UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

NÁVRH A VYPRACOVANIE WEBOVÉHO ROZHRANIA PRE INTERAKTÍVNU MOŽNOSŤ RIEŠENIA DEA ÚLOH

BAKALÁRSKA PRÁCA

2013 **Radomír PROFANT**

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

NÁVRH A VYPRACOVANIE WEBOVÉHO ROZHRANIA PRE INTERAKTÍVNU MOŽNOSŤ RIEŠENIA DEA ÚLOH

BAKALÁRSKA PRÁCA

Študijný program:	Ekonomická a finančná matematika
Študijný odbor:	1114 Aplikovaná matematika
Školiace pracovisko:	Katedra aplikovanej matematiky a štatistiky
Vedúci práce:	prof. RNDr. Daniel Ševčovič, CSc.

Bratislava 2013

Radomír PROFANT

62343450

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Radomír Profant
Študijný program: ekonomická a finančná matematika (Jednoodborové štúdium, bakalársky I. st., denná forma)
Študijný odbor: 9.1.9. aplikovaná matematika
Typ záverečnej práce: bakalárska
Jazyk záverečnej práce: slovenský

Názov: Návrh a vypracovanie webového rozhrania pre interaktívnu možnosť riešenia DEA úloh

Cieľ: Cieľom bude vytvoriť webovú aplikáciu v jazyku PHP na báze otvoreného GNU softvéru Octave (klon systému Matlab) so zameraním na riešenie úloh Data Envelopment Analýzy efektivity.

Vedúci: prof. RNDr. Daniel Ševčovič, CSc.
Katedra: FMFI.KAMŠ - Katedra aplikovanej matematiky a štatistiky
Vedúci katedry: prof. RNDr. Daniel Ševčovič, CSc.

Dátum zadania: 10.10.2012

Dátum schválenia: 03.11.2012 doc. RNDr. Margaréta Halická, CSc.
garant študijného programu

....
študent vedúci práce

Poďakovanie

Touto cesto sa chcem poďakovať svojmu vedúcemu bakalárskej práce prof. RNDr. Danielovi Ševčovičovi, CSc. za ochotu, podnetné pripomienky a odbornú pomoc, ktorá mi pomohla pri písaní tejto práce. Poďakovanie patrí aj doc. RNDr. Margaréte Halickej, CSc. za podnetné pripomienky pri finalizácii práce. V neposlednom rade ďakujem aj svojej rodine a priateľom za ich trpezlivosť a podporu.

Abstrakt v štátnom jazyku

PPROFANT, Radomír: *Návrh a vypracovanie webového rozhrania pre interaktívnu možnosť riešenia DEA úloh* [Bakalárska práca], Univerzita Komenského v Bratislave, Fakulta matematiky, fyziky a informatiky, Katedra aplikovanej matematiky a štatistiky; školiteľ: prof. RNDr. Daniel Ševčovič, CSc., Bratislava, 2013, 49 s.

Bakalárska práca sa zaoberá vytvorením webového rozhrania pre interaktívnu možnosť riešenia DEA (Data Envelopment Analysis) úloh. V práci sú zadefinované a vysvetlené dôležité pojmy a tvrdenia, stručný prehľad základných modelov a spôsob ich implementácie do webového rozhrania. Súčasťou práce je vysvetlenie logických schém, ktoré sú použité pri programovaní webovej stránky a modelov. Hlbšie sa venujeme programovaniu jednotlivých modelov v programovacom jazyku GNU Octave a dôležitých častí webovej stránky v programovacom jazyku HTML s využitím jazyka PHP. Vytvorená aplikácia je použitá na riešenie praktickej úlohy hodnotenia bankových pobočiek Slovenskej sporiteľne. Na tomto príklade je zároveň demonštrovaná funkčnosť a využitie webovej stránky.

Kľúčové slová: DEA (Data Envelopment Analysis), webové rozhranie, GNU Octave, PHP

Abstrakt v cudzom jazyku

PPROFANT, Radomír: *Design and develop web interface for interacting possibility of solving DEA task* [Bachelor Thesis], Comenius University in Bratislava, Faculty of Mathematics, Physics and Informatics, Department of Applied Mathematics and Statistics; Supervisor: prof. RNDr. Daniel Ševčovič, CSc., Bratislava, 2011, 49p.

My Bachelor's Thesis is dedicated to the development of a web interface for interacting possibility of solving DEA (Data Envelopment Analysis) tasks. In this work are defined and explained important concepts and statements, brief overview of the basic models and their implementation in web interface. Part of the work is composed of the explanation of logic schemes that are used in programming of the website and models. More in detail are described programming of respective models in GNU Octave programming language and relevant parts of the website in the programming language HTML using PHP language. The developed application is used to solve practical task of evaluation of bank branches of Slovak Savings Bank. The functionality and use of the website is also demonstrated on this example.

Keywords: DEA (Data Envelopment Analysis), web interface, GNU Octave, PHP

Obsah

Zoznam ilustrácií

Zoznam tabuliek

Úvod

Súčasná doba, v ktorej žijeme, je veľmi poznačená ekonomickou krízou. Kríza sa dotkla väčšiny ekonomických a výrobných sfér. Často skloňovaným slovom v tomto období je efektivita. Štáty sú nútené efektívnejšie hospodáriť s verejnými financiami, firmy optimalizujú a zefektívňujú výrobné procesy. Efektivitu by sme mohli chápať ako praktickú účinnosť nejakej činnosti. Postupne sa stáva hlavným kritériom pri posudzovaní resp. hodnotení.

Hodnoteniu efektivity sa venuje rozsiahla teória multikriteriálneho manažmentu: *Data Envelopment Analysis* (ďalej DEA). Táto teória vznikla v sedemdesiatych rokoch minulého storočia. Základom bol veľmi známy článok od Farrella z roku 1957. DEA sa zameriava na výpočet technickej efektivity rôznych organizačných jednotiek napríklad firiem, bánk, škôl, zdravotníckych zariadení a mnohých iných. Sú to v podstate modely matematického programovania aplikované na rôzne vstupné a výstupné dáta s cieľom zistiť efektivitu konkrétnych jednotiek.

V súčasnej dobe je už známych veľa modelov a článkov, ktoré boli prezentované a citované v rôznych vedeckých prácach. Existujú aj rozličné softvéry na riešenie DEA úloh. Niektoré sú však drahé, alebo sú zložité pre používateľov, ktorí nie sú dostatočne matematicky zdatní.

Cieľom tejto bakalárskej práce bolo priblížiť niektoré zo základných modelov DEA a vypracovať interaktívne webové rozhranie na riešenie DEA úloh so zameraním na jednoduché zadávanie vstupov a zrozumiteľnú, prehľadnú prezentáciu výstupov užívateľovi. Cieľom bolo čo možno najjednoduchšie a najpohodlnejšie zobraziť výstup ako efektivitu zo vstupných dát.

Bakalárska práca je rozdelená na dve hlavné kapitoly: teoretickú a praktickú. Cieľom teoretickej časti bolo korektným spôsobom prezentovať základné matematické modely DEA s poukázaním na dôležitosť odhaľovania neefektívnosti jednotlivých útvarov. Kapitola je rozdelená na tri podkapitoly. V prvej sa venujeme úvodu do DEA problematiky, kde si vysvetlíme pojem efektívnosti. Druhá kapitola je venovaná príkladom, na ktorých si vysvetlíme pojmy používané v DEA analýze. V tretej sa budeme venovať základným DEA modelom. Pri prezentácií teórie a skúmaní jednotlivých modelov sa budeme voľne opierať o knihu Coopera, Seiforda, Toneho [1], knihu Zhua [10], článok Ševčoviča, Halickej a Brunovského [8] a poznámky

Halickej [4]. V praktickej časti sme sa sústredili na implementovanie jednotlivých modelov do webovej stránky tak, aby výsledky boli jasné a zrozumiteľné pre používateľov. Táto časť je rozdelená na tri podkapitoly. V prvej si vysvetlíme logický model webovej stránky. V ďalšej kapitole sa budeme venovať jednotlivým modelom naprogramovaným v programovacom jazyku GNU Octave. V tretej kapitole si ukážeme samotné programovanie webovej stránky.

1 Teoretická časť

1.1 Úvod do DEA modelov

DEA metóda, tak ako sme spomenuli v úvode, je významný prostriedok multikriteriálneho manažmentu. Umožňuje porovnávať a vyhodnocovať homogénne útvary (jednotky) v rámci danej skupiny. Jednotlivé útvary sa označujú ako DMU z anglického *Decision making units.* Ako príklad môžu slúžiť filiálky bánk, pobočky supermarketov, nemocničné zariadenia alebo školy. Veľmi významným predpokladom pre jednotlivé DMU je homogenita, teda že sú z nejakej rovnorodej skupiny, majú identické alebo ekvivalentné vlastnosti, zaoberajú sa rovnakou činnosťou. Tieto útvary sú charakterizované vstupmi a výstupmi. Vstupmi môžu byť napr. materiál, počet zamestnancov, počiatočné náklady alebo predajná plocha. Vo všeobecnosti je to niečo, čo sa spotrebúva alebo predstavuje náklady. Výstupom môže byť napr. celkový zisk alebo počet spokojných zákazníkov. Vo všeobecnosti niečo, čo predstavuje úžitok alebo kvalitu. Dá sa usúdiť, že sa snažíme pri minimálnych vstupoch dosahovať maximálne výstupy, a teda mať čo najväčšiu efektivitu.

1.1.1 Pojem efektívnosti

Cieľom DEA metódy je rozdeliť jednotlivé útvary na efektívne a neefektívne podľa toho, ako dokážu jednotlivé vstupy premeniť na výstupy. Veľkou výhodou DEA modelov je, že dokážu odhaliť zdroj neefektívnosti a v podstate navrhnúť riešenie ako dosiahnuť hranicu efektívnosti.

Najskôr si vysvetlíme pojem produktivity, pretože veľmi úzko súvisí s efektivitou. V najjednoduchšom prípade, teda pre jeden vstup a jeden výstup sa produktivita vyjadruje všeobecne ako pomer:

$$produktivita = \frac{výstup}{vstup}. \tag{1}$$

Efektivita je potom vyjadrená podľa vzorca:

$$efektivita = \frac{produktivita\ jednotky}{maximálna\ produktivita}. \tag{2}$$

Zo vzťahu (1) a (2) je veľmi jednoduché vypočítať efektivitu pre prípad jedného vstupu a jedného výstupu. Ako by sme však počítali efektivitu pre viac vstupov alebo výstupov? Riešením je do vzorca (1) dosadiť za vstup vážený súčet vstupov a za výstup vážený súčet výstupov, teda násobiť vstupy a výstupy určitými kladnými číslami (váhami). Do popredia sa dostáva otázka voľby váh pre jednotlivé vstupy a výstupy. Tomu sa budeme podrobnejšie venovať v kapitole 1.2.4.1.

Kvôli prezentácii výsledku má zmysel uvažovať efektivitu v hraniciach 0 a 1, vtedy môžeme ľahko previesť efektivitu na percentá. Takto definovanú efektivitu dokážeme prezentovať tak, že na koľko percent jednotka pracuje, resp. ako sa daná jednotka približuje k efektívnej jednotke. Pomocou DEA metód nevieme určiť maximálnu možnú teoretickú efektivitu, vieme určiť len jej odhad pomocou útvaru s najväčšou produktivitou. DEA metódy odhaľujú len relatívnu efektivitu, teda efektivitu vzhľadom na dané DMU.

Definícia 1 *(Efektivita) [1 str.45]: Maximálna (100%) efektivita pre jednotlivé DMU je dosiahnutá vtedy a len vtedy, ak žiadny zo vstupov alebo výstupov nemožno zlepšiť bez zhoršenia iných vstupov alebo výstupov.*

1.2 Ilustračné príklady

V tejto časti sa budeme sústrediť na nami vytvorené ilustračné príklady, na ktorých si zadefinujeme a vysvetlíme pojmy dôležité pri budovaní teórie. Výsledky a pozorovania budeme zobrazovať graficky. Pri zobrazení sa pre názornosť a jednoduchosť obmedzíme na dva rozmery, keďže je zložité zobrazovať viacero rozmerov.

1.2.1 Prípad jedného vstupu a jedného výstupu

Príklad 1 *(Sieť supermarketov): Majme 7 pobočiek supermarketu. Každá pobočka je charakterizovaná počtom zamestnancov a priemernými dennými tržbami v tisíckach eur. Vypočítajme produktivitu a efektivitu pre dané jednotky. Potrebné údaje sú uvedené v Tab. 1:*

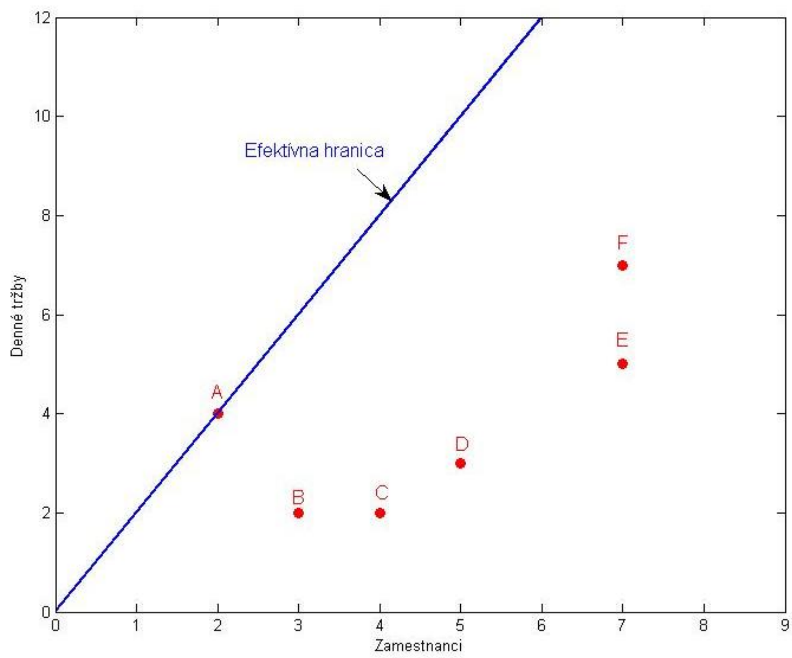
Tab. 1: Údaje k príkladu 1 – sieť supermarketov

Supermarkety	A	B	C	D	E	F
Zamestnanci (x)	2	3	4	5	7	7
Tržby (y)	4	2	2	3	5	7
Produktivita $(\frac{y}{x})$	2	0,67	0,5	0,6	0,7	1
Efektivita $(\frac{\frac{y}{x}}{\max(\frac{y}{x})})$	1	0,335	0,25	0,3	0,35	0,5

V prvom riadku sú jednotlivé DMU, teda pobočky supermarketu. Ako vstupy sme si logicky zvolili počet zamestnancov, ako výstupy priemerné denné tržby. Produktivitu sme potom počítali podľa vzorca (1) ako podiel tržieb a zamestnancov. Najväčšiu produktivitu dosiahol supermarket A. Efektivitu sme počítali podľa vzorca (2) ako podiel produktivity a maximálnej produktivity. Efektívny je teda supermarket A, ktorý má efektivitu 1, ostatné sú neefektívne, číslo v poslednom riadku vyjadruje relatívnu mieru efektivity.

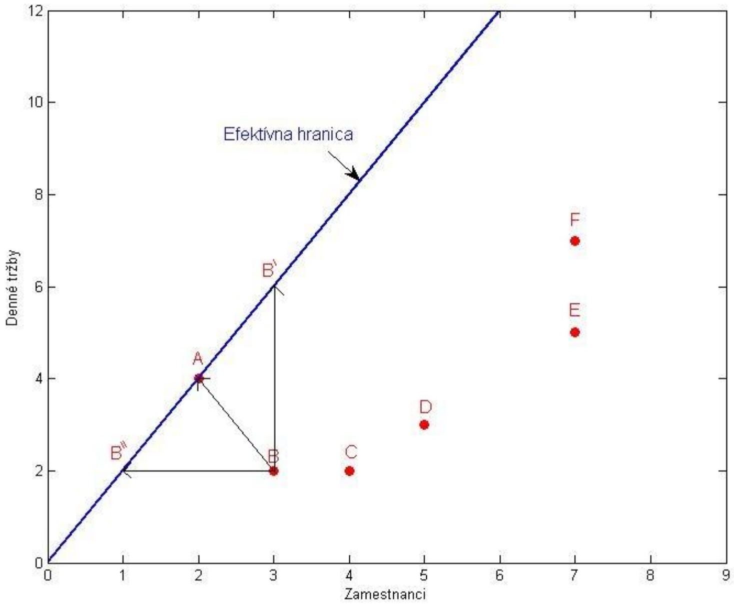
Na nasledujúcom Obr. 1, sú znázornené jednotlivé DMU aj s hranicou efektivity. V našom prípade konštantných výnosov z rozsahu je to priamka prechádzajúca bodom [0,0] a bodom efektívnej jednotky A[2,4]. Ak by chceli byť ostatné jednotky efektívne, museli by dosiahnuť rovnakú produktivitu ako jednotka A. Museli by sa teda dostať na hranicu efektivity.

Definícia 2 *(Hranica efektivity): Hranicou efektivity alebo efektívnou hranicou nazývame množinu efektívnych DMU v rámci danej skupiny.*



Obr. 1: Graf jednotlivých DMU a efektívna hranica

Často sa uvažuje, že organizačné jednotky DMU sa snažia dostať na efektívnu hranicu. Ide teda o akúsi projekciu na efektívnu hranicu, resp. ako sa majú zmeniť vstupy alebo výstupy, aby sa jednotka, v našom prípade supermarket, zefektívnili. Ako príklad si zoberme supermarket B. Existuje rada možností ako danú jednotku zefektívniť: Môžeme znížiť vstupy pri zachovaní výstupu. V našom prípade by sme museli znížiť počet zamestnancov z pôvodných 3 na 1 (Obr. 2 bod B``) s tým, že tržby by zostali rovnaké. Ďalšou možnosťou je zachovať počet zamestnancov a zvýšiť tržby z pôvodnej hodnoty 2 na hodnotu 6 (Obr. 2 bod B`). Ako posledná možnosť sa javí znížiť vstupy a zvýšiť výstupy, tým sa dostaneme niekde medzi prvú a druhú možnosť, napr. do bodu A [2,4] (Obr. 2 bod A). Všetky možnosti sú znázornené na Obr. 2.



Obr. 2: Projekcia na efektívnu hranicu

Poznámka 1 *(Projekcia na efektívnu hranicu): Nech vstupy sú x, výstupy y, zmena vstupov* x'*, zmena výstupov* y'*. Vo všeobecnosti projektovať na efektívnu hranicu znamená, že znižujeme vstupy alebo zvyšujeme výstupy. Jednotku* (x, y) *teda projektujeme na jednotku* (x', y') *tak, že platí* $x' \leq x, y' \geq y$.

1.2.2 Množina produkčných možností a výnosy z rozsahu

DEA metódy sú založené na tom, že pre daný problém existuje produkčná množina (množina produkčných možností) tvorená všetkými prípustnými kombináciami vstupov a výstupov. Produkčná množina je teda určená efektívnou hranicou. Má však zmysel uvažovať o nejakých obmedzeniach produkčnej množiny. Predstavme si, že by sme do Tab. 1 v príklade 1 doplnili supermarket Z. Bol by v menšom meste, zamestnával by napr. 70 pracovníkov a jeho tržby by boli 100. Dostať ho na efektívnu hranicu by bolo veľmi ťažké resp. nemožné. Keby zamestnal ďalších ľudí, tak tržby by sa nemuseli zmeniť, pretože dopyt v malom meste je obmedzený počtom obyvateľov. Vynára sa teda otázka, čo je, a čo nie je možné vyrobiť.

Definícia 2 *(Axiómy produkčnej množiny) [5 str.7]: Nech x je vektor vstupov, y vektor výstupov. Technológiou nazveme dvojicu* (x, y)*, ktorá označuje výstupy y, ktoré je možné vyrobiť spotrebovaním vstupov x za nejakú jednotku času. Potom p je produkčná množina (množina všetkých technológií, množina produkčných možností) ak sú splnené axiómy:*

A1: Pozorované technológie (x^j, y^j) *pre j=1,2,...,n jednotlivých DMU patria do produkčnej množiny P.*

A2: Ak $(x, y) \in P$*, tak* $\forall\, x', y' : x' \leq x, y' \geq y$ *platí, že* $(x', y') \in P$*. Teda ak neznížime vstupy alebo nezvýšime výstupy, dostaneme realizovateľnú technológiu.*

A3: P je konvexná, t.j. konvexné kombinácie technológii sú tiež realizovateľnými technológiami.

Produkčná množina má konštantné výnosy z rozsahu, ak navyše platí:

A4: Nezáporný skalárny násobok $(\alpha x, \alpha y) \in P$ *pre každú technológiu* $\forall (x, y) \in P$ *a pre ľubovoľné* $\alpha \in \mathbb{R}, \alpha \geq 0$.

Z definície a z predchádzajúcich kapitol vyplýva, že hranica efektivity môže mať rôzny tvar. Tento tvar závisí na tom, či v danom probléme uvažujeme konštantné alebo variabilné výnosy z rozsahu.

V definícii 2 sme si zadefinovali pojem konštantné výnosy z rozsahu. Voľne by sa táto definícia dala vysvetliť aj tak, že pokiaľ je jednotka, ktorá vznikla určitou kombináciou vstupov a výstupov efektívna, tak bude efektívna aj jednotka, ktorej vstupy a výstupy sú α násobkom vstupov a výstupov pôvodnej efektívnej jednotky. Pri konštantných výnosoch je teda efektívna hranica priamka. Tento prípad je zobrazený na Obr. 1.

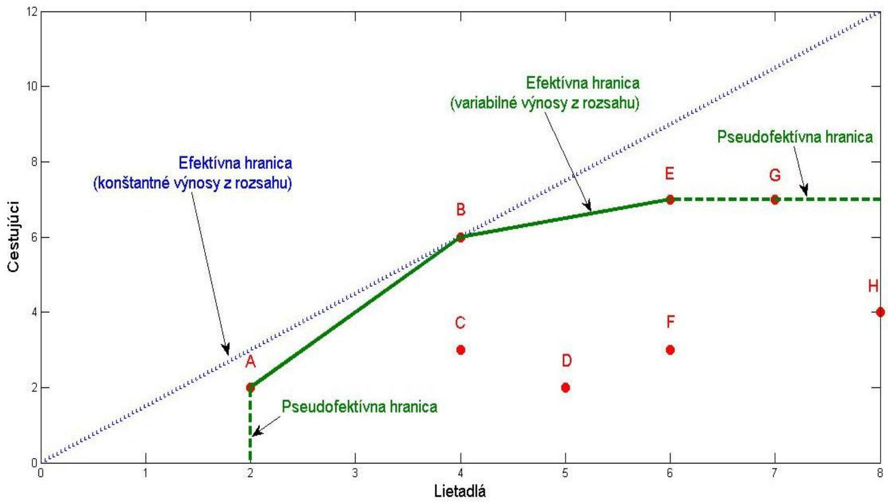
Pri variabilných výnosoch z rozsahu neplatí, že α násobok vstupu musí byť vyvážený nárastom výstupu o rovnaký násobok. Jednotka teda môže byť efektívna, i keď pomerný nárast výstupu bude nižší alebo vyšší ako nárast vstupu. Kvôli názornosti sme vytvorili príklad 2, kde si výpočtom aj graficky ukážeme možnosť variabilných výnosov z rozsahu a porovnanie s konštantnými výnosmi z rozsahu.

Príklad 2 *(Letecká preprava): Predstavme si, že máme 8 leteckých prepravcov označených A až H v 8 mestách s rovnakým počtom obyvateľov. Úlohou je vypočítať efektivitu a graficky zobraziť efektívnu hranicu. Údaje k príkladu sú v Tab. 2.*

Tab. 2: Údaje k príkladu 2 - letecký prepravcovia

Lietadlá	A	B	C	D	E	F	G	H
Počet lietadiel	2	4	4	5	6	6	7	8
Počet cestujúcich	2	6	3	2	7	3	7	4
Produktivita	1	1,5	0,75	0,4	1,17	0,5	1	0,5
Efektivita (konštantné)	0,67	1	0,5	0,27	0,78	0,34	0,67	0,34
Efektivita (variabilné)	áno	áno			áno			

V prípade leteckých prepravcov by sme mali uvažovať variabilné výnosy z rozsahu, pretože počet prepravených cestujúcich je obmedzený počtom obyvateľov. To znamená, že aj keby prepravca nakúpil nové lietadlá, nemal by nimi kto cestovať. Na Obr. 3 sme graficky zobrazili daných prepravcov a vyznačili sme hranice efektivity pre variabilné výnosy, a kvôli názornosti aj pre konštantné výnosy z rozsahu. V prípade prepravcu B dosahujeme efektivitu pre oba typy výnosov. Táto jednotka má teda globálne najväčšiu efektivitu. Jednotky A a E dosahujú taktiež efektívnu hranicu, majú lokálne najvyššiu produktivitu. Pre variabilné výnosy máme tri efektívne jednotky A,B,E. Efektívna hranica tvorí teda obal dát, ktorý je konvexný. Všeobecne platí, že keď je jednotka efektívna pre konštantné výnosy z rozsahu, tak je efektívna aj pre variabilné, naopak to však z povahy veci neplatí. Veľmi zaujímavou jednotkou je jednotka G, ktorá síce leží na hranici efektívnosti, ale v porovnaní s jednotkou E má pri tom istom výstupe väčšie vstupy. Jednotka G by sa dala vyrábať s menším počtom vstupov, nemôže byť teda efektívna. Takéto jednotky nazývame pseudoefektívne.



Obr. 3: Efektívna hranica, konštantné a variabilné výnosy z rozsahu

1.2.3 Označenie dát

Doteraz sme sa venovali jednoduchým úlohám s jedným vstupom a výstupom. Ak chceme prejsť od jednoduchých úloh k zložitejším, musíme zaviesť niekoľko označení.

Jednotlivé DMU budeme označovať spodnými indexmi $DMU_1, DMU_2, \dots , DMU_n$. Počet DMU bude teda n. Optimalizovanú jednotku označíme ako DMU_o. Údaje, ktoré sa týkajú jednotlivých DMU, budeme označovať takto:

$$vstupy : x^o = (x_1^o, x_2^o, \dots., x_m^o)^T \tag{3}$$

$$výstupy : y^o = (y_1^o, y_2^o, \dots., y_s^o)^T. \tag{4}$$

Vstupov bude m, výstupov s a $x_j^o(y_j^o)$ budeme niekedy označovať aj $x_{oj}(y_{oj})$. $x^o(y^o)$ budeme označovať ako vektor vstupov (výstupov) pre DMU_o. Jeho technológiu budeme označovať ako (x^o, y^o). Celú sadu dát teda môžeme označiť do dvoch matíc, vstupnej $X \in \mathbb{R}^{n\times m}$ a výstupnej $Y \in \mathbb{R}^{n\times s}$.

$$X = \begin{Bmatrix} x_1^1 & x_2^1 & x_3^1 & \dots & x_m^1 \\ x_1^2 & x_2^2 & x_3^2 & \dots & x_m^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_1^n & x_2^n & x_3^n & \dots & x_m^n \end{Bmatrix} \tag{5}$$

$$Y = \begin{Bmatrix} y_1^1 & y_2^1 & y_3^1 & \dots & y_m^1 \\ y_1^2 & y_2^2 & y_3^2 & \dots & y_m^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_1^n & y_2^n & y_3^n & \dots & y_m^n \end{Bmatrix}. \tag{6}$$

Vnútorné ocenenie vstupov a výstupov, resp. váhy jednotlivých vstupov a výstupov budeme označovať pomocou vektorov:

$$váhy\ vstupov : u^o = (u_1^o, u_2^o, \dots., u_m^o)^T \tag{7}$$

$$váhy\ výstupov : v^o = (v_1^o, v_2^o, \dots., v_s^o)^T. \tag{8}$$

Takto zadefinované označenia nám pomôžu pri riešení viacrozmerných úloh alebo úloh s viacerými vstupmi a výstupmi.

1.2.4 Prípad s viacerými vstupmi a výstupmi

Predpokladajme, že máme n organizačných jednotiek DMU_o, pre $o = \{1, \dots, n\}$, ďalej predpokladajme, že každá organizačná jednotka je charakterizovaná m vstupmi zadefinovanými v (3) a s výstupmi zadefinovanými v (4). Vstupy a výstupy sú homogénne, kladné (niekedy stačí nezáporné) a každé DMU_o má hodnotu aspoň jedného vstupu a výstupu. Každý zo vstupov a výstupov má určité kladné ocenenia resp. váhy, ktoré sú zadefinované podľa (7) a (8). Na začiatok si pevne zvoľme váhy u a v, potom môžeme dosadiť vážený súčet vstupov a váh, resp. výstupov a váh do vzorca (1), vypočítať produktivitu a s pomocou vzorca (2) vypočítať efektivitu.

Definícia 3 *(Produktivita) [5 str. 10]: Nech* $v \in \mathbb{R}_{+}^{m}$, $u \in \mathbb{R}_{+}^{s}$ *sú dané kladné ohodnotenia (váhy) vstupov a výstupov. Produktivitou útvaru* DMU_o *(pre* $o \in \{1,2,\dots,n\}$*) pri daných váhach u, v budeme rozumieť:*

$$P^o(u,v) = \frac{u^T y^o}{v^T x^o} = \frac{\sum_{j=1}^{s} u_j y_j^o}{\sum_{i=1}^{m} v_i x_i^o} \tag{9}$$

kde x^o *je vektor vstupov a* y^o *je vektor výstupov jednotky o a u, v sú váhy pre konkrétne vstupy a výstupy.*

Definícia 4 *(Efektivita) [5 str. 10]: Efektivitou jednotky pri daných váhach u, v budeme rozumieť pomer jej produktivity a najväčšej produktivity v skupine pri daných váhach, teda:*

$$E^o(u,v) = \frac{P^o(u,v)}{\max\limits_{\forall k} P^o(u,v)} = \frac{\frac{u^T y^o}{v^T x^o}}{\max\limits_{\forall k} \frac{u^T y^k}{v^T x^k}} \tag{10}$$

kde $\forall k$ *označuje celú skupinu n útvarov, takže* $k \in \{1,2,\dots,n\}$.

Z definícií (9) a (10) vyplýva, že $E^0(u,v) \in \langle 0,1 \rangle$, čo je veľmi dobrá vlastnosť pri prezentácií výsledkov. Triviálne z definícií platí aj: $E^o(u,v) = 1 \Leftrightarrow P^o(u,v) = max_{\forall k} P^o(u,v)$, teda keď je útvar efektívny, jeho produktivita je v danej skupine maximálna. Na príklade 3 si ukážeme výpočet efektivity.

Príklad 3 *(Požičovňa áut): Nech máme 6 požičovní áut označených A až F. K dispozícií máme mesačné údaje o dostupnom počte áut, počte zamestnancov, počte požičaných áut a počte spokojných zákazníkov. Úlohou je vypočítať efektivitu jednotlivých požičovní. Potrebné dáta sú v Tab. 3.*

Tab. 3: Údaje k príkladu 3 – požičovňa áut

Požičovňa áut	A	B	C	D	E	F
Dostupný počet áut	120	97	87	131	79	91
Počet zamestnancov	14	19	12	23	9	15
Počet požičaných áut	40	35	56	37	12	62
Počet spokojných zákazníkov	23	31	56	25	9	60

Predtým ako začneme riešiť daný problém podľa vzorcov (9) a (10), musíme si ujasniť, čo budú naše vstupy resp. výstupy a zvoliť primerané váhy. Rozumné je zvoliť si za vstupy počet áut a počet zamestnancov, pretože je to niečo, čo sa do procesu dodáva a predstavuje nejaké náklady. Váhy teda budeme mať dve: v_1, v_2. Výstupom bude počet požičaných áut a počet spokojných zákazníkov, pretože to sa v podstate produkuje, vyrába alebo predstavuje kvalitu (spokojnosť). Váhy výstupu teda budú: u_1, u_2.

Váhy si pre názornosť zvolíme dvoma spôsobmi:

- Jednotkové váhy: $\mathrm{v}_1 = 1, \mathrm{v}_2 = 1, \mathrm{u}_1 = 1, \mathrm{u}_2 = 1$
- Váhy odrážajúce dôležitosť: $\mathrm{v}_1 = 1, \mathrm{v}_2 = 2, \mathrm{u}_1 = 1, \mathrm{u}_2 = 2$

Na nasledujúcej Tab. 4 sa nachádza výsledok výpočtu produktivity podľa vzorca (9) a efektivity podľa vzorca (10) pre obe varianty váh. Ľahko sa dá vyčítať, že zmena váh spôsobila zmenu efektívnej jednotky. Kým pri jednotkových váhach bola efektívna jednotka F, pri váhach odrážajúcich dôležitosť bola efektívna jednotka C. Efektívne jednotky teda nevieme spoľahlivo určiť. Z tohto dôvodu by sme mali venovať zvýšenú pozornosť voľbe váh.

Tab. 4: Výsledky k príkladu 3 – požičovňa áut

Požičovňa áut	A	B	C	D	E	F
Produktivita (jednotkové váhy)	0,4701	0,5690	1,1313	0,4026	0,2386	1,1509
Efektivita (jednotkové váhy)	0,4085	0,4943	0,9829	0,3498	0,2073	**1,0000**
Produktivita (váhy odrážajúce dôležitosť)	0,5811	0,7185	1,5135	0,4915	0,3093	1,5041
Efektivita (váhy odrážajúce dôležitosť)	0,3839	0,4747	**1,0000**	0,3248	0,2043	0,9938

1.2.4.1 Voľba vhodných váh

Voľba vhodných váh je dôležitou otázkou. Ako zvoliť váhy, aby sme pristupovali k analyzovaným DMU spravodlivo? Nápadov by sme našli hneď niekoľko: napr. jednotkové váhy, váhy odrážajúce dôležitosť, prevrátenú hodnotu priemeru alebo prevrátenú maximálnu hodnotu. O žiadnej z týchto možností by sme však nemohli povedať, že je jediná alebo najlepšia. Použitím rôznych váh by sme dostali rôznu mieru efektivity podobne ako v príklade 3. Dobrým nápadom by bolo, aby si jednotlivé DMU samé zvolili váhy, ktoré by spĺňali určité podmienky. Dôležitou podmienkou je, aby miera efektivity bola zhora ohraničená nejakou kladnou konštantou (má zmysel uvažovať ako konštantu číslo jedna). Efektívne jednotky by boli potom tie, ktorých miera efektivity by sa rovnala danej konštante. Ďalšou podmienkou by bola kladnosť vstupov a výstupov, pretože všetky nami zvolené vstupy a výstupy považujeme za dôležité a musíme ich brať do úvahy. Za týchto predpokladov si môže každé DMU_o zvoliť vlastné váhy.

1.2.4.2 Príklad analýzy efektívnosti bankových pobočiek

V nasledujúcom praktickom príklade sa budeme venovať analýze bankových pobočiek Slovenskej sporiteľne vypracovanej podľa [8]. Z poskytnutých dát si musíme najprv ujasniť počet DMU, vstupy a výstupy. Máme 106 pobočiek, teda počet DMU je 106.

Vstupy poskytnuté bankou:

- suma debetných zostatkov na bežných účtoch ku koncu mesiaca
- počet debetných účtov
- suma zostatkov kontokorentných účtov
- počet kontokorentných účtov
- úvery celkom
- úvery v Sk
- úvery poskytnuté v cudzej mene
- náklady na bankové činnosti v roku yy
- celkové náklady spojené s prevádzkou filiálky
- osobné náklady
- všeobecné prevádzkové náklady
- náklady na prenájom priestorov filiálky
- mimoriadne náklady
- ostatné prevádzkové náklady

Vstupy – hlavné ukazovatele:

- celkové náklady spojené s prevádzkou filiálky
- osobné náklady
- ostatné prevádzkové náklady

Výstupy poskytnuté bankou:

- suma kreditných zostatkov na bežných účtoch ku koncu mesiaca
- počet bežných účtov
- suma všetkých typov zostatkov
- celkový počet všetkých účtov
- suma zostatkov všetkých typov vkladov spolu ku koncu mesiaca
- počet všetkých typov účtov agendy vkladov spolu ku koncu mesiaca
- hospodársky výsledok v roku
- výnosy z bankových činností v roku

Výstupy – hlavné ukazovatele:

- suma všetkých typov zostatkov
- celkový počet všetkých účtov
- suma zostatkov všetkých typov vkladov spolu ku koncu mesiaca
- počet všetkých typov účtov agendy vkladov spolu ku koncu mesiaca

Vybrali sme si z dát hlavné ukazovatele vstupov a výstupov, ktorých efektivitu budeme merať. Výsledky a riešenie daného príkladu spolu s grafmi a vysvetleniami si ukážeme v praktickej časti v kapitole 2.3.3.

1.3 DEA modely

V nasledujúcej kapitole sa budeme venovať základným modelom DEA, ktoré sme neskôr požívali aj pri realizácii praktickej časti. Pridŕžajúc sa [1] a [4] ukážeme odvodenie CCR a BCC modelu z koncepčného modelu. Podľa článku [8] si ukážeme modely s váhami.

1.3.1 Koncepčný model

V predchádzajúcich kapitolách sme uviedli úvahy, ktoré nám budú slúžiť na sformulovanie koncepčného modelu. Od tohto modelu sa v DEA teórií odvíja veľké množstvo modelov, preto je veľmi dôležitý. Chceme analyzovať efektívnosť jednotlivých DMU_{o}, pre $o = \{1, \dots, n\}$ pomocou koncepčného modelu, to znamená riešiť úlohu matematického programovania formulovanú takto:

(MP$_{\mathrm{o}}$)[1]

$$\max_{u \in \mathbb{R}^s, v \in \mathbb{R}^m} \frac{u^T y^o}{v^T x^o} \tag{11}$$

$$\frac{u^T y^j}{v^T x^j} \leq 1, \forall j = 1,2, \dots, n \tag{12}$$

$$v > 0 \tag{13}$$

$$u > 0. \tag{14}$$

Označme dvojicu (u^*, v^*) ako optimálne riešenie a $E_o^* = E_o(u^*, v^*)$ ako optimálnu hodnotu účelovej funkcie úlohy (MP$_{\mathrm{o}}$). Ak hodnota $E_o^* = 1$ hovoríme, že DMU_o je efektívny, inak je neefektívny a E_o^* udáva mieru efektívnosti. Ostré nerovnosti v (13) a (14) spôsobujú, že úloha nemusí mať vždy riešenie. V sofistikovanejších modeloch sa kladnosť u, v nahrádza nezápornosťou, prípadne sa váhy zdola ohraničujú veľmi malými kladnými číslami. Všimnime si, že úloha (11) - (14) je úlohou zlomkového programovania. Vyššie modely sú formulované v úlohe lineárneho programovania, to podrobnejšie rozvedieme v ďalších kapitolách.

1.3.2 CCR model

CCR model je najstarším DEA modelom, bol vytvorený Charnesom, Cooperom a Rhodesom v roku 1978. V CCR modeli uvažujeme konštantné výnosy z rozsahu. Najskôr si treba všimnúť, že pre optimálne riešenie (u^*, v^*) úlohy (MP$_{\mathrm{o}}$) triviálne platí, že aj $(\alpha u^*, \alpha v^*), \alpha > 0$ je optimálnym riešením. Vzniknutú nejednoznačnosť môžeme odstrániť podmienkou $v^T x^o = 1$. Normalizovali sme teda vstupy, a to nám umožní vynechať menovateľ z (11):

$$\max_{u \in \mathbb{R}^s, v \in \mathbb{R}^m} u^T y^o.$$

Ohraničenie (12) jednoducho prevedieme:

$$\frac{u^T y^j}{v^T x^j} \leq 1, \forall j = 1,2, \dots, n \Rightarrow u^T y^j - v^T x^j \leq 0, \forall j = 1,2, \dots, n.$$

[1] Koncepčný model

Ostré nerovnosti v (13), (14) nahradíme neostrými:

$$v \geq 0, u \geq 0,$$

čo treba zohľadniť, keď budeme interpretovať výsledky riešenia úloh. Keďže sme normalizovali vstupy, dostávame tvar vstupne orientovaného CCR modelu v primárnom tvare ako úlohu lineárneho programovania.

(CCR I P)[2]$_o$

$$\max_{u \in \mathbb{R}^s, v \in \mathbb{R}^m} u^T y^o$$
$$v^T x^o = 1$$
$$u^T y^j - v^T x^j \leq 0, \forall j = 1,2,\dots,n \tag{15}$$
$$v \geq 0$$
$$u \geq 0.$$

Definícia 5 *(Efektivita primárneho CCR modelu) [1 str. 24]: Nech* (u^*, v^*) *je optimálne riešenie úlohy (15) a platí, že váhy sú kladné (*$u^* > 0, v^* > 0$*) vo všetkých zložkách. Hovoríme, že testovaný útvar* DMU_o *je efektívny, ak hodnota účelovej funkcie* $E_o^* = 1$, *inak je neefektívny (hovoríme teda o miere efektivity).*

Poznámka 2 *(Pseudoefektívnosť): Nech* (u^*, v^*) *je optimálne riešenie úlohy (15) a platí, že aspoň jedna z váh je nulová (*$u^* = 0 \vee v^* = 0$*) v niektorej zložke. Hovoríme, že testovaný útvar* DMU_o *je pseudoefektívny, ak hodnota účelovej funkcie* $E_o^* = 1$, *inak hovoríme o miere pseudoefektivity. Táto vlastnosť sa nedá presne ekonomicky interpretovať. Dôležité je, že existujú spôsoby ako predísť tomuto problému, napr. metódou vnútorného bodu, váhami odrazenými od nuly alebo pomocou duálnej úlohy.*

Duálna úloha k úlohe (15) má tvar:

(CCR I D)[3]$_o$

$$\min_{\theta \in \mathbb{R}, \lambda \in \mathbb{R}^n} \theta \tag{16}$$

$$\sum_{i=1}^{n} \lambda_i x^i \leq \theta x^o \tag{17}$$

$$\sum_{i=1}^{n} \lambda_i y^i \geq y^o \tag{18}$$

$$\lambda \geq 0. \tag{19}$$

[2] Primárna úloha ku vstupne orientovanému CCR modelu
[3] Duálna úloha ku vstupne orientovanému CCR modelu

Duálna úloha má úzky súvis s produkčnou množinou, ktorú sme definovali v 1.2.2. Prostredníctvom λ_i n-rozmerného vektora skladáme ideálny útvar ako kombináciu iných útvarov, pričom tento útvar tiež patrí do množiny produkčných možností.

Definícia 6 *(Efektivita duálneho CCR modelu) [1 str. 24]: Nech θ^* je optimálne riešenie úlohy (19), a ohraničenia (17), (18) sú aktívne (dosahuje sa rovnosť) vo všetkých zložkách. Hovoríme, že testovaný útvar DMU_o je efektívny, ak hodnota $\theta^* = 1$, inak je neefektívny (hovoríme teda o miere efektivity).*

Odvodili sme teda primárnu a duálnu úlohu pre CCR vstupne orientovaný model. Odvodenie výstupne orientovaného modelu by bolo symetrické, tak ako je to uvedené v [7 str. 11]. Vo výstupných modeloch si však treba dať pozor na prezentáciu výsledkov, pretože nesmieme zabudnúť, že miera efektivity sa počíta ako prevrátená hodnota.

1.3.2.1 Rezervy v CCR modely

V duálnej úlohe by sme mohli mierne pozmeniť podmienky (17), (18), tým, že by sme pridali dve doplnkové premenné $s^x \in \mathbb{R}_+^m, s^y \in \mathbb{R}_+^s$. Dostali by sme ohraničenia v tvare rovností. Duálny CCR model by mal tvar:

(CCR I D)[4]$_o$

$$\min_{\theta \in \mathbb{R}, \lambda \in \mathbb{R}^n} \theta \tag{20}$$

$$\sum_{i=1}^{n} \lambda_i x^i + s^x = \theta x^o \tag{21}$$

$$\sum_{i=1}^{n} \lambda_i y^i - s^y = y^o \tag{22}$$

$$\lambda \geq 0 \tag{23}$$

$$s^x \geq 0 \tag{24}$$

$$s^y \geq 0. \tag{25}$$

Definícia 7 *(Efektivita duálneho CCR modelu podľa doplnkových premenných) [1 str. 45]: Nech θ^* je optimálne riešenie úlohy (20) a pre doplnkové premenné platí, že sú nulové ($s^x = 0, s^y = 0$) vo všetkých zložkách. Hovoríme, že testovaný útvar DMU_o je efektívny, ak hodnota $\theta^* = 1$, inak je neefektívny (hovoríme teda o miere efektivity).*

[4] Duálna úloha ku vstupne orientovanému CCR modelu s použitím rezerv

Premenné $s^x \in \mathbb{R}_+^m, s^y \in \mathbb{R}_+^s$ sa nazývajú aj rezervy *(angl. slacks)*, s^x vyjadruje nadmernú spotrebu vstupov a s^y vyjadruje stratenú výrobu, teda nedostatky pri výstupoch. Keby sme chceli odhaľovať neefektívnosť, mohli by sme tieto premenné pridať aj do účelovej funkcie, a to tak, že by sme ich násobili malou váhou ε. Tým by sme odstránili pseudoefektivitu spomínanú v poznámke 2. Účelovú funkciu (20) by sme nahradili funkciou:

$$\min_{\theta\in\mathbb{R},\lambda\in\mathbb{R}^n,s^x\in\mathbb{R}_+^m,s^y\in\mathbb{R}_+^s} \theta - \varepsilon\left(\sum_{i=1}^{m} s_i^X + \sum_{i=1}^{s} s_i^Y\right). \tag{26}$$

Treba si všimnúť, že ε musí byť dostatočne malé, aby neprevážilo θ, pretože v tom prípade by sme dostali neohraničenosť.

1.3.3 BCC model

V kapitole 1.2.2 sme spomenuli problém variabilných výnosov z rozsahu. Tomuto problému sa venuje BBC model, ktorý bol vytvorený v roku 1984 Bankerom, Charnesom a Cooperom. BCC model je odvodený z CCR modelu, presnejšie z úlohy (20) - (25), kde sme spomenuli akési skladanie (kombináciu) ideálneho útvaru pomocou λ_i. Ak chceme, aby táto kombinácia patrila do produkčnej množiny definovanej v 1.2.2, tak musí byť podľa axiómy (A3) konvexná. Konvexnosť zabezpečíme jednoducho, a to tak, že do podmienok (21) - (25) pridáme: $\sum_{i=1}^{n} \lambda_i = 1$. V prípade, že by sme o výnosoch z rozsahu vedeli, že sú rastúce, klesajúce alebo konštantné, mohli by sme podmienky upraviť nasledovne:

- rastúce výnosy z rozsahu - $\sum_{i=1}^{n} \lambda_i \geq 1$
- klesajúce výnosy z rozsahu - $\sum_{i=1}^{n} \lambda_i \leq 1$
- konštantné výnosy z rozsahu - $\sum_{i=1}^{n} \lambda_i = 1$

Duálna úloha k BCC modelu teda vyzerá nasledovne:

(BCC I D)[5]$_o$

$$\min_{\theta\in\mathbb{R},\lambda\in\mathbb{R}^n} \theta$$
$$\sum_{i=1}^{n}\lambda_i x^i + s^x = \theta x^o$$
$$\sum_{i=1}^{n}\lambda_i y^i - s^y = y^o \tag{27}$$
$$\sum_{i=1}^{n}\lambda_i = 1$$
$$\lambda \geq 0$$
$$s^x \geq 0$$
$$s^y \geq 0.$$

Ako definíciu efektivity môžeme využiť definíciu 7 pre duálnu úlohu a definíciu 5 pre primárnu úlohu. Primárna úloha ma tvar:

(BCC I P)[6]$_o$

$$\max_{u\in\mathbb{R}^s,v\in\mathbb{R}^m,v_o\in\mathbb{R}} u^T y^o + v_o$$
$$v^T x^o = 1$$
$$u^T y^j - v^T x^j + v_o \leq 0, \forall j = 1,2,\dots,n \tag{28}$$
$$v \geq 0$$
$$u \geq 0.$$

Odvodili sme teda primárnu a duálnu úlohu pre BCC vstupný model. Výstupný model je bližšie popísaný v [6 str. 23].

1.3.4 Modely s váhami

V tejto časti si vysvetlíme základné modely s váhami, t.j. aditívny model s váhami, vstupný a výstupný model s váhami. Všetky tieto modely sa odvíjajú od základného koncepčného modelu, a to v prvom rade tak, že podmienky (13) a (14) nahradíme takto:

$$v \geq w^x, u \geq w^y. \tag{29}$$

Vynára sa otázka, ako správne zvoliť minimálne váhy w^x, w^y.

[5] Duálna úloha ku vstupne orientovanému CCR modelu s použitím rezerv
[6] Primárna úloha ku vstupne orientovanému BCC modelu

1.3.4.1 Voľba vhodných minimálnych váh

Poznámka 3 *(Invariantnosť vzhľadom na zmenu jednotiek): Invariantnosť na zmenu jednotiek je veľmi dôležitá vlastnosť modelov (nie je však samozrejmosťou u všetkých modelov). Túto vlastnosť by sme mohli interpretovať tak, že hodnota účelovej funkcie sa nezmení pri zmene jednotiek. (napr. kilogramov na gramy). Hodnota účelovej funkcie sa teda nezmení, ak vynásobíme vstupy resp. výstupy určitou kladnou konštantou.*

Pri voľbe vhodných minimálnych váh $w^x,\ w^y$ musíme mať na pamäti poznámku 3. Váhy sa snažíme voliť ako nejakú štatistiku resp. funkciu dát. Najvhodnejším spôsobom voľby váh sa javí voľba váh ako prevrátenej hodnoty štandardnej odchýlky dát, ktoré máme k dispozícií:

$$w_i^x = \frac{1}{\sigma_i^x}, \forall i = 1, .., m \tag{30}$$

$$w_i^y = \frac{1}{\sigma_i^y}, \forall i = 1, .., s. \tag{31}$$

Štandardnú odchýlku σ_i^y i-teho výstupu vypočítame cez nevychýlený odhad podľa vzorca:

$$\sigma_i^y = \sqrt{\frac{1}{n-1}\sum_{o=1}^{n}(x_i^o - \overline{x_i})^2} \tag{32}$$

kde $\overline{x_i}$ je aritmetický priemer vypočítaný ako:

$$\overline{x_i} = \frac{1}{n}\sum_{o=1}^{n} x_i^o. \tag{33}$$

Ďalšie možnosti voľby váh sú popísané napr. v [7 str. 13].

1.3.4.2 Aditívny model s váhami

V úvode kapitoly 1.3.4 sme spomenuli, že modely z váhami sa dajú odvodiť z koncepčného modelu. Prvým krokom bola voľba minimálnych váh, ktorú sme ukázali v kapitole 1.3.4.1. Ďalej upravíme podmienku do tvaru rozdielu, podobne ako pri CCR modeli. Účelovú funkciu zmeníme z tvaru zlomku tak, že sa budeme zároveň snažiť minimalizovať vážený vstup a maximalizovať vážený výstup prostredníctvom ich rozdielu. V ekonomickej interpretácií budeme teda maximalizovať zisk. Nezameriavame sa preto len na vstupy alebo len na výstupy. Aditívny model ako úloha lineárneho programovania potom bude vyzerať nasledovne:

(AD P)[7]$_o$

$$\max_{u\in\mathbb{R}^s, v\in\mathbb{R}^m, v_0\in\mathbb{R}} u^T y^o - v^T x^o$$

$$u^T y^j - v^T x^j \leq 0, \forall j = 1,2,\dots,n$$

$$v \geq w^x \tag{34}$$

$$u \geq w^y.$$

Keďže sa jedná o úlohu lineárneho programovania, vieme k nej odvodiť príslušnú duálnu úlohu. Na duálnu úlohu sa však dá pozerať aj ako na zlepšenie modelu (20) - (25) tým, že vynecháme θ z ohraničení a účelovej funkcie (26). ε v účelovej funkcii nahradíme minimálnymi váhami w^x, w^y:

(AD D)[8]$_o$

$$\min_{\lambda\in\mathbb{R}^n, s^x\in\mathbb{R}^m_+, s^y\in\mathbb{R}^s_+} -((w^x)^T s^x + (w^y)^T s^y)$$

$$\sum_{i=1}^{n} \lambda_i x^i + s^x = x^o$$

$$\sum_{i=1}^{n} \lambda_i y^i - s^y = y^o \tag{35}$$

$$\lambda \geq 0$$

$$s^x \geq 0$$

$$s^y \geq 0.$$

Modely (34) a (35) popisujú situáciu konštantných výnosov z rozsahu. Situáciu pri variabilných výnosoch z rozsahu budeme riešiť podobne ako v kapitole 1.3.3 pridaním podmienky $\sum_{i=1}^{n} \lambda_i = 1$ do duálneho modelu (35), resp. pridaním premennej $z \in \mathbb{R}$ do ohraničenia v primárnom modeli (34).

Definícia 8 *(Efektivita AD modelu) [5 str. 21]: Nech E_o^* je hodnota účelovej funkcie. Hovoríme, že útvar je efektívny podľa aditívneho modelu, ak $E_o^* = 0$, inak je neefektívny, $E_o^* \in (-\infty, 0)$.*

Vidíme, že v tomto prípade je miera efektivity veľmi ťažko interpretovateľná. Našou snahou bude preškálovať túto mieru do intervalu $E_o^* \in \langle 0,1 \rangle$. Máme niekoľko možností ako to môžeme urobiť. Ukážeme si možnosť, ktorú sme následne použili aj v praktickej časti, iné možnosti sú popísané v [8]:

- podľa [8 str. 7]: $\mathrm{E}' = \frac{1}{\mathrm{m+s}}\left[\sum_{\mathrm{i=1}}^{\mathrm{m}}\left(\frac{\mathrm{x_i^o - s_i^x}}{\mathrm{x_i^o}}\right) + \sum_{\mathrm{i=1}}^{\mathrm{m}}\left(\frac{\mathrm{y_i^o}}{\mathrm{y_i^o + s_i^y}}\right)\right]$

[7] Primárna úloha k aditívnemu modelu (konštantné výnosy z rozsahu)
[8] Duálna úloha aditívnemu modelu (konštantné výnosy z rozsahu)

1.3.4.3 Vstupný a výstupný model s váhami

V tejto časti si ukážeme vstupný a výstupný model tak, ako sú definované v [8]. Tieto modely v princípe vychádzajú z CCR a BBC modelov. Obmedzíme sa na duálnu úlohu a variabilné výnosy z rozsahu, pretože v predchádzajúcich kapitolách sme jasne definovali prechod medzi primárnou a duálnou úlohou, a tak isto výnosy z rozsahu.

Vstupný model má tvar:

(I D)[9] $_o$

$$\min_{\theta\in\mathbb{R},\lambda\in\mathbb{R}^n,s^x\in\mathbb{R}_+^m,s^y\in\mathbb{R}_+^s} \theta - \varepsilon((w^x)^T s^x + (w^y)^T s^y)$$

$$\begin{aligned} \sum_{i=1}^{n} \lambda_i x^i + s^x &= \theta x^o \\ \sum_{i=1}^{n} \lambda_i y^i - s^y &= y^o \\ \sum_{i=1}^{n} \lambda_i &= 1 \\ \lambda &\geq 0 \\ s^x &\geq 0 \\ s^y &\geq 0. \end{aligned} \tag{36}$$

Výstupný model má tvar:

(O D)[10] $_o$

$$\max_{\theta\in\mathbb{R},\lambda\in\mathbb{R}^n,s^x\in\mathbb{R}_+^m,s^y\in\mathbb{R}_+^s} \theta + \varepsilon((w^x)^T s^x + (w^y)^T s^y)$$

$$\begin{aligned} \sum_{i=1}^{n} \lambda_i x^i + s^x &= x^o \\ \sum_{i=1}^{n} \lambda_i y^i - s^y &= \theta y^o \\ \sum_{i=1}^{n} \lambda_i &= 1 \\ \lambda &\geq 0 \\ s^x &\geq 0 \\ s^y &\geq 0. \end{aligned} \tag{37}$$

[9] Duálna úloha pre vstupný model (variabilné výnosy z rozsahu)
[10] Duálna úloha pre výstupný model (variabilné výnosy z rozsahu)

Efektivita vstupného modelu je definovaná podobne ako pre vstupné modely v predchádzajúcich kapitolách, t.j., že útvar je efektívny, ak hodnota účelovej funkcie $E_O^* = 1$, inak je neefektívny a miera efektivity je z intervalu $E_O^* \in \langle 0,1 \rangle$. Efektivita pre výstupný model je definovaná tak, že útvar je efektívny, ak hodnota účelovej funkcie $E_O^* = 1$, inak je neefektívny a miera efektivity je z intervalu $E_O^* \in \langle 1, \infty)$. Ak chceme dostať efektivitu do intervalu $E_O^* \in \langle 0,1 \rangle$, musíme vypočítať prevrátenú hodnotu.

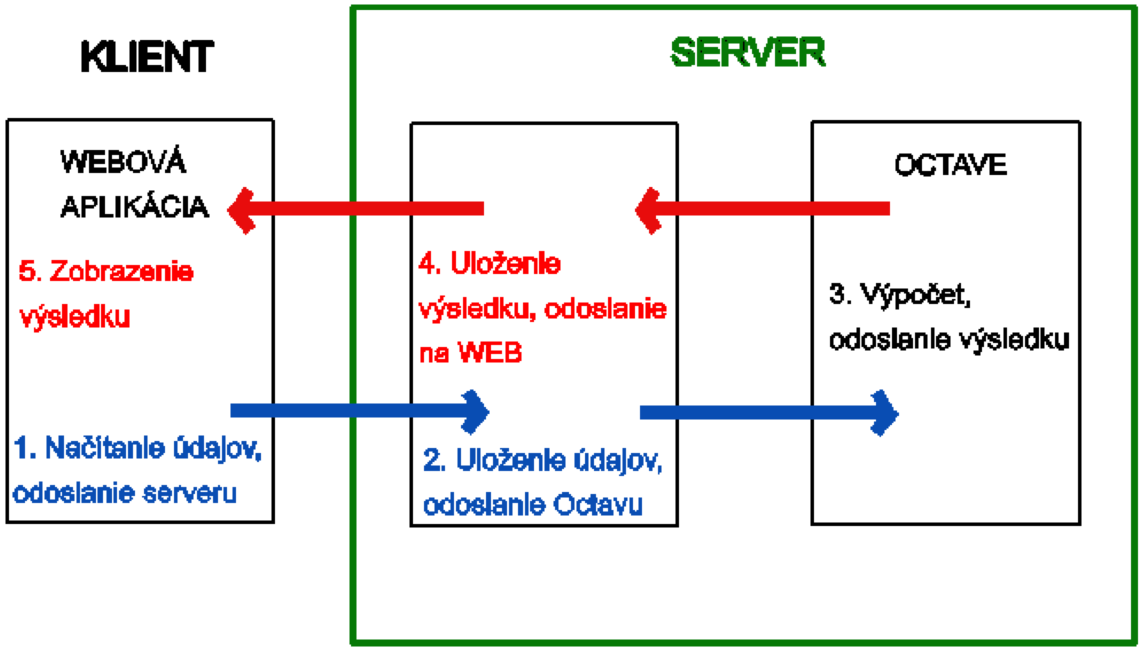
2 Praktická časť

V tejto kapitole si ukážeme, ako sme na základe teoretických poznatkov vytvorili webovú aplikáciu, ktorá dokáže z načítaných vstupov a výstupov vyhodnotiť efektivitu resp. neefektivitu a zobraziť príslušné výsledky v jednoduchej a zrozumiteľnej forme.

2.1 Logický model internetovej aplikácie

Na Obr. 4 je schematicky popísaný logický model, podľa ktorého sme programovali webovú stránku. Na takýto druh aplikácie sa hodí použiť systém klient/server. Tento systém sa skladá zo softvéru klienta a softvéru servera. Klient/server opisuje vzťah medzi dvoma počítačovými programami resp. počítačmi v sieti, z ktorých jeden, klient, odošle požiadavku na službu druhému, serveru, ktorý požiadavku splní. V našom prípade je klientom počítač, na ktorom si prostredníctvom internetového prehliadača zobrazíme stránku, ktorá je uložená na serveri a zadáme do nej vstupné dáta (Obr. 4 1). Z pohľadu DEA budú vstupné dáta predstavovať vstupy, výstupy a model, ktorý chceme použiť. Server tieto údaje zoberie, uloží a pošle programu GNU Octave na spracovanie (Obr. 4 2). V tomto programe prebehne výpočet spolu s vrátením výsledku (Obr. 4 3). Server výsledok uloží a odošle späť na stranu klienta (Obr. 4 4), kde sa výsledok zobrazí (Obr. 4 5). Komunikáciu klienta so serverom sme zabezpečili prostredníctvom textových súborov, ktoré pri uložení na server dostanú jedinečné meno, aby nevznikali problémy pre viac pripojených užívateľov (klientov) v jednom čase. Komunikáciu v rámci servera sme vyriešili tak, že sme do naprogramovaných modelov v programovacom jazyku GNU Octave posielali, ako parameter, cestu k uloženým údajom. Program potom uložil výsledky do súboru na serveri a následne sa zobrazili používateľovi. Webovú stánku sme programovali v jazyku HTML s využitím CSS [11]. Komunikáciu a dynamické časti stránky sme programovali v jazyku PHP, jednotlivé modely v jazyku GNU Octave.

[11] Cascading Style Sheets" - dizajn stránky



Obr. 4: Logický model internetovej aplikácie

2.2 Programovanie DEA modelov

2.2.1 GNU Octave

GNU Octave je voľne šíriteľný vyšší programovací jazyk zameraný na numerické operácie. Autorom jazyka je John W. Eaton. Jazyk poskytuje rozhranie príkazového riadku na numerické riešenie lineárnych a nelineárnych problémov. GNU Octave má rozsiahle nástroje pre riešenie spoločných problémov numerickej lineárnej algebry, hľadanie koreňov nelineárnych rovníc, integrovanie bežných funkcií, manipuláciu s polynómami a integráciu diferenciálnych a diferenčných rovníc. Vo väčšine prípadov je jeho syntax kompatibilná s programovacím jazykom Matlab.

V teoretickej časti sme sformulovali jednotlivé modely ako úlohy lineárneho programovania, preto je nevyhnutné nájsť funkciu, ktorá rieši príslušné lineárne programy.

2.2.1.1 Funkcia glpk

Funkcia glpk dokáže riešiť úlohy lineárneho programovania, ktoré sú zadané nasledovne:

$$[\,min\ |max\ ]\ C^T x$$

$$Ax\ ["="|">"|"<"|"\leq"|"\geq"]b$$

$$x \geq LB$$

$$x \leq UB.$$

Funkcia sa podľa [2 str. 485] uvádza v tvare:

$$[xopt, fmin, status, extra] = \boldsymbol{glpk}(c, a, b, lb, ub, ctype, vartype, sense, param).$$

Vstupné argumenty sú:

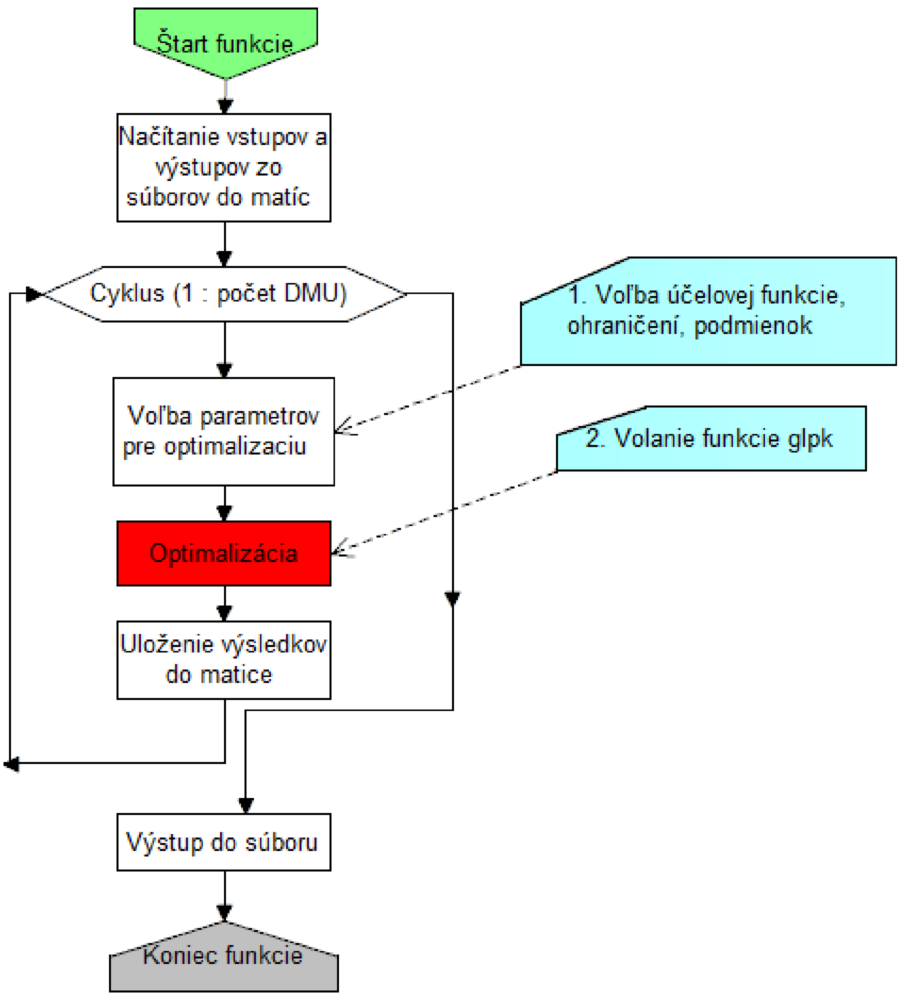
c	Vektor obsahujúci koeficienty účelovej funkcie
a	Matica obsahujúca koeficienty ohraničení
b	Vektor obsahujúci koeficienty pravej strany ohraničení
lb	Vektor obsahujúci dolné ohraničenie každej premennej
ub	Vektor obsahujúci horné ohraničenie každej premennej
$ctype$	Vektor znakov vzťahujúcich sa na porovnanie matice A a pravej strany b (“F“ voľné, “U“ nerovnosť s hornou hranicou, “L“ nerovnosť s dolnou hranicou, “S“ rovnosť)
$vartype$	Vektor znakov vzťahujúcich sa na typy premenných (“C“ reálne, “I“ prirodzené)
$sense$	Číselná hodnota minimalizácie (1) alebo maximalizácie (-1)
$param$	Štruktúra, ktorou sa definuje riešiteľnosť problému

Výstupné argumenty sú:

$xopt$	Vektor optimálnych hodnôt premenných
$fopt$	Optimálna hodnota funkcie
$status$	Status optimalizácie
$extra$	Extra výstup

2.2.2 Postup programovania

Predtým, ako sme začali programovať jednotlivé modely, sme si museli ujasniť niekoľko vecí. Všetky modely sme programovali ako funkcie, preto bolo nevyhnutné určiť si vstupné a výstupné parametre. Za vstupné parametre sme si zvolili názvy súborov so vstupmi a výstupmi jednotlivej úlohy, ďalším parametrom bol názov súboru, kde sa majú ukladať výsledky, teda výstupy z funkcie. Ďalej sme sa rozhodovali akú úlohu lineárneho programovania zvoliť. Rozhodli sme sa programovať duálne úlohy pre všetky modely. Výsledky primárnej úlohy nám potom funkcia glpk poskytla pomocou rozšíreného výstupu (extra). Každý model sme programovali podľa vývojového diagramu uvedeného na Obr. 5. Jednotlivé naprogramované modely sa v podstate líšia len v častiach označených ako komentáre 1, 2 modrou farbou.



Obr. 5: Vývojový diagram programovania modelov

Na príklade aditívneho modelu s variabilnými výnosmi z rozsahu, ktorý je popísaný v kapitole 1.3.4.2 si ilustrujeme voľbu parametrov do optimalizácie. Najskôr si musíme uvedomiť, aké premenné optimalizujeme. Riadkový vektor Z bude označovať optimalizované premenné:

$$Z = [\lambda_{1\times n} \quad {s^x}_{1\times m} \quad {s^y}_{1\times s}].$$

Po tom, ako sme zvolili vektor, ktorý optimalizujeme, dokážeme zvoliť účelovú funkciu F ako riadkový vektor, maticu ohraničení A, riadkový vektor pravých strán b, riadkový vektor dolných ohraničení lb a horných ub. Tieto parametre teda majú nasledovný tvar:

$$F = -[0_{1\times n} \quad {w^x}_{1\times m} \quad {w^y}_{1\times s}]$$

$$A = \begin{bmatrix} X_{m\times n} & I_{m\times m} & 0_{m\times s} \\ Y_{s\times n} & 0_{s\times m} & I_{s\times s} \\ 1_{1\times n} & 0_{1\times m} & 0_{1\times s} \end{bmatrix}$$

$$b = [x^o_{1\times m} \quad y^o_{1\times s} \quad 1]$$

$$lb = [0_{1\times n} \quad 0_{1\times m} \quad 0_{1\times s}]$$

$$ub = [\,].$$

Pre označenie vstupov a výstupov sme použili značenie z kapitoly 1.2.3. Funkciu glpk potom voláme v tvare:

$$[X, fval, status, extra] = \; glpk(f, Af, bf, lb, ub, ctype).$$

Všetky naprogramované modely sa dajú spúšťať aj samostatne v príkazovom riadku, takéto zadávanie môže vyzerať napr.: *ADvariabvr ('G:\vstup.txt' ,'G:\vystup.txt' ,'G:\vysledok.txt').*

2.2.3 Zdrojový kód

V nasledujúcej časti je ukážka zdrojového kódu pre prípad aditívneho modelu s variabilnými výnosmi z rozsahu. Zdrojové kódy pre ostatné modely sa nachádzajú v prílohe na CD.

```
function [ efektivita ] = ADvariabvr( vstup,vystup,zobraz)
%funkcia na výpočet aditívneho modelu s váhami (variabilné výnosy z rozsahu)
%načítanie vstupov, výstupov                x=load(vstup);        y=load(vystup);
%maximum z jednotlivých vstupov, výstupov   xmax=max(x);          ymax=max(y);
%počet vstupov         %počet výstupov      %počet jednotiek
m=size(x,2);           s=size(y,2);         n=size(x,1) ;
```

```
%minimálne váhy w+(wy) a w-(wx)        wplus=1./(std(y));      wminus=1./(std(x));
%matica minimálnych váh         maticaminvahy=ones(n,1)*[wminus wplus];
%rozlíšenie rovnosti/nerovnosti v matici A
for i=1:(m+s+1)         ctype(i)="S";           endfor;


%for cyklus pre jednotlivé DMU
for i=1:n
%DMUo,  pre ktoré meriam efektivitu                 x0=x(i,:);                y0=y(i,:);
%účelová funkcia                                f=-[zeros(1,n),wminus,wplus]';
%ohraničenia
Aeq=[x',eye(m,m),zeros(m,s);y',zeros(s,m),-eye(s,s);ones(1,n),zeros(1,s+m)];
%pravá strana ohraničení
beq=[x0,y0,1];
%dolné ohraničenie lambda>0, splus>0, sminus>0           %horné ohraničenie
lb=zeros(m+s+n,1);                                       ub=[];
%zhotovenie matice A, ohraničenia b         Af=Aeq;         bf=beq;
%lineárny program            [X,fval,status,extra] = glpk(f,Af,bf,lb,ub,ctype);
%riešenie primárnej úlohy    maticaextra(:,i)=abs([extra.lambda]);
%efektivita bez škálovania   ucelova(i)=fval;
%výpočet s+(sy), s-(sx) a lambda
sminus=X([(n+1):(n+m)],1);  splus=X([(n+m+1):(n+m+s)],1); lambda=X([(1):(n)],1);
%lambda, s+(sy) a s-(sx) v matici
maticasminus(:,i)=[sminus]; maticasplus(:,i)=[splus]; maticalambda(:,i)=[lambda];
%inicializácia a výpočet pomx, pomy         pomx=0;         pomy=0;
%pomx          for k=1:m       pomx=pomx+((x0(k)-sminus(k))/x0(k));       endfor
%pomy          for k=1:s       pomy=pomy+(y0(k)/(y0(k)+splus(k)));        endfor
%preškálovaná efektivita na interval <0,1>
efekt(i)=(1/(m+s))*(pomx+pomy);    efektivita=efekt;
end %koniec cyklu for
```

```
%ideálne vstupy, výstupy
zmenavstupov=x'*maticalambda;      zmenavstupov=x'-zmenavstupov;
zmenavystupov=y'*maticalambda;    zmenavystupov=-y'+zmenavystupov;
%váhy u,v, v0 - riešenie duálnej úlohy
maticaextrauv=maticaextra(1:(m+s),:);
v0=maticaextra((m+s+1),:);
%výstupná matica
zapis=[efektivita;maticasminus;maticasplus;maticalambda;maticaextrauv;zmenavstupo;
zmenavystupov;ucelova;v0;maticaminvahy'];

%výstup do súboru
x=x';    y=y';
dlmwrite(vstup,x, ' ');             dlmwrite(vstup,xmax, ' ',"-append");
dlmwrite(vystup,y, ' ');            dlmwrite(vystup,ymax, ' ',"-append");
dlmwrite(zobraz,m, ' ');            dlmwrite(zobraz,s, ' ',"-append");
dlmwrite(zobraz,zapis, ' ',"-append");
fclose('all');
end
```

2.3 Programovanie webovej stránky

Tak, ako bolo spomenuté v kapitole 2.1, webové rozhranie sme programovali v jazyku HTML. Za základ našej stránky sme si zobrali predpripravenú šablónu podľa [3], do ktorej sme potom postupne dotvárali celé prostredie stránky. Jazyk PHP nám slúžil na spracovanie zadávaných údajov, vykreslenie tabuliek, grafov, ale hlavným prínosom bolo, že sme mohli z prostredia stránky spúšťať programy napísané v jazyku GNU Octave. V tejto kapitole si uvedieme ukážky najdôležitejších častí kódu, ktoré tvoria základ webovej stránky. Pri programovaní sme si pomáhali stránkou [9].

2.3.1 Načítanie vstupov a výstupov

Načítanie vstupov a výstupov sa na stránke dá realizovať dvomi spôsobmi:

- Načítanie zadávaním
- Načítanie prostredníctvom súboru

2.3.1.1 Načítanie zadávaním

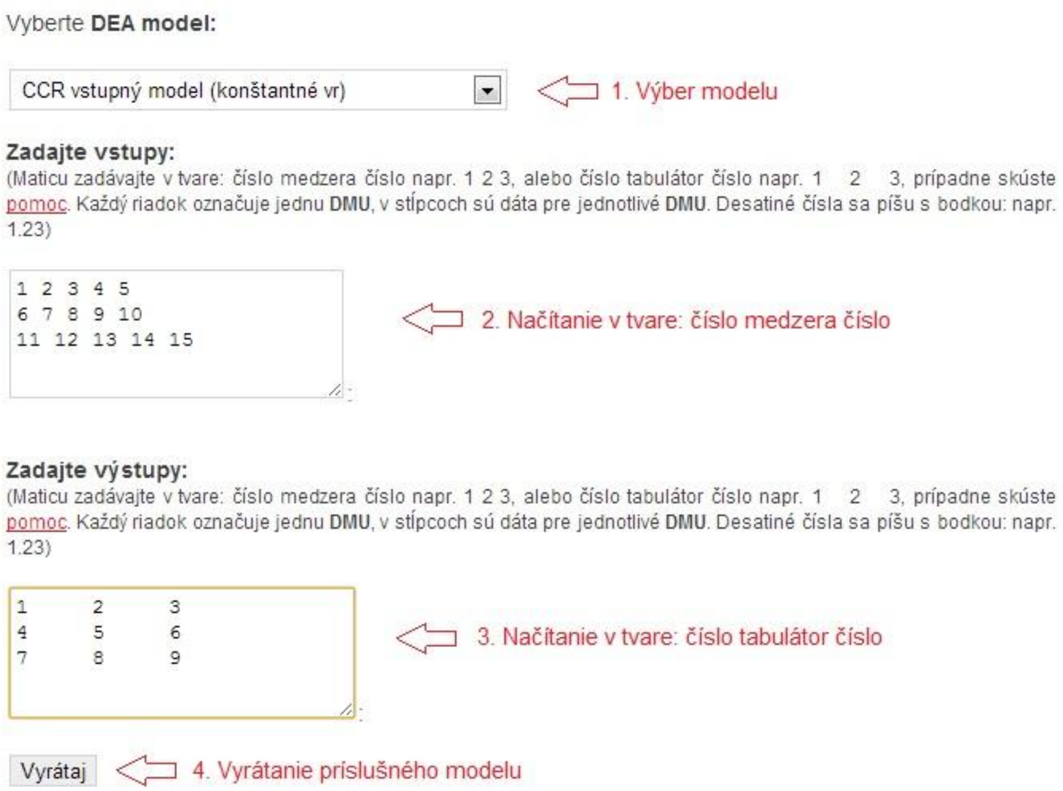
Načítanie vstupov a výstupov priamo na stránke zadávaním sme programovali prostredníctvom párového HTML tagu *<textarea>*. Následne sme potom načítané dáta uložili do súboru s jedinečným menom. Nasledujúci kód vykonáva zadávanie vstupu:

```
<textarea class="input" rows="10" cols="50" name="vstupy" wrap="soft"
placeholder="Matica so vstupmi" required autofocus>
</textarea>
```

Na strane PHP potom uložíme načítavané údaje do textového súboru:

```
$f = FOpen("runserver/uploads/$vstupy", "w");
$maticavstup = $_POST['vstupy'];
FWrite($f,$maticavstup);
FClose ($f);
```

Na Obr. 6 je znázornené ako na stránke vyzerá načítanie zadávaním.



Obr. 6: Načítanie zadávaním

2.3.1.2 Načítanie prostredníctvom súboru

Načítanie prostredníctvom súboru sa realizuje pomocou dvoch textových súborov, v jednom sú uložené vstupy, v druhom výstupy. Na tento účel sme si zvolili tag <input>. Nasledujúci HTML kód opisuje načítanie:

```
Vyber súbor so vstupmi: <input class="button" name="vstupy" type="file" value="Načítaj vstupy" required/>.
```

Na strane PHP potom spracujeme súbor a uložíme ho na server:

```
$target_path = "runserver/uploads/";
$pom=date("F_j_Y_H_i_s");
$_FILES['vstupy']['name']= "vstup".$pom.".txt" ;
$vstupy = "vstup".$pom.".txt" ;
$target_path = $target_path . basename( $_FILES['vstupy']['name']);.
```

Na Obr. 7 je znázornené ako v praxi vyzerá načítanie údajov prostredníctvom súboru.

Obr. 7: Načítanie prostredníctvom súboru

2.3.2 Komunikácia s GNU Octave

Najdôležitejšia časť celej webovej stránky je práve komunikácia s GNU Octave. Zabezpečili sme ju pomocou PHP funkcie exec, ktorá nám umožní na pozadí spúšťať jednotlivé modely. V praxi to vyzerá tak, že v PHP sme vytvorili jeden textový súbor so vstupmi a druhý s výstupmi. Tieto súbory sú uložené v zložke na serveri a majú jedinečné meno. Mená týchto dvoch súborov potom pošleme ako parameter do modelu. Model sa vykoná a vytvorí v zložke na serveri súbor s výsledkami. Kód funkcie vyzerá nasledovne:

```
$command = $octavefolder." --eval \"addpath ('".$rootfolder." /runserver /programoctave/'); ".$model." ('".$intputDir. "/".$vstupy. "','".$intputDir. "/".$vystupy."' , '".$outputDir." /".$filename."') \" --persist" ;
exec($command);.
```

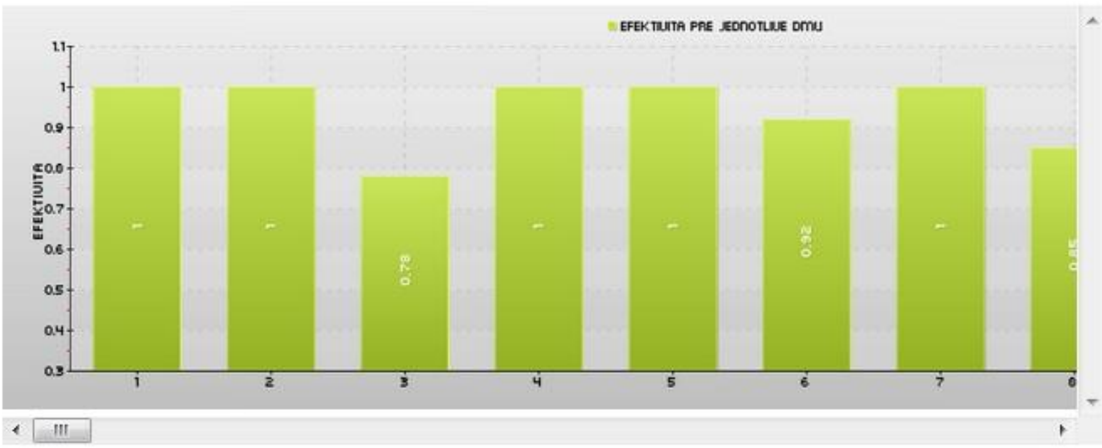
Rozoberme si nasledovné časti premennej *$command*. V premennej *$octavefolder* máme uloženú cestu k programu GNU Octave. Funkcia *addpath ('".$rootfolder." /runserver /programoctave/')* nám pridá zdrojový adresár, pre naše uložené funkcie (programy). V premennej *$model* je uložený model, ktorý si zvolil používateľ, meno modelu je rovnaké ako meno funkcie. V nasledujúcej zátvorke *('".$intputDir. "/".$vstupy. "','".$intputDir. "/".$vystupy."' , '".$outputDir." /".$filename."')* sú uvedené parametre jednotlivých funkcií. Po vykonaní tejto časti máme na serveri vytvorený súbor s výsledkami.

2.3.3 Zobrazenie výsledkov

Zobrazenie výsledkov na stránke máme rozdelené na tri časti. Kvôli názornosti budeme zobrazovať výsledky príkladu analýzy bankových pobočiek z kapitoly 1.2.4.2 pre aditívny model s variabilnými výnosmi z rozsahu. Hneď ako prebehne výpočet, zobrazí sa tabuľka efektivity, ktorá je znázornená na Obr. 8. Pod tabuľkou sa zobrazí hneď aj graf jednotlivých efektivít tak, ako na Obr. 9. Ako rozšírený výstup sme ponúkli možnosť kliknúť na rezervy, kde sa zobrazia rezervy vstupov, výstupov a podiel daného útvaru v optimálnej kombinácii útvarov aj s príslušnými grafmi. Pre názornosť sme si vybrali 28. útvar, ktorého výsledky (rezervy) sme zobrazili do Obr. 10 - Obr. 15. Treťou časťou sú celkové výsledky, kde sme zosumarizovali všetky informácie, ktoré nám poskytol naprogramovaný model. Do tabuľky sme teda zachytili efektivitu, hodnotu účelovej funkcie, vstupy, výstupy, rezervy, váhy a lambdy pre každé DMU. Ukážka celkových výstupov sa nachádza na Obr. 16. Pri veľkých súboroch dát, tak ako je to v tomto prípade, je zložité prehľadne zobraziť celkový výstup, preto sme používateľovi ponúkli možnosť stiahnutia výsledkov do csv súboru.

DMU	Efektivita pre Adítívny model s váhami (variabilné vr)	Efektivita v %	Rezervy
1	1	100	Rezervy
2	1	100	Rezervy
3	0.7838	78.38	Rezervy
4	1	100	Rezervy
5	1	100	Rezervy
6	0.9181	91.81	Rezervy
7	1	100	Rezervy
8	0.8484	84.84	Rezervy
9	1	100	Rezervy
10	1	100	Rezervy
11	0.8954	89.54	Rezervy
12	0.749	74.9	Rezervy
13	0.7103	71.03	Rezervy
14	1	100	Rezervy
15	1	100	Rezervy
16	0.8688	86.88	Rezervy
17	0.8107	81.07	Rezervy

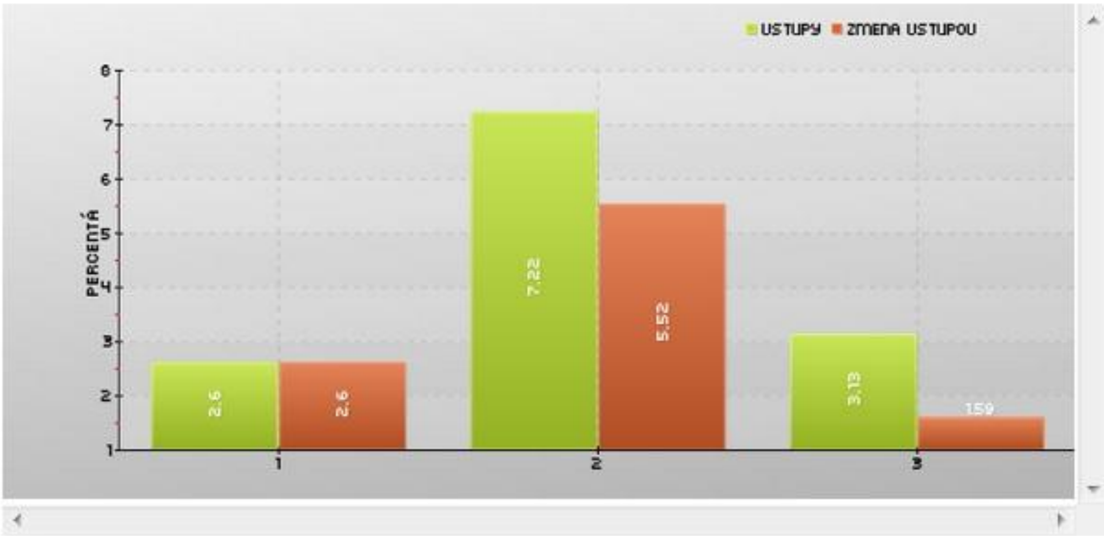
Obr. 8: Úsek efektivity pre jednotlivé DMU



Obr. 9: Časť grafu efektivity pre jednotlivé DMU

	Vstupy	Rezervy	Ideálne vstupy
1	70598.9	0	70598.9
2	182710	42990.02	139719.98
3	32467.6	15969.97	16497.63

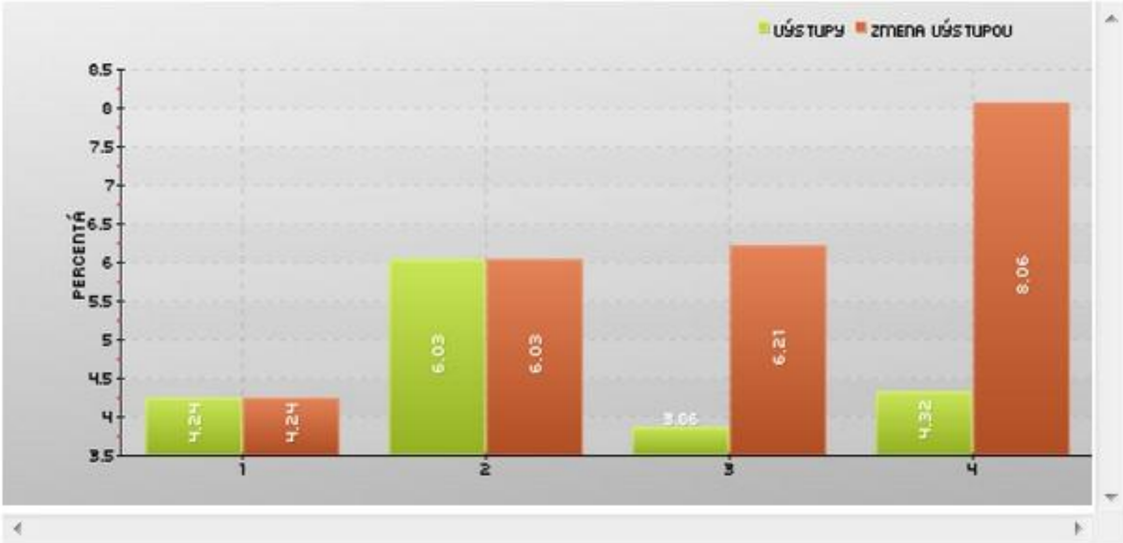
Obr. 10: Vstupy a ich rezervy



Obr. 11: Graf vstupov a ich rezerv

	Výstupy	Rezervy	Ideálne výstupy
1	2185990	0	2185990
2	645	0	645
3	11144099	6792694.79	17936793.79
4	1643	1424.19	3067.19

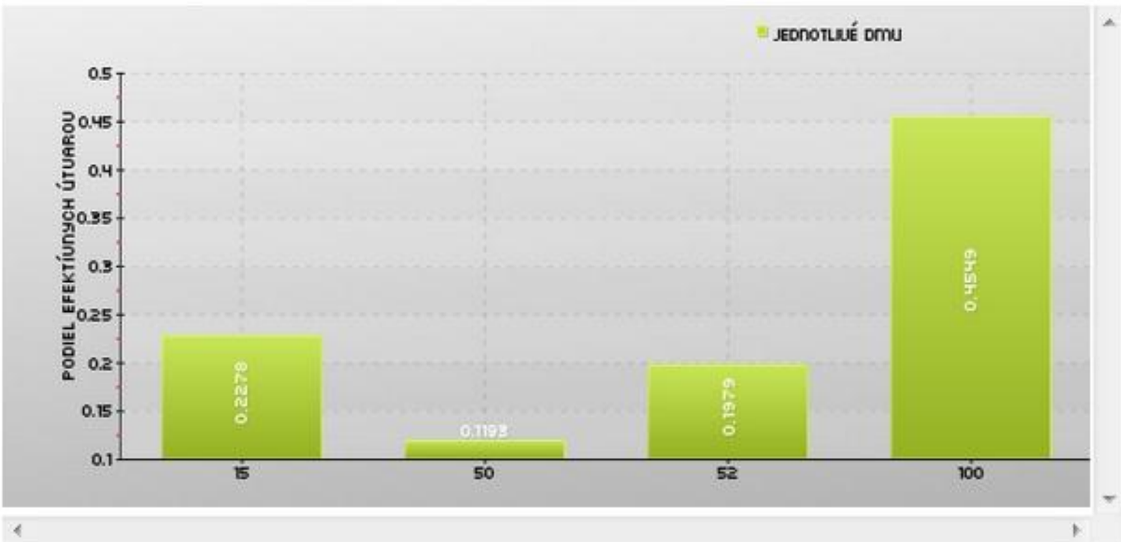
Obr. 12: Výstupy a ich rezervy



Obr. 13: Graf výstupov a ich rezerv

	DMU	Proporcionálne zastúpenie ef. DMU	Zastúpenie v %
1	15	0.2278	22.78
2	50	0.1193	11.93
3	52	0.1979	19.79
4	100	0.4549	45.49

Obr. 14: Podiel daného útvaru v optimálnej kombinácií útvarov



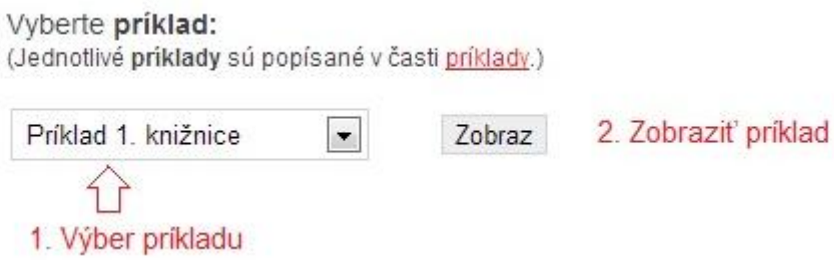
Obr. 15: Graf podielu daného útvaru v optimálnej kombinácií útvarov

DMU	Efektivita	Účelová funkcia	Vstupy			Výstupy				S
1	1	0	2582618	2530099	37027.4	34359033	8381	281921703	37523	0
2	1	-0	1122877	1244833	28099.8	20602647	3061	71228122	9630	0
3	0.7838	-0.1387	32082.1	24944.4	1038.01	171113	37	4446894	642	0
4	1	0	48123.4	34386.4	1502.1	304461	95	12570405	1178	-0
5	1	-0	80205.5	76182.4	1643.12	1462297	679	9691587	1840	0
6	0.9181	-0.1713	80205.5	79774.2	2182.01	1108206	334	17472629	2057	4438.8
7	1	0	80205.5	56588.8	1474.09	907319	386	15499026	2107	-0
8	0.8484	-0.0623	48123.4	35339.7	1290.43	138017	67	12539199	1263	8101.4
9	1	0	16041.2	13198.7	491.152	22776.1	5	2916802	370	0
10	1	0	894722	1259935	399901	22656464	4823	171343585	23935	0
11	0.8954	-0.9877	492620	641552	151323	7261338	2455	82077207	12708	0
12	0.749	-0.6063	70684.2	118807	31297.8	610384	249	20859214	3540	0
13	0.7103	-0.6299	52235.9	60675	13449.9	550215	156	8617529	1498	0
14	1	-0	25732.2	32304.3	11110.6	1370037	102	8200879	1701	0
15	1	-0	44915.4	59379.8	23196.2	2013678	321	17556491	1855	0
16	0.8688	-0.2282	52992.5	73335.7	24723.4	1032907	288	19428458	2459	0
17	0.8107	-0.3621	31744.7	27648.7	6507.53	418417	125	4175177	846	0
18	0.6898	-1.3522	145382	337408	50611.3	2177445	841	22489604	3443	0
19	0.585	-0.3524	25686.7	63907.4	19909.3	96924.7	32	13545322	1119	0
20	0.7302	-0.1266	27618.5	53217	1037.05	54396.9	18	9875785	1034	0
21	0.8597	-3.3727	1323045	1826462	392794	18993263	5112	178836421	26161	0

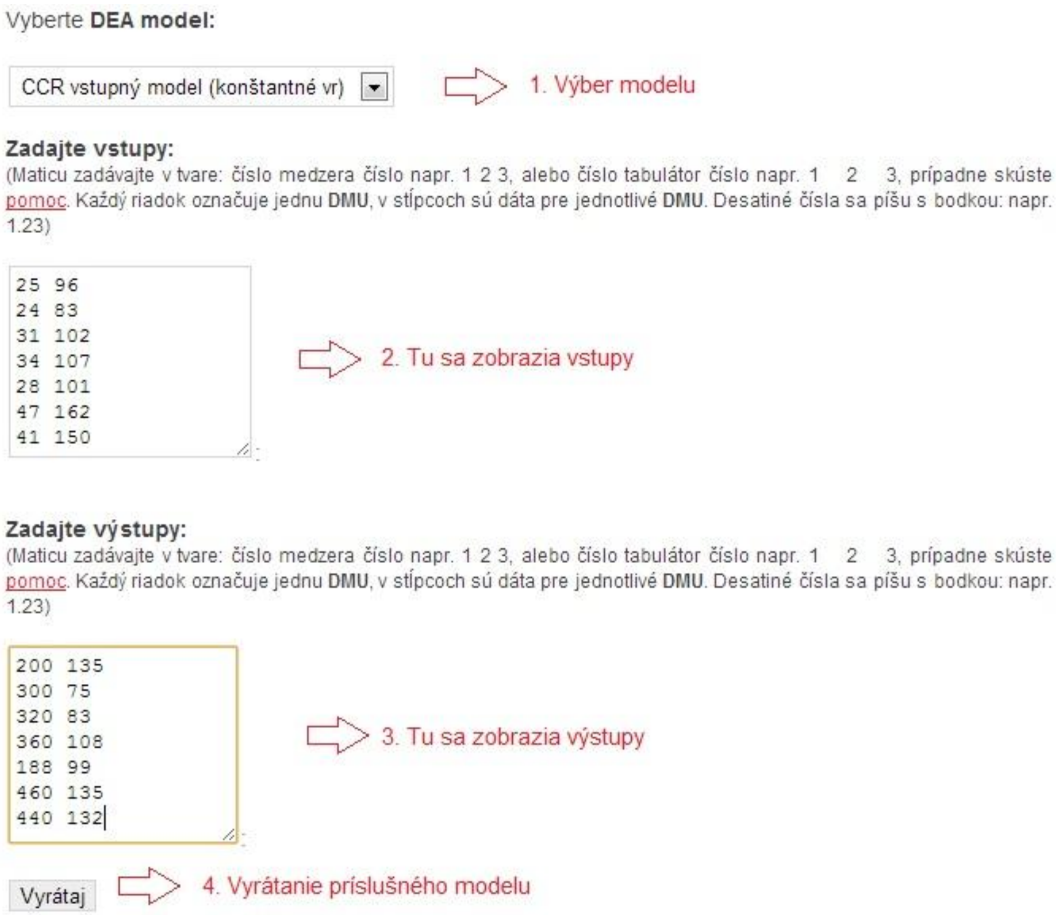
Obr. 16: Časť celkových výstupov úlohy

2.3.4 Vzorové úlohy

Na webovej stránke sme vytvorili sekciu vzorové úlohy, v ktorých si môže používateľ nájsť ako vyzerajú zadania a riešenia vopred zadefinovaných úloh pre rôzne modely. Je to najjednoduchší spôsob ako sa naučiť pracovať so stránkou, zároveň môže používateľ porovnávať výsledky pre rôzne modely. Najprv si vyberie akú úlohu má záujem vyrátať (Obr. 17), následne sa mu zobrazia údaje, zvolí si model a vypočíta úlohu (Obr. 18).



Obr. 17: Voľba úlohy a zobrazenie dát na stránke



Obr. 18: Výpočet zvolenej úlohy

Záver

Data envelopment analysis ponúka veľa možností ako hodnotiť efektivitu útvarov v rámci danej skupiny. V bakalárskej práci sme sa zaoberali odvodením základných teoretických poznatkov, ktoré sme vysvetľovali na jednoduchých príkladoch, o čom svedčí kapitola 1.2. Postupne sme prešli cez základné CCR a BBC modely, ktoré sme sa snažili matematicky korektne odvodiť z koncepčného modelu. Na záver teoretickej časti sme v kapitole 1.3.4 prezentovali modely s váhami.

Hlavným cieľom bakalárskej práce bolo na základe teoretických poznatkov vypracovať webové rozhranie, ktoré využíva naprogramované modely na riešenie DEA úloh a výsledky dokáže zobraziť vo vhodnej, zrozumiteľnej forme. Z pohľadu realizácie tohto cieľa je dôležitá kapitola 2, v ktorej sme logicky odôvodnili najdôležitejšie časti programovania DEA modelov v jazyku GNU Octave a webovej stránky v jazyku PHP. V kapitole 2.3.3 sme poukázali na funkčnosť webovej aplikácie na konkrétnom príklade hodnotenia bankových pobočiek zadefinovaných v teoretickej časti v kapitole 1.2.4.2.

Prínosom našej práce bolo prehľadné a zrozumiteľné spracovanie základnej teórie z oblasti DEA modelov. Hlavným prínosom našej práce bolo však webové rozhranie, ktoré sme vytvorili ako jednoduchý nástroj hodnotenia efektivity v rámci danej skupiny. Toto rozhranie poskytuje používateľom na výber viacero možnosti zadávania vlastných dát. Výpočet efektivity sa dá realizovať niekoľkými modelmi, výsledky sú zobrazené v zrozumiteľnej a prehľadnej forme, doplnené grafmi a vysvetleniami. Stránka je vhodnou pomôckou pre študentov na predmete DEA modely, kde si môžu na praktických príkladoch otestovať a porovnávať výsledky pre jednotlivé modely.

Rozšírením tejto bakalárskej práce by bolo implementovanie sofistikovanejších DEA modelov. Webová stránka je pripravená na rozšírenie o dvojkrokové modely, nakoľko poskytuje rozšírený výstup, ktorý by sa dal neskôr využiť ako vstup pre tieto modely. Možným rozšírením by bolo ponúknuť užívateľovi voľbu vloženia vlastného modelu. Oboje však vyžaduje širší priestor, ako ponúka táto práca.

Zoznam použitej literatúry

[1] Cooper, W. W., Seiford, L. M., Tone, K.: *Data Envelopment Analysis. A Comprehensive Text with Models*, Applications, References and DEA-Solver Software, Kluwer Academic Publishers, Boston, 2000

[2] Eaton, J., Bateman, B., Hauberg, S.: *GNU Octave. Free Your Numbers, Free Software Foundation*, Boston, 2011, dostupné na internete (24.5.2013): http://www.gnu.org/software/octave/octave.pdf

[3] Free css templates: css templates, dostupné na internete (24.5.2013): http://www.freecsstemplates.org/

[4] Halická, M.: *DEA modely*, prednášky, FMFI UK, Bratislava, 2012

[5] Lauko, M.: *DEA analýza efektívnosti a SBM model*, bakalárska práca, FMFI UK, Bratislava, 2007, dostupné na internete (24.5.2013): http://diplomovka.sme.sk/praca/3182/deaanalyzaefektivnosti-sbm-model.php

[6] Lennerová, V., *DEA modely a meranie eko-efektívnosti*, diplomová práca, FMFI UK, Bratislava, 2008, dostupné na internete (24.5.2013): http://www.iam.fmph.uniba.sk/studium/efm/diplomovky/2008/lennerova/diplomovka.pdf

[7] Nemethová, A., *DEA modely a meranie efektívnosti*, diplomová práca, FMFI UK, Bratislava, 2001, dostupné na internete (24.5.2013): http://www.iam.fmph.uniba.sk/studium/efm/diplomovky/2008/lennerova/diplomovka.pdf

[8] Ševčovič, D., Halická, M., Brunovský, P.: *DEA analysis for large structured bank branch network*, Central Euro. J. of Opreational Res. 9 (2001), 329-342, dostupné na internete (24.5.2013): http://www.iam.fmph.uniba.sk/institute/sevcovic/papers/cl19.pdf

[9] W3schools: PHP, dostupné na internete (24.5.2013): http://www.w3schools.com/php/default.asp

[10] Zhu, J., *Quantitative Models for Performance Evaluation and Benchmarking. Data. Envelopment Analysis with Spreadsheets*, Second Edition, Science Business Media, LLC, New York, 2009

Prílohy

Príloha A: CD médium – prílohy v elektronickej podobe.