, vtype=GRB.CONTINUOUS, lb=0, name="lambda
        ")
73      z_vars = model.addVars(n, vtype=GRB.BINARY, name="z")
74
75      model.setObjective(
76          gp.quicksum(s_x[i] for i in range(len(x))) +
77          gp.quicksum(s_y[r] for r in range(len(y))),
78          GRB.MAXIMIZE
79      )
80
81      # Obmedzenia pre vstupy
82      for i in range(x.shape[0]):
83          if i in theta_inputs and i in volume_inputs:
84              model.addConstr(sum(x[i,j] * lambda_vars[j] for j in range(n))
         + s_x[i] <= theta_opt * x_o[i])
85          elif i in theta_inputs and i in ratio_inputs:
86              for j in range(n):
87                  model.addConstr(lambda_vars[j] <= z_vars[j])
88                  model.addConstr(x[i,j] - (theta_opt * x_o[i] - s_x[i]) <=
        M1 * (1 - z_vars[j]))
89          elif i in volume_inputs:
90              model.addConstr(sum(x[i,j] * lambda_vars[j] for j in range(n))
         + s_x[i] <= x_o)
91          elif i in ratio_inputs:
92              for j in range(n):
93                  model.addConstr(lambda_vars[j] * (x[i,j] - (x_o[i] - s_x[i
        ])) <= 0)
94      # Obmedzenia pre výstupy
95      for r in range(len(y)):
96          if r in volume_outputs:
97              model.addConstr(sum(y[r, j] * lambda_vars[j] for j in range(n))
         - s_y[r] >= y_o[r])
98          elif r in ratio_outputs:
99              for j in range(n):
100                 model.addConstr(lambda_vars[j] <= z_vars[j])
```

```
                model.addConstr(y[r, j] - (y_o[r] + s_y[r]) >= -M2 * (1 -
    z_vars[j]))

    model.addConstr(sum(lambda_vars[j] for j in range(n)) == 1)

    for i in range(len(x)):
        model.addConstr(s_x[i] >= 0)
    for r in range(len(y)):
        model.addConstr(s_y[r] >= 0)

    model.optimize()

    if model.status == GRB.OPTIMAL:
        for i in range(len(x)):
            input_slack[o, i] = s_x[i].x
        for r in range(len(y)):
            output_slack[o, r] = s_y[r].x
        for j in range(n):
            lambda_values[o, j] = lambda_vars[j].x
```

A.4 Výstupne orientovaný R-VRS model

```
volume_inputs = [0,1]   # Indexy objemových vstupov
ratio_inputs = [2]     # Indexy pomerových vstupov
volume_outputs = [0]  # Indexy objemových výstupov
ratio_outputs = [1,2]   # Indexy pomerových výstupov

# Definovanie indexov premenných ovplyvnených psi (O_psi)
psi_outputs = [0,1,2]

# čDostatone ľveká škontanta M
M1=float(np.max(x))
M2=float(np.max(y))
efficiency = np.zeros(n)
psi_opt = np.zeros(n)

for o in range(n):
```

```
    x_o = x[:, o]
    y_o = y[:, o]

    model = gp.Model("DEA_R_VRS")

    p = model.addVar(vtype=GRB.CONTINUOUS, name="p")  # Efektivita
    lambda_vars = model.addVars(n, vtype=GRB.CONTINUOUS, lb=0, name="lambda
") # lambda_j >= 0
    z_vars = model.addVars(n, vtype=GRB.BINARY, name="z")  # Binárne
premenné

    model.setObjective(p, GRB.MAXIMIZE)

    # Obmedzenia pre vstupy
    for i in range(len(x)):
        if i in volume_inputs:
            model.addConstr(sum(x[i, j] * lambda_vars[j] for j in range(n))
 <= x_o[i])
        elif i in ratio_inputs:
            for j in range(n):
                model.addConstr(lambda_vars[j] * (x[i, j] - x_o[i]) <= 0)

    # Obmedzenia pre výstupy
    for i in range(y.shape[0]):
        if i in psi_outputs and i in volume_outputs:
            model.addConstr(sum(y[i, j] * lambda_vars[j] for j in range(n))
 >= p * y_o[i])
        elif i in psi_outputs and i in ratio_outputs:
            for j in range(n):
                model.addConstr(lambda_vars[j] <= z_vars[j])
                model.addConstr(y[i, j] - p * y_o[i] >= -M2 * (1 - z_vars[j
]))
        elif i in volume_outputs:
            model.addConstr(sum(y[i, j] * lambda_vars[j] for j in range(n))
 >= y_o[i])
        elif i in ratio_outputs:
            for j in range(n):
                model.addConstr(lambda_vars[j] * (y[i, j] - y_o[i]) >= 0)
```

```
48
49      model.addConstr(sum(lambda_vars[j] for j in range(n)) == 1)
50
51      model.optimize()
52
53      if model.status == GRB.OPTIMAL:
54          efficiency[o] = 1 / p.x
55          psi_opt[o] = p.x
56
57  # Výsledky
58  print("Efektivita:\n", efficiency)
59  print("Psi optimal:\n", psi_opt)
60
61  input_slack = np.zeros((n, len(x)))
62  output_slack = np.zeros((n, len(y)))
63  lambda_values = np.zeros((n, n))
64
65  for o in range(n):
66      x_o = x[:, o]
67      y_o = y[:, o]
68
69      psi_opt = 1/efficiency[o]
70
71      model = gp.Model("R_VRS_I_Second_Phase")
72
73      s_x = model.addVars(len(x), vtype=GRB.CONTINUOUS, lb=0, name="s_x")
74      s_y = model.addVars(len(y), vtype=GRB.CONTINUOUS, lb=0, name="s_y")
75      lambda_vars = model.addVars(n, vtype=GRB.CONTINUOUS, lb=0, name="lambda
        ")
76      z_vars = model.addVars(n, vtype=GRB.BINARY, name="z")
77
78      model.setObjective(
79          gp.quicksum(s_x[i] for i in range(len(x))) +
80          gp.quicksum(s_y[r] for r in range(len(y))),
81          GRB.MAXIMIZE
82      )
83
84      # Obmedzenia pre vstupy
```

```
    for i in range(len(x)):
        if i in volume_inputs:
            model.addConstr(sum(x[i, j] * lambda_vars[j] for j in range(n))
    + s_x[i] <= x_o[i])
        elif i in ratio_inputs:
            for j in range(n):
                model.addConstr(lambda_vars[j] <= z_vars[j])
                model.addConstr(x[i, j] - (x_o[i] - s_x[i]) <= M1 * (1 -
    z_vars[j]))

    # Obmedzenia pre výstupy
    for r in range(y.shape[0]):
        if r in psi_outputs and r in volume_outputs:
            model.addConstr(sum(y[r, j] * lambda_vars[j] for j in range(n))
    - s_y[r] >= psi_opt * y_o[r])
        elif r in psi_outputs and r in ratio_outputs:
            for j in range(n):
                model.addConstr(lambda_vars[j] <= z_vars[j])
                model.addConstr(y[r, j] - (psi_opt * y_o[r] + s_y[r]) >= -
    M2 * (1 - z_vars[j]))
        elif r in volume_outputs:
            model.addConstr(sum(y[r, j] * lambda_vars[j] for j in range(n))
    - s_y[r]>= y_o[i])
        elif r in ratio_outputs:
            for j in range(n):
                model.addConstr(lambda_vars[j] <= z_vars[j])
                model.addConstr(y[r, j] - (y_o[r] + s_y[r]) >= -M2 * (1 -
    z_vars[j]))


    model.addConstr(sum(lambda_vars[j] for j in range(n)) == 1)

    for i in range(len(x)):
        model.addConstr(s_x[i] >= 0)
    for r in range(len(y)):
        model.addConstr(s_y[r] >= 0)

    model.optimize()
```

```
117
118     if model.status == GRB.OPTIMAL:
119         for i in range(len(x)):
120             input_slack[o, i] = s_x[i].x
121         for r in range(len(y)):
122             output_slack[o, r] = s_y[r].x
123         for j in range(n):
124             lambda_values[o, j] = lambda_vars[j].x
125
126 result = pd.DataFrame()
127 for j in range(n):
128     result[f'lambda_{j+1}'] = lambda_values[:, j]
129 for i in range(len(x)):
130     result[f'input_slack_{i+1}'] = input_slack[:, i]
131 for r in range(len(y)):
132     result[f'output_slack_{r+1}'] = output_slack[:, r]
133 pd.set_option('display.max_columns', None)
134 result
```

A.5 Vstupne orientovaný FDH model

```
1  import numpy as np
2  import pandas as pd
3  import gurobipy as gp
4  from gurobipy import GRB
5
6  # Inicializácia výsledkov
7  efficiency = np.zeros(n)
8  lambda_values = np.zeros((n, n))
9
10 for o in range(n):
11     x_o = x[:, o]
12     y_o = y[:, o]
13
14     model = gp.Model("FDH")
15     model.setParam('OutputFlag', 0)
16
```

```
    # Premenné
    theta = model.addVar(vtype=GRB.CONTINUOUS, name="theta")
    lambdas = model.addVars(n, vtype=GRB.BINARY, name="lambda")

    model.setObjective(theta, GRB.MINIMIZE)

    # Obmedzenia pre vstupy
    for i in range(m):
        model.addConstr(gp.quicksum(x[i, j] * lambdas[j] for j in range(n))
    <= theta * x_o[i])

    # Obmedzenia pre výstupy
    for r in range(s):
        model.addConstr(gp.quicksum(y[r, j] * lambdas[j] for j in range(n))
    >= y_o[r])

    model.addConstr(gp.quicksum(lambdas[j] for j in range(n)) == 1)

    model.optimize()

    if model.status == GRB.OPTIMAL:
        efficiency[o] = theta.X
        for j in range(n):
            lambda_values[o, j] = lambdas[j].X

result = pd.DataFrame()
for j in range(n):
    result[f'lambda_{j+1}'] = lambda_values[:, j]

result['Eff'] = efficiency
print("Efektivity pre vsetky DMU:")
print(result.round(4))

input_slack = np.zeros((n, m))
output_slack = np.zeros((n, s))
lambda_values = np.zeros((n, n))

for o in range(n):
```

```
    x_o = x[:, o]
    y_o = y[:, o]

    theta_opt = efficiency[o]

    model_phase2 = gp.Model("FDH_I-II")

    lambda_vars_phase2 = model_phase2.addVars(n, vtype=GRB.BINARY, name="lambda")
    sx_phase2 = model_phase2.addVars(m, vtype=GRB.CONTINUOUS, lb=0, name="sx")
    sy_phase2 = model_phase2.addVars(s, vtype=GRB.CONTINUOUS, lb=0, name="sy")

    model_phase2.setObjective(
        gp.quicksum(sx_phase2[i] for i in range(m)) + gp.quicksum(sy_phase2[r] for r in range(s)),
        GRB.MAXIMIZE
    )

    for i in range(m):
        model_phase2.addConstr(gp.quicksum(x[i, j] * lambda_vars_phase2[j] for j in range(n)) + sx_phase2[i] == theta_opt * x_o[i])

    for r in range(s):
        model_phase2.addConstr(gp.quicksum(y[r, j] * lambda_vars_phase2[j] for j in range(n)) - sy_phase2[r] == y_o[r])

    model_phase2.addConstr(gp.quicksum(lambda_vars_phase2[j] for j in range(n)) == 1)

    model_phase2.optimize()

    if model_phase2.status == GRB.OPTIMAL:
        for i in range(m):
            input_slack[o, i] = sx_phase2[i].x
        for r in range(s):
            output_slack[o, r] = sy_phase2[r].x
```

```
        for j in range(n):
            lambda_values[o, j] = lambda_vars_phase2[j].x

result = pd.DataFrame()
for j in range(n):
    result[f'lambda_{j+1}'] = lambda_values[:, j]
for i in range(m):
    result[f'input_slack_{i+1}'] = input_slack[:, i]
for r in range(s):
    result[f'output_slack_{r+1}'] = output_slack[:, r]

print("Lambda hodnoty a slacky pre vsetky DMU:")
print(result)
```

A.6 Výstupne orientovaný FDH model

```
efficiency = np.zeros(n)
psi_values = np.zeros(n)
lambda_values = np.zeros((n, n))

# Iterácia cez vsetky DMU
for o in range(n):
    x_o = x[:, o]
    y_o = y[:, o]

    model = gp.Model("BCC-O")

    psi = model.addVar(vtype=GRB.CONTINUOUS, name="psi")
    lambdas = model.addVars(n, vtype=GRB.BINARY, name="lambda")
    sx = model.addVars(m, vtype=GRB.CONTINUOUS, lb=0, name="sx")
    sy = model.addVars(s, vtype=GRB.CONTINUOUS, lb=0, name="sy")

    model.setObjective(psi, GRB.MAXIMIZE)

    for i in range(m):
        model.addConstr(gp.quicksum(x[i, j] * lambdas[j] for j in range(n))
    <= x_o[i])
```

```
21
22      for r in range(s):
23          model.addConstr(gp.quicksum(y[r, j] * lambdas[j] for j in range(n))
        >= psi * y_o[r])
24
25      model.addConstr(gp.quicksum(lambdas[j] for j in range(n)) == 1)
26
27      model.optimize()
28
29      if model.status == GRB.OPTIMAL:
30          efficiency[o] = 1 / psi.x if psi.x > 0 else np.nan
31          psi_values[o] = psi.x
32          for j in range(n):
33              lambda_values[o, j] = lambdas[j].x
34
35  result = pd.DataFrame()
36  for j in range(n):
37      result[f'lambda_{j+1}'] = lambda_values[:, j]
38
39  print("Efektivity pre vsetky DMU:")
40  print(efficiency)
41  print(result)
42
43  input_slack = np.zeros((n, m))
44  output_slack = np.zeros((n, s))
45  lambda_values = np.zeros((n, n))
46
47  for o in range(n):
48      x_o = x[:, o]
49      y_o = y[:, o]
50
51      psi_opt = psi_values[o]
52
53      model_phase2 = gp.Model("FDH-II")
54
55      lambda_vars_phase2 = model_phase2.addVars(n, vtype=GRB.BINARY, name="
       lambda")
56      sx_phase2 = model_phase2.addVars(m, vtype=GRB.CONTINUOUS, lb=0, name="
```

```
    sx")
57      sy_phase2 = model_phase2.addVars(s, vtype=GRB.CONTINUOUS, lb=0, name="
    sy")
58
59      model_phase2.setObjective(
60          gp.quicksum(sx_phase2[i] for i in range(m)) + gp.quicksum(sy_phase2
    [r] for r in range(s)),
61          GRB.MAXIMIZE
62      )
63
64      for i in range(m):
65          model_phase2.addConstr(gp.quicksum(x[i, j] * lambda_vars_phase2[j]
    for j in range(n)) + sx_phase2[i] == x_o[i])
66
67      for r in range(s):
68          model_phase2.addConstr(gp.quicksum(y[r, j] * lambda_vars_phase2[j]
    for j in range(n)) - sy_phase2[r] == psi_opt * y_o[r])
69
70      model_phase2.addConstr(gp.quicksum(lambda_vars_phase2[j] for j in range
    (n)) == 1)
71
72      model_phase2.optimize()
73
74      if model_phase2.status == GRB.OPTIMAL:
75          for i in range(m):
76              input_slack[o, i] = sx_phase2[i].x
77          for r in range(s):
78              output_slack[o, r] = sy_phase2[r].x
79          for j in range(n):
80              lambda_values[o, j] = lambda_vars_phase2[j].x
81
82  # Výsledky v ľtabuke
83  result = pd.DataFrame()
84  for j in range(n):
85      result[f'lambda_{j+1}'] = lambda_values[:, j]
86  for i in range(m):
87      result[f'input_slack_{i+1}'] = input_slack[:, i]
88  for r in range(s):
```

```
    result[f'output_slack_{r+1}'] = output_slack[:, r]

# Zobrazenie výsledkov
print("Lambda hodnoty a slacky pre vsetky DMU:")
print(result)
```