



Centro Universitário de Brasília
Instituto CEUB de Pesquisa e Desenvolvimento - ICPD

JOGI TAKECHI

AVALIAÇÃO DE EFICIÊNCIA EM UNIDADES DE ATENDIMENTO AO ELEITOR
ESTUDO DE CASO DE CARTÓRIOS ELEITORAIS DO DF

Brasília
2013

JOGI TAKECHI

**AVALIAÇÃO DE EFICIÊNCIA EM UNIDADES DE ATENDIMENTO AO ELEITOR
ESTUDO DE CASO DE CARTÓRIOS ELEITORAIS DO DF**

Monografia apresentada ao Centro
Universitário de Brasília (UniCEUB/ICPD)
como uma das atividades programadas pelo
módulo Metodologia Científica do curso de
Pós-graduação Latu-Sensu em Governança
de Tecnologia de Informação

Orientador: Prof. Flávio Antonio Klein

**Brasília
2013**

JOGI TAKECHI

AVALIAÇÃO DE EFICIÊNCIA EM UNIDADES DE ATENDIMENTO AO ELEITOR

ESTUDO DE CASO DE CARTÓRIOS ELEITORAIS DO DF

Monografia apresentada ao Centro
Universitário de Brasília (UniCEUB/ICPD)
como uma das atividades programadas pelo
módulo Metodologia Científica do curso de
Pós-graduação Latu-Sensu em Governança
de Tecnologia de Informação

Brasília, ____ de _____ de 2013

Banca Examinadora

Prof. orientador: Flávio Antonio Klein

Prof. examinador 1

Prof. examinador 2

RESUMO

Este estudo de caso pretende demonstrar a utilização de uma metodologia que combina a Tecnologia de Informação (TI) e o modelo DEA – Data Envelopment Analysis (Análise Envoltória de Dados) aplicada na avaliação de eficiência técnica dos Cartórios Eleitorais do Distrito Federal. Demonstrar como esta metodologia pode ser usada como uma ferramenta de TI para comparar unidades de atendimentos, identificando as mais eficientes e menos eficientes e os fatores que as tornam eficientes ou ineficientes; e por ser uma ferramenta de uso iterativo, podemos simular cenários alterando o nível de utilização dos fatores de produção. Em resumo, um modelo de análise e tomada de decisão, no qual podemos avaliar as unidades de atendimentos, classifica-las por sua eficiência relativa, identificar os níveis de utilização dos fatores de produção (falta ou excesso), identificar um padrão de eficiência e assim permitir a melhoria contínua da prestação de serviços. Os resultados deste estudo mostram que 70% dos Cartórios Eleitorais do Distrito Federal estão dentro da fronteira de eficiência relativa.

Palavras-chaves: Análise Envoltória de Dados. Programação linear. Eficiência relativa. Cartórios eleitorais. Unidade de atendimento. Fatores de produção.

ABSTRACT

This study case aims to demonstrate the use of a methodology that combines the Information Technology (IT) and DEA - Data Envelopment Analysis applied in evaluating of the technical efficiency in Federal District Electoral Notaries. Demonstrate how this methodology can be used as an IT tool to compare production units, identifying the most efficient and least efficient and the factors that make it efficient or inefficient, and for being a tool for interactive usage, we can simulate scenarios by changing the level of production factors. In summary, a model of analysis and decision making, in which we evaluate units, sorts them by their relative efficiency, identify the levels of utilization of production factors (lack or excess), identify a standard of efficiency and thus enabling continuous improvement of service. The results of this study show that 70% of the Federal District Electoral Notaries are within the boundary of relative efficiency.

Keywords: Data envelopment analysis. Linear programming. Relative efficiency. Notary election. Production unit. Factors of production.

SUMÁRIO

| | |
|---|----|
| INTRODUÇÃO | 1 |
| CAPÍTULO I - REVISÃO DE BIBLIOGRAFIA | 4 |
| 1.1. Programação Linear | 8 |
| 1.2. Análise Envoltória de Dados (DEA) | 11 |
| 1.3. Abordagem sistêmica do DEA | 12 |
| 1.4. Implementação da Análise Envoltória de Dados (DEA) | 14 |
| 1.4.1. Seleção de DMU's..... | 14 |
| 1.4.2. Seleção de fatores | 16 |
| 1.4.2.1. Seleção criterial | 17 |
| 1.4.2.2. Análises quantitativas não-DEA | 17 |
| 1.4.2.3. Análises baseadas em DEA | 21 |
| 1.4.3. Modelos/formulações DEA | 18 |
| 1.4.3.1. Modelo CCR/CRS | 18 |
| 1.4.3.2. Modelo BCC/VRS | 20 |
| CAPÍTULO II - PROCEDIMENTO METODOLÓGICO | 22 |
| 2.1. Modelagem do problema | 23 |
| 2.2. Definição do modelo | 28 |
| 2.3. Seleção dos Inputs e Outputs | 28 |
| 2.4. Desenvolvimento do software para geração do modelo e sua solução | 32 |
| 2.5. Análise dos relatórios..... | 33 |
| 2.5.1. Relatório de eficiência relativa | 33 |
| 2.5.2. Relatório de sobra de recursos | 35 |
| CONSIDERAÇÕES FINAIS | 36 |
| REFERÊNCIA BIBLIOGRÁFICA..... | 38 |
| APÊNDICE | 41 |

INTRODUÇÃO

O presente trabalho pretende avaliar os Cartórios Eleitorais do Distrito Federal – Unidade de Atendimento ao Eleitor -, integrante do Tribunal Regional Eleitoral do Distrito Federal, em termos de sua eficiência relativa, ou seja, classifica os Cartórios por nível de eficiência e identifica as mais eficientes que servirão de modelo de comparação para as ineficientes.

O trabalho utiliza a Tecnologia da Informação (TI) combinada com o modelo de Análise de Envoltória de Dados (DEA – *Data Envelopment Analysis*) para determinar a eficiência relativa pelo seu aspecto técnico.

Neste estudo cada Cartório Eleitoral do DF é considerado como uma Unidade Produtiva (ou do inglês DMU – *Decision Making Units*). Cada DMU recebe recursos/INPUT (orçamento aplicado, quantitativo de pessoal, material aplicado, etc.) que serão utilizados pelo PROCESSO de trabalho dando como resultados diversos produtos ou serviços/OUTPUT, por exemplo: o serviço de atendimento ao eleitor.

A organização necessita conhecer o nível de eficiência das DMUs, identificar as DMUs mais eficientes e os fatores e/ou processos que contribuem para o alcance dessa eficiência com o intuito de melhorar o desempenho geral e promover a melhoria contínua de processos (*benchmarking*) das DMUs.

Diante deste problema, seguem abaixo as questões listadas de acordo com o nível de ênfase deste trabalho:

a) Dentro do conjunto de Cartórios Eleitorais, quais são os mais eficientes e quais são os menos eficientes?

b) Quais são os fatores que contribuem ou atrapalham a eficiência dos Cartórios?

c) Como as ferramentas de TI podem apoiar na avaliação da eficiência?

d) As DMUs eficientes segundo a percepção da direção da organização, serão confirmadas com a avaliação feita a partir desta ferramenta de TI?

Em organizações públicas que possuem unidades produtivas descentralizadas (Cartório Eleitoral), por exemplo, é necessária uma avaliação da sua eficiência para apontar não somente a excelência do atendimento ao público, mas também, a excelência no uso dos recursos aplicados. Este trabalho pretende mostrar a aplicação de uma metodologia que demonstre a eficiência de cada unidade produtiva.

Do ponto de vista social, como órgão público de atendimento ao cidadão, o estudo contribui para identificar uma melhor aplicação do dinheiro público para proporcionar os maiores retornos à sociedade, como consequência da melhoria no atendimento do público, devido ao aumento de eficiência das unidades menos produtivas.

Do ponto de vista acadêmico, a aplicação e validação desta ferramenta para a avaliação de Unidades Produtivas (UP ou DMU). Demonstrar o uso da ferramenta DEA juntamente com as tecnologias de TI, aplicadas em situações reais. Analogamente, poderá ser utilizada em outras áreas, como por exemplo, na área da saúde, educação, previdência, serviços bancários e postais, entre outras prestações de serviços ou fornecimento de produtos onde haja um conjunto de Unidades Produtivas semelhantes.

A identificação dos processos internos que contribuem para a eficiência pode ser adicionada ao banco de dados de Boas Práticas.

Do ponto de vista do pesquisador, o trabalho demonstra o uso de ferramentas de programação matemática (otimização de problemas de programação linear – PPL e DEA) e de TI (software), para conhecer, identificar e resolver problemas reais de forma que minimizem a utilização de recursos escassos e que atendam os requisitos/resultados desejados.

O trabalho está estruturado da seguinte forma: no Capítulo I será apresentada uma revisão da literatura referente à métodos de programação linear e Análise Envoltória de Dados; no Capítulo II os procedimentos metodológicos adotados seguida das considerações finais.

CAPÍTULO I

REVISÃO DE LITERATURA

Para atender à crescente demanda da sociedade por serviços de qualidade, realizados com os recursos disponíveis e de forma transparente, tornou-se essencial para a administração pública modernizar a sua gestão, buscando novos modelos.

Afinal, não basta ter bom planejamento, estabelecer objetivos, metas e estratégias, se não há ferramentas eficientes para monitorar, avaliar, corrigir e redirecionar os seus processos em busca da eficiência.

No Brasil, para atender essas exigências da sociedade e do mercado, vale ressaltar o pioneirismo das empresas estatais, como a Petrobrás e Banco do Brasil (apud LOBATO, 2007), seguida atualmente por outros órgãos da administração pública e principalmente o governo do Estado de São Paulo com a utilização do DEA.

A Petrobrás adotou a gestão de processos para gerir várias de suas atividades na produção e escoamento, usando para tanto metodologias de mapeamento de processos (BPM – Business Process Management), BSC - Balanced Score-Card bem como de ERP – Enterprise Resource Planning.

O Banco do Brasil, na sua área de Tecnologia de Informação (TI), aplicou a metodologia AHP - Processo de Análise Hierárquica, no apoio à resolução de problemas complexos, notadamente, na priorização de projetos.

A Secretaria da Fazenda do Estado do Amazonas que através de uma reengenharia de seus processos alcançou a reduções de custos, melhoria da qualidade de seus serviços e procedimentos.

O TSE – Tribunal Superior Eleitoral com seu projeto de mapeamento, modelagem e documentação dos processos de votação eletrônica.

Com exemplo de utilização do DEA – Data Envelopmet Analysis, podemos citar o *Poupatempo* (apud CAMANHO). Criado em 1997, pelo governo do Estado de São Paulo, o serviço disponibiliza, em 11 postos espalhados pelo Estado de São Paulo, vários órgãos e empresas prestadoras de serviço de natureza pública, além de serviços de apoio (fotocopiadoras, fotos). Isso permite que as pessoas possam tirar seus documentos (carteira profissional, RG, seguro desemprego etc.) de forma rápida e prática, em um único local. Para maximizar a performance dos postos, foi aplicada a metodologia DEA que permite calcular a eficiência de cada posto (ou hospital, escola, secretaria), comparando os recursos consumidos (funcionários, custos, áreas) com os resultados obtidos (números e velocidades de atendimentos, tipos de serviços), obtendo-se um único índice para cada unidade avaliada. Assim, tem-se uma completa noção de quais são os fatores que afetam a *performance* e quais são os postos que podem ser considerados *Modelo de referência/benchmarking* para cada tipo de serviço oferecido. O resultado desse trabalho, que atende em média 75 mil pessoas diariamente, está refletido no nível de satisfação de 99%, de acordo com pesquisa realizada pelo IBOPE – Instituto Brasileiro de Pesquisa e Estatística. Como reconhecimento pela sua qualidade, o serviço se tornou exemplo para outras áreas de atendimento ao público.

O CNJ – Conselho Nacional da Justiça utilizou o DEA para analisar do desempenho global dos Tribunais de Justiça do Brasil, no ano de 2011, sendo que estas informações estão publicadas no CNJ – Justiça em Números – Indicadores do Poder Judiciário 2011.

Estes exemplos mostram que os órgãos governamentais que adotaram metodologias como o BSC, gestão de processos, DEA, a Gestão Eletrônica de Documentos conseguiram melhorar a *performance* em suas iniciativas e projetos.

As empresas devem buscar maiores índices de produtividade para conseguirem se manter num mercado tão competitivo ou para oferecer serviços com qualidade. Emerge nessa perspectiva a seguinte questão: Como medir na prática, a produtividade? As dificuldades na medição da produtividade podem ser desdobradas em três partes:

- quais são as entradas (INPUT) apropriadas para o sistema e os medidores para as mesmas?
- quais são as saídas (OUTPUT) apropriadas do sistema e os medidores para as mesmas?
- quais são as formas (PROCESSO) apropriadas para medir o relacionamento entre essas entradas e saídas?

Felizmente, desenvolveu-se uma técnica com capacidade de comparar a eficiência de múltiplas unidades de serviço que fornecem serviços similares mediante a consideração explícita do uso de suas múltiplas entradas (recursos) na produção de múltiplas saídas (serviços). O método oferece subsídios às empresas para realizarem diagnósticos de eficiência em suas unidades, em termos do uso de insumos (inputs) para obtenção de produtos (outputs).

A literatura se refere a dois tipos básicos de métodos que trabalham com o objetivo de mensurar eficiência e produtividade e, embora usem técnicas distintas para efetuar a mensuração, os dois tipos convergem no fato de estimar uma fronteira relativa, ou seja, ao máximo de produto possível de se obter utilizando os insumos disponíveis. O primeiro conjunto de métodos é formado por modelos paramétricos –

não está no escopo deste trabalho. O segundo conjunto, de métodos baseada em programação matemática (técnicas não-paramétricas) no qual está incluída o DEA.

Quando se discute a *performance* de uma unidade de produção, é comum descrever como sendo mais ou menos "eficiente" ou, mais ou menos "produtiva". As unidades de produção empregam múltiplos *inputs* para produzir múltiplos *outputs* e esta complexidade pode ser incorporada em uma medida de produtividade que pode ser definida como:

$$\text{Produtividade} = (\text{composição ponderada dos } \textit{outputs}) / (\text{composição ponderada dos } \textit{inputs})$$

O DEA (*Data Envelopment Analysis*) originalmente desenvolvida por Charnes, Cooper e Rhodes (1978) é uma metodologia que objetiva comparar a eficiência relativa de unidades produtivas complexas, tais como, setores departamentais, escolas, hospitais, agências bancárias, dentre outras. Desde a primeira publicação em 1978, a metodologia DEA tem-se desenvolvido acentuadamente, registrando inúmeras aplicações que vêm mostrando o quanto esta técnica pode ser considerada uma ferramenta importante para avaliação de eficiência.

Esta metodologia mede a eficiência relativa de unidades de tomada de decisão (*DMU-Decision Making Units*), que desempenham tarefas de transformar múltiplos *inputs* em múltiplos *outputs*. Quando interpretada no contexto da teoria de produção, ela ajuda a transpor a brecha entre a noção teórica de função de produção com uma relação de *inputs* e *outputs* e suas estimativas empíricas.

A análise DEA envolve selecionar *inputs* e *outputs* para produzir uma função de produção empírica que é baseada no comportamento "ótimo" observado. O modelo DEA compara cada uma das DMUs com a melhor prática observada, para

obter a medida de eficiência relativa. Cada DMU é então classificada, como sendo eficiente ou ineficiente.

Como o DEA utiliza um conjunto de metodologias conhecidas da Pesquisa Operacional/Métodos Quantitativos, vejamos uma rápida revisão bibliográfica a seguir.

1.1. Programação Linear

Segundo Fitzsimmons e Fitzsimmons (2000), a Programação Linear (PL) é uma ferramenta computacional de modelagem para tomadas de decisão associadas à alocação de recursos que transcendem todos os aspectos de gerenciamento de gerações de serviços. Ela se refere ao planejamento que utiliza modelos matemáticos que consistem em expressões lineares. Um modelo é uma visão bem estruturada da realidade, ou seja, é uma abstração seletiva da realidade. A modelagem seleciona as características da realidade mais importantes para o problema de interesse. No que diz respeito à tomada de decisão é o ato de selecionar, dentre várias decisões possíveis, a mais adequada para o alcance de certo objetivo.

Ainda conforme Fitzsimmons e Fitzsimmons (2000), modelos de programação linear são uma classe especial de modelos de otimização, para que um determinado sistema possa ser representado por meio de um modelo PL todas as relações entre variáveis são expressas com funções lineares e todos os modelos de PL possuem a seguinte forma algébrica:

$$\begin{array}{l}
 \text{Maximize (ou Minimize):} \quad c_1x_1 + c_2x_2 + \dots + c_nx_n \\
 \text{Sujeito a(S.a.):} \quad \begin{array}{l}
 a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \left\{ \begin{array}{l} \leq \\ = \\ \geq \end{array} \right. b_1 \\
 a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \left\{ \begin{array}{l} \leq \\ = \\ \geq \end{array} \right. b_2 \\
 \vdots \\
 a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \left\{ \begin{array}{l} \leq \\ = \\ \geq \end{array} \right. b_n
 \end{array}
 \end{array}$$

e às restrições de não-negatividade

$$x_1, x_2, \dots, x_n \geq 0$$

Esta estrutura de problema contém as seguintes características:

- Variáveis de Decisão: as variáveis x_1, x_2, \dots, x_n assumem valores reais maiores ou iguais a zero.
- Função Objetivo: a função $c_1x_1 + c_2x_2 + \dots + c_nx_n$ pode ser maximizada (por exemplo, lucros) ou minimizada (por exemplo, custos), isso vai depender da natureza dos coeficientes c_1, c_2, \dots, c_n , onde esta função deve ser maior ou menor possível.
- Restrições: quando valores numéricos são designados para as variáveis de decisão x_1, x_2, \dots, x_n para influenciar a função objetivo, estes valores também influenciam as restrições. Os modelos requerem que os valores numéricos sejam tais que não violem nenhuma restrição. O conjunto de valores de b_1, b_2, \dots, b_n é denominado Lado Direito da Inequação, (RHS - Right-Hand Sides) que limitam os possíveis valores das variáveis de decisão.
- Parâmetros: os coeficientes na função objetivo e os valores RHS são parâmetros. Os parâmetros são entidades cujo valor permanece fixo durante a resolução do problema.

- Constantes: os coeficientes a_{11} , a_{12}, \dots , a_{1n} representam o consumo do recurso RHS por unidade de cada variável de decisão, que refletem uma taxa constante de utilização do recurso.

Portanto o modelo de Programação Linear reduz um sistema real a um conjunto de equações ou inequações onde pretendemos otimizar uma função objetivo. Uma das grandes contribuições à programação matemática desse século, segundo Goldbarg e Luna (2000) é o Algoritmo Simplex. O Simplex é um algoritmo que utiliza um ferramental baseado na Álgebra Linear para determinar, por um método iterativo, a solução ótima de um PPL (Problema de Programação Linear).

De acordo com os autores, dualidade é um conceito amplo que engloba a possibilidade do tratamento de duas naturezas distintas de uma mesma entidade, ou seja, eles definem duais como um par de modelos de programação matemática *primal* e *dual*. Este par de modelos preservam as seguintes condições:

- as funções objetivos são simétricas, isto é, se o primal for de minimização, o dual será de maximização, reciprocamente.
- são simétricas as descrições das restrições, ou seja, se na forma canônica o primal possui restrições \leq então o dual terá restrições \geq .
- os termos independentes no primal surgem como os coeficientes da função objetivo no dual, reciprocamente.
- a matriz de restrição do primal é a transposta da matriz de restrição do dual, reciprocamente.

Os PPLs abaixo generalizam o par de modelos primal x dual.

*Primal**Dual**Minimizar:* $Z_0 = cx$ *Maximizar:* $W_0 = ub$ *Sujeito a:* $Ax \geq b$ *Sujeito a* $uA^T \leq c$ $x \geq 0$ $u \geq 0$

onde x é um vetor coluna e u é um vetor linha; A é a matriz de restrição do *primal* e A^T é a matriz de restrição do *dual*; e b é o coeficiente no *primal* que surge como coeficiente na função objetivo.

Estes PPLs apresentam propriedades importantes como a que referencia ao valor das funções objetivo Z_0 e W_0 quando encontram seus valores ótimos x' e u' . Além desses valores numéricos, existe uma dependência entre a condição de viabilidade de cada um desses modelos, que pode ser verificada através do Teorema das Folgas Complementares, que segundo o autor diz:

“Dado um par de programas duais, uma condição necessária e suficiente para que as soluções x' e u' sejam ótimas é que se verifiquem as seguintes relações de complementaridade de folga: $u(Ax - b) = 0$ e $(c - uA)x = 0$.” (GOLDBARG E LUNA, 2000, p.138).

O que foi dito até aqui, é que existem métodos aritméticos simples capazes de maximizar (ou minimizar) funções lineares sujeitas a restrições em forma de desigualdades lineares.

1.2. Análise Envoltória de Dados (DEA)

Segundo Pereira (1995), os estudos das técnicas de DEA foram iniciadas por Farrel (1957) e ampliadas por Charnes, Cooper e Rhodes (CHARNES, A; COOPER, W.W.; RHODES, E., 1978). O DEA recebeu grande destaque depois da

publicação do artigo introdutório de Charnes, Cooper e Rhodes (1978) para a obtenção de grau de Ph.D de Rhodes.

O objetivo da tese foi desenvolver um método para comparar a eficiência de escolas públicas norte-americanas (Decision Making Units – DMU's) levando em conta “*outputs*” como:

- Scores aritméticos;
- Melhoria de auto-estima medida em testes psicológicos;
- Habilidade psicomotora;

e “*inputs*” como:

- Número de professor-hora;
- Tempo gasto pela mãe em leituras com o filho.

A DEA representa uma das mais adequadas ferramentas para avaliar a eficiência, em comparação com ferramentas convencionais, sendo assim, são destacadas as seguintes características:

- Não requer *a priori* uma função de produção explícita;
- Examina a possibilidade de diferentes, mas igualmente eficientes, combinações de *inputs* e *outputs*;
- Localiza a fronteira eficiente dentro de um grupo analisado e as unidades incluídas;
- Determina, para cada unidade ineficiente, subgrupos de unidades eficientes, os quais formam seu conjunto de referência.

1.3. Abordagem Sistêmica do DEA

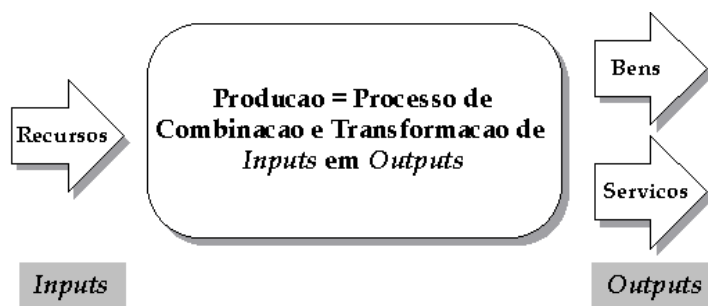
A Administração Contemporânea dissemina a ideia de que para avaliar um sistema organizacional é necessário definir quais são os seus principais *inputs* e

outputs (ALMEIDA; MARIANO; REBELATTO, 2006). Os *inputs* são os recursos ou insumos consumidos pelos processos internos da unidade de produção para produzir produtos ou serviços (*outputs*) aos clientes.

Visando amenizar essa complexidade, as ferramentas matemáticas, mediante o uso dos indicadores, de maneira quantitativa, têm a função de identificar os investimentos mais eficientes e, assim, contribuir para a otimização das tomadas de decisões estratégicas envolvendo a distribuição dos recursos públicos.

Ao longo dos anos, aumentou-se substancialmente a demanda por métodos que avaliassem o índice de eficiência dos gastos públicos e ao mesmo tempo classificar e identificar as mais eficientes e as menos eficientes dentro do conjunto de Unidades Produtivas.

Figura 1 – Diagrama input-processo-output



Fonte: Almeida, Mariano, Rebellato, 2006

Utilizando de ferramentas de TI, modelos matemáticos (usando PL e DEA entre outros) podem ser construídos e avaliados de forma iterativa, alterando o valor de uma variável ou restrição, retirando ou adicionando variáveis ou restrições, proporcionando análise de forma ágil (simulação de diversos cenários).

1.4. Implementação da Análise Envoltória de Dados (DEA)

Lins e Moreira (2000) apresentam uma implementação da metodologia DEA que, segundo eles, foi desenvolvida por Golany e Roll (ROLL,Y; COOK,W.D.; GOLANY,B; 1989), e que é utilizada largamente de maneira formal como intuitivamente. De acordo com os autores, Golany e Roll estabelecem três principais fases:

Fase 1. Determinação do conjunto de DMUs homogêneas a serem avaliadas, ou seja, define e seleciona DMUs a entrarem na análise. Lembrando que uma vez definidas as DMUs, estas devem ser no mínimo o dobro do número de variáveis utilizadas no modelo.

Fase 2. Seleciona as variáveis (*input e output*), considerando a princípio uma grande lista de possíveis variáveis a entrar no modelo. Estas variáveis podem ser controláveis ou não, quantitativas ou qualitativas. Vale a pena ressaltar que, a introdução de um grande número de variáveis reduz a capacidade do DEA de distinguir as DMUs eficientes das ineficientes e, portanto, o modelo deve ser o mais compacto possível para maximizar o poder discriminatório do DEA.

Fase 3. Aplicação do modelo DEA. Conforme os autores, a literatura sobre DEA estão extensivamente referenciada por Charnes (1995) e Coelli (1996). Os principais modelos propostos por Charnes (1995) estão citados no item 1.4.3. Modelos/Formulações DEA, enquanto Coelli (1996) apresenta um guia de operação para um software de DEA.

1.4.1. Seleção de DMUs

As unidades fazem parte de um conjunto homogêneo onde comparações das DMUs fazem sentido. Um grupo de unidades pode ser considerado homogêneo para atender ao objetivo do DEA quando:

- as unidades em consideração desempenham as mesmas tarefas, com os mesmos objetivos;
- os fatores (*input* e *output*) que caracterizam o desempenho de todas as unidades do grupo são iguais, exceto por diferenças em intensidade e magnitude.

Antes da definição dos fatores é necessário definir o tamanho do grupo de comparação, cuja determinação está sujeita a uma situação de conflito. Há uma tendência de se aumentar o número de unidades, pois, numa população grande tem-se uma maior probabilidade de unidades de alto nível de desempenho determinar a fronteira de eficiência (como definida na teoria microeconômica). Um conjunto maior de unidades possibilita, além disso, uma identificação das relações típicas entre *input* e *output* no conjunto. Além disso, com o aumento das unidades é possível incorporar mais fatores dentro da análise. Uma regra estabelecida é que o número de unidades deve ser no mínimo duas vezes o número de *inputs* e *outputs* considerados. Por outro lado, um grande número de unidades no conjunto analisado pode diminuir a homogeneidade dentro do conjunto, aumentando a possibilidade dos resultados serem afetados por fatores exógenos que não são de interesse (ROLL, Y; COOK, W.D.; GOLANY, B; 1989, p. 239).

A determinação de DMUs para entrar no processo de avaliação DEA é afetada por dois tipos de fronteiras. Uma inclui as fronteiras organizacionais, físicas ou regionais que definem as unidades individuais. A outra relata os períodos de tempo usados para medir as atividades das DMUs. Preferencialmente o período de

tempo a ser considerado deve ser o período "natural". Considerando a duração de tais períodos, deve-se considerar que períodos longos podem obscurecer mudanças importantes ocorrendo com as unidades, enquanto períodos curtos podem dar uma figura incompleta das atividades das DMUs.

Deve-se observar que eficiência relativa é medida com referência às DMUs e fatores selecionados. Não há garantia de que a seleção inicial sirva melhor para o objetivo da análise. Portanto, as considerações mencionadas podem requerer que parte da aplicação dos procedimentos propostos sejam iterativamente processadas (ROLL,Y; COOK,W.D.; GOLANY,B; 1989, p. 239).

1.4.2. Seleção de Fatores

A lista inicial de fatores/variáveis – inputs e outputs - a serem consideradas para avaliar o desempenho das DMUs deve ser a maior possível. Em todas as dimensões, as mudanças que podem afetar as DMUs a serem avaliadas devem ser incluídas na lista inicial. Tais fatores podem ser controláveis (fatores gerenciais) ou não-controláveis pelas DMUs (fatores ambientais) e ainda quantitativos ou qualitativos. Todos os fatores que tem relação com a *performance* das DMUs, devem ser listados.

O próximo passo está direcionado para a redução da lista inicial, incluindo somente os mais relevantes. Os fatores selecionados devem claramente distinguir entre as unidades comparadas e trabalhar eficientemente o objetivo da análise. Este refinamento pode ser realizado em três estágios [Golany/Roll, 1989, p.242]:

- seleção criterial;
- análises quantitativas não-DEA;

- análises baseadas no DEA.

1.4.2.1. Seleção criterial

O primeiro estágio para a redução da lista inicial de fatores é um exame criterial pelos tomadores de decisão na área onde as DMUs operam. Já que uma lista de fatores é usualmente compilada, alguns podem estar repetindo virtualmente as mesmas informações, alguns podem não ser considerados como cruciais, enquanto outros podem parecer conflitantes ou conjugados. Um problema nesse estágio é selecionar corretamente os fatores que determinam eficiência. Para ajudar esta seleção, as seguintes perguntas podem ser feitas:

- O fator selecionado contribui para um ou mais dos conjuntos dos objetivos estabelecidos para a análise?
- O fator transporta informações pertinentes não incluídas em outros fatores?
- Os dados referentes ao fator são prontamente disponíveis e geralmente confiáveis?

1.4.2.2. Análises quantitativas não-DEA

O primeiro passo aqui é atribuir valores numéricos aos fatores. Para muitos desses fatores a escolha natural seria as unidades físicas pelos quais eles são medidos. Em princípio, o DEA pode manipular casos com valor zero para alguns dos fatores, contanto que existam no mínimo um *input* e um *output* para cada uma das DMUs que não sejam zero. Tais casos devem ser manuseados com cuidado, pois os algoritmos computacionais podem ser sensíveis ao valor zero.

Um outro passo dentro desse estágio seria descrever a relação de produção que governa as DMUs a serem analisadas e classificar os fatores em *input* e *output*. Qualquer recurso usado por uma DMU deve ser incluído como *input*. Uma DMU converterá recursos para produzir *outputs*, os quais, devem incluir a quantidade de produtos ou serviços produzidos pelas DMUs (vantagens mensuráveis produzidas) e estes produtos ou serviços podem ser produzidos em diferentes níveis de qualidade.

Pode-se também fazer uma análise de correlação para identificar a relação que existe entre os *inputs* e *outputs*, indicando que devem permanecer na lista aqueles fatores que estão bem correlacionados. A análise de correlação tende a eliminar as inconsistências existentes nas informações coletadas.

1.4.2.3. Análises baseadas no DEA

O último passo no processo para examinar e refinar a lista de fatores consiste da realização de testes usando modelos DEA. Os fatores que permaneceram na lista final devem ser considerados no modelo, cujos resultados, podem então, ser examinados estritamente.

1.4.3. Modelos/Formulações DEA

Os modelos DEA são representados por dois modelos básicos a partir do qual foram desenvolvidas variações:

1.4.3.1. Modelo CCR/CRS

Modelo CCR/CRS (CHARNES, COOPER, RHODES, 1978), permite avaliar a eficiência relativa, identificar as DMUs eficientes e ineficientes e determinar a distância da fronteira de eficiência das unidades ineficientes. Trabalha com retornos

constante de escala, ou seja, qualquer variação no *Input* produz uma variação proporcional no *Output*.

Equação que representa o Modelo CCR/CRS - orientado a input – (SANTOS, 2012)

$$\begin{aligned} \max h_o &= \sum_{j=1}^s u_j y_{jo} \\ \text{sujeito a} \\ \sum_{i=1}^r v_i x_{io} &= 1 \\ \sum_{j=1}^s u_j y_{jk} - \sum_{i=1}^r v_i x_{ik} &\leq 0, \quad k = 1, \dots, n \\ u_j, v_i &\geq 0 \quad \forall i, j \end{aligned}$$

Onde:

h_o é a medida de eficiência que será maximizada ou minimizada

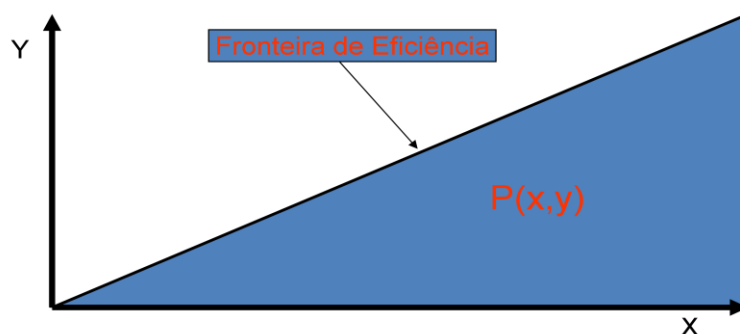
u_i é a importância relativa do *output* i ;

y_i é o valor do *output* i ;

v_j é a importância relativa do *input* j ;

x_j é o valor do *input* j ;

Figura 2 - Diagrama da representação do Modelo CCR/CRS - orientado a *input*



Fonte: SANTOS, 2012, slide 46

Propriedades de P:

- As atividades observadas (x_j, y_j) ($j=1,2,\dots,n$) pertencem a P
- Se uma atividade (x,y) pertence a P, então a atividade (kx,ky) pertence a P para $k \geq 0$. Propriedade dos retornos constante de escala.

- Para uma atividade (x,y) em P , qualquer atividade (x',y') com $x' \geq x$ e $y' \leq y$ esta incluído em P
- Qualquer combinação linear das atividades em P pertencem a P

1.4.3.2. Modelo BCC/VRS

O Modelo BCC/VRS (BANKER, CHARNES E COOPER, 1984), permite a projeção de cada DMU ineficiente sobre a superfície de fronteira (envoltória) determinada pelas DMUs eficientes. Considera retornos variáveis de escala, isto é, substitui o axioma da proporcionalidade entre *inputs* e *outputs* pelo axioma da convexidade. Por isso, esse modelo também é conhecido como VRS – *Variable Returns to Scale*. Ao obrigar que a fronteira seja convexa, o modelo BCC permite que DMUs que operam com baixos valores de *inputs* tenham retornos crescentes de escala e as que operam com altos valores tenham retornos decrescentes de escala.

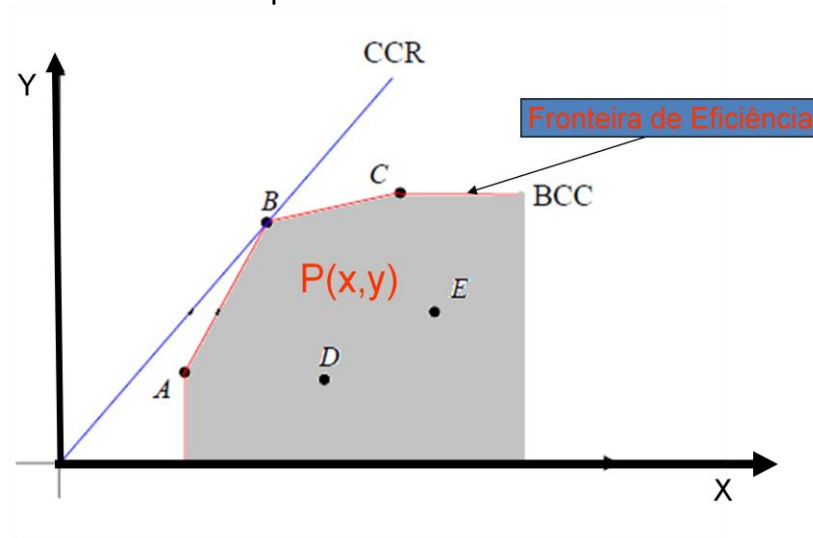
Equação que representa o Modelo BCC/VRS - orientado a input – (SANTOS, 2012)

$$\begin{aligned} & \text{Min } h_0 \\ & \text{sujeito à:} \\ & h_0 x_{io} \geq \sum_{k=1}^n x_{ik} \lambda_k, \forall i, \\ & y_{j0} \leq \sum_{k=1}^n y_{jk} \lambda_k, \forall j, \\ & \lambda_k \geq 0, \forall k. \end{aligned}$$

Onde:

h_o é a medida de eficiência que será maximizada ou minimizada
 u_i é a importância relativa do *output* i ;
 y_i é o valor do *output* i ;
 v_j é a importância relativa do *input* j ;
 x_j é o valor do *input* j ;

Figura 3 - Diagrama que representa o Modelo BCC/VRS - orientado a input -, comparado com o CCR/CRS



Fonte: SANTOS, 2006, slide 61

Propriedades de P:

- As atividades observadas (x_j, y_j) ($j=1,2,\dots,n$) pertencem a P
- Se uma atividade (x_j, y_j) ($j=1,2,\dots,n$) pertencem a P e λ_j ($j=1,2,\dots,n$) são escalares não negativos tais que $\sum_j \lambda_j = 1$, então $\left(\sum_j x_j \lambda_j, \sum_j y_j \lambda_j \right)$ pertence a P, propriedade dos retornos de variáveis de escala ou da convexidade
- Para uma atividade (x,y) em P, qualquer atividade (x',y') com $x' \geq x$ e $y' \leq y$ esta incluído em P
- Qualquer combinação linear das atividades em P pertencem a P

CAPÍTULO II

PROCEDIMENTOS METODOLÓGICOS

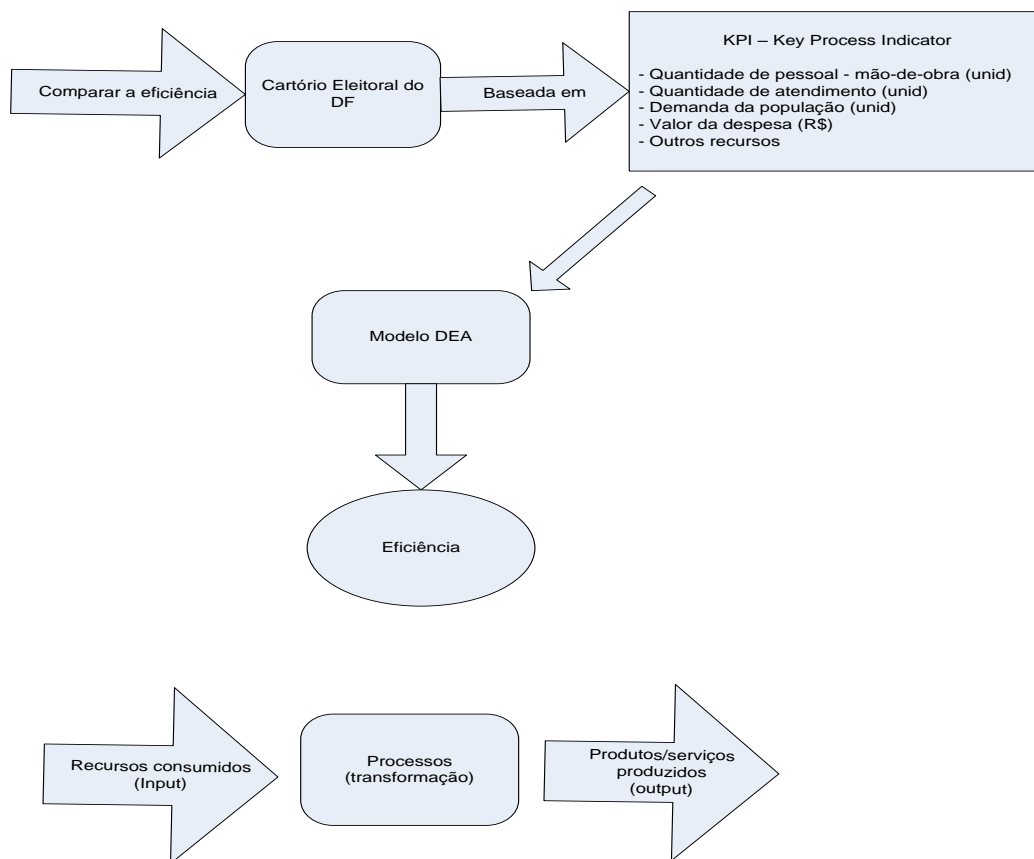
Este estudo utiliza como metodologia principal as técnicas da DEA – *Data Envelopment Analysis* (Análise Envoltória de Dados).

As etapas envolvidas no trabalho seguem os passos consagrados no uso das técnicas do DEA:

- 5.1. Modelagem do problema
- 5.2. Levantar os dados
- 5.3. Definir o modelo de DEA
- 5.4. Selecionar os inputs e outputs
- 5.5. Desenvolvimento do software para gerar as equações, otimizar e gerar os relatórios
- 5.6. Executar o software com os parâmetros (simulação de cenários)
- 5.7. Analisar os resultados da otimização
- 5.8. Analisar os dados de input e output.

2.1. Modelagem do problema

Figura 4 – diagrama do modelo do problema



Os dados foram coletados a partir de informações orçamentárias, despesas com materiais de consumo, gastos com energia, água e esgoto, entre outros custos fixos e variáveis, quantidade de pessoas lotadas no cartório eleitoral, treinamento de pessoal, equipamentos e móveis, resultados de *Correções* (auditoria interna e fiscalização periódica efetuada dentro do Poder Judiciário) nos Cartórios Eleitorais e da estatística do eleitorado do Distrito Federal, no período de janeiro a dezembro de 2009.

A partir das informações coletadas no Quadro 01 que contém os dados dos Cartórios Eleitorais do DF, foram definidos os indicadores utilizados para a análise da eficiência.

Tabela 01 – Dados dos Cartórios Eleitorais do DF

| Cartório eleitoral do DF | (2) Atendimento | (3) Processos Administrativos (quant) | (4) Processos judiciais (quant) | (5) Treinamento (quant) | (6) População (quant) | (7) Eleitorado (quant) | (8) Pessoal (quant) | (9) Água (R\$) | (10) Energia (R\$) | (11) Material (R\$) | (12) Área (m2) |
|--------------------------|-----------------|---------------------------------------|---------------------------------|-------------------------|-----------------------|------------------------|---------------------|----------------|--------------------|---------------------|----------------|
| 01ZE | 3.327 | 748 | 112 | 4 | 103.925 | 106.394 | 12 | 3.075 | 16.976 | 8.115 | 840 |
| 02ZE | 3.609 | 126 | 76 | 3 | 104.965 | 73.599 | 8 | 735 | 4.207 | 6.569 | 290 |
| 03ZE | 5.302 | 313 | 24 | 7 | 98.689 | 75.140 | 10 | 4.569 | 8.796 | 6.984 | 550 |
| 04ZE | 2.686 | 567 | 448 | 4 | 172.172 | 106.466 | 10 | 2.399 | 10.968 | 8.584 | 1.300 |
| 05ZE | 7.278 | 311 | 34 | 6 | 204.091 | 102.148 | 9 | 2.078 | 5.326 | 6.041 | 260 |
| 06ZE | 9.810 | 518 | 26 | 8 | 227.342 | 104.308 | 10 | 1.723 | 5.251 | 7.801 | 260 |
| 07ZE | 3.267 | 524 | 26 | 4 | 61.096 | 44.954 | 7 | 1.699 | 5.493 | 6.610 | 260 |
| 0812ZE | 10.967 | 882 | 545 | 7 | 181.761 | 151.030 | 17 | 7.264 | 32.155 | 13.071 | 900 |
| 09ZE | 5.319 | 296 | 183 | 3 | 138.880 | 111.146 | 9 | 3.505 | 5.876 | 7.147 | 260 |
| 10ZE | 3.002 | 345 | 179 | 2 | 130.977 | 78.310 | 9 | 2.044 | 4.773 | 5.457 | 260 |
| 11ZE | 3.198 | 317 | 35 | 6 | 80.351 | 64.442 | 7 | 5.479 | 7.699 | 9.024 | 290 |
| 13ZE | 9.510 | 428 | 419 | 3 | 141.807 | 79.160 | 9 | 1.976 | 7.221 | 7.916 | 290 |
| 14ZE | 6.015 | 604 | 56 | 2 | 100.911 | 105.393 | 8 | 7.205 | 19.542 | 5.547 | 900 |
| 15ZE | 2.309 | 573 | 72 | 2 | 78.524 | 77.821 | 9 | 2.190 | 9.815 | 4.539 | 550 |
| 16ZE | 5.484 | 253 | 600 | 1 | 87.675 | 75.501 | 8 | 6.634 | 11.507 | 7.673 | 500 |
| 17ZE | 6.036 | 435 | 243 | 6 | 98.188 | 85.671 | 9 | 1.190 | 11.489 | 4.036 | 500 |
| 18ZE | 1.806 | 256 | 50 | 4 | 147.318 | 76.910 | 6 | 0 | 5.550 | 9.631 | 500 |
| 19ZE | 7.626 | 311 | 72 | 4 | 99.067 | 78.467 | 10 | 3.057 | 20.832 | 7.227 | 2.000 |
| 20ZE | 5.484 | 160 | 201 | 3 | 91.011 | 75.558 | 7 | 1.398 | 5.670 | 7.013 | 500 |
| 21ZE | 8.428 | 328 | 341 | 4 | 189.382 | 78.039 | 7 | 2.097 | 7.000 | 6.980 | 290 |
| TOTAL | 110.463 | 8.295 | 3.742 | 83 | 2.538.132 | 1.750.457 | 181 | 60.317 | 206.146 | 145.965 | 11.500 |

Fonte: levantamento feito no TRE-DF

Explicação do Tabela 01 - Dados dos Cartórios Eleitorais do DF

- Coluna 1 – Cartórios Eleitorais. Os Cartórios Eleitorais do DF estão nomeados por número de ZE (Zona Eleitoral, conforme o Quadro 02), e corresponde a uma área que inclui integral ou parcialmente um ou mais RA - Região Administrativa do DF. As ZE 01, 04, 10, 14 e 18 incluem PE - Postos Eleitorais que são unidades subordinadas às ZE e não fazem parte deste trabalho por ter uma estrutura diferenciada em relação às ZE. As ZE 08 e 12 foram analisadas juntas por compartilharem o mesmo espaço físico. A ZE 18 não tem despesa de água por utilizar um espaço cedido pela Administração Regional da Asa Sul.
- Coluna 2 – Quantidade de Atendimento Eleitoral. São todos os atendimentos eleitorais efetuados durante o ano de 2009 nos Cartórios Eleitorais/Zonas Eleitorais do DF. Entende-se por atendimento eleitoral as operações de Alistamento, Transferência, Revisão e 2o. via, É a atividade-fim do Cartório Eleitoral.
- Coluna 3 – quantidade de Processos Administrativos tramitados durante o período.
- Coluna 4 – quantidade de Processos Judiciais tramitados durante o período.
- Coluna 5 – quantidade de treinamentos efetuados no período.
- Coluna 6 – quantidade estimada de habitantes na área de abrangência do Cartório Eleitoral (dados baseados nas estimativas fornecidas pela CODEPLAN – Companhia de Desenvolvimento do Planalto Central, 2009).
- Coluna 7 - quantidade de eleitores alistados (Cadastro Eleitoral do TSE – Tribunal Superior Eleitoral)
- Coluna 8 – quantidade de pessoas no Cartório Eleitoral (força de trabalho)
- Coluna 9 – despesa com água e esgoto (em R\$)

- Coluna 10 – despesa com energia elétrica (em R\$)
- Coluna 11 – despesa com material de expediente, higiene, limpeza e alimentação - café, chá, açúcar, etc - (em R\$)
- Coluna 12 – área construída (em m2)

Segundo o Tribunal Superior Eleitoral (TSE), Zona Eleitoral (ZE) é a região geograficamente delimitada dentro de um determinado Estado (Unidade da Federação) e administrada por um Cartório Eleitoral, que trata da realização das eleições e situação cadastral dos eleitores ali registrados. Pode ser composta por mais de um município, ou ter área coincidente à do município, ou ocupar apenas uma parte do município (no caso de cidades populosas e com muitos eleitores).

Tabela 02 – Zonas Eleitorais do DF (ZE) e área de abrangência das Regiões Administrativas (RA)

| ZE – Zona Eleitoral do DF | RA – Região Administrativa |
|----------------------------------|--|
| 01ZE | Asa Sul |
| 02ZE | Paranoa, Lago Norte, Itapoã, Varjão, Granja do Torto |
| 03ZE | Taguatinga |
| 04ZE | Gama leste, Santa Maria |
| 05ZE | Sobradinho, Sobradinho II |
| 06ZE | Planaltina |
| 07ZE | Brazlândia |
| 08ZE | Ceilândia |
| 09ZE | Guará, Cidade Estrutural |
| 10ZE | N. Bandeirantes, Candango., Riacho Fundo e Fundo II, ParkWay |
| 11ZE | Cruzeiro, SMU, Octogonal, Sudoeste |
| 12ZE | Ceilândia |
| 13ZE | Samanbaia |
| 14ZE | Asa norte |
| 15ZE | Taguatinga, Águas Claras, Park Way |
| 17ZE | Ceilândia |
| 17ZE | Gama |
| 18ZE | Lago Sul, São Sebastião, Cidade Jardim Botânico |
| 19ZE | Taguatinga |
| 20ZE | Ceilândia |
| 21ZE | Rec. Das Emas, Samanbaia |

Fonte: levantamento feito no TRE-DF

2.2. Definição do modelo

Dentre os dois modelos básicos do DEA, foi utilizado o Modelo CCR/CRS neste estudo, que permite avaliar a eficiência relativa, identificar as DMUs eficientes e ineficientes e determinar a distância da fronteira de eficiência das unidades ineficientes além de trabalhar com retornos constante de escala, ou seja, qualquer variação no *Input* produz uma variação proporcional no *Output*.

2.3. Seleção dos Inputs e Outputs

A lista inicial de fatores inclui:

- Quantidade de Atendimento Eleitoral
- Quantidade de Processos Administrativos tramitados durante o período.
- Quantidade de Processos Judiciais tramitados durante o período.
- Quantidade de treinamentos efetuados no período.
- Quantidade estimada de habitantes
- Quantidade de pessoas no Cartório Eleitoral (força de trabalho)
- Despesa com água e esgoto (em R\$)
- Despesa com energia elétrica (em R\$)
- Despesa com material de expediente, higiene, limpeza e alimentação - café, chá, açúcar, etc - (em R\$)
- Área construída (em m²)

Após a análise dos fatores foram definidos os seguintes fatores que mais influenciam para a *performance*:

Fatores de INPUT:

- Quantidade estimada de habitantes

- Quantidade de pessoas no Cartório Eleitoral (força de trabalho)
- Despesa com água, esgoto e energia elétrica (em R\$) – foi aglutinado em único fator
- Despesa com material de expediente, higiene, limpeza e alimentação - café, chá, açúcar, etc - (em R\$)

Fatores de OUTPUT:

- Quantidade de Atendimento Eleitoral
- Quantidade de Processos Administrativos tramitados durante o período.
- Quantidade de Processos Judiciais tramitados durante o período.

Os fatores de Input (recursos) e Output (produtos/serviços) foram selecionados levando em conta os seguintes critérios:

- O Planejamento Estratégico do Judiciário (CNJ – Conselho Nacional de Justiça) para o quadriênio 2010/2014 determina uma redução de despesas operacionais em 5%;
- Utilizar os fatores que contribuem para o objetivo deste trabalho;
- Foi executado várias simulações para aproximar com a realidade.

O Tabela 03 apresenta os dados coletados junto à Administração do TRE-DF no exercício de 2009. As informações estão divididas por Zonas Eleitorais e inclui o quantitativo de: atendimentos eleitorais efetuados, processos administrativos processados, processos judiciais processados, eleitorado cadastrado, pessoal lotado; a despesa de água, elétrica e esgoto e a despesa com material de consumo.

Tabela 03 – Fatores de INPUT/OUTPUT selecionados

| DMU | OUTPUT | | | | INPUT | | |
|--------------|----------------|--------------|--------------|------------------|------------|----------------|----------------|
| | ATENDIM | PA | PJ | ELEITOR | PESSOAL | DESPESA | MATERIAL |
| | (UNIT) | (UNIT) | (UNIT) | (UNIT) | (UNIT) | (R\$) | (R\$) |
| 01ZE | 3.327 | 748 | 112 | 106.394 | 12 | 20.051 | 8.115 |
| 02ZE | 3.609 | 126 | 76 | 73.599 | 8 | 4.942 | 6.569 |
| 03ZE | 5.302 | 313 | 24 | 75.140 | 10 | 13.365 | 6.984 |
| 04ZE | 2.686 | 567 | 448 | 106.466 | 10 | 13.367 | 8.584 |
| 05ZE | 7.278 | 311 | 34 | 102.148 | 9 | 7.404 | 6.041 |
| 06ZE | 9.810 | 518 | 26 | 104.308 | 10 | 6.974 | 7.801 |
| 07ZE | 3.267 | 524 | 26 | 44.954 | 7 | 7.192 | 6.610 |
| 0812ZE | 10.967 | 882 | 545 | 151.030 | 17 | 39.419 | 13.071 |
| 09ZE | 5.319 | 296 | 183 | 111.146 | 9 | 9.380 | 7.147 |
| 10ZE | 3.002 | 345 | 179 | 78.310 | 9 | 6.817 | 5.457 |
| 11ZE | 3.198 | 317 | 35 | 64.442 | 7 | 13.178 | 9.024 |
| 13ZE | 9.510 | 428 | 419 | 79.160 | 9 | 9.197 | 7.916 |
| 14ZE | 6.015 | 604 | 56 | 105.393 | 8 | 26.747 | 5.547 |
| 15ZE | 2.309 | 573 | 72 | 77.821 | 9 | 12.005 | 4.539 |
| 16ZE | 5.484 | 253 | 600 | 75.501 | 8 | 18.141 | 7.673 |
| 17ZE | 6.036 | 435 | 243 | 85.671 | 9 | 12.680 | 4.036 |
| 18ZE | 1.806 | 256 | 50 | 76.910 | 6 | 5.550 | 9.631 |
| 19ZE | 7.626 | 311 | 72 | 78.467 | 10 | 23.889 | 7.227 |
| 20ZE | 5.484 | 160 | 201 | 75.558 | 7 | 7.068 | 7.013 |
| 21ZE | 8.428 | 328 | 341 | 78.039 | 7 | 9.097 | 6.980 |
| TOTAL | 110.463 | 8.295 | 3.742 | 1.750.457 | 181 | 266.462 | 145.965 |
| MÉDIA | 5.523 | 415 | 187 | 87.523 | 9 | 13.323 | 7.298 |

Fatores de Input:

- PESSOAL – Quantidade de pessoas no Cartório Eleitoral (unitário)
- DESPESA – Despesa com água, esgoto e energia elétrica (em R\$) – foi aglutinado em único fator

- MATERIAL – Despesa com material de consumo/expediente, higiene, limpeza e alimentação - café, chá, açúcar, etc - (em R\$)

Fatores de Output:

- ATENDIM – Quantidade de atendimento de eleitores (unitário)
- PA – Quantidade de Processos Administrativos (unitário)
- PJ – Quantidade de Processos Judiciais (unitário)
- ELEITOR – Quantidade de eleitores (unitário)

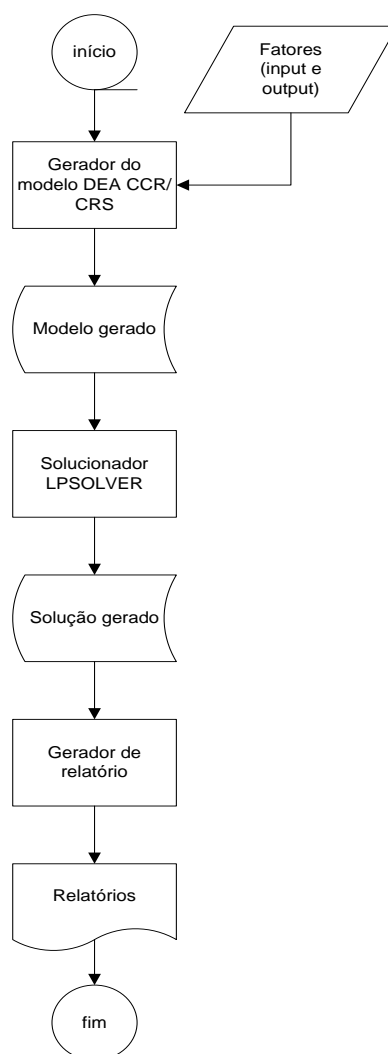
2.4. Desenvolvimento do software para geração do modelo e sua solução

Optamos por desenvolver um gerador automatizado do modelo, um solucionador e gerador de relatório específico para este trabalho, o que permitiu agilizar a simulação de vários cenários, com alteração de inputs. Com o gerador automatizado necessitamos somente dos dados de entrada em formato de planilha, a partir disto, o software gera o modelo, soluciona o problema e emite o relatório simplificado. Os detalhes do desenvolvimento do software estão no Apêndice.

Foi utilizado o pacote open-source LP_SOLVE ver. 5.5.2.0

(<http://www.cs.sunysb.edu/~algorithm/implement/lpsolve/implement.shtml>) para calcular a solução.

Figura 05 - fluxograma de execução do sistema



2.5. Análise dos resultados

2.5.1. Relatório de eficiência relativa

O Tabela 04 mostra que 70% dos Cartórios Eleitorais do Distrito Federal (ZE – Zonas Eleitorais) estão situadas dentro da fronteira de eficiência relativa, confirmando assim a percepção real da administração do TRE - Tribunal Regional Eleitoral do DF e da sociedade.

ZE eficientes (100%): 02, 04, 05, 06, 07, 09, 10, 13, 14, 15, 16, 17, 18, 21 (14 ZE's, 70%)

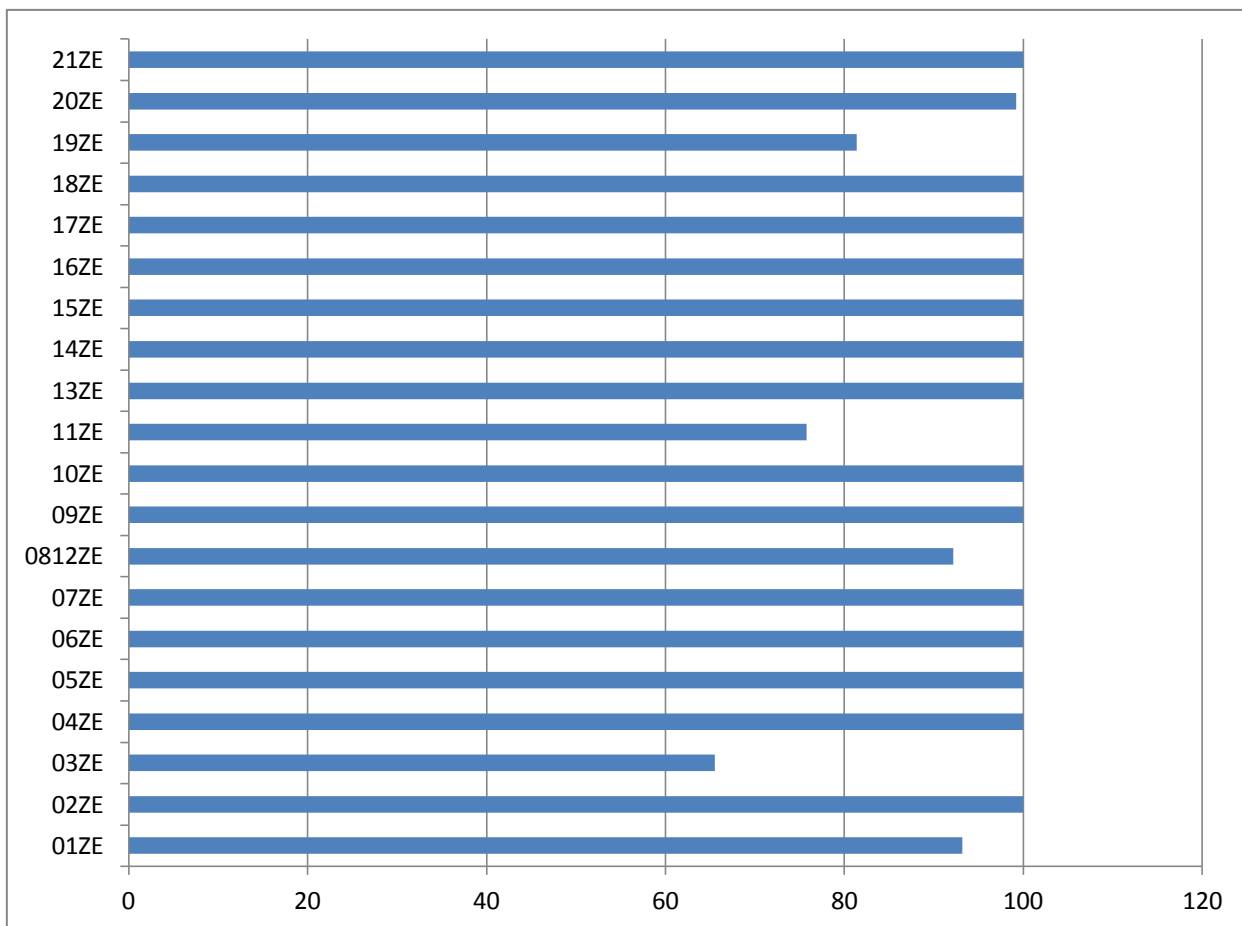
ZE ineficientes (<100%): 01, 03, 08/12, 11, 19 e 20 (6 ZE's, 30%)

Tabela 04 - Relatório de eficiência relativa e peso dos recursos e produtos

| | (%) | PESO | | | | | | |
|---------------|------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| | EFICIENCIA | OUTPUT | | | | INPUT | | |
| DMU | RELATIVA | ATENDIM | PA | PJ | ELEITOR | PESSOAL | DESPESA | MATERIAL |
| 01ZE | 93 | 0,0000 | 0,0008 | 0,0001 | 0,0000 | 0,0583 | 0,0000 | 0,0000 |
| 02ZE | 100 | 0,0000 | 0,0000 | 0,0001 | 0,0000 | 0,0000 | 0,0002 | 0,0000 |
| 03ZE | 66 | 0,0000 | 0,0006 | 0,0000 | 0,0000 | 0,0487 | 0,0000 | 0,0000 |
| 04ZE | 100 | 0,0000 | 0,0009 | 0,0007 | 0,0000 | 0,0000 | 0,0000 | 0,0001 |
| 05ZE | 100 | 0,0000 | 0,0000 | 0,0000 | 0,0000 | 0,0000 | 0,0000 | 0,0001 |
| 06ZE | 100 | 0,0001 | 0,0000 | 0,0000 | 0,0000 | 0,0000 | 0,0000 | 0,0001 |
| 07ZE | 100 | 0,0000 | 0,0018 | 0,0000 | 0,0000 | 0,0000 | 0,0001 | 0,0001 |
| 0812ZE | 92 | 0,0000 | 0,0006 | 0,0005 | 0,0000 | 0,0364 | 0,0000 | 0,0000 |
| 09ZE | 100 | 0,0000 | 0,0000 | 0,0007 | 0,0000 | 0,0000 | 0,0000 | 0,0001 |
| 10ZE | 100 | 0,0000 | 0,0001 | 0,0010 | 0,0000 | 0,0000 | 0,0001 | 0,0001 |
| 11ZE | 76 | 0,0000 | 0,0012 | 0,0001 | 0,0000 | 0,1112 | 0,0000 | 0,0000 |
| 13ZE | 100 | 0,0001 | 0,0000 | 0,0002 | 0,0000 | 0,0000 | 0,0000 | 0,0001 |
| 14ZE | 100 | 0,0000 | 0,0009 | 0,0000 | 0,0000 | 0,0118 | 0,0000 | 0,0002 |
| 15ZE | 100 | 0,0000 | 0,0007 | 0,0000 | 0,0000 | 0,0000 | 0,0000 | 0,0001 |
| 16ZE | 100 | 0,0000 | 0,0000 | 0,0012 | 0,0000 | 0,0000 | 0,0000 | 0,0001 |
| 17ZE | 100 | 0,0000 | 0,0000 | 0,0000 | 0,0000 | 0,0000 | 0,0000 | 0,0002 |
| 18ZE | 100 | 0,0000 | 0,0000 | 0,0006 | 0,0000 | 0,0309 | 0,0001 | 0,0000 |
| 19ZE | 81 | 0,0001 | 0,0000 | 0,0000 | 0,0000 | 0,0250 | 0,0000 | 0,0001 |
| 20ZE | 99 | 0,0000 | 0,0000 | 0,0010 | 0,0000 | 0,0468 | 0,0001 | 0,0000 |
| 21ZE | 100 | 0,0001 | 0,0000 | 0,0005 | 0,0000 | 0,0078 | 0,0000 | 0,0001 |
| MÉDIA | 95 | 0,0000 | 0,0004 | 0,0003 | 0,0000 | 0,0188 | 0,0000 | 0,0001 |

O Gráfico 01 mostra a tabela do Quadro 04, acima, em formato de gráfico em que o eixo horizontal apresenta a eficiência relativa (0 a 100) e o eixo vertical as ZE.

Gráfico 01 - Eficiência Relativa por Zona Eleitoral (visualização gráfica do Quadro 04)



2.5.2. Relatório de sobras de recursos

O Tabela 05 mostra os recursos (input) utilizados pelos processos para gerar os produtos/serviços. Na coluna ATUAL mostra a quantidade consumida de input, o IDEAL mostra a quantidade mínima necessária para produzir o produto/serviço e a coluna SOBRA o percentual de sobra em relação ao IDEAL. (o valor negativo indica o excesso). Com a diminuição do excesso de recurso podemos direcionar estas ZE's ineficientes para aproximar da *performance* das ZE's eficientes.

Observação: os valores fracionários na coluna PESSOAL é uma restrição dessa forma de cálculo. Como indica o número de pessoas, não podemos considerar uma fração, para que este dado tenha mais significância podemos alterar a unidade de medida para horas de trabalho por exemplo.

Tabela 05 - Relatório de sobras em relação aos recursos utilizados (inputs)

| DMU | PESSOAL (UNIT) | | | DESPESA (R\$) | | | MATERIAL (R\$) | | |
|--------|----------------|-------|-----------|---------------|--------|-----------|----------------|--------|-----------|
| | ATUAL | IDEAL | SOBRA (%) | ATUAL | IDEAL | SOBRA (%) | ATUAL | IDEAL | SOBRA (%) |
| 01ZE | 12 | 11,2 | -7,33 | 20.051 | 18.688 | -7,3 | 8.115 | 7.564 | -7,3 |
| 02ZE | 8 | 8 | 0 | 4.942 | 4.942 | 0 | 6.569 | 6.569 | 0 |
| 03ZE | 10 | 6,6 | -52,67 | 13.365 | 8.754 | -52,68 | 6.984 | 4.574 | -52,68 |
| 04ZE | 10 | 10 | 0 | 13.367 | 13.367 | 0 | 8.584 | 8.584 | 0 |
| 05ZE | 9 | 9 | 0 | 7.404 | 7.404 | 0 | 6.041 | 6.041 | 0 |
| 06ZE | 10 | 10 | 0 | 6.974 | 6.974 | 0 | 7.801 | 7.801 | 0 |
| 07ZE | 7 | 7 | 0 | 7.192 | 7.192 | 0 | 6.610 | 6.610 | 0 |
| 0812ZE | 17 | 15,7 | -8,49 | 39.419 | 36.332 | -8,49 | 13.071 | 12.048 | -8,49 |
| 09ZE | 9 | 9 | 0 | 9.380 | 9.380 | 0 | 7.147 | 7.147 | 0 |
| 10ZE | 9 | 9 | 0 | 6.817 | 6.817 | 0 | 5.457 | 5.457 | 0 |
| 11ZE | 7 | 5,3 | -32,08 | 13.178 | 9.981 | -32,02 | 9.024 | 4.972 | -81,48 |
| 13ZE | 9 | 9 | 0 | 9.197 | 9.197 | 0 | 7.916 | 7.916 | 0 |
| 14ZE | 8 | 8 | 0 | 26.747 | 26.747 | 0 | 5.547 | 5.547 | 0 |
| 15ZE | 9 | 9 | 0 | 12.005 | 12.005 | 0 | 4.539 | 4.539 | 0 |
| 16ZE | 8 | 8 | 0 | 18.141 | 18.141 | 0 | 7.673 | 7.673 | 0 |
| 17ZE | 9 | 9 | 0 | 12.680 | 12.680 | 0 | 4.036 | 4.036 | 0 |
| 18ZE | 6 | 6 | 0 | 5.550 | 5.550 | 0 | 9.631 | 9.631 | 0 |
| 19ZE | 10 | 8,1 | -22,85 | 23.889 | 11.018 | -116,82 | 7.227 | 5.880 | -22,91 |
| 20ZE | 7 | 7 | -0,72 | 7.068 | 7.013 | -0,79 | 7.013 | 6.958 | -0,79 |
| 21ZE | 7 | 7 | 0 | 9.097 | 9.097 | 0 | 6.980 | 6.980 | 0 |

CONSIDERAÇÕES FINAIS

A aplicação do modelo DEA sobre os dados de INPUT/OUTPUT do Quadro 03 mostram que 70% dos Cartórios Eleitorais do Distrito Federal (ZE – Zonas Eleitorais) estão situadas dentro da fronteira de eficiência relativa, confirmando assim a percepção real da administração do TRE - Tribunal Regional Eleitoral do DF e da sociedade (Tabela 04). Esta grande quantidade de DMU's situadas na fronteira da eficiência demonstra a homogeneidade das Unidades (Cartórios) em relação à estrutura física, composição da força de trabalho e outros recursos, incluindo uma organização interna e de rotina de trabalho padronizado que vem sendo implementada durante anos.

Por outro lado, a 11ZE que é reconhecida pela administração como um dos Cartórios eficientes, isso baseado no fato de ter como Chefe de Cartório uma funcionária considerada modelo de eficiência; de acordo com os resultados da análise DEA ficou posicionado como um dos menos eficientes. Isto pode ser explicado pelo seguinte motivo, o Cartório possui uma estrutura similar ao dos outros Cartórios, porém a demanda pelos seus serviços é menor que a da maioria.

Os resultados mostram um excesso/desperdício de recursos (input) de pessoal, despesa e material nas unidades 11ZE, 03ZE, 19ZE, 01ZE, 0812ZE e 20ZE sugerindo-se assim um remanejamento destes recursos, trazendo melhoraria no nível de eficiência destas unidades.

Analisando a Tabela 05 – relatório de sobras, verificamos que alguns recursos destoam da média geral, por exemplo:

- Na 11ZE há uma sobra grande de material (consumo, expediente, higiene, limpeza, alimentos) que deverá ser avaliada.

- Na 19ZE há sobra excessiva de despesa com água e energia relativa a outras unidades.

Da mesma forma que os processos de melhoria contínua são processos iterativos, esta metodologia pode ser aplicada de forma iterativa principalmente para simular cenários com diferentes níveis de uso de recurso. Simulações de vão mostrar os impactos provocados pela alteração no nível de inputs.

Este trabalho demonstrou a utilização de várias ferramentas como DEA – Data Envoltery Analysis, PL - Modelos de Programação Linear e software gerador de matriz e solucionador de PL, para prover um sistema que pode ser utilizada de forma iterativa no apoio a decisão em organizações privadas ou governamentais.

REFERÊNCIAS BIBLIOGRÁFICAS

ALMEIDA, M.R.; MARIANO, E.B.; REBELATTO, D.A.N. **A Nova Administração da Produção**: Uma Seqüência de Procedimentos Pela Eficiência. In: IX Seminário de administração da FEA –USP (SEMEAD), São Paulo-SP, 2006, *Anais*.

CAMANHO, Roberto. **A Gestão Pública pode ser mais eficiente?**. Fórum Brasileiro de Processos. Disponível em:

http://www.fbp.org.br/v2/artigos/GetArtigo.asp?t=A_GEST%C3O_P%DABLICA_POD_E_SER_MAIS_EFICIENTE?&ID=20. Acessado em: 05/03/2012, 2012. Hora 21:00.

CHARNES, A.; COOPER, W. W.; RHODES, E. **.Measuring efficiency of decision making units**. European Journal of Operational Research, v. 1, p. 429-444, 1978.

CNJ – Conselho Nacional da Justiça. Disponível em:

<http://www.cnj.jus.br/programas-de-a-a-z/eficiencia-modernizacao-e-transparencia/pj-justica-em-numeros>. Acessado em: 01/12/2012, 2012 Hora 14:00.

COELLI, T.J. **A Guide to DEAP Version 2.1: A Data Envelopment Analysis (Computer) Program**, CEPA Working Paper 96/8, Department of Econometrics, University of New England, Armidale, NSW, Australia, 1996.

FARRELL, M.J. **The Measurement of Productive Efficiency**. Journal of the Royal Statistical Society. Series A (General), vol. 120, n. 3, p. 253-290, 1957.

FITZSIMMONS, James A. e FITZSIMMONS, Mona J. **Administração de serviços: operações, estratégias e tecnologia de informação**. Editora Bookman, Porto Alegre-RS 2000.

GILLEN, D.; LALL, A. **Developing measures of airport productivity and performance**: an application of data envelopment analysis, *Transportation Research–E* 33, p. 261–273, 1997.

GOLDBARG, M. C. LUNA, H. P. **Otimização Combinatória e Programação Linear**: Modelos e Algoritmos. Editora Campus, Rio de Janeiro, Brazil, 2000.

LINS, M.P.E.; MOREIRA, M.C.B. **Método I-O stepwise para seleção de variáveis em modelos de análise envoltória de dados**. In: *Revista Pesquisa Operacional*, v. 19, n. 1, p. 39-50, 1999.

LOBATO, Ricardo L. de Q. **Avaliação de Desempenho de Empresas Estatais no novo contexto de retomada do desenvolvimento**. In: Prêmio DEST/MP de monografias. 2007.

PEREIRA, M. F. **Mensuramento de Eficiência Multidimensional Utilizando Análise de Envelopamento de Dados**: revisão da teoria e aplicações. In: Tese de Mestrado em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis, 1995.

SANTOS, Marcos Aurélio Reis dos. **DEA – Data Envelopment Analysis**. Disponível em:

[\[\\[UEIPrvxnZpeT88aHQ\\]\\(http://www.google.com.br/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&ved=0CCIQFjAA&url=http%3A%2F%2Fwww.feg.unesp.br%2F~fmarins%2Fpo%2Fslides%2F2o.s%2FAula%2520sobre%2520DEA%2FDEA%2520\\(6\\)%2520\\(1\\).ppt&ei=d6G5Ko9gTzgoHwDA&usq=AFQjCNEYEBkRyyOdtMXpaqvBfh1xJltDAQ&sig2=Uj5cnUEIPrvxnZpeT88aHQ\\). Acessado em: 07/03/2012. Hora 10:00.\]\(http://www.google.com.br/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&ved=0CCIQFjAA&url=http%3A%2F%2Fwww.feg.unesp.br%2F~fmarins%2Fpo%2Fslides%2F2o.s%2FAula%2520sobre%2520DEA%2FDEA%2520\(6\)%2520\(1\).ppt&ei=d6G5Ko9gTzgoHwDA&usq=AFQjCNEYEBkRyyOdtMXpaqvBfh1xJltDAQ&sig2=Uj5cn</p></div><div data-bbox=\)](http://www.google.com.br/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&ved=0CCIQFjAA&url=http%3A%2F%2Fwww.feg.unesp.br%2F~fmarins%2Fpo%2Fslides%2F2o.s%2FAula%2520sobre%2520DEA%2FDEA%2520(6)%2520(1).ppt&ei=d<u>uZqUL-</u></p></div><div data-bbox=)

SENGUPTA, J. K. **Efficiency Analysis by Production Frontiers: The Nonparametric Approach**. Kluwe Academics Publishers. 246 p., 1989.

SOARES DE MELLO, J.C.C.B; MEZA, L.A.; GOMES, E.G. et al. **Curso de Análise de Envoltória de Dados**. In: XXXVII SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL. Gramado, RS, 2005.

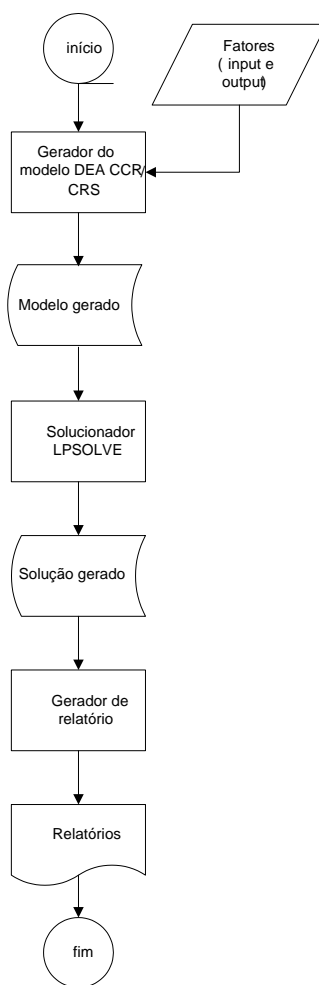
ROLL, Y.; COOK, W.D.; GOLANY, B. **Controlling factor weights in DEA**. In: IIE Transactions, v. 23, n. 1, p. 2-9, 1991.

SOUZA, C. **Estado do Campo da pesquisa em políticas públicas no Brasil**. Revista Brasileira de Ciências Sociais, vol. 18, n.51, p. 15-20, 2003.

APÊNDICE

1. Fluxograma de execução do sistema

Figura 01 – Fluxograma de execução



O gerador do modelo e o gerador de relatórios foram desenvolvidas em linguagem de programação C#, usando o MS-Visual Studio 2010. O solucionador utilizado foi o PL_SOLVE, software livre desenvolvido em linguagem C.

2. Entrada de dados – Fatores de Input e Output

Tabela 01 - Entrada de dados

| | O | O | O | O | I | I | I |
|---------------|---------|-----|-----|---------|---------|---------|----------|
| | ATENDIM | PA | PJ | ELEITOR | PESSOAL | DESPESA | MATERIAL |
| 01ZE | 3.327 | 748 | 112 | 106.394 | 12 | 20.051 | 8.115 |
| 02ZE | 3.609 | 126 | 76 | 73.599 | 8 | 4.942 | 6.569 |
| 03ZE | 5.302 | 313 | 24 | 75.140 | 10 | 13.365 | 6.984 |
| 04ZE | 2.686 | 567 | 448 | 106.466 | 10 | 13.367 | 8.584 |
| 05ZE | 7.278 | 311 | 34 | 102.148 | 9 | 7.404 | 6.041 |
| 06ZE | 9.810 | 518 | 26 | 104.308 | 10 | 6.974 | 7.801 |
| 07ZE | 3.267 | 524 | 26 | 44.954 | 7 | 7.192 | 6.610 |
| 0812ZE | 10.967 | 882 | 545 | 151.030 | 17 | 39.419 | 13.071 |
| 09ZE | 5.319 | 296 | 183 | 111.146 | 9 | 9.380 | 7.147 |
| 10ZE | 3.002 | 345 | 179 | 78.310 | 9 | 6.817 | 5.457 |
| 11ZE | 3.198 | 317 | 35 | 64.442 | 7 | 13.178 | 9.024 |
| 13ZE | 9.510 | 428 | 419 | 79.160 | 9 | 9.197 | 7.916 |
| 14ZE | 6.015 | 604 | 56 | 105.393 | 8 | 26.747 | 5.547 |
| 15ZE | 2.309 | 573 | 72 | 77.821 | 9 | 12.005 | 4.539 |
| 16ZE | 5.484 | 253 | 600 | 75.501 | 8 | 18.141 | 7.673 |
| 17ZE | 6.036 | 435 | 243 | 85.671 | 9 | 12.680 | 4.036 |
| 18ZE | 1.806 | 256 | 50 | 76.910 | 6 | 5.550 | 9.631 |
| 19ZE | 7.626 | 311 | 72 | 78.467 | 10 | 23.889 | 7.227 |
| 20ZE | 5.484 | 160 | 201 | 75.558 | 7 | 7.068 | 7.013 |
| 21ZE | 8.428 | 328 | 341 | 78.039 | 7 | 9.097 | 6.980 |

3. Gerador do modelo DEA

O gerador automatizado utiliza os dados de entrada em formato de planilha, a partir disto, o software gera o modelo, soluciona o problema e emite o relatório simplificado.

O gerador do modelo DEA utiliza as funções existentes no LP_SOLVE (descrito abaixo), constroe a matriz em memória, usando as API (*Application Program Interface*) disponibilizados pelo LP_SOLVE.

4. Solucionador LPSOLVE

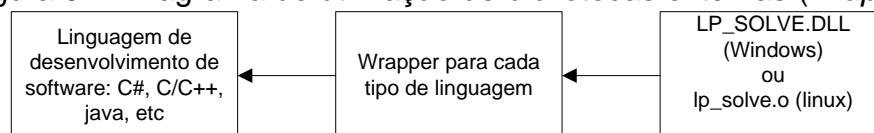
Foi utilizado o pacote open-source LP_SOLVE ver. 5.5.2.0 (<http://www.cs.sunysb.edu/~algorithm/implement/lpsolve/implement.shtml>) para calcular a solução.

O LP_SOLVE é um software que soluciona sistemas de Programação Linear e Programação Linear Inteira (*MILP -Mixed Integer Linear Programming*). O LP_SOLVE é um software livre (*LGPL GNU lesser general public license*), disponibilizando o código-fonte e a documentação.

O LP-SOLVE pode ser chamado como uma biblioteca por diferentes linguagens de desenvolvimento. Foi escrito em ANSI C e pode ser compilado para diferentes plataformas como Linux e Windows. Foi desenvolvido originalmente por Michel Berkelaar da Universidade de Tecnologia de Eindhoven. Ele é basicamente uma biblioteca, um conjunto de rotinas (funções, *API – Application Program Interface*) que pode ser utilizado por diversas linguagens de programação para resolver problemas de Programação Linear.

O LP-SOLVE não é a única biblioteca com estas características, existem outras bibliotecas livres e comerciais como: GLPK, CPLEX, LINDO, MINOS, etc.

Figura 02 - Diagrama de utilização de bibliotecas externas (*wrapper*)



O *wrapper* é a técnica para mapear funções de biblioteca externa (software compilada e linkeditada) e permite o seu uso em linguagens hospedeiros de alto nível.

5. Listagem do programa fonte do gerador, chamada do solucionador e gerador de relatórios do sistema

```

using System;
using System.Collections.Generic;
using System.Text;
using System.Runtime.InteropServices;
using System.IO;
using System.Data;
using aproxUtil;
using aproxCTRL.DEA;

using Excel = Microsoft.Office.Interop.Excel;

namespace lpteste
  
```



```

[DllImport(glpkLibrary, SetLastError = true)]
static extern void glp_simplex(IntPtr lp, int[] options);

[DllImport(glpkLibrary, SetLastError = true)]
static extern void glp_intopt(IntPtr lp, IntPtr options);

[DllImport(glpkLibrary, SetLastError = true)]
private static extern double glp_mip_col_val(IntPtr lp, int col);

[DllImport(glpkLibrary, SetLastError = true)]
static extern double glp_print_sol(IntPtr lp, string filename);

[DllImport(glpkLibrary, SetLastError = true)]
static extern double glp_write_sol(IntPtr lp, string filename);

[DllImport(glpkLibrary, SetLastError = true)]
static extern double glp_write_lp(IntPtr lp, int[] options, string
filename);

[DllImport(glpkLibrary, SetLastError = true)]
static extern double glp_read_mps(IntPtr lp, int fmt, int[] param,
string filename);

[DllImport(glpkLibrary, SetLastError = true)]
static extern double glp_write_mps(IntPtr lp, int fmt, int[] param,
string filename);

[DllImport(glpkLibrary, SetLastError = true)]
static extern void glp_set_obj_dir(IntPtr lp, int dir);

[DllImport(glpkLibrary, SetLastError = true)]
static extern void glp_set_prob_name(IntPtr lp, string name);

[DllImport(glpkLibrary, SetLastError = true)]
static extern string glp_get_prob_name(IntPtr lp);

[DllImport(glpkLibrary, SetLastError = true)]
static extern void glp_set_row_name(IntPtr lp, int irow, string
name);

[DllImport(glpkLibrary, SetLastError = true)]
static extern void glp_set_col_name(IntPtr lp, int icol, string
name);

[DllImport(glpkLibrary, SetLastError = true)]
static extern void glp_set_obj_coef(IntPtr lp, int icol, double
valor);

[DllImport(glpkLibrary, SetLastError = true)]
static extern void glp_delete_prob(IntPtr lp);

[DllImport(glpkLibrary, SetLastError = true)]
static extern int glp_find_row(IntPtr lp, string name);

[DllImport(glpkLibrary, SetLastError = true)]
static extern int glp_find_col(IntPtr lp, string name);

[DllImport(glpkLibrary, SetLastError = true)]
static extern int glp_set_obj_name(IntPtr lp, string name);

```

```
[DllImport(glpkLibrary, SetLastError = true)]
static extern void glp_create_index(IntPtr lp);

[DllImport(glpkLibrary, SetLastError = true)]
static extern int glp_get_num_rows(IntPtr lp);

[DllImport(glpkLibrary, SetLastError = true)]
static extern int glp_get_num_cols(IntPtr lp);

[DllImport(glpkLibrary, SetLastError = true)]
static extern int glp_get_status(IntPtr lp);

[DllImport(glpkLibrary, SetLastError = true)]
static extern double glp_get_obj_val(IntPtr lp);

[DllImport(glpkLibrary, SetLastError = true)]
static extern double glp_get_col_prim(IntPtr lp, int icol);

[DllImport(glpkLibrary, SetLastError = true)]
static extern double glp_get_row_prim(IntPtr lp, int irow);

[DllImport(glpkLibrary, SetLastError = true)]
static extern double glp_get_col_dual(IntPtr lp, int icol);

[DllImport(glpkLibrary, SetLastError = true)]
static extern double glp_get_row_dual(IntPtr lp, int irow);

[DllImport(glpkLibrary, SetLastError = true)]
static extern int glp_get_col_stat(IntPtr lp, int icol);

[DllImport(glpkLibrary, SetLastError = true)]
static extern int glp_get_row_stat(IntPtr lp, int irow);

[DllImport(glpkLibrary, SetLastError = true)]
static extern double glp_get_row_lb(IntPtr lp, int irow);

[DllImport(glpkLibrary, SetLastError = true)]
static extern double glp_get_row_ub(IntPtr lp, int irow);

[DllImport(glpkLibrary, SetLastError = true)]
static extern double glp_get_col_lb(IntPtr lp, int icol);

[DllImport(glpkLibrary, SetLastError = true)]
static extern double glp_get_col_ub(IntPtr lp, int icol);

[DllImport(glpkLibrary, SetLastError = true)]
static extern double glp_get_unbnd_ray(IntPtr lp);

[DllImport(glpkLibrary, SetLastError = true)]
static extern string glp_get_row_name(IntPtr lp, int irow);

[DllImport(glpkLibrary, SetLastError = true)]
static extern string glp_get_col_name(IntPtr lp, int icol);

[DllImport(glpkLibrary, SetLastError = true)]
private static extern void glp_free_env(IntPtr lp);

[DllImport(glpkLibrary, SetLastError = true)]
private static extern void glp_adv_basis(IntPtr lp, int flags);

[DllImport(glpkLibrary, SetLastError = true)]
```

```

        private static extern void glp_set_mat_row(IntPtr lp, int irow, int
ilen, int[] ind, double[] val);

        [DllImport(glpkLibrary, SetLastError = true)]
        private static extern void glp_set_mat_col(IntPtr lp, int icol, int
ilen, int[] ind, double[] val);

        [DllImport(glpkLibrary, SetLastError = true)]
        private static extern void glp_get_mat_row(IntPtr lp, int irow, ref
int[] ind, ref double[] val);

        [DllImport(glpkLibrary, SetLastError = true)]
        private static extern void glp_get_mat_col(IntPtr lp, int icol, ref
int[] ind, ref double[] val);

        [DllImport(glpkLibrary, SetLastError = true)]
        private static extern int glp_get_num_nz(IntPtr lp);

        [DllImport(glpkLibrary, SetLastError = true)]
        private static extern void glp_get_obj_coef(IntPtr lp, int icol);

class Matriz
{
    public int[] i;
    public int[] j;
    public double[] v;

    public Matriz(int max)
    {
        i = new int[max];
        j = new int[max];
        v = new double[max];
    }

    public void popula(int ordem, int idx, int jdx, double val)
    {
        i[ordem] = idx;
        j[ordem] = jdx;
        v[ordem] = val;
    }
}

    public int insereLinha(IntPtr lp, int irow, string nome, int
tipoRestricao, double limmin, double limmax )
    {
        glp_add_rows(lp, 1); glp_set_row_name(lp, irow, nome);
        glp_set_row_bnds(lp, irow, tipoRestricao, limmin, limmax);

        return irow++;
    }

    public int insereColuna(IntPtr lp, int icol, string nome, double
valCoef,
    int tipoRestricao, double limmin, double limmax )
    {
        glp_add_cols(lp, 1); glp_set_col_name(lp, icol, nome);

        glp_set_col_bnds(lp, icol, tipoRestricao, limmin, limmax);
        glp_set_obj_coef(lp, icol, valCoef );
    }
}

```

```

        return icol++;
    }

    public string verInfinito(double valor)
    {
        string svalor = "";

        if (valor < 0)
            svalor = "-";
        else
            svalor = "+";

        if (Math.Abs(valor) > 1.0e+10)
            svalor = "NULL"; //svalor + "inf";
        else
            svalor = string.Format("{0:F6}", valor);

        return svalor;
    }

    public string pegaStatus(int status)
    {
        string[] msg = { "BS", "NL", "NU", "NF", "NS" };

        return msg[status - 1];
    }

    public string pegaEficiencia(string projeto, string dmU, string
tipo)
    {
        aproxCTRL.DEA.Linha lin = new aproxCTRL.DEA.Linha("DBDEA",
"TPDEA");
        aproxCTRL.DEA.solucao sol = new aproxCTRL.DEA.solucao("DBDEA",
"TPDEA");

        // pega indice de eficiencia
        lin.selecionaNome(projeto, "AUXILIAR");
        string idx = lin.indice;
        sol.seleciona(projeto, dmU, idx, tipo);

        string dual = sol.dual;

        dual = (Convert.ToDouble(dual)*100.0).ToString("F2");

        lin.fecha();
        sol.fecha();

        return dual;
    }

    public GLPK(string projeto, string orientacao)
    {
        aproxCTRL.DEA.Coluna DEAcoluna = new
aproxCTRL.DEA.Coluna("DBDEA", "TPDEA");
        aproxCTRL.DEA.Linha DEALinha = new aproxCTRL.DEA.Linha("DBDEA",
"TPDEA");
        aproxCTRL.DEA.dmu DEAdmu = new aproxCTRL.DEA.dmu("DBDEA",
"TPDEA");
        aproxCTRL.DEA.solucao DEAsol = new
aproxCTRL.DEA.solucao("DBDEA", "TPDEA");
    }

```

```

int linha_max = DEAlinha.quantidade(projeto) - 1;

string filename = "resultado.txt";

DEAsol.exclui(projeto);

for (int linha_dmu_base = 0; linha_dmu_base < linha_max;
linha_dmu_base++)
{
    // linha base
    lp = glp_create_prob();

    string numDMU = (linha_dmu_base + 1).ToString();

    string problema = "DMU" + numDMU;

    glp_set_prob_name(lp, problema);

    glp_create_index(lp);

    glp_set_obj_name(lp, "OBJ" + numDMU);

    IDataReader drLinha = DEAlinha.seleciona(projeto);
    while (drLinha.Read())
    {
        int ilin = Convert.ToInt32(Campo.pegaValor(drLinha,
"indice"));
        string nomeLinha = Campo.pegaValor(drLinha, "linha");

        if (ilin < linha_max)
            insereLinha(lp, ilin + 1, nomeLinha, GLP_UP, 0.0,
0.0);
        else
            insereLinha(lp, linha_max + 1, nomeLinha, GLP_FX,
1.0, 1.0);
    }
    drLinha.Close();

    IDataReader drColuna = DEAcoluna.seleciona(projeto);
    while (drColuna.Read())
    {
        int icol = Convert.ToInt32(Campo.pegaValor(drColuna,
"indice"));
        string nomeColuna = Campo.pegaValor(drColuna,
"coluna");
        string tipoColuna = Campo.pegaValor(drColuna, "tipo");

        if (tipoColuna == orientacao)
        {
            double valor = DEAdmu.seleciona(projeto,
linha_dmu_base.ToString(), icol.ToString());
            insereColuna(lp, icol + 1, nomeColuna, valor,
GLP_LO, 0.0, 0.0);
        }
        else
        {
            insereColuna(lp, icol + 1, nomeColuna, 0.0, GLP_LO,
0.0, 0.0);
        }
    }
}

```

```

int numRows = glp_get_num_rows(lp);
int numcols = glp_get_num_cols(lp);

int numtotals = numRows * numcols;

Console.WriteLine(string.Format("num rows={0} cols={1}",
numRows, numcols));

Matriz m = new Matriz(numtotals + 1);

int ordem = 1;

IDataReader drDmu2 = DEAdmu.seleciona(projeto);
while (drDmu2.Read())
{
    int icol = Convert.ToInt32(Campo.pegaValor(drDmu2,
"coluna"));
    int ilin = Convert.ToInt32(Campo.pegaValor(drDmu2,
"linha"));
    double valor = Convert.ToDouble(Campo.pegaValor(drDmu2,
"valor"));

    DEAcoluna.selecionaIndice(projeto, icol.ToString());

    if (DEAcoluna.tipo == orientacao)
    {
        m.popula(ordem++, ilin + 1, icol + 1, valor);
    }
    else
    {
        m.popula(ordem++, ilin + 1, icol + 1, -valor);
    }
}
drDmu2.Close();

IDataReader drColuna3 = DEAcoluna.seleciona(projeto);
while (drColuna3.Read())
{
    int icol = Convert.ToInt32(Campo.pegaValor(drColuna3,
"indice"));
    string tipoColuna = Campo.pegaValor(drColuna3, "tipo");

    if (tipoColuna != orientacao)
    {
        double valor = DEAdmu.seleciona(projeto,
linha_dmu_base.ToString(), icol.ToString());
        m.popula(ordem++, linha_max + 1, icol + 1, valor);
    }
}
drColuna3.Close();

glp_load_matrix(lp, ordem - 1, m.i, m.j, m.v);

//glp_read_mps(lp, GLP_MPS_FILE, null, "PROSUINO.MPS");

/*
if (orientacao == "O")
{
    glp_set_obj_dir(lp, GLP_MAX);
}

```

```

else
{
    glp_set_obj_dir(lp, GLP_MIN);
}
*/

glp_set_obj_dir(lp, GLP_MAX);

glp_write_mps(lp, GLP_MPS_DECK, null, "MATRIZ" + numDMU +
".MPS");

glp_adv_basis(lp, 0);

glp_simplex(lp, null);

double z = glp_get_obj_val(lp);

Console.WriteLine();

util.wr2Exclui(filename);

//string problema = glp_get_prob_name(lp);
string objnome = string.Format("{0:F6}", z);

//aproxUtil.util.wr2Stream(filename,problema+"
"+objnome,true);
//aproxUtil.util.wr2Stream(filename, "ATIVIDADE", true);

for (int i = 1; i <= glp_get_num_cols(lp); i++)
{
    string col_name = glp_get_col_name(lp, i);
    int col_stat1 = glp_get_col_stat(lp, i);

    double col_prim1 = glp_get_col_prim(lp, i);
    double col_dual1 = glp_get_col_dual(lp, i);
    double col_lb = glp_get_col_lb(lp, i);
    double col_ub = glp_get_col_ub(lp, i);

    string lb = verInfinito(col_lb);
    string ub = verInfinito(col_ub);

    DEAsol.projeto = projeto;
    DEAsol.dmu = linha_dmu_base.ToString();
    DEAsol.indice = (i - 1).ToString();
    DEAsol.tipo = "2";
    DEAsol.estado = col_stat1.ToString();
    DEAsol.atividade = col_prim1.ToString("F6");
    DEAsol.dual = col_dual1.ToString("F6");
    DEAsol.limmin = lb;
    DEAsol.limmax = ub;

    DEAsol.insere();
}

//DEAsol.fecha();

for (int i = 1; i <= glp_get_num_rows(lp); i++)
{
    string row_name = glp_get_row_name(lp, i);
    int row_stat1 = glp_get_row_stat(lp, i);

```

```

        double row_prim1 = glp_get_row_prim(lp, i);
        double row_dual1 = glp_get_row_dual(lp, i);
        double row_lb = glp_get_row_lb(lp, i);
        double row_ub = glp_get_row_ub(lp, i);

        string lb = verInfinito(row_lb);
        string ub = verInfinito(row_ub);

        //string restricao = string.Format("{0,-16} {1,3} {2,9}
{3,9} {4,9} {5,9}",
        //
        row_name,
pegaStatus(row_stat1), row_prim1.ToString("F6"),
        //
row_dual1.ToString("F6"), lb, ub);

        DEAsol.projeto = projeto;
        DEAsol.dmu = linha_dmu_base.ToString();
        DEAsol.indice = (i - 1).ToString();
        DEAsol.tipo = "1";
        DEAsol.estado = row_stat1.ToString();
        DEAsol.atividade = row_prim1.ToString("F6");
        DEAsol.dual = row_dual1.ToString("F6");
        DEAsol.limmin = lb;
        DEAsol.limmax = ub;

        DEAsol.insere();
    }

    glp_print_sol(lp, "solucao" + numDMU + ".txt");
    //glp_write_sol(lp, "solucaowrite" + numDMU + ".txt");
    //glp_write_lp(lp, null, "solucaolp" + numDMU + ".txt");

    glp_delete_prob(lp);

}

TABELA tb = new TABELA();

tb.insereColuna("DMU", typeof(string));
tb.insereColuna("EFICIENCIA(%)", typeof(string));

IDataReader drCol = DEAcoluma.seleciona(projeto);
while (drCol.Read())
{
    string nomeColuna = Campo.pegaValor(drCol, "coluna");
    tb.insereColuna(nomeColuna, typeof(string));
}
drCol.Close();

IDataReader drPeso = DEAsol.seleciona(projeto, "2");

string DMUant = "";

int tam = tb.tamColunas();
string[] valorColuna = new string[tam];

int tamCol = 2;

while (drPeso.Read())
{
    string idxDMU = Campo.pegaValor(drPeso, "dmu");

```

```

        if (DMUant == "")
            DMUant = idxDMU;

        if (idxDMU == DMUant)
        {
            valorColuna[tamCol] = Campo.pegaValor(drPeso,
"atividade");
            tamCol++;
        }
        else
        {
            DEAlinha.selecionaIndice(projeto, DMUant);

            valorColuna[0] = DEAlinha.linha;
            valorColuna[1] = pegaEficiencia(projeto, DMUant, "1");

            tb.insereLinha(valorColuna);

            tamCol = 2;
            valorColuna[tamCol] = Campo.pegaValor(drPeso,
"atividade");
            tamCol++;

            DMUant = idxDMU;
        }
    }

    DEAlinha.selecionaIndice(projeto, DMUant);
    valorColuna[0] = DEAlinha.linha;
    valorColuna[1] = pegaEficiencia(projeto, DMUant, "1");
    tb.insereLinha(valorColuna);

    tb.geraHTML("testel.html", "Relatorio de eficiencia relativa e
peso dos recursos e produtos");
    tb.geraExcel("testel.xls", "", 1);

    drPeso.Close();

    DEAdmu.fecha();
    DEAlinha.fecha();
    DEAcoluma.fecha();

    //string filename = "resultado.txt";

    //aproxUtil.util.wr2Stream(filename, "NIVEL DE INPUT IDEAL E
DESPERDICIO", true);
    //aproxUtil.util.wr2Stream(filename, "DMU          COLUNA
(IDEAL, ATUAL E EXCESSO) ", true);

    IDataReader drSol2 = DEAsol.calcFolga(projeto, "1", "I");

    string linAnt = "";
    string colAnt = "";
    string idxlinAnt = "";
    string idxcolAnt = "";

    TABELA tb2 = new TABELA();

    IDataReader drCol9 = DEAcoluma.seleciona(projeto);
    tb2.insereColuna("DMU", typeof(string));

```

```

while (drCol9.Read())
{
    string nomeColuna = Campo.pegaValor(drCol9, "coluna");
    tb2.insereColuna(nomeColuna+"(ATUAL)", typeof(string));
    tb2.insereColuna(nomeColuna+"(IDEAL)", typeof(string));
}
drCol9.Close();

int tam9 = tb2.tamColunas();
string[] valorColuna9 = new string[tam9];

int tamCol9 = 1;

bool rc = drSol2.Read();

while (rc)
{
    string linha = Campo.pegaValor(drSol2, "linha");
    string coluna = Campo.pegaValor(drSol2, "coluna");
    double valor = Convert.ToDouble(Campo.pegaValor(drSol2,
"valor"));
    double dual = Convert.ToDouble(Campo.pegaValor(drSol2,
"dual"));

    string idxcol = Campo.pegaValor(drSol2,"indice");
    string idxlin = Campo.pegaValor(drSol2,"dmu");

    if (linAnt == "")
        linAnt = linha;
    if (colAnt == "")
        colAnt = coluna;
    if (idxcolAnt == "")
        idxcolAnt = idxcol;
    if (idxlinAnt == "")
        idxlinAnt = idxlin;

    double valCol = 0;

    while (rc & (colAnt == coluna) & (linAnt == linha))
    {
        valCol = valCol + valor * dual;

        rc = drSol2.Read();

        if (rc)
        {
            linha = Campo.pegaValor(drSol2, "linha");
            coluna = Campo.pegaValor(drSol2, "coluna");
            valor = Convert.ToDouble(Campo.pegaValor(drSol2,
"valor"));
            dual = Convert.ToDouble(Campo.pegaValor(drSol2,
"dual"));

            idxcol = Campo.pegaValor(drSol2, "indice");
            idxlin = Campo.pegaValor(drSol2, "dmu");

            if (colAnt != coluna)
            {
                DEAdmu.seleciona(projeto, idxlinAnt,
idxcolAnt);
            }
        }
    }
}

```

```

        valorColuna9[tamCol9] = DEAdmu.valor;
        valorColuna9[tamCol9+1] =
valCol.ToString("F2");
        tamCol9 = tamCol9 + 2;

        colAnt = coluna;
        idxcolAnt = idxcol;
        valCol = 0;
    }

    if (linAnt != linha)
    {
        valorColuna9[0] = linAnt;
        tamCol9 = 1;

        tb2.insererLinha(valorColuna9);

        linAnt = linha;
        idxlinAnt = idxlin;
    }
}
else
{
    DEAdmu.seleciona(projeto, idxlinAnt, idxcolAnt);

    valorColuna9[0] = linAnt;
    valorColuna9[tamCol9] = DEAdmu.valor;
    valorColuna9[tamCol9 + 1] = valCol.ToString("F2");

    tb2.insererLinha(valorColuna9);
}
}

if (rc)
    rc = drSol2.Read();
}

tb2.geraHTML("teste2.html", "Relatorio");
tb2.geraExcel("teste2.xls", "", 2);

drSol2.Close();

//aproxUtil.util.wr2Stream(filename, "NIVEL DE INEFICIENCIA",
true);
//aproxUtil.util.wr2Stream(filename, "DMU          NIVEL INEFIC
CUSTO REDUZIDO ", true);

TABELA tb3 = new TABELA();

tb3.insererColuna("DMU", typeof(string));

IDataReader drCol3 = DEAcoluma.seleciona(projeto);
while (drCol3.Read())
{
    string nomeColuna = Campo.pegaValor(drCol3, "coluna");
    tb3.insererColuna(nomeColuna, typeof(string));
}
drCol3.Close();

int tam8 = tb3.tamColunas();
string[] valorColuna8 = new string[tam8];

```

```

int tamCol8 = 1;

aproxCTRL.DEA.solucao DEAsol3 = new
aproxCTRL.DEA.solucao("DBDEA", "TPDEA");

string lin3Ant = "";

IDataReader drSol3 = DEAsol.calcCustoReduzido(projeto);
while (drSol3.Read())
{
    string linha = Campo.pegaValor(drSol3, "linha");
    int coluna = Convert.ToInt32(Campo.pegaValor(drSol3,
"indice"));
    double valor = Convert.ToDouble(Campo.pegaValor(drSol3,
"dual"));

    if (lin3Ant == "")
        lin3Ant = linha;

    if (lin3Ant == linha)
    {
        valorColuna8[coluna + 1] =
(Math.Abs(valor)).ToString("F2");
    }
    else
    {
        valorColuna8[0] = lin3Ant;
        tb3.inseraLinha(valorColuna8);
        lin3Ant = linha;
    }
}
drSol3.Close();

valorColuna8[0] = lin3Ant;
tb3.inseraLinha(valorColuna8);

tb3.geraHTML("teste3.html", "Relatorio de custo reduzido");
tb3.geraExcel("teste3.xls", "", 3);

DEAsol3.fecha();

//aproxUtil.util.wr2Stream(filename, "LAMBDA", true);

TABELA tb4 = new TABELA();

IDataReader drCol4 = DEAlinha.seleciona(projeto);
tb4.inseraColuna("DMU", typeof(string));
while (drCol4.Read())
{
    string nomeColuna = Campo.pegaValor(drCol4, "linha");
    if (nomeColuna != "AUXILIAR")
        tb4.inseraColuna(nomeColuna, typeof(string));
}
drCol4.Close();

int tam4 = tb4.tamColunas();
string[] valorColuna4 = new string[tam4];

int tamCol4 = 1;

```

```

string lin4Ant = "";
double valor4 = 0;

IDataReader drSol4 = DEAsol.calcLambida(projeto);
while (drSol4.Read())
{
    string linha = Campo.pegaValor(drSol4, "linha");
    string idxlin = Campo.pegaValor(drSol4, "dmu");
    valor4 = Convert.ToDouble(Campo.pegaValor(drSol4, "dual"));

    if (lin4Ant == "")
        lin4Ant = linha;

    if (lin4Ant == linha)
    {
        valorColuna4[tamCol4] = valor4.ToString("F4");
        tamCol4++;
    }
    else
    {
        valorColuna4[0] = lin4Ant;
        tb4.insereLinha(valorColuna4);

        tamCol4 = 1;
        valorColuna4[tamCol4] = valor4.ToString("F4");
        tamCol4++;

        lin4Ant = linha;
    }
}

valorColuna4[0] = lin4Ant;
tb4.insereLinha(valorColuna4);

drSol4.Close();

tb4.geraHTML("teste4.html", "Relatorio LAMBDA");

tb4.geraExcel("teste4.xls", "", 4);

DEAsol.fecha();
}

~GLPK()
{
    glp_free_env(lp);
    Console.WriteLine("Terminado com sucesso!");
}

}

public class TABELA
{
    public DataTable dt;

    public TABELA()
    {
        dt = new DataTable();
    }

    public int tamColunas()

```

```

    {
        return dt.Columns.Count;
    }

    public void insereColuna(string coluna, Type tipo)
    {
        dt.Columns.Add(new DataColumn(coluna, tipo));
    }

    public void insereLinha( string[] valores )
    {
        DataRow rw = dt.NewRow();

        for (int i = 0; i < valores.Length; i++)
        {
            rw[i] = valores[i];
        }

        dt.Rows.Add( rw );
        //dt.Rows.Add("DMU2", 3123.22, 299.11);
    }

    public void lista()
    {
        DataRow[] rc = dt.Select();

        Console.WriteLine();

        foreach (DataColumn cc in dt.Columns)
        {
            Console.Write("{0,12}", cc.ColumnName);
        }

        Console.WriteLine();

        foreach (DataRow rw in rc)
        {
            for (int i = 0; i < rw.ItemArray.Length; i++)
            {
                Console.Write("{0,12}", rw[i]);
            }

            Console.WriteLine();
        }
    }

    public void geraHTML(string filename, string titulo)
    {
        DataRow[] rc = dt.Select();

        util.wr2Exclui(filename);

        string str = "<html> " +
            " <body> <h1> " + titulo + " </h1> ";

/*
        string s = " <hr> " +
            " <form action=''> " +
            " Pick a User: " +
            " <select name='username'> " +
            " <option value='volvo'>Volvo</option>" +

```

```

        " <option value='saab'>Saab</option>" +
        " </select> " +
        " <br /> Pick an Account Status: " +
        " <input type='radio' name='account' value='valid'
/> Current " +
        " <input type='radio' name='account' value='not
valid' /> Delinquent <br /> " +
        " <input type='submit' value='Submit' /> " +
        " </form> " +
        " <hr> ";
    util.wr2Stream(filename, s, true);
*/
    str = str + "<table border=1>";
    str = str + "<tr>";

    foreach (DataColumn cc in dt.Columns)
    {
        str = str + "<th align=center width=10%>" + cc.ColumnName +
"</th>";
    }

    str = str + "</tr>";

    foreach (DataRow rw in rc)
    {
        str = str + "<tr>";
        for (int i = 0; i < rw.ItemArray.Length; i++)
        {
            if (i == 0)
                str = str + "<th align=center>" + rw[i] + "</th>";
            else
                str = str + "<td align=right>" + rw[i] + "</td>";
        }
        str = str + "</tr>";
    }

    str = str + "</table>";

    str = str + " </body></html> ";

    util.wr2Stream(filename, str, true);
}

public void geraExcel(string filename, string titulo, int plan)
{
    Excel.Application xlApp;
    Excel.Workbook xlWorkBook;
    Excel.Worksheet xlWorkSheet;
    object misValue = System.Reflection.Missing.Value;
    Excel.Range chartRange;

    xlApp = new Excel.ApplicationClass();
    xlWorkBook = xlApp.Workbooks.Add(misValue);

    if (xlWorkBook.Worksheets.Count < plan)
        xlWorkSheet = (Excel.Worksheet)xlWorkBook.Worksheets.Add();
    else
        xlWorkSheet =
(Excel.Worksheet)xlWorkBook.Worksheets.get_Item(plan);

    xlWorkSheet.Select();
}

```

```

//range = xlWorkSheet.UsedRange;
//for (rCnt = 1; rCnt <= range.Rows.Count; rCnt++)
//{
//    for (cCnt = 1; cCnt <= range.Columns.Count; cCnt++)
//    {
//        str = (string)(range.Cells[rCnt, cCnt] as
Excel.Range).Value2;
//        MessageBox.Show(str);
//    }
//}

DataRow[] rc = dt.Select();

int lin = 3;
int col = 1;

xlWorkSheet.Cells[2, col] = titulo;

foreach (DataColumn cc in dt.Columns)
{
    xlWorkSheet.Cells[lin, col] = cc.ColumnName;
    col++;
}

lin++;

foreach (DataRow rw in rc)
{
    col = 1;

    for (int i = 0; i < rw.ItemArray.Length; i++)
    {
        xlWorkSheet.Cells[lin, col] = rw[i];
        col++;
    }
    lin++;
}

/*
xlWorkSheet.get_Range("b2", "e3").Merge(false);

chartRange = xlWorkSheet.get_Range("b2", "e3");
chartRange.FormulaR1C1 = "MARK LIST";
chartRange.HorizontalAlignment = 3;
chartRange.VerticalAlignment = 3;

chartRange = xlWorkSheet.get_Range("b4", "e4");
chartRange.Font.Bold = true;
chartRange = xlWorkSheet.get_Range("b9", "e9");
chartRange.Font.Bold = true;

chartRange = xlWorkSheet.get_Range("b2", "e9");
chartRange.BorderAround(Excel.XlLineStyle.xlContinuous,
Excel.XlBorderWeight.xlMedium, Excel.XlColorIndex.xlColorIndexAutomatic,
Excel.XlColorIndex.xlColorIndexAutomatic);
*/

xlWorkBook.SaveAs(@"c:\aproxima\aprox50\prg\"+filename,
Excel.XlFileFormat.xlWorkbookNormal, misValue, misValue, misValue,

```

```

misValue, Excel.XlSaveAsAccessMode.xlExclusive, misValue, misValue,
misValue, misValue, misValue);
    xlWorkbook.Close(true, misValue, misValue);
    xlApp.Quit();

    releaseObject(xlApp);
    releaseObject(xlWorkbook);
    releaseObject(xlWorksheet);

    //MessageBox.Show ("File created !");
}

private void releaseObject(object obj)
{
    try
    {
        System.Runtime.InteropServices.Marshal.ReleaseComObject(obj);
        obj = null;
    }
    catch (Exception ex)
    {
        obj = null;
        //MessageBox.Show("Unable to release the Object " +
ex.ToString());
    }
    finally
    {
        GC.Collect();
    }
}

}

class test
{
    public static void Main(string[] args)
    {
        string projeto = "1";

        migraDEA p = new migraDEA();

        p.leExcel("1",@"c:\aproxima\aprox50\prg\DMU-TRE.xls",1);
        GLPK lp = new GLPK(projeto, "O");
    }
}
}

```