UNIVERSIDADE FEDERAL DE GOIÁS
INSTITUTO DE INFORMÁTICA

CARLOS ANTÔNIO CAMPOS JORGE

Algoritmo Evolutivo Multi-Objetivo de Tabelas para Seleção de Variáveis em Calibração Multivariada

Goiânia
2014

UNIVERSIDADE FEDERAL DE GOIÁS

INSTITUTO DE INFORMÁTICA

AUTORIZAÇÃO PARA PUBLICAÇÃO DE DISSERTAÇÃO EM FORMATO ELETRÔNICO

Na qualidade de titular dos direitos de autor, **AUTORIZO** o Instituto de Informática da Universidade Federal de Goiás – UFG a reproduzir, inclusive em outro formato ou mídia e através de armazenamento permanente ou temporário, bem como a publicar na rede mundial de computadores (*Internet*) e na biblioteca virtual da UFG, entendendo-se os termos "reproduzir" e "publicar" conforme definições dos incisos VI e I, respectivamente, do artigo 5º da Lei nº 9610/98 de 10/02/1998, a obra abaixo especificada, sem que me seja devido pagamento a título de direitos autorais, desde que a reprodução e/ou publicação tenham a finalidade exclusiva de uso por quem a consulta, e a título de divulgação da produção acadêmica gerada pela Universidade, a partir desta data.

Título: Algoritmo Evolutivo Multi-Objetivo de Tabelas para Seleção de Variáveis em Calibração Multivariada

Autor(a): Carlos Antônio Campos Jorge

Goiânia, 08 de Abril de 2014.

Carlos Antônio Campos Jorge – Autor

Dr. Anderson da Silva Soares – Orientador

CARLOS ANTÔNIO CAMPOS JORGE

Algoritmo Evolutivo Multi-Objetivo de Tabelas para Seleção de Variáveis em Calibração Multivariada

Dissertação apresentada ao Programa de Pós–Graduação do Instituto de Informática da Universidade Federal de Goiás, como requisito parcial para obtenção do título de Mestre em Programa de Pós-Graduação em Computação.

Área de concentração: Reconhecimento de Padrões e Análise Multivariada.
Orientador: Prof. Dr. Anderson da Silva Soares

Goiânia
2014

CARLOS ANTÔNIO CAMPOS JORGE

Algoritmo Evolutivo Multi-Objetivo de Tabelas para Seleção de Variáveis em Calibração Multivariada

Dissertação defendida no Programa de Pós–Graduação do Instituto de Informática da Universidade Federal de Goiás como requisito parcial para obtenção do título de Mestre em Programa de Pós-Graduação em Computação, aprovada em 08 de Abril de 2014, pela Banca Examinadora constituída pelos professores:

Prof. Dr. Anderson da Silva Soares
Instituto de Informática – UFG
Presidente da Banca

Prof. Dr. Alexandre Cláudio Botazzo Delbem
Instituto de Ciências Matemáticas e de Computação – USP

Prof. Dr. Clarimar José Coelho
Departamento de Computação – PUCGO

Carlos Antônio Campos Jorge

Graduou-se em Engenharia da Computação na PUC Goiás - Pontifícia Universidade Católica de Goiás. Durante sua graduação, foi monitor no departamento de Computação da PUC Goiás. Durante o Mestrado, foi bolsista da CAPES e obteve algumas publicações importantes, as quais contribuíram para o desenvolvimento desta dissertação.

À minha família.

Agradecimentos

Agradeço aos meus pais, pelo apoio em todos os sentidos. Sem eles, eu não teria conseguido chegar até aqui.

Ao meu orientador, Prof^{o}. Dr^{o}. Anderson da Silva Soares, por sua paciência e contribuição para o desenvolvimento deste trabalho, por todos os conselhos que me forneceu e por me ajudar na escrita dos artigos, os quais foram importantes para a realização deste trabalho.

À Prof^{a}. Dr^{a}. Telma Woerle de Lima, também por sua paciência e contribuição para o desenvolvimento deste trabalho

Agradeço ao Prof^{o}. Dr^{o}. Clarimar pela colaboração neste trabalho, pela paciência e por me proporcionar a oportunidade de participar do grupo de pesquisa em matemática computacional.

À minha namorada, Wanessa Rodrigues de Sousa, pelo carinho, pelo apoio, pela paciência, pela motivação e compreensão em todos os momentos.

Agradeço à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelo fornecimento de minha bolsa de estudo. Sem esse apoio financeiro, dificilmente eu teria conseguido realizar este trabalho e adquirir os equipamentos que foram necessários para a obtenção de resultados.

Enfim, agradeço a todos aqueles que, de alguma forma, contribuíram para a realização deste trabalho, que, na verdade, é um meio para se atingir um fim.

"Tornamos-nos significantes pela coragem de nossas perguntas e pela profundidade de nossas respostas."

Carl Sagan,
Cosmos.

Resumo

Jorge, Carlos Antônio Campos. **Algoritmo Evolutivo Multi-Objetivo de Tabelas para Seleção de Variáveis em Calibração Multivariada**. Goiânia, 2014. 66p. Dissertação de Mestrado. Instituto de Informática, Universidade Federal de Goiás.

Este trabalho propõe o uso de algoritmo multi-objetivo evolutivo que faz uso de subconjuntos armazenados em uma estrutura de dados chamada tabela em que os melhores indivíduos de cada objetivo são preservadas. Esta abordagem é comparada neste trabalho com o algoritmo evolutivo tradicional mono-objetivo e outros algoritmos clássicos (MONO-GA-MLR, PLS, APS-MLR) e com o algoritmo multi-objetivo clássico NSGA-II-MLR. Como estudo de caso, o problema de calibração multivariada envolve a previsão da concentração de proteínas em amostras de trigo a partir das medições espectrofotométricas. Os resultados mostraram que a formulação proposta seleciona um número menor de variáveis e apresenta um erro de predição menor quando comparada com o algoritmo evolutivo mono-objetivo. Quando comparado com os algoritmos clássicos PLS e APS-MLR e com o algoritmo multi-objetivo clássico NSGA-II-MLR, o algoritmo proposto apresenta um erro de predição menor, porém com um número maior de variáveis selecionadas. Finalmente, um estudo de sensibilidade à ruído foi realizado. A solução obtida pela formulação proposta apresentou melhores resultados quando comparado com o algoritmo mono-objetivo e NSGA-II-MLR e desempenho similar à solução obtida com o SPA-MLR.

Palavras–chave

Seleção de Variáveis, Algoritmos Evolutivos, Calibração, Algoritmos Multi-Objetivos

Abstract

Jorge, Carlos Antônio Campos. **Multi-Objective Evolutionary Algorithm in Tables for Variable Selection in Multivariate Calibration**. Goiânia, 2014. 66p. MSc. Dissertation. Instituto de Informática, Universidade Federal de Goiás.

This work proposes the use of a multi-objective evolutionary algorithm that makes use of subsets stored in a data structure called table in which the best individuals from each objective considered are preserved. This approach is compared in this work with the traditional mono-objective evolutionary algorithm (GA), classical algorithms (PLS and SPA) and another classic multi-objective algorithm (NSGA-II). As a case study, a multivariate calibration problem is presented which involves the prediction of protein concentration in samples of whole wheat from the spectrophotometric measurements. The results showed that the proposed formulation has a smaller prediction error when compared to the mono-objective formulation and with a lower number of variables. Finally, a study of noise sensitivity obtained by the multi-objective formulation showed a better result when compared to the other classical algorithm for variable selection.

Keywords

Multivariate Calibration, Variable Selection, Evolutionary Algorithms, Multi-Objective Algorithms

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Algoritmos

Lista de Símbolos

Lista de Abreviaturas e Siglas

CAPÍTULO 1

Introdução

Em diversos problemas das áreas médica, biológica, industrial, química entre outras, é de grande interesse estabelecer um modelo matemático que explique a relação entre variáveis independentes e dependentes [45][41].

A quimiometria é uma área que se refere à aplicação de métodos estatísticos e matemáticos, tais como regressão, a dados de origem química. O uso de computadores para analisar dados químicos cresceu nos últimos vinte anos, em parte devido aos recentes avanços em hardware. Por outro lado, a aquisição de dados principalmente na área de química analítica, atingiu um ponto bastante sofisticado com o interfaceamento de instrumentos aos computadores produzindo uma enorme quantidade de informação, muitas vezes complexa e variada. De posse de tal quantidade de dados, a necessidade de ferramentas novas e mais sofisticadas para tratá-los e extrair informações relevantes cresceu muito rapidamente.

Para expressar a relação entre as variáveis pode-se estabelecer um modelo matemático que infira a relação entre as variáveis dependentes e independentes do problema. Variáveis dependentes é o conjunto de valores de referência obtido em laboratório que servirá como parâmetro para a calibração do modelo sendo representado como um vetor, denominado vetor de variáveis dependentes ou vetor de parâmetro de referência [5]. Nesse contexto, a Regressão Linear Múltipla (RLM) é uma técnica estatística que pode ser utilizada para construir esses modelos que descrevem de maneira razoável as relações entre várias variáveis explicativas com variáveis independentes [41].

Quanto maior a resolução do hardware espectrofotométrico utilizado para obtenção das variáveis independentes maior é a quantidade de variáveis geradas por amostra. Por estarem fisicamente relacionadas entre si tais variáveis geralmente apresentam um fenômeno em que duas ou mais variáveis apresentam informação similar, consequentemente reduzindo a confiabilidade da estimativa dos coeficientes do modelo de regressão. Para contornar tal problema é necessário fazer uso de alguma técnica de seleção ou descarte de variáveis, determinando assim um subconjunto de variáveis independentes que melhor explique a variável resposta [41].

Em Soares et. al [52], os autores afirmam que modelos PLS (*Partial Least*

Square) podem não permitir uma interpretação física direta em virtude da regressão ser realizada no domínio dos dados transformados, sendo que a MLR (*Multiple Linear Regression*) a opera no domínio original obtendo-se modelos mais simples e fáceis de interpretar.

A literatura sobre esse problema (veja referências [2, 42, 59, 53]) indica que o algoritmo evolutivo possui duas limitações no contexto do problema analisado: (i) seleciona um número de variáveis maior do que modelos clássicos como o algoritmo das projeções sucessivas (APS) e (ii) os modelos gerados possuem uma maior sensibilidade à presença de ruídos instrumentais. Recentemente, Lucena [12], com o objetivo de reduzir o problema (i), propôs uma formulação multi-objetivo utilizando o algoritmo NSGA-II que minimiza o erro de predição e o número de variáveis selecionadas de forma simultânea. Os resultados demonstraram que a formulação com dois objetivos conseguiu minimizar simultaneamente os objetivos analisados a níveis melhores do que os algoritmos tradicionais. Entretanto, o problema de sensibilidade à ruído não foi considerado.

A formulação original do algoritmo NSGA-II utilizada por Lucena et. al [12] foi proposta por Deb et. al [16], em que não é possível inferir se um determinado indivíduo é melhor do que o outro em todos os objetivos considerados. Apesar de apresentar uma técnica eficiente em termos de otimização multi-objetivo com algoritmos genéticos, o autor afirma que tal algoritmo pode não ser eficaz para problemas em que três ou mais objetivos são considerados.

Em Galvão Filho[19] foi proposto a análise multi-objetivo para seleção de variáveis envolvendo a capacidade de predição e a sensibilidade à ruído. Verificou-se que a associação de tal objetivo resultou em modelos mais robustos para determinadas configurações das amostras para construção do modelo. Porém os resultados obtidos com os objetivos se restringiram a análise do modelo já construído. Uma abordagem interessante seria incorporar tais objetivos na seleção de variáveis.

Em Sipoli[48] e Santos[17], uma nova proposta de algoritmo evolutivo que seja capaz de trabalhar com três ou mais objetivos é citada, com o uso de armazenamento em tabelas e que se obteve resultados satisfatórios para sistemas de distribuição de energia elétrica.

Neste cenário este trabalho propõe um algoritmo evolutivo multi-objetivo baseado em uma nova forma de seleção de indivíduos que faz uso de seleção de subpopulações em estrutura de dados denominada tabelas. Espera-se que o algoritmo seja capaz de gerenciar mais de dois objetivos,sendo eles: (1) o erro de predição, (2) o número de variáveis, (3) a sensibilidade a ruído e (4) função de agregação para ponderação das anteriores; de forma a encontrar soluções satisfatórias para o problema de seleção de variáveis. Tal proposta tem por objetivo preservar os melhores indivíduos para cada objetivo considerado.

Como estudo de caso, o algoritmos proposto foi aplicado ao processo de seleção de variáveis em um conjunto de dados do trigo obtidos através da espectroscopia no infravermelho próximo (NIR) por reflectância difusa na faixa de 1100 a 2500 nm com resolução de 2 nm [32]. Utiliza-se como base de comparação mono-objetivo, os algoritmos de seleção de variáveis APS-MLR, MONO-GA-MLR e PLS-MLR. E como base de comparação entre algoritmos multi-objetivos, o NSGA-II-MLR.

Os resultados mostraram que o algoritmo proposto foi capaz de obter uma solução com menor sensibilidade ao ruído instrumental e capaz de melhorar o erro de predição em aproximadamente 65% quando comparado aos algoritmos clássicos mono-objetivos. Em comparação ao algoritmo multi-objetivo, foi capaz de obter uma solução com valor similar de erro de predição, porém com menor sensibilidade ao ruído.

1.1 Organização da Dissertação

O Capítulo 2 traz uma revisão sobre a calibração multivariada, o problema da multicolinearidade e seleção de variáveis. O Capítulo 3 mostra os conceitos sobre algoritmos evolutivos e as principais abordagens de algoritmos evolutivos. O Capítulo 4 mostra a fundamentação teórica sobre os algoritmos multi-objetivos NSGA-II, SPEA-II e VEGA. O Capítulo 5 apresenta o algoritmo proposto neste trabalho e os experimentos realizados. O Capítulo 6 apresenta os resultados obtidos com o uso do algoritmo proposto e faz um comparativo com os resultados obtidos com os algoritmos MONO-GA-MLR, PLS-MLR, APS-MLR e NSGA-II-MLR. O Capítulo 7 apresenta as considerações finais e aponta as sugestões de trabalhos futuros.

CAPÍTULO 2

Análise Multivariada

Este Capítulo descreve os principais detalhes relacionados à análise multivariada. Na seção 2.1 detalha alguns conceitos relacionados à calibração multivariada, assim como a capacidade preditiva de um modelo de regressão. Na seção 2.2 mostra o problema da colinearidade do conjunto de dados da amostra e a necessidade da seleção de variáveis. Nas seguintes seções, é mostrado alguns algoritmos clássicos para seleção de variáveis, tais como Regressão em Mínimos Quadrados Parciais (*Partial Least Square*) (PLS), Algoritmo de Projeções Sucessivas (APS) e o Algoritmos Evolutivos (AE).

2.1 Calibração Multivariada

Análise química quantitativa é a ciência da determinação da concentração de um ou mais substancias presentes em uma amostra. O estado da arte dessa aplicação inclui o uso de técnicas espectrofotométricas que medem a interação entre o objeto em análise e a energia radiada [51]. Essa interação é baseada pela Lei de Lambert-Beer [50], ilustrada na Figura 2.1, na qual mostra uma amostra recebendo uma radiação P_0 , e atravessando com uma energia menor P. A energia absorvida da amostra pode ser medida com um espectrofotômetro e relacionado com a concentração de propriedade [25].

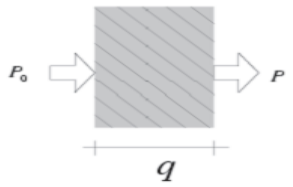

Figura 2.1: *Processo de espectroscopia de absorção.*

Portanto, a intensidade de absorbância é numericamente dada por

$$x(\lambda) = \log \frac{P_0(\lambda)}{P(\lambda)} \tag{2-1}$$

onde $P_0(\lambda)$ é a radiação emitida pelo equipamento e $P(\lambda)$ é a radiação emitida pela amostra no comprimento de onda λ.

Para obter a concentração da amostra, é necessário irradiar diferentes comprimentos de onda simultaneamente. Neste cenário, é normal comprimentos de onda se sobreporem e, consequentemente, dois ou mais sinais de enviarem a mesma informação. Em termos algébricos as ondas sobrepostas significam alta correlação entre as variáveis e podem induzir a problemas matemáticos no processo de regressão [43].

Seja uma amostra incluindo duas absorbância (A and B) com com sobreposição espectral $\lambda(1)$ e $\lambda(2)$, é possível obter y_A and y_B tal que

$$\begin{aligned} x(\lambda_1) &= k_A(\lambda_1)y_A + k_B(\lambda_1)y_B \\ x(\lambda_2) &= k_A(\lambda_2)y_A + k_B(\lambda_2)y_B \end{aligned} \tag{2-2}$$

$$\begin{gathered} \begin{bmatrix} x(\lambda_1) \\ x(\lambda_2) \end{bmatrix} = \begin{bmatrix} k_A(\lambda_1) & k_B(\lambda_1) \\ k_A(\lambda_2) & k_B(\lambda_2) \end{bmatrix} \begin{bmatrix} y_A \\ y_B \end{bmatrix} \\ \begin{bmatrix} y_A \\ y_B \end{bmatrix} = \begin{bmatrix} k_A(\lambda_1) & k_B(\lambda_1) \\ k_A(\lambda_2) & k_B(\lambda_2) \end{bmatrix}^{-1} \begin{bmatrix} x(\lambda_1) \\ x(\lambda_2) \end{bmatrix} \\ y_A = b_A(\lambda_1)(\lambda_1) + b_A(\lambda_2)(\lambda_2) \\ y_B = b_B(\lambda_1)(\lambda_1) + b_B(\lambda_2)(\lambda_2) \end{gathered} \tag{2-3}$$

Em termos gerais, o modelo multivariado é dado por

$$y = x_0 b_0 + x_1 b_1 + \ldots + x_{J-1} b_{J-1} + \varepsilon \tag{2-4}$$

ou em notação vetorial,

$$Y = X\beta + \varepsilon \tag{2-5}$$

com $x = [x_0\ x_1 \ldots x_{J-1}]$ é o vetor de valores medidos, $\beta = [b_0\ b_1 \ldots b_{J-1}]^T$ é o vetor a ser determinado e ε faz parte de erro aleatório.

Nesse caso de i amostras estarem disponíveis com n comprimento de onda, podemos organizar em pares $(x_i, y_i) \in \mathbf{R}^J \times \mathbf{R}$ tal que

$$Y = \begin{bmatrix} y_1^a \\ y_2^a \\ \vdots \\ y_i^a \end{bmatrix} X = \begin{bmatrix} x_1^1(\lambda_1) & \ldots & x_1^j(\lambda_n) \\ x_2^1(\lambda_1) & \ldots & x_2^j(\lambda_n) \\ \vdots & \ddots & \vdots \\ x_i^1(\lambda_1) & \ldots & x_i^j(\lambda_n) \end{bmatrix}, \tag{2-6}$$

onde $x_i^j(\lambda_n)$ é a i-ésima amostra no comprimento de onda λ_n e y_i^a é a concentração de a (energia absorvida) na i-ésima amostra e pode-se obter um modelo matemático que relacione as matrizes por meio de um vetor de coeficientes β tal como mostrado na Equação (2-7).

$$\beta = (X_{cal}^T X_{cal})^{-1} X_{cal}^T Y_{cal} \tag{2-7}$$

As matrizes X e Y são separadas em X_{cal} e Y_{cal} para obter a matriz de coeficientes β, X_{teste} e Y_{teste} são usados para testar a precisão do modelo de predição. A variável resposta estimada é definida pela combinação linear entre a matriz de dados X_{teste} e os coeficientes de regressão estimado β [46], logo $\hat{Y}$ pode ser estimado tal que

$$\hat{Y} = X_{teste}\beta \tag{2-8}$$

Como mostra a Equação (2-9), a capacidade preditiva de um modelo de regressão linear múltipla é calculada pela raiz do erro quadrático médio de predição (*Root Mean Squared Error of Prediction* - RMSEP):

$$\text{RMSEP} = \sqrt{\frac{\sum_{i=0}^{N}(y_i - \hat{y}_i)^2}{N}}, \tag{2-9}$$

em que y é o i-ésimo valor da propriedade de interesse, N é o número de observações, e $\hat{\mathbf{y}}$ é o valor estimado.

2.2 O problema da multicolinearidade e Seleção de Variáveis

A colinearidade das variáveis de um conjunto de dados ocorre quando qualquer combinação linear das variáveis preditoras resulta em um valor igual a 0. Isso quer dizer, do ponto de vista da análise de dados, que mais de uma coluna representam o mesmo dado ou a mesma informação.

O efeito colateral do ponto de vista da operação com matrizes é que a operação de multiplicação de uma matriz transposta por ela mesma não possui matriz inversa. A combinação linear em regressão linear não é exatamente 0, mas pode chegar próximo desse valor. Quando isso acontece, a inversa torna-se numericamente instável [7] [11]. Isso implica que a variância dos coeficientes de regressão é muito alta. Assim, aumenta a dificuldade para encontrar um coeficiente de regressão significativo para ser utilizado na Regressão Linear Múltipla.

Outro problema com a RLM é a necessidade do número de amostras exceder o número de variáveis. A RLM consiste na resolução de um sistema de equações lineares simultâneas, portanto, para que o sistema não se torne indeterminado o número de equações do conjunto de dados deve ser superior ao número de variáveis. Assim, seleciona-se as variáveis mais informativas e não redundantes para o modelo de calibração.

A seleção de um conjunto reduzido de variáveis, que influenciam positivamente no modelo, é importante para melhorar a eficiência dos algoritmos utilizados para a construção de modelos de RLM. Ainda, a identificação de um pequeno conjunto de variáveis explicativas é, normalmente, desejada em problemas de regressão [24].

O problema da determinação de uma equação apropriada associada a um subconjunto de variáveis independentes depende do critério usado para: *i*) analisar as variáveis; *ii*) selecionar o subconjunto; e *iii*) estimar os coeficientes na Equação (2-7). De acordo com Miller [40], as razões para utilizar somente algumas das variáveis disponíveis incluem:

- As estimativas de baixo custo ou previsões podem ser alcançadas por meio da redução do número de variáveis;
- A precisão pode ser aprimorada eliminando variáveis não informativas;
- Um conjunto de dados multivariados pode ser parsimoniosamente descrito.

Existem algoritmos matemáticos como Regressão em Mínimos Quadrados (PLS), Algoritmo das Projeções Sucessivas (APS) e Algoritmos Evolutivos que são utilizados para Seleção de Variáveis nos problemas de Regressão Linear Múltipla e capazes de encontrar soluções próximas do ótimo para esses dois problemas citados. Nas seções seguintes, são apresentados esses algoritmos.

2.3 Regressão em Mínimos Quadrados Parciais

Mínimos Quadrados Parciais (PLS) é um método para a construção de modelos preditivos, quando existe um grande número de variáveis independentes colineares. Note-se que a ênfase está em predizer as respostas e não necessariamente em tentar entender a relação subjacente entre as variáveis [57][1].

A base fundamental do PLS é o PCA. O PCA consiste na manipulação da matriz de dados de tal forma a representar as variações presentes em muitas variáveis através de um número menor de novas variáveis denominadas fatores [31][30]. Os fatores representam o novo sistema de eixos também chamados de componentes principais, variáveis latentes ou autovetores, para representar as amostras. A principal diferença do PLS em relação ao PCA é que ao invés de se utilizar apenas a variância das variáveis originais, o PLS utiliza o vetor $\mathbf{Y}$ para rotacionar os eixos de componentes principais. Tal rotação permite obter uma maior capacidade preditiva [1].O primeiro fator (variável latente) descreve a direção de máxima variância, também correlacionada com a concentração. Estas variáveis latentes são combinações lineares dos componentes principais calculados pelo método PCA [18][23].

O modelo geral do PLS é dado por:

$$X = TP^{\top} + E \tag{2-10}$$

$$Y = UQ^{\top} + F \tag{2-11}$$

em que X é uma matriz preditora e Y é uma matriz resposta, T e U são matrizes projetadas de X e Y, respectivamente. E P e Q são as matrizes dos pesos das matrizes X e de Y e as matrizes E e F são matrizes de erro residuais. As decomposições de X e de Y são executadas de forma a maximizar a covariância de T e U. Há vários algoritmos para calcular os componentes principais utilizados no algoritmo PLS dos quais vale citar os mais comuns NIPALS e SVD [23][18].

Para se construir as variáveis latentes, os dados são transformados para um novo domínio e que como efeito, na prática, é necessário que usar as variáveis originais do conjunto de entrada, uma vez que não é possível identificar quais variáveis originais foram utilizadas para se construir as variáveis latentes e isso pode ser visto como uma desvantagem, pois o PLS não permite uma interpretação direta nos resultados, já que essa técnica realiza a regressão no domínio dos dados transformados.

2.4 Algoritmo de Projeções Sucessivas (APS)

O algoritmo das projeções sucessivas é uma técnica de seleção de variáveis para minimizar problemas de colinearidade em regressão linear múltipla [3][22].

APS é uma técnica do tipo *forward* (*passo a frente*) que dado uma variável inicial, a cada iteração, inseri-se uma nova variável até que um número máximo n de seja atingido. O objetivo principal do APS é selecionar as variáveis que contenham o mínimo de redundância possível minimizando o problema de colinearidade [3]. APS é composto por três fases principais:

1. A primeira consiste em operações de projeção realizadas na matriz **X** de respostas instrumentais. Estas projeções são usadas para gerar cadeias de variáveis com cada vez mais elementos. Cada elemento de uma cadeia é selecionado de modo a obter a menor colinearidade com a anterior.
2. Na fase seguinte, os subconjuntos de variáveis candidatas são avaliados de acordo com o desempenho preditivo RMSEP no modelo MLR.
3. A última fase consiste no procedimento de eliminação de variáveis, em que por meio de um teste estatístico verifica-se se a eliminação de uma dada variável não compromete significativamente o erro RMSEP. Tal procedimento visa melhorar a simplicidade do modelo.

O APS não modifica os vetores de dados originais pois as projeções são utilizadas apenas para propósitos de seleção. Portanto, a relação entre os vetores de dados e

variáveis espectrais é preservada [3].Os últimos resultados da literatura sobre calibração multivariada mostrou que o APS-MLR tem os melhores resultados em termos de RMSEP e parcimônia, quando comparado com o algoritmo genético clássico e PLS [54][55].

2.5 Algoritmos Evolutivos

Algoritmos Evolutivos (AE) é uma área de pesquisa da ciência da computação que se baseia no processo da evolução natural. Princípios como a mutação, recombinação genética, seleção natural e a sobrevivência do mais forte; são inspirações vindas da biologia que auxiliam na construção de algoritmos capazes de resolver problemas insolúveis para a computação. Um Algoritmo Evolutivo é um processo iterativo estocástico para gerar tentativas de solução para certo problema.

A aplicação dos AEs em química foi inicialmente utilizada por Lucasius e Kateman [39] na seleção de comprimentos de onda em análises de sistemas multicomponentes. A partir disso, vários estudos tem utilizado essa abordagem na seleção de dados espectrofotométricos. Em Costa Filho [20] e em Lucena [12], mostra-se que o uso de AEs foi eficaz para calibração multivariada em problemas de quimiometria, possibilitando encontrar soluções que fossem próximas da solução ótima.

Em um AE, a principal estrutura de dados é um cromossomo, que é uma cadeia de bits ou caracteres que representa uma possível solução para o problema e apresenta-se como na Figura 2.2,

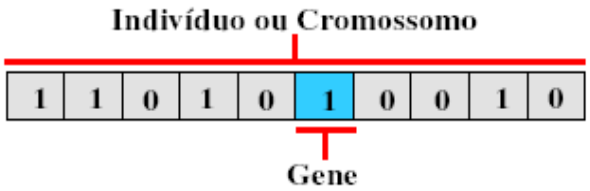

Figura 2.2: *Estrutura de um cromossomo binário genérico de um AE.*

Cada coluna dessa estrutura representa a escolha de uma variável do conjunto original que será utilizada no modelo de regressão. A avaliação das variáveis escolhidas pelo cromossomo é feita através de uma função objetivo, que é o erro de predição dado pela Equação (2-9).

A vantagem de se utilizar tal estratégia para problemas de calibração multivariada está em poder trabalhar com várias possíveis soluções simultaneamente, uma vez que várias estruturas como citado anteriormente são geradas, formando o que é chamado de população. Os cromossomos gerados dentro dessa população serão avaliados e, ou podem ser descartados, caso não sejam pertinentes ao conjunto de soluções existentes, ou pode-

rão ser selecionados para gerarem novos cromossomos, através de operadores genéticos, ou poderão permanecer na população até uma nova geração.

CAPÍTULO 3

Algoritmos Evolutivos

Algoritmos Evolutivos (AE) é um conjunto de algoritmos inspirados no processo da evolução natural e de reprodução biológica, tais como a mutação, recombinação genética, seleção natural e a sobrevivência do mais forte [37].

O AE manipula um conjunto de indivíduos P (população), cada um dos quais compreende um ou mais cromossomos. Esses cromossomos permitem que cada indivíduo representa uma solução potencial para o problema em consideração. Um processo de codificação / decodificação é responsável pelo mapeamento da solução em um cromossomo e vice-versa. Cromossomos são divididos em unidades menores chamadas genes. Os valores diferentes que um determinado gene pode assumir são chamado de alelos.

Os operadores genéticos, de um modo geral, consistem em aproximações computacionais de fenômenos vistos na natureza, como reprodução sexuada, a mutação genética entre outros. Os AEs são dependentes de fatores estocásticos, ou seja, probabilísticos, tanto na fase de inicialização da população quanto na fase de evolução. Isto faz com os resultados raramente sejam perfeitamente reprodutíveis [37].

A idéia básica de funcionamento desses algoritmos é a de tratar as possíveis soluções do problema como "indivíduos"de uma "população", que irá "evoluir"a cada iteração ou "geração". Para isso é necessário construir um modelo de evolução onde os indivíduos sejam soluções de um problema.

3.1 Componentes de um Algoritmo Evolutivo

Os componentes básicos de um AE são:

- Um problema para ser resolvido pelo algoritmo;
- Um método para codificar e decodificar soluções do problema através de cromossomos;
- Uma função de avaliação que mede quão bem, cada solução é capaz de resolver o problema;
- Um método para selecionar indivíduos.

- Um método para criar a população inicial de cromossomos;
- Um conjunto de parâmetros para o algoritmo;
- Um conjunto de operadores que atuam no processo de reprodução;

A partir disso pode-se apresentar uma breve discussão a respeito de cada um desses componentes.

3.1.1 Representação das Soluções

A representação das possíveis soluções do espaço de busca de um problema define a estrutura do cromossomo a ser manipulado pelo algoritmo e depende do tipo de problema e do que se deseja manipular. Os principais tipos de representação são [14][37]:

- Binária: quando numérico ou inteiro;
- Números Reais: quando numérico;
- Permutação de Símbolos: quando baseado em ordem;
- Símbolos repetidos: quando por grupamento.

3.1.2 Codificação e Decodificação das Soluções

A codificação e decodificação do cromossomo consiste basicamente na construção da solução do problema a partir do cromossomo. Em outras palavras, seria o método do algoritmo de traduzir o cromossomo para um formato mais simples de se entender tanto pelo usuário quanto para o próprio algoritmo [37].

O processo de codificação constrói o cromossomo para seja avaliado pelo algoritmo e o processo de decodificação simplifica o cromossomo em um valor que possa ser entendido mais facilmente. Aqui, a representação binária consegue ter vantagem em relação às demais, pois é fácil a transformação do valor binária em inteiro ou real [37].

3.1.3 Método ou Função de Avaliação

A avaliação é o elo entro o AE e o mundo externo, pois através dela é que o problema pode ser representado e tem por objetivo fornecer uma medida de aptidão de cada indivíduo na população corrente, que irá guiar o processo de busca. Fazendo uma analogia com o sistema natural, pode-se dizer que a função de avaliação é como o meio ambiente, através dele que podemos nos adaptar e a natureza selecionar os mais aptos para aquele determinado ambiente. Por causa disso, podemos dizer que cada problema significa um ambiente diferente e assim cada problema tem sua própria função de avaliação [37].

3.1.4 Técnicas de Reprodução

Existem algumas formas ainda de reprodução após a seleção dos pais. Essas técnicas determinam como os indivíduos serão substituídos pela nova geração. Em Deb [14], são citadas os seguintes métodos:

- Troca de Toda a População: a cada ciclo os novos indivíduos irão substituir toda a geração corrente. Dessa forma, se temos 12 indivíduos, 6 serão selecionados que gerarão novos 12 indivíduos.
- Troca de População com Elitismo: a cada ciclo todos os indivíduos da população corrente serão substituídos com exceção do mais apto.
- Troca Parcial da População: a cada ciclo é gerado uma quantidade de indivíduos que irá substituir apenas os piores da população corrente.
- Troca Parcial da População Sem Duplicados: assim como o anterior, mas sem permitir indivíduos duplicados.

3.1.5 Inicialização da População

A inicialização da população determina o processo de criação dos indivíduos para o primeiro ciclo do algoritmo. Tipicamente, a população inicial é formada a partir de indivíduos aleatoriamente criados. Se já se tem certo conhecimento do problema em questão, é possível inserir indivíduos já bons dentro da população inicial para uma evolução mais rápida [37].

3.1.6 Operadores Genéticos

Os operadores genéticos transformam a população através de sucessivas gerações, estendendo a busca até chegar a um resultado satisfatório. Sendo estes necessários para que a população se diversifique e mantenha características de adaptação adquiridas durante as etapas de processo anteriores [28][37].

Destacam-se nesta categoria dois principais agentes:

- Seleção;
- Mutação;
- Cruzamento.

Seleção

O processo de seleção dentro de AEs tem como objetivo de selecionar quais são os indivíduos dentro da população já avaliada para a reprodução. Essa seleção tem base

na aptidão dos indivíduos de acordo com a função de avaliação: indivíduos mais aptos tem maior probabilidade de serem escolhidos para a reprodução [28][37].

Em algoritmos genéticos os principais métodos de seleção são:

- Roleta: seleciona os indivíduos aleatoriamente, proporcionando maiores chances de reprodução aos indivíduos mais aptos da população;
- Torneio: consiste em selecionar uma série de indivíduos da população de fazer com que eles entrem em competição pelo direito de ser pai, usando sua avaliação;
- Amostragem Universal Estocástica: todos os indivíduos são mapeados em segmentos contíguos de uma linha, sendo que o tamanho de cada segmento é proporcional ao valor da avaliação do indivíduo que está sendo mapeado [37].

Mutação

Os operadores de mutação são necessários para a introdução e manutenção da diversidade genética da população, alterando arbitrariamente um ou mais componentes de uma estrutura escolhida, fornecendo assim, meios para a introdução de novos elementos na população. Dessa forma, a mutação assegura que a probabilidade de se chegar a qualquer ponto do espaço de busca nunca será zero [14].

O operador de mutação é aplicado aos indivíduos com uma probabilidade dada pela taxa de mutação. O exemplo mais simples e clássico de operação de mutação é o *bit flip* ou Inversão de Bit, que consiste basicamente em escolher um gene aleatoriamente dentro do cromossomo binário e inverte o valor no gene escolhido [14].

Cruzamento ou Crossover

O cruzamento é o operador responsável pela recombinação de características dos pais durante a reprodução, permitindo que as próximas gerações herdem essas características. Ele é considerado o operador genético predominante, por isso é aplicado com probabilidade dada pela taxa de crossover, que deve ser maior que a taxa de mutação [14].

As formas mais comuns de crossover existentes são:

- Crossover em Único Ponto: é definido um único ponto de cruzamento dentro dos cromossomos dos pais, a sequência do início até o primeiro ponto é copiada do primeiro pai, o resto é copiado do segundo pai.
- Crossover de Dois Pontos: são definidos dois pontos, do início até o primeiro ponto é copiada do primeiro pai, a partir do primeiro ponto até o segundo ponto, é copiado do segundo pai e o restante copiado do primeiro pai novamente.
- Crossover Uniforme: os genes são copiados aleatoriamente dos dois pais.

- Crossover Aritmético: é realizada uma operação aritmética entre os genes dos dois pais para a nova geração.

Algoritmo 3.1: Pseudocódigo de um AE genérico.

1. $g \leftarrow 0$; `// inicializa o contador de gerações`
2. $P(g) \leftarrow Populacao\_Inicial()$; `// gera aleatoriamente uma população inicial P(g)`
3. $Avalia(P(g))$; `// avalia os indivíduos da população inicial segundo uma função de adequação`
4. **Enquanto** $g \leq gmax$ **faça** `// teste do critério de parada (gmax, por exemplo)`
5. $Pi \leftarrow Selecione\_Aleatoriamente(P(g))$; `// aleatoriamente selecione uma sub-população de (Pi) para gerar descendentes.`
6. $P' \leftarrow Aplica\_Operadores(Pi)$; `// alteração desses indivíduos através dos Operadores para gerar nova população.`
7. $Avalie(P')$; `// avaliação dos novos indivíduos de P' segundo a função adequação.`
8. $P(g+1) \leftarrow Sobreviventes(P(g), P')$; `// selecione sobreviventes entre P(g)e P'`
9. $g \leftarrow g+1$; `// incrementa o contador de gerações`
10. **Fim Enquanto**
11. **retorne** $min(P(gmax))$;

CAPÍTULO 4

Algoritmos Evolutivos Multi-Objetivos

Problemas de otimização multi-objetivo têm despertado grande interesse na área de otimização. Nesses problemas, a qualidade da solução é definida com base na sua adequação em relação a diversos objetivos possivelmente conflitantes. Uma função objetivo f_1 é conflitante com uma outra função f_2 quando não é possível melhorar o valor de f_1 sem piorar o valor da função f_2. Em Deb [14], apresenta-se o enunciado geral de uma Otimização Multi-Objetivo (OMO):

$$\left.\begin{array}{lll} \textit{minimizar/maximizar} & f_m(x), & m = 1,2,...,N_{obj} \\ \text{restrita a} & g_j(x) \geq 0, & j = 1,2,...,NR_{des}; \\ & h_k(x) = 0, & k = 1,2,...,NR_{igu}; \\ & x_i^{(inf)} \leq x_i \leq x_i^{sup}, & i = 1,2,...,N_{var}, \end{array}\right\} \tag{4-1}$$

onde x é um vetor de tamanho N_{var} variáveis de decisão $x = (x_1, x_2, ..., x_{N_{var}})^T$ também denominado de solução. Os valores $x_i^{(inf)}$ e $x_i^{(sup)}$ representam os limites inferior e superior, respectivamente, para a variável x_i. Esses limites definem o espaço de variáveis de decisão ou espaço de decisão S_{dec} . As NR_{des} desigualdades (g_j) e as NR_{igu} igualdades (h_k) são chamadas de funções de restrição. Uma solução factível satisfaz as $NR_{igu} + NR_{des}$ funções de restrição e os $2N_{var}$ limites. Caso contrário, a solução não será factível. O conjunto de todas as soluções factíveis formam espaço de busca S_{fact}. Cada função $f_m(x)$ pode ser maximizada ou minimizada.

Porém, para trabalhar com os algoritmos de otimização, é necessário converter todas as funções para serem apenas maximização ou minimização. O vetor funções objetivo $f(x) = [f_1(x), f_2(x), ..., f_{N_{obj}}(x)]$ compõe um espaço multidimensional chamado espaço de objetivos S_j. Para cada solução no espaço de decisão, existe um $f(x)$ em S_{obj}. Esta é uma diferença fundamental em relação à otimização de objetivos simples, cujo espaço de objetivos é unidimensional. O mapeamento ocorre então entre um vetor x (de dimensão N_{var}) e um vetor $f(x)$ (de dimensão N_{obj}). Por exemplo, se cada elemento de x e $f(x)$ são números reais, então $f(x)$ estaria mapeada como $f(x) : \mathbb{R}^{N_{var}} \rightarrow \mathbb{R}^{N_{obj}}$.

Em problemas de OMO, emprega-se o conceito de dominância de Pareto para

comparar duas soluções factíveis do problema. O conceito de dominância de Pareto pode ser empregado da seguinte forma. Dadas duas soluções x e y, diz-se que x domina y se as seguintes condições forem satisfeitas:

- A solução x é não pior que y em todos os objetivos;
- A solução x é melhor que y em pelo menos um objetivo.

Dessa forma, podemos dizer que existe um conjunto de alternativas ótimas que são não dominadas entre si nos objetivos. Nesse caso, não existe somente uma solução para o problema, mas sim um conjunto de soluções ótimas, denominado conjunto de Pareto ótimo ou fronteira de Pareto. Observando a Figura 4.1 pode-se observar que uma solução x_1 domina a solução x_2, se x_1 é pelo menos igual a x_2, em todos os objetivos ou se x_1 é superior x_2 em pelo menos um objetivo.

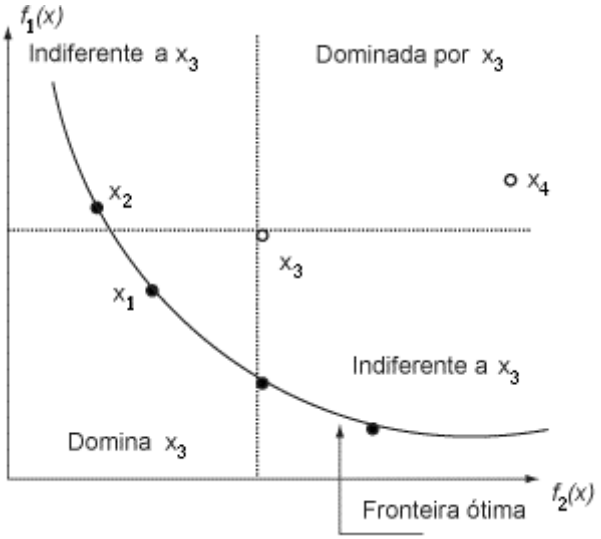

Figura 4.1: *Ilustração de um exemplo de fronteira de Pareto.*

A busca de soluções no espaço multi-objetivo pode ser computacionalmente dispendioso e muitas vezes é inviável, pois a complexidade de um problema pode impedir que métodos exatos consigam alcançar soluções satisfatórias em tempo hábil. Por este motivo, estratégias de buscas estocásticas como algoritmos evolutivos, busca tabu, *simulated annealing* e otimização por colônia de formigas podem ser desenvolvidos para encontrar o conjunto de Pareto. Apesar de não garantir as melhores soluções, podem obter, na maioria dos casos, encontrar uma boa aproximação, ou seja, um conjunto de soluções cujos objetivos são próximos das soluções ótimas [48].

À grosso modo, um algoritmo de busca geral estocástico consiste em três partes:

- Uma memória de trabalho que contém os candidatos à solução atualmente consideradas;
- Um módulo de seleção;

- Um módulo variação.

A principal diferença entre os AEs tradicionais, como o Algoritmo Genético no Capítulo 2, e os AEMOs é o operador de seleção, dado que a comparação entre duas soluções deve realizar-se de acordo com o conceito de dominância. Em algumas propostas, como o SPEA-II, o valor de aptidão é proporcional à dominância da solução [48].

Os modelos de AEMO são usualmente classificados em dois grupos:

- Não elitistas: compreendem os algoritmos que como o próprio nome indica, não utiliza nenhuma forma de elitismo nas suas interações;
- Elitistas: compreendem os modelos que empregam alguma forma de elitismo. Por exemplo, como o SPEA-II, utiliza uma população externa para armazenar as soluções não dominadas encontradas até o momento. O NSGA-II combina a população atual com a população gerada e preserva as melhores soluções de ambas.

A tabela 4.1 apresenta os principais modelos de AEMO e seus autores.

Tabela 4.1: *Diferentes Modelos de AEMO.*

Sigla	Nome do Modelo	Autores
VEGA	Vector Evaluated Genetic Algorithm	Schaffer [49]
WBGA	Weight Based Genetic Algorithm	Hajela [27]
MOGA	Multiple Objective Genetic Algorithm	Fonseca [21]
NSGA	Non-Dominated Sorting Genetic Algorithm	Deb [56]
NPGA	Niched-Pareto Genetic Algorithm	Horn [29]
PPES	Predator-Prey Evolution Strategy	Laumanns [36]
REMOEA	Rudoph's Elitist Multi-Objective Evolutionay Algorithm	Rudolph [47]
NSGA-II	Elitist Non-Dominated Sorting Genetic Algorithm	Deb [15]
SPEA, SPEA-2	Strenght Pareto Evolutionary Algorithm 1 e 2	Zitzler [63], Zitzler [62]
TGA	Thermodynamical Genetic Algorithm	Kita [34]
PAES	Pareto-Archived Evolutionary Strategy	Knowles [35]
MONGA-I, MONGA-II	Multi-Objective Messy Genetic Algorithm	Velduizen [61]
Micro-GA	Multi-Objective Micro-Genetic Algorithm	Coelho [8]
PESA-I, PESA-II	Pareto Envelope-Base Selection Algorithm	Corne [10], Corne [9]

Nas seções a seguir, é feita uma breve descrição à respeito dos algoritmos que foram utilizados para a inspiração desse trabalho.

4.1 Algoritmo Genético de Avaliação de Vetores (VEGA)

Em inglês, *Vector Evaluated Genetic Algorithm* (VEGA) é o mais simples Algoritmo Genético Multi-objetivo, e surge como uma extensão natural de um AG mono-objetivo. Schaffer [49] propõe a divisão aleatória da população do AG em k subpopulações de dimensão idêntica em cada geração, sendo M o número de funções objetivo. As soluções de cada uma das subpopulações é atribuído um valor de aptidão, de acordo com o respectivo objetivo. Desta forma, cada uma das funções serve para avaliar apenas alguns indivíduos da população. As soluções da população são posteriormente selecionadas por um mecanismo de seleção proporcional à aptidão [4][38][64].

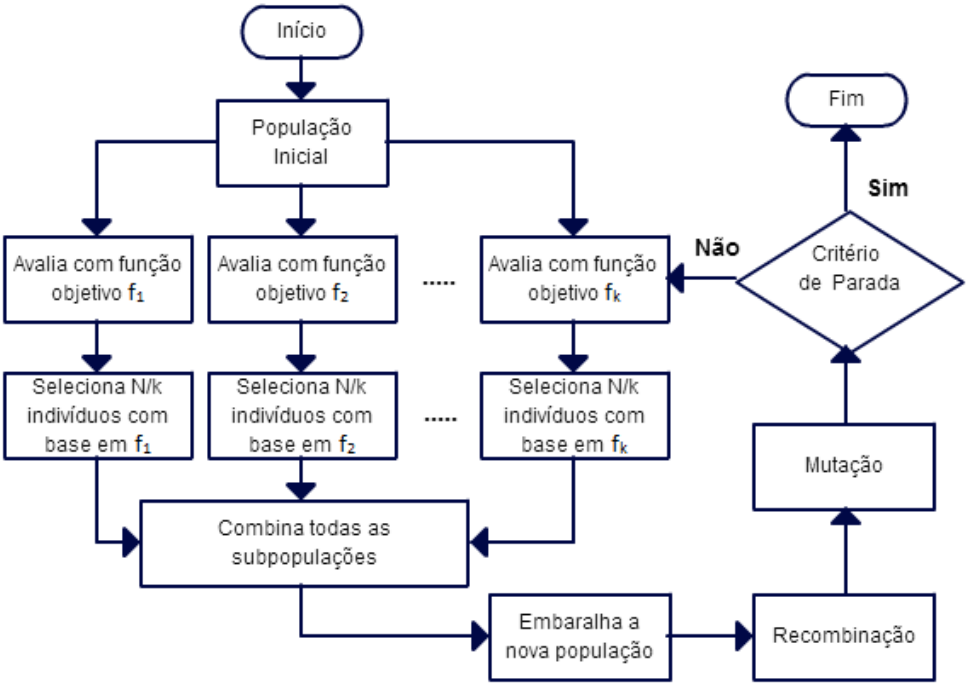

Figura 4.2: *Fluxograma geral do VEGA.*

4.2 Algoritmo Genético de Classificação por Não Dominância II (NSGA-II)

Em Deb [56], é apresentado o NSGA, uma versão anterior ao NSGA-II, mas que possui limitações, tais quais: 1) alta complexidade e custo computacional; 2) não faz uso de elitismo. Já o NSGA-II (*Non-dominated Sorting Genetic Algorithm II*) é um algoritmo multi-objetivo, baseado em Algoritmos Genéticos e que implementa o conceito de Dominância, ou seja, classifica a População Total em fronteiras de acordo com o grau de dominância. Segundo o NSGA II, os indivíduos que estão localizados na primeira

fronteira são considerados as melhores soluções daquela geração, enquanto que na última fronteira encontram-se as piores. Usando esse conceito, pode-se encontrar resultados mais consistentes (pontos mais próximos da região de Pareto) e que se adaptam melhor ao tipo do problema [48][13].

A Figura 4.3 mostra uma fluxograma básico da execução de um NSGA-II.

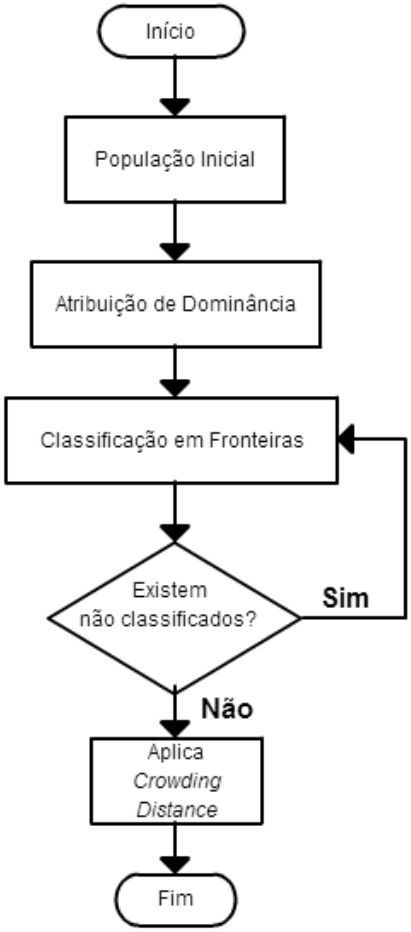

Figura 4.3: *Fluxograma geral do NSGA-II.*

A principal diferença do NSGA-II e um AE simples é a forma como o operador de seleção é aplicado, sendo este operador subdividido em dois processos: Classificação Rápida por Não Dominância (*Fast Non-Dominated Sorting*) e o Distância de Agrupamento (*Crowding Distance*), sendo o primeiro atuando na determinação do grau de dominância e consequentemente na classificação dos indivíduos: indivíduos não-dominados formam novas fronteiras a cada iteração do algoritmo, formando várias fronteiras até todos da população serem classificados; e o segundo, baseada na comparação de aglomerado, para ordenar as soluções dentro de uma mesma fronteira. Os demais operadores são aplicados de maneira tradicional [13].

Para melhor compreender a abordagem do Distância de Agrupamento, é necessário definir a métrica para estimação de densidade e o operador de comparação.

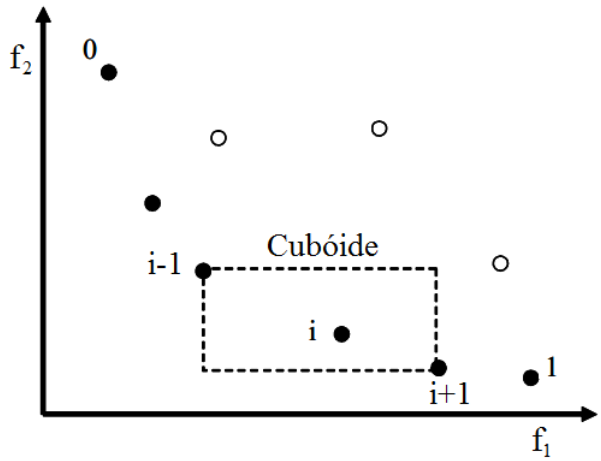

Figura 4.4: *Cálculo crowding-distance. Pontos marcados em círculos preenchidos são soluções da fronteira não dominada.*

A estimativa da densidade de soluções em torno de uma solução particular da população, é obtida através do cálculo da distância média de dois pontos de cada lado deste ponto ao longo de cada um dos objetivos. O valor i serve como uma estimativa do perímetro do cuboide formado usando os vizinhos mais próximos como os vértices, conforme Figura 4.4 [13].

O operador de comparação ($\prec_n$) tem o objetivo de orientar o processo de seleção nas várias fases do algoritmo em direção a uma fronteira Pareto-ótima uniformemente espalhada. Supondo que cada indivíduo na população tem dois atributos:

1. *Rank* de não dominância (i_{rank});
2. Distância de Agrupamento (*Crowding Distance*) ($i_{distance}$).

Uma ordem parcial $\prec_n$ é definido por:

$$\begin{gathered} i \prec_n \text{se } (i_{rank} < j_{rank}) \\ ou \\ ((i_{rank} = j_{rank}) \text{ e } (i_{distance} > j_{distance})) \end{gathered} \tag{4-2}$$

Para duas soluções entre diferentes fronteiras não-dominadas, este modelo dá preferência à escolha da solução com menor *rank*, caso contrário, é escolhida a solução localizada em uma região de menor aglomeração [13].

A Figura 4.5 exemplifica o processo descrito no NSGA-II.

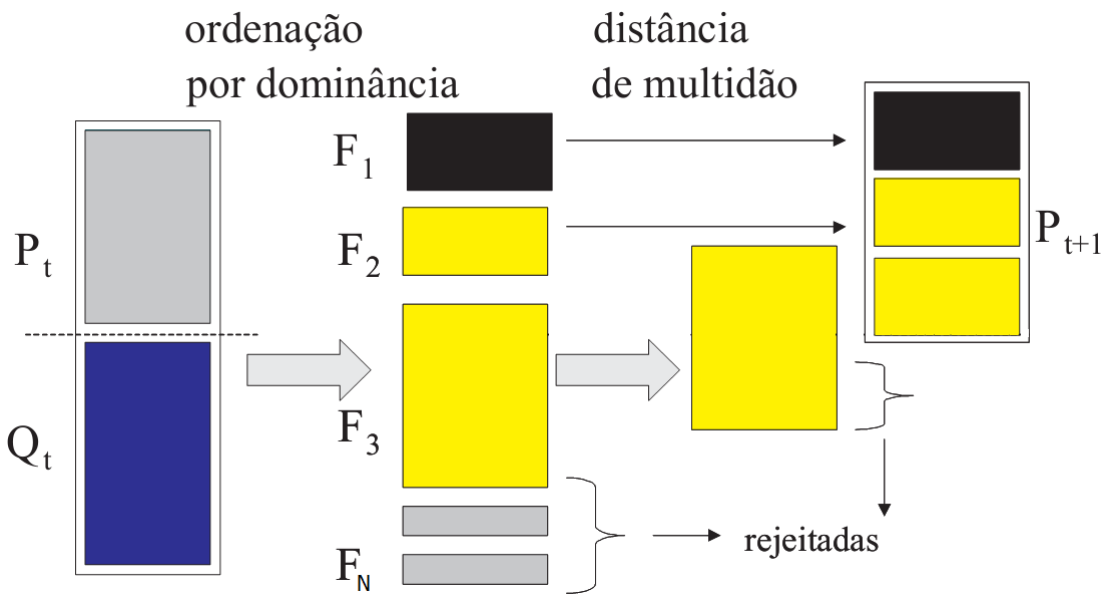

Figura 4.5: *Procedimento NSGA-II.*

As duas populações P_t e Q_t são classificadas e ao término, são separadas em fronteiras de dominância. Os indivíduos pertencentes a primeira fronteira F_1 são não-dominados e dominam os indivíduos pertencentes as demais fronteiras $F_2, F_3, \cdots, F_n$. F_2 domina as demais fronteiras $F_3, F_4, \cdots, F_n$ e assim por diante. Na etapa seguinte, são selecionados indivíduos para compor a próxima geração a partir dos que compõem a primeira fronteira, até completar a nova população. Se uma fronteira não puder ser totalmente inserida na população o algoritmo usa como critério o processo de *crowding distance* ou distância de agrupamento para escolher quais indivíduos estarão na nova população [13].

4.3 Algoritmo Evolutivo da Força de Pareto II (SPEA-II)

Proposto por Zitzler [65] o algoritmo SPEA-II é uma abordagem evolutiva multi-objetivo que também inclui o conceito de elitismo e assim como o NSGA-II, utiliza duas populações P e Q. Sendo P a população que armazena os indivíduos da população inicial, assim como os indivíduos das próximas gerações e a população Q, denotada como população externa, armazena apenas as soluções não-dominadas encontradas pelo algoritmo. É fornecido como parâmetro o tamanho da população Q, denotado como N_{next}. As populações P e Q, em cada iteração $t = 1, 2, \cdots, N_{iter}$, são denotadas com P_t e Q_t, respectivamente [65]. É possível se observar um fluxograma geral do SPEA-II na Figura 4.6.

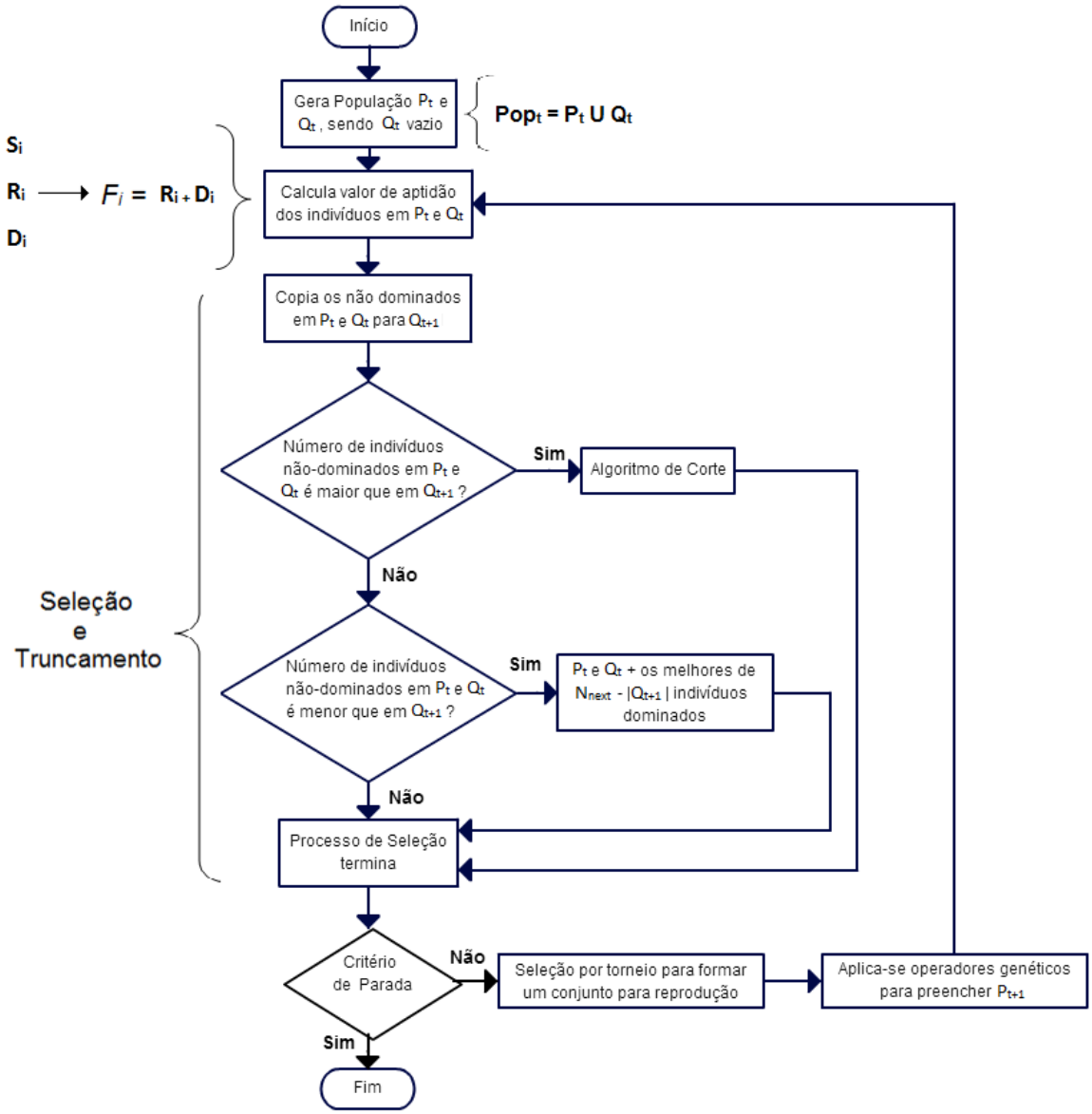

Figura 4.6: *Fluxograma geral do SPEA-II.*

Inicia-se com a criação de uma população inicial aleatória P_t e uma população externa Q_t vazia, em que $t = 0$. A cada iteração t, é calculada a função de aptidão para cada solução i em P_t e Q_t e $Pop_t = P_t \cup Q_t$. No cálculo da função de aptidão, são usados os conceitos de dominância e de densidade, definida na Equação (4-5). O objetivo é minimizar o valor da função de aptidão, definida na Equação (4-6). Quanto menor o valor da função de aptidão de um indivíduo, melhor é a adaptação do indivíduo [65]. A força do indivíduo, denotada por S_i na Equação (4-3), é dada pelo número de soluções que ele domina:

$$S_i = |\{j, j \in Pop_t \wedge i \succ j\}|. \tag{4-3}$$

O número de soluções em Pt dominadas pela solução i é representado pelo valor

N_i. Logo, as soluções que não dominam nenhuma outra possui o valor de S_i igual a zero. O valor de aptidão bruto do indivíduo, denotado por R_i, também é calculado somando as forças de todos os indivíduos que o dominam, conforme apresentada pela Equação (4-4) [65]. O valor de aptidão bruto R_i das soluções não dominadas é igual a zero enquanto as soluções dominadas tem o valor R_i alto.

$$R_i = \sum_{j \in Pop_t, j \succ i} S_j \tag{4-4}$$

A densidade do indivíduo é uma função decrescente em relação ao k-ésimo vizinho mais próximo. Para os casos onde existem muitas soluções não dominadas, o valor S_i aproxima-se de zero para todas as soluções. Assim, é necessário haver um mecanismo para privilegiar soluções dentre as não dominadas chamado fator de densidade, denotado por D_i, mostrado na Equação (4-5) [65].

$$D_i = \frac{1}{dist_{ij}^k + 2} \tag{4-5}$$

Para cada indivíduo i, as distâncias (no espaço dos objetivos) entre i e todos os indivíduos $j \in Pop_t$ são calculadas e armazenadas em uma lista. Logo após a ordenação da lista em ordem crescente, o k-ésimo elemento representa o termo $dist_{ij}^k$. É sugerido para k o valor $k = \sqrt{|Pop_t|}$. Enfim, a aptidão final para cada solução i em Pop_t, denotada F_i, é dada pela Equação (4-6).

$$F_i = R_i + D_i \tag{4-6}$$

O método de seleção do SPEA-II primeiramente copia todos os indivíduos não dominados de P_t e de Q_t para a população externa da próxima geração $\overline{P_{t+1}}$. Neste cenário, existem três situações possíveis:

1. O número de indivíduos no conjunto não dominado é exatamente o mesmo da população externa ($|Q_{t+1}| = N_{ext}$);
2. O número de indivíduos no conjunto não dominado é menor que o tamanho da população externa ($|Q_{t+1}| < N_{ext}$);
3. O número de indivíduos no conjunto não dominado é menor que o tamanho da população externa ($|Q_{t+1}| > N_{ext}$);

No primeiro caso, o processo de seleção está completo. No segundo caso, são copiados para a nova população externa, os melhores $N_{ext} - |Q_{t+1}|$ indivíduos dominados, incluindo a população regular e a população externa na geração anterior. No terceiro caso, utiliza-se um algoritmo de corte. O objetivo do algoritmo de corte do SPEA-II é restringir o tamanho de Q_{t+1} a N_{ext} soluções. Em cada iteração, é removida a solução cuja distância

para seu vizinho mais próximo seja menor dentre as distâncias existentes. Em caso de empate, calcula-se a segunda menor distância e assim sucessivamente [65]. A solução i é escolhida para ser removida se $i \leq_d j$ para todo $j \in Q_{t+1}$ satisfazendo as seguintes condições:

$$\begin{aligned} i \leq_d j \Leftrightarrow \quad & \forall \quad 0 < k < |Q_{t+1}| : \sigma_i^k = \sigma_j^k \ \vee \\ & \exists \quad 0 < k < |Q_{t+1}| : \left[(\forall\, 0 < l < k : \sigma_i^l = \sigma_j^l) \wedge \sigma_i^k = \sigma_j^k \right] \end{aligned} \tag{4-7}$$

A Figura 4.7 apresenta à esquerda, um conjunto de soluções pertencentes à população externa Q_{t+1}. À direita, após a aplicação do algoritmo de corte, algumas soluções são eliminadas. Além disso, o algoritmo de corte garante que as soluções extremas para cada objetivo sejam mantidas.

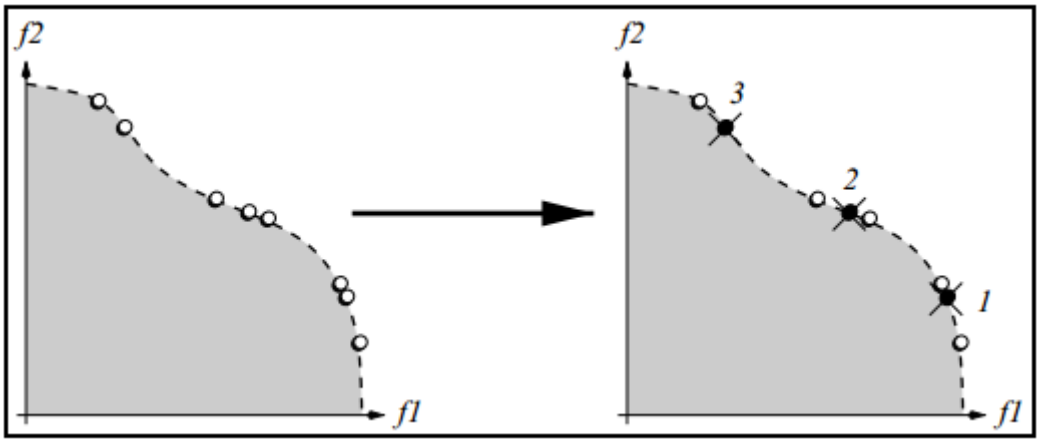

Figura 4.7: *Algoritmo de Corte do algoritmo SPEA-II (O símbolo • representa as soluções que foram eliminadas) [65].*

CAPÍTULO 5

Metodologia Proposta e Experimentos

Este Capítulo apresenta o AE proposto, chamado de AE Multi-objetivo em Tabelas (AEMT). Este algoritmo trabalha com várias subpopulações armazenadas em tabelas, onde os melhores indivíduos para cada característica do problema são armazenados em sua respectiva subpopulação. Por essa razão, o AE é denominado de AE Multi-Objetivo em Tabelas [17][48].

Mesmo que muitos algoritmos evolutivos sigam diferentes linhas de raciocínio, quase todos são desenvolvidos com base em uma única população de indivíduos, que interage de alguma forma para gerar novos indivíduos dentro da mesma população. Entretanto, modelos baseados em "ilhas"e algoritmos em células, quando extendidos à incluir conceitos de subpopulações, podem ser vantajosos [58]. Esses algoritmos são chamados de modelos estruturados, mas o uso de modelos estruturados em otimização multi-objetivo tem sido pouco explorado [44]. Em Vargas [60], mostra-se que uma simples dinâmica de subpopulações pode melhorar consideravelmente o desempenho da busca de soluções quando usado em algoritmos evolutivos clássicos. As soluções apresentadas pelas diferentes dinâmicas na presença de múltiplas subpopulações mostraram que os algoritmos se comportam se forma menos sensível à ótimos locais, expandido o campo de busca no espaço de soluções.

Na Figura 5.1 é possível observar a estrutura geral do algoritmo proposto. Cada subpopulação armazena indivíduos de modo a minimizar individualmente um objetivo. Neste trabalho serão considerados: (1) erro de predição; (2) número de variáveis; (3) sensibilidade à ruído instrumental. Uma 4ª subpopulação é criada para armazenar indivíduos avaliados por uma função de agregação, também conhecido como método da soma ponderada [14][8]. Neste trabalho utilizou-se somente de uma única função de agregação. É possível utilizar tabelas de dominância e força, presentes nos algoritmos NSGA-II e SPEA-II, mas tal idéia não foi explorada no trabalho.

O indivíduo selecionado para a reprodução pode ser proveniente de qualquer subpopulação da tabela e de qualquer tabela. Essa estratégia de seleção aumenta a diversidade entre os indivíduos que reproduzem de forma que as características de um indivíduo de uma subpopulação possam migrar para as demais subpopulações da tabela.

Em consequência, aumenta-se a possibilidade do algoritmo escapar de ótimos locais, aproximando-se de soluções com avaliações próximas de um ótimo global na fronteira de Pareto-ótima [17][48].

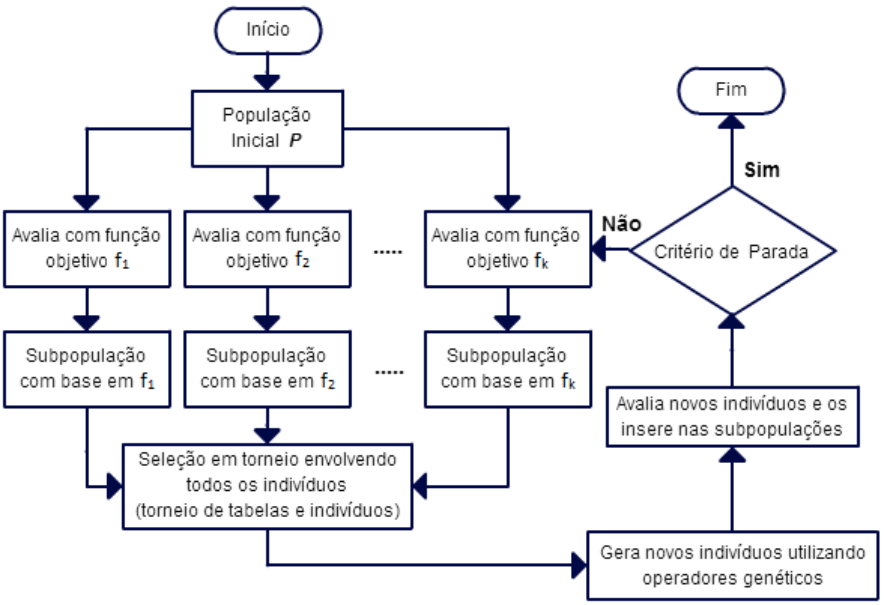

Figura 5.1: *Fluxograma do algoritmo proposto.*

Soluções geradas pelo AEMT podem ser armazenadas ou descartadas, dependendo do grau de adaptação do indivíduo a cada objetivo do problema. No processo de seleção de sobreviventes, um novo indivíduo é acrescentado a uma subpopulação se sua adequação ao objetivo for melhor que pelo menos um indivíduo da mesma. O mesmo indivíduo pode ser incluído em mais de uma tabela de acordo com esse critério de seleção. Como a população é estacionária, os novos indivíduos substituem os piores. Nesse caso, a adequação do indivíduo é um vetor de seus valores relativos a cada objetivo ou restriçoes [17][48].

A principal diferença entre o AEMT e o VEGA está no processo de seleção. No VEGA as subpopulações são concatenadas e depois misturadas para se produzir uma nova população e assim se aplicar os operadores genéticos com todos os indivíduos da nova população gerada. Já no AEMT, é realizado o torneio entre as tabelas e selecionado apenas dois indivíduos por vez no processo de seleção, o que diminui o custo computacional e previne a perda de informações relevantes já existentes no espaço de busca.

O pseudocódigo do AEMT é apresentado no Algoritmo 5.1. Inicia-se uma população inicial P com uma quantidade de indivíduos Q_i, $vfit$ é um vetor com todos os valores de aptidão das funções objetivo consideradas. Cada indivíduo em P é avaliado e inserido em sua respectiva subpopulação *SubPops*, sendo essa uma tabela que integra todas as subpopulações das funções de aptidão. Após avaliação, é feita a seleção por

meio de torneio, que está descrita no Algoritmo 5.2. Depois da seleção, são aplicados os operadores genéticos sobre os dois indivíduos selecionados e os novos indivíduos gerados são avaliados e inseridos nas respectivas subpopulações, caso sejam válidos para tal. Esse processo se repete *gmax* gerações. A quantidade de novos indivíduos por geração é definido pelo usuário, assim como o tamanho de cada subpopulação e o número de gerações.

Algoritmo 5.1: Pseudocódigo do AE Multi-objetivo em Tabelas.

1. **Seja P a População,**
2. **SubPops, a Tabela de Objetivos,**
3. **vfit, o vetor com valores de aptidão,**
4. **Qi, a quantidade de indivíduos,**
5. **Qfg, a quantidade de filhos gerados**
6. **gmax, a quantidade máxima de gerações,**
7. $P = gera\_populacao\_inicial(Qi)$
8. **Para** $i = 1$ **até** Qi **faça**
9. $\quad individuo = P(i)$
10. $\quad vfit = avalia(individuo)$
11. $\quad inserir\_subpop(SubPops, vfit, individuo)$
12. **Fim para**
13. **Para** $g = 0$ **até** $gmax$ **faça**
14. $\quad$ **Para** $tam = 1$ **até** $size(SubPops)$ **faça**
15. $\quad\quad pai, mae = torneio(SubPops)$
16. $\quad\quad$ **Se** $rand > pc$ **faça**
17. $\quad\quad\quad novos\_filhos = cruzamento(pai, mae);$
18. $\quad\quad$ **Fim se**
19. $\quad$ **Fim para**
20. $\quad$ **Se** $rand > pm$ **faça**
21. $\quad\quad$ **Para** $i = 1$ **até** Qfg **faça**
22. $\quad\quad\quad mutacao(novos\_filhos(i));$
23. $\quad\quad$ **Fim para**
24. $\quad$ **Fim se**
25. **Fim para**
26. **Para** $j = 1$ **até** Qfg **faça**
27. $\quad individuo = novos\_filhos(j)$
28. $\quad vfit = avalia(individuo)$
29. $\quad inserir\_subpop(SubPops, vfit, individuo)$
30. **Fim para**

No Algoritmo 5.2, a seleção de indivíduos é feita em duas etapas: 1ª) Torneio de tabelas e 2ª) Torneio de indivíduos. O torneio de tabelas consiste em selecionar duas tabelas, ou seja, duas subpopulações aleatoriamente, sendo que cada tabela possuem um identificador que determina quantas vezes a tabela foi sorteada para a seleção. Quanto maior o número desse identificador, maior a influência na geração de novos indivíduos [6]. Uma vez selecionadas as tabelas, passa-se para a segunda etapa de torneio de indivíduos, em que das tabelas selecionadas, dois indivíduos de cada tabela são selecionados e prevalece o que tiver melhor valor da função de aptidão de acordo com a tabela selecionada. Com isso, são selecionados dois indivíduos para a aplicação dos operadores genéticos.

Algoritmo 5.2: Pseudocódigo do Torneio de Tabelas e de Indivíduos.

1. **Seja SubPops o conjunto de tabelas**
2. $rand > tabela1$;
3. $rand > individuo1$ **e** $rand > individuo2$;
4. $Pop \leftarrow SubPops(tabela1)$;
5. **Se** $Pop(individuo1) < Pop(individuo2)$ **faça**
6. $\quad pai \leftarrow Pop(individuo1)$;
7. **senão** $pai \leftarrow Pop(individuo2)$;
8. **Fim se**
9. $rand > tabela2$;
10. $rand > individuo1$ **e** $rand > individuo2$;
11. $Pop \leftarrow SubPops(tabela2)$;
12. **Se** $Pop(individuo1) < Pop(individuo2)$ **faça**
13. $\quad mae \leftarrow Pop(individuo1)$;
14. **senão** $mae \leftarrow Pop(individuo2)$;
15. **Fim se**

5.1 Funções Objetivo Consideradas

A avaliação dos resultados foi obtida utilizando o RMSEP, o número de variáveis e a norma-2 dos coeficientes de regressão. O erro quadrático médio obtido com o conjunto de predição (*Root Mean Square Error of Prediction*, RMSEP) foi definido na Equação (2-9). O número de variáveis é dado por um escalar k indicando a quantidade de variáveis utilizadas no modelo. No método de mínimos quadrados não se consideram incertezas sobre os valores das variáveis independentes. Contudo em problemas de calibração multivariada para análises espectrofotométricas, deve-se ter em mente que os valores das variáveis independentes correspondem a medidas instrumentais contaminadas por

ruído. Desse modo convém ter uma medida da sensibilidade das predições do modelo com respeito a tal ruído instrumental [19].

A norma-2, representada por b, é uma maneira de ponderar tal medida. Em Galvão Filho [19], foi demonstrado que a norma dos coeficientes está relacionada com a sensibilidade do modelo à ruído e é descrita por

$$\|b\| = \sqrt{\sum_{i=1}^{k} b_i^2} \tag{5-1}$$

em que k é a quantidade de variáveis utilizadas no modelo.

Conforme a Equação (5-1), a norma-2 calcula a magnitude somada dos coeficientes de regressão. Essa magnitude são os ruídos e quanto mais próxima de zero (seja positivo ou negativo) menos ruidosos serão os coeficientes de regressão.

Para função de agregação, sabendo que $F = \{f_1, f_2, f_3, \cdots, f_k\}$, sendo F o vetor que contém os demais valores das funções citadas acima, é desenvolvida na Equação (5-2):

$$OP = \frac{F_k - \mu(F)}{\sigma(F)} \tag{5-2}$$

o que gera um novo vetor $F' = \{f'_1, f'_2, f'_3, \cdots, f'_k\}$ com valores escalonados, uma vez que os objetivos possuem magnitudes diferentes e é necessário que os valores estejam em escalas próximas, e por seguinte é somado, como na Equação (5-3):

$$\sum_{i=1}^{k} \left|F'_i\right| \tag{5-3}$$

5.2 Materiais e Métodos

Nesta seção estão descritas os dados utilizados, as funções objetivo consideradas, as ferramentas e o ambiente utilizados para desenvolvimento desse trabalho.

5.2.1 Dados do Trigo

Todas as amostras são de trigo de grão inteiro, obtidos a partir de material vegetal de produtores ocidentais canadenses. Os dados de referência foram determinados no Laboratório de Pesquisa de Grãos, em Winnipeg. Os dados de referência são: a concentração de proteína (%); teste de peso (kg/hl); PSI (textura do grão de trigo) (%); absorção de água por farinografia (%), tempo de desenvolvimento de massa por farinografia (em minutos), e índice de tolerância à mistura por farinografia (Unidades Brabender). O conjunto de dados para o estudo de calibração multivariada consiste de 775

espectros VIS-NIR de amostras de todo o grão de trigo, que foram utilizados como dados *shoot-out* em 2008 na Conferência Internacional Reflectância Difusa (http://www.idrc-chambersburg.org/shootout.html).

A concentração de proteína foi escolhida como a propriedade de interesse. Os espectros foram adquiridos na faixa de 400-2500nm, com uma resolução de 2nm. No presente trabalho, a região NIR na faixa de 1100-2500nm foi empregado. A fim de eliminar as características indesejáveis, foram calculadas a primeira derivada dos espectros usando um filtro Savitzky-Golay com um polinômio de 2ª ordem e uma janela de 11-pontos. Mas apenas os dados referentes a concentração de proteína foram usadas nestes testes.

O algoritmo de Kennard-Stone (KS) [33] foi aplicada ao espectro resultante para dividir os dados em conjuntos de validação, calibração e predição de amostras 389, 193 e 193, respectivamente, contendo cada conjunto 690 variáveis independentes. O conjunto de validação foi utilizado para orientar a seleção de variáveis em APS-MLR, MONO-GA-MLR, NSGA-II-MLR e SPEA-II-MLR.

O conjunto de predição foi apenas utilizado na avaliação do desempenho final dos modelos resultantes MLR.

5.2.2 Ferramentas e Ambiente

A configuração computacional deste trabalho consiste em um computador de mesa equipado com processador Intel® Core i7, 16 GB de memória RAM. Matlab 8.0.0.783 (R2012b) foi o software utilizado para implementação desse trabalho.

5.2.3 Algoritmos de Comparação

Para fins de comparação, os algoritmos clássicos utilizados neste trabalho são PLS, APS, Algoritmo Genético Mono-Objetivo e Algoritmo Genético Mono-Objetivo Ponderado, sendo este uma adaptação do Algoritmo Genético Mono-Objetivo para minimizar a Equação (5-3) que minimiza as Equações (**??**), (5-1) e a quantidade de variáveis utilizadas no modelo. A função objetivo do Algoritmo Genético Mono-Objetivo Ponderado é a mesma da tabela de agregação do algoritmo proposto. Para comparação multi-objetivo, apenas o NSGA-II foi utilizado. As configurações dos experimentos aqui utilizados estão descritos na Tabela 5.1 para o PLS; na Tabela 5.2 para os SPA-MLR, MONO-GA-MLR, MONO-GA-POND-MLR e o NSGA-II; e na Tabela 5.3, as configurações para o algoritmo proposto. No estudo de PLS, os conjuntos de calibração e de validação foram unidos num único conjunto de modelagem, o qual foi utilizado no procedimento de validação cruzada *leave-one-out*. O número de variáveis latentes foi selecionado com base no erro de validação cruzada, utilizando o critério de teste F de Haaland e Thomas com

$\alpha = 0,25$, tal como sugerido por Haaland [26]. O conjunto de predição só foi empregado na avaliação final do modelo de PLS.

Tabela 5.1: *Tabela com configurações dos experimentos realizados com o PLS.*

	Configuração PLS
Validação Cruzada	*leave-one-out**
Número de Variáveis Latentes	Teste F com $\alpha = 0,25$

*Os conjuntos de calibração e de validação foram unidos num único conjunto de modelagem.

Tabela 5.2: *Tabela com configurações dos experimentos realizados com os algoritmos clássicos e o algoritmo NSGA-II.*

	Configuração 1	Configuração 2
População Inicial	50	100
Número de Gerações	100	200

Tabela 5.3: *Tabela com configurações dos experimentos realizados com o Algoritmo Proposto.*

	Configuração 1	Configuração 2	Configuração 3	Configuração 4
População Inicial	50	100	50	100
Número de Gerações	100	200	100	200
Tamanho das Subpopulações	10		20	

CAPÍTULO 6

Resultados e Discussões

Neste capítulo são apresentados os resultados obtidos com os algoritmos PLS, APS-MLR, MONO-GA-MLR, MONO-GA-POND-MLR, NSGA-II-MLR e o algoritmo proposto. Os algoritmos foram aplicados ao problema de seleção de variáveis em calibração multivariada descrito no Capítulo 2. A Seção 6.1 apresenta os resultados obtidos pelos algoritmos clássicos de calibração. As seções seguintes detalham os resultados obtidos com o algoritmo proposto, comparando-o com os algoritmos clássicos e com o algoritmo multi-objetivo clássico NSGA-II.

6.1 Resultados dos algoritmos mono-objetivos

Primeiramente, apresenta-se os resultados dos algoritmos clássicos, PLS, APS-MLR, MONO-GA-MLR, MONO-GA-POND-MLR, sendo a minimização do erro de predição o único objetivo considerado. Estes resultados são apresentados na Tabela 6.1.

Tabela 6.1: *Resultados das técnicas tradicionais PLS, APS-MLR, MONO-GA-MLR e MONO-GA-POND-MLR. Os resultados estão expressos em valores de RMSEP.*

	RMSEP	Número de Variáveis
PLS	0.21	15*
APS-MLR	0.20	13
MONO-GA-MLR	0.21	146
MONO-GA-POND-MLR	0.18	95

*Número de variáveis latentes.

Como pode ser observado, os resultados são similares para os quatro algoritmos em relação aos valores de RMSEP. Entretanto, os algoritmos MONO-GA-MLR e MONO-GA-POND-MLR utilizam um número expressivo de variáveis quando comparados ao APS-MLR. Esse resultado pode ser explicado pelo fato de que o MONO-GA-MLR usa somente um objetivo, o RMSEP do conjunto de validação e o MONO-GA-POND-MLR, considera todos os objetivos, porém numa única função objetivo. Na prática, o APS-MLR

é utilizado pois o algoritmo utiliza menos variáveis que o MONO-GA-MLR, MONO-GA-POND-MLR e PLS.

O algoritmo genético clássico é projetado para minimizar a mesma função do APS-MLR, isto é, a Equação RMSEP (2-9). No entanto, a medida que reduz o RMSEP, mais variáveis são incluídas no modelo. Por exemplo, em Lucena [12], usando os mesmos dados da Seção 5.2.1, o AG mono-objetivo foi executado com número diferente de variáveis máximas a serem incluídas. Como pode ser visto na Figura 6.1, o RMSEP e o número de variáveis são objetivos conflitantes.

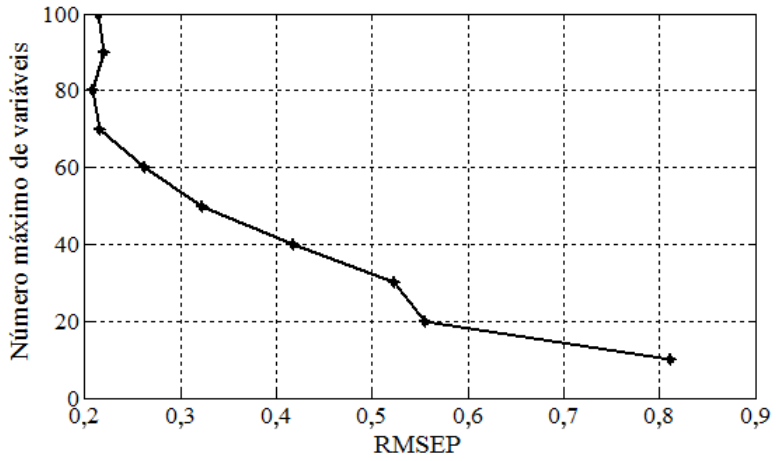

Figura 6.1: *Comportamento do RMSEP com diferentes números máximos de variáveis no algoritmo genético mono-objetivo (Extraído de Lucena [12]).*

Pode-se notar que o RMSEP é reduzido tão logo mais variáveis são incluídas no modelo. Por outro lado se o número de variáveis é grande, a Equação (2-7) pode ter um mau condicionamento em que pequenas perturbações nos dados de entrada produzem grandes variações na variável de saída.

6.2 Resultados obtidos com os algoritmos multi-objetivos

Nesta seção, apresenta-se os resultados obtidos com o NSGA-II e o algoritmo proposto. Como o algoritmo proposto trabalha com várias subpopulações (tabelas), cada uma dessas armazena os melhores indivíduos para cada uma das funções objetivo consideradas. Na Tabela 6.2 apresenta-se o melhor indivíduo de cada subpopulação do algoritmo proposto.

Tabela 6.2: *Valores das funções objetivo do melhor indivíduo de cada tabela do algoritmo proposto.*

	Subpopulação 1: RMSEP	Subpopulação 2: Número de Variáveis	Subpopulação 3: Norma-2	Subpopulação 4: Função de Agregação
RMSEP	**0,043**	RMSEP	1,7	0,07
Número de Variáveis	46	**5**	5	49
Norma-2	99,79	32,56	**3,97**	24,43
Função de Agregação	2,26	2,13	2,22	**2**

Como pode ser observado na Tabela 6.2, em cada coluna tem-se os valores das funções objetivo referente ao melhor indivíduo de cada tabela considerada respectivamente e em negrito o valor da função objetivo referente à tabela do algoritmo. Observe que na diagonal principal, tem-se os menores valores obtidos por cada uma das funções objetivo, o que podemos considerar que são os valores pelos quais as soluções deverão se aproximar. É possível notar também que os valores das outras funções objetivo são distoantes dos outros indivíduos, como por exemplo, no indivíduo da Subpopulação 1, o menor erro de 0,043 foi obtido com 46 variáveis, mas na Subpopulação 2, o número de variáveis foi quase 5 vezes menor, porém o erro de predição é 10 vezes maior que o da Subpopulação 1.

Em Lucena [12], a sensibilidade à ruído não foi considerada no trabalho de Lucena, apenas o erro de predição e o número de variáveis foram consideradas, portanto, era necessário um estudo considerando também a sensibilidade à ruído. Assim, para esse trabalho o algoritmo NSGA-II foi adaptado para três objetivos, considerando a sensibilidade à ruído.

Na Figura 6.2, tem-se a sensibilidade ao ruído de cada indivíduo das subpopulações do algoritmo proposto. Para análise de sensibilidade ao ruído instrumental, cada variável foi contaminada com um ruído branco aleatório em função do desvio padrão da variável com taxa variada entre 1% e 10%. Embora o melhor desempenho (o RMSEP se manter estável à medida que o ruído aumenta) à sensibilidade ao ruído esteja nos cromossomos da Subpopulação "Norma do Vetor de Coeficientes", é possível inferir que as informações genéticas que fazem com que os cromossomos de tal tabela possuam uma menor sensibilidade foram propagadas para as demais tabelas. É possível ainda observar que apesar do indivíduo da Tabela RMSEP ainda apresentar problemas com a sensibilidade à ruído, tem-se que a sensibilidade dos indivíduos das demais tabelas é menor quando comparado à sensibilidade do APS-MLR, mesmo não utilizando o objetivo diretamente,

apesar do erro de predição ser maior.

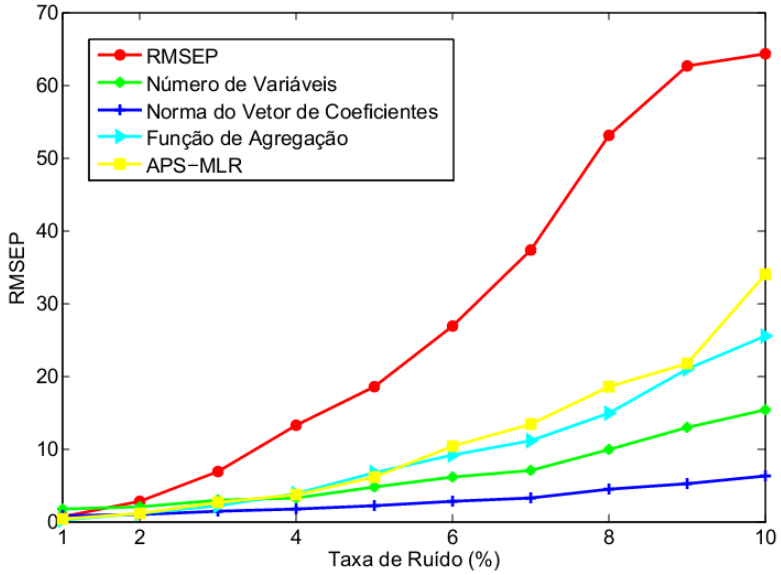

Figura 6.2: *Análise da Taxa de Ruído entre os melhores indivíduos de cada tabela do AEMT-MLR e APS-MLR.*

A subpopulação Função de Agregação armazena os melhores indivíduos que se sobressaíram em todos os objetivos considerados nesse trabalho, portanto, deste ponto em diante, usa-se o melhor indivíduo dessa subpopulação como referência para os resultados apresentados a seguir. Observa-se na Figura 6.3 a comparação da sensibilidade à ruídos entre o algoritmo proposto e os algoritmos clássicos. Nas Figuras 6.3(a) e 6.3(b) pode-se observar que a solução proposta pelo AEMT-MLR obteve valores melhores de robustez se comparado ao algoritmo MONO-GA-MLR e MONO-GA-POND-MLR e na Figura 6.3(c) valores semelhantes em relação APS-MLR. Como pode ser observado na Figura 6.3(d), a solução obtida pelo NSGA-II-MLR possui uma maior sensibilidade à ruído quando comparado à solução do AEMT-MLR.

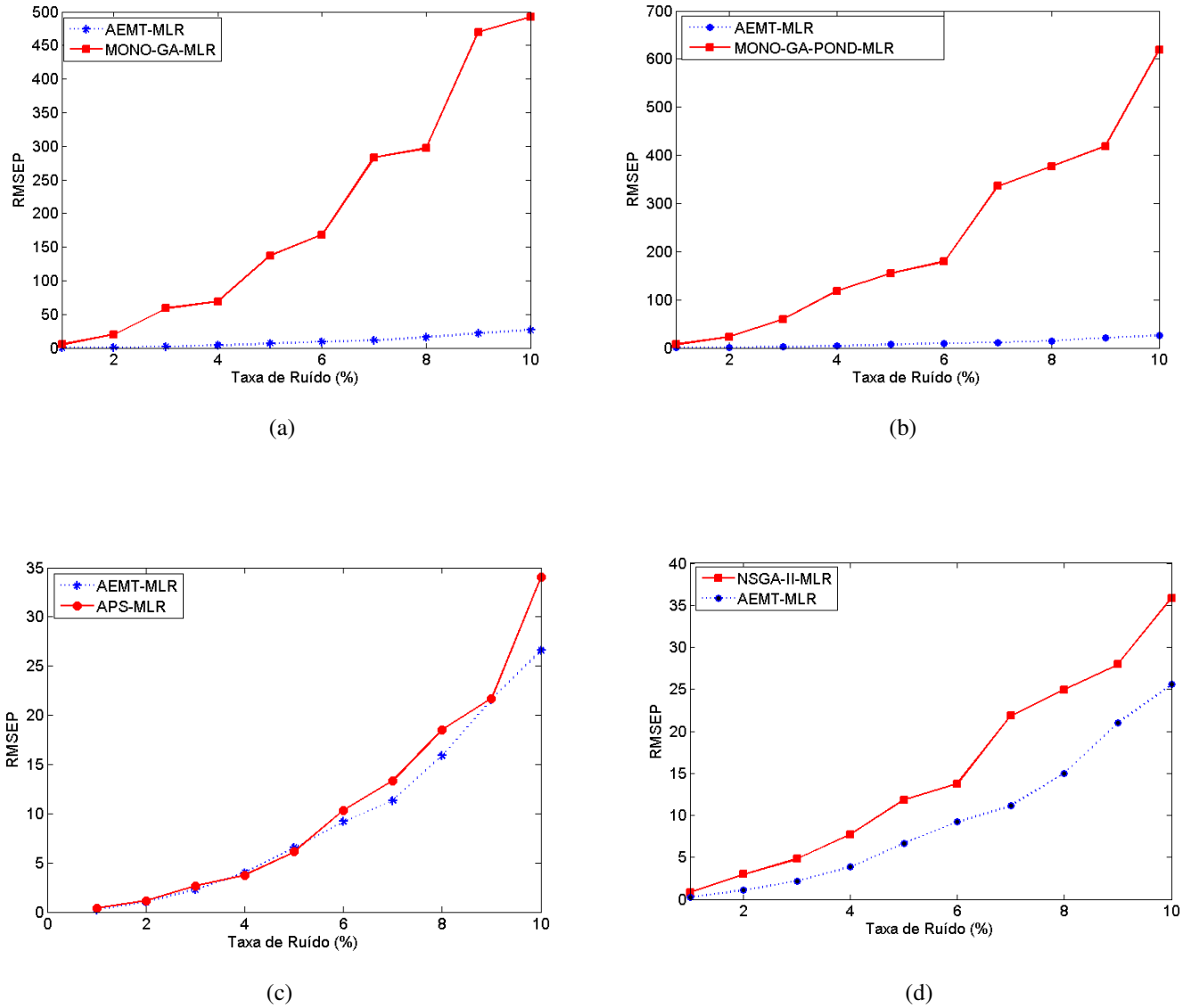

Figura 6.3: *Análise da Taxa de Ruído entre (a) AEMT-MLR e MONO-GA-MLR, (b) MONO-GA-POND-MLR e (c) AEMT-MLR e APS-MLR e (d) AEMT-MLR e NSGA-II-MLR.*

Na Tabela 6.3 apresenta-se o resultado médio dos algoritmos NSGA-II-MLR e AEMT-MLR do conjunto de predição. É possível observar que houve uma pequena melhora no erro de predição e que, apesar da quantidade de variáveis da solução do AEMT-MLR ser maior do que a do NSGA-II-MLR, foi possível demonstrar pelo resultado da Figura 6.3 que o problema da sensibilidade à ruído apresentado por Lucena [12] nas soluções do NSGA-II foi contornado.

Tabela 6.3: *Resultado médio das soluções dos algoritmos NSGA-II-MLR e AEMT-MLR no conjunto de predição.*

	NSGA-II-MLR		AEMT-MLR	
	RMSEP	N^o Variáveis	RMSEP	N^o Variáveis
Média	0,09	25	0,07	45
Menor	0,06 (32*)	15 (0,15**)	0,05 (50*)	33 (0,22**)
Maior	0,21 (21*)	73 (0,04**)	0,09 (41*)	50 (0,05**)

*número de variáveis selecionadas, **RMSEP.

Na Figura 6.4, tem-se as variáveis selecionadas por cada algoritmo. Observa-se que o AEMT-MLR seleciona uma quantidade de variáveis significantemente menor que a quantidade do MONO-GA-MLR e MONO-GA-POND-MLR, porém maior que o NSGA-II, no entanto os algoritmos possuem variáveis em regiões semelhantes do espectro.

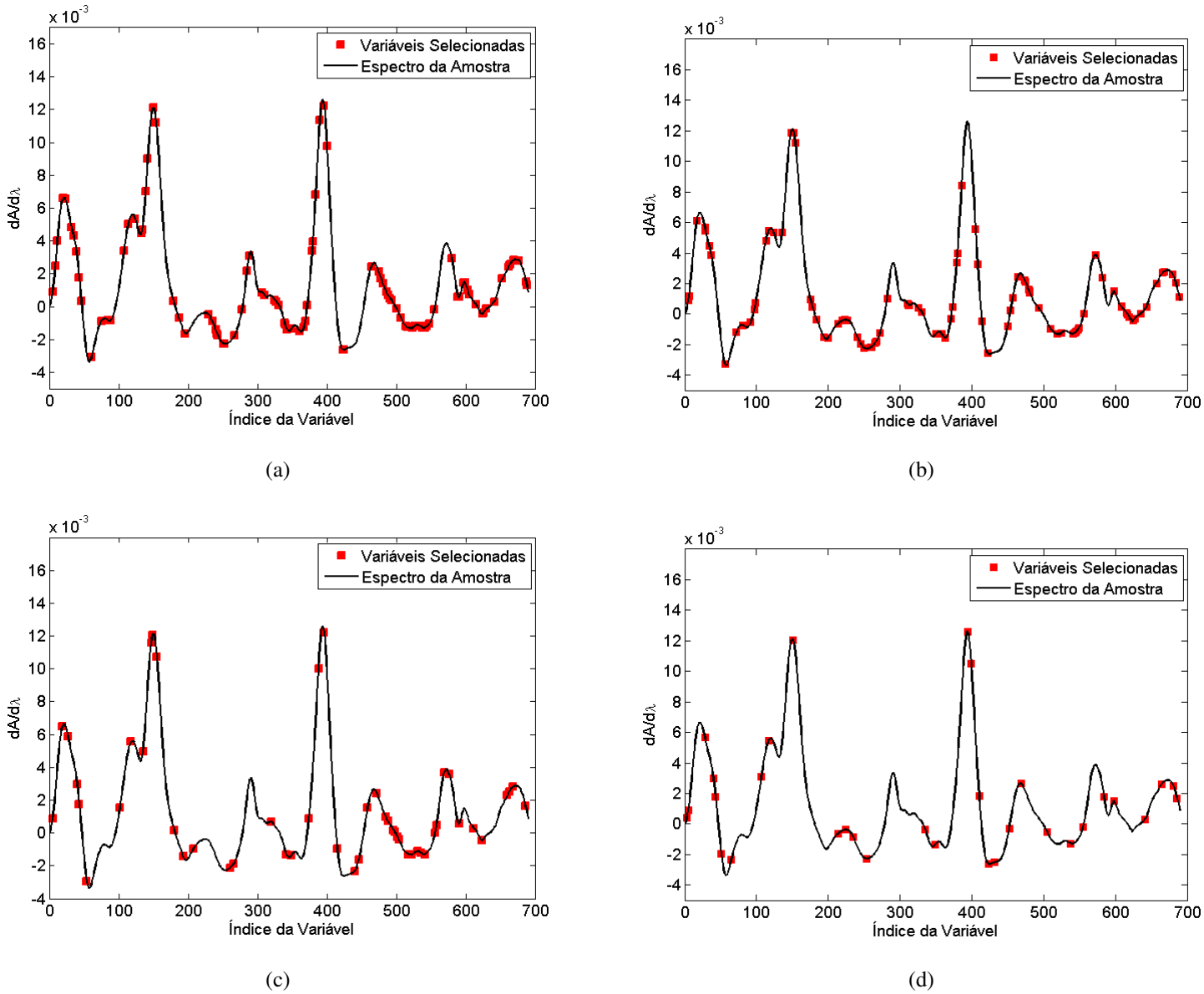

Figura 6.4: *Espectro da Amostra e Variáveis Selecionadas pelo (a) MONO-GA-MLR, (b) MONO-GA-POND-MLR , (c) AEMT-MLR e (d) NSGA-II-MLR.*

A Figura 6.5(a) mostra as soluções da população final do AEMT-MLR no espaço de busca dos três objetivos considerados. Nas Figuras 6.5(b), 6.5(c) e 6.5(d) apresenta-se as soluções considerando os objetivos dois a dois. Observa-se que quando são consideradas os objetivos de número de variáveis e RMSEP, as soluções da população final formam uma fronteira distinta. No entanto, quando o objetivo da norma-2 é utilizado comparando os objetivos dois a dois, as soluções não formaram uma fronteira distinta. Dessa forma, é necessária a utilização dos três objetivos para obter uma população totalmente não-dominada, uma vez que a norma 2 não consegue essa distinção e a concentração dos indivíduos em uma mesma região pode não ser vantajosa, deixando

a população com pouca diversidade.

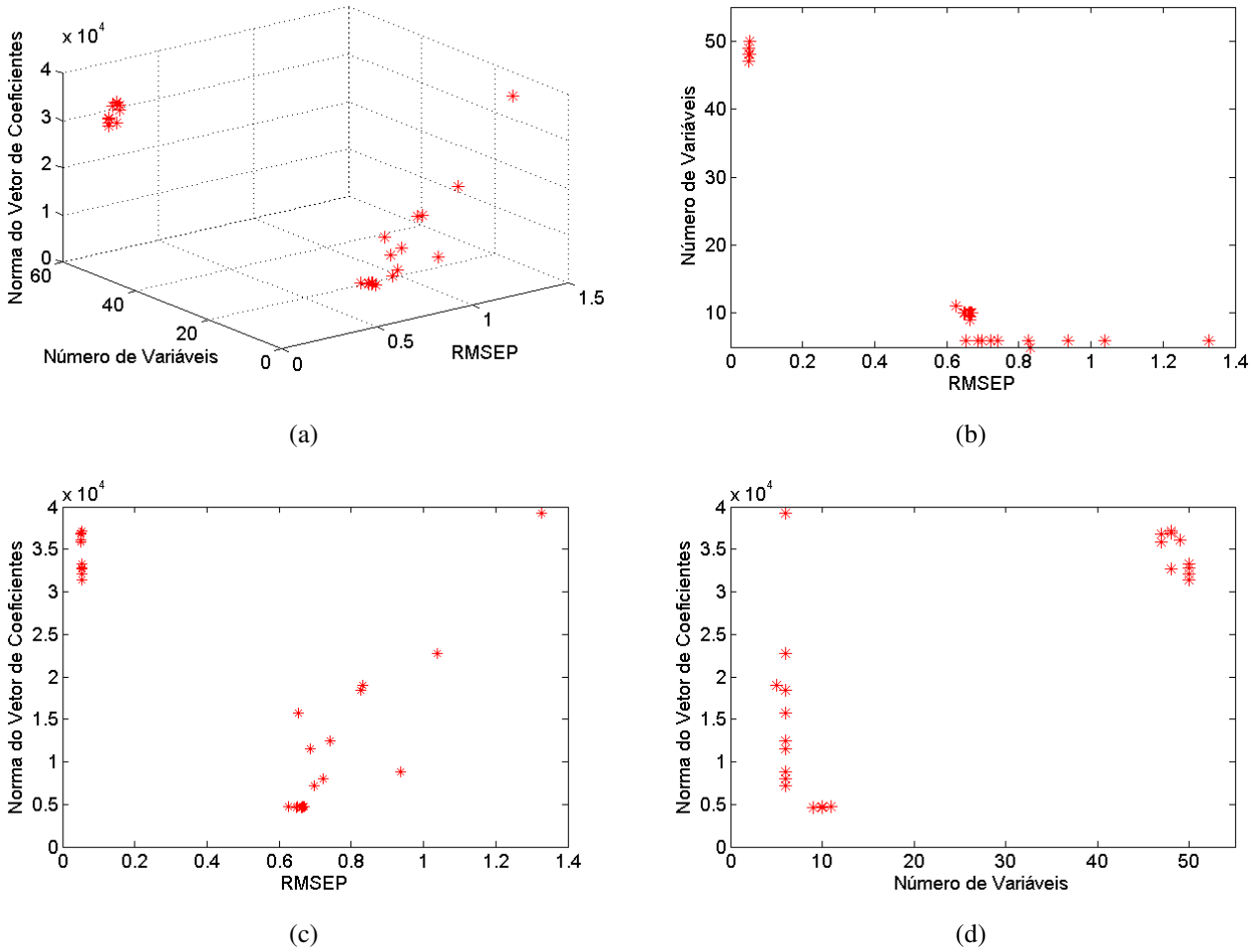

Figura 6.5: *(a) Fronteira de Pareto do Algoritmo Proposto, (b) Relação RMSEP por Número de Variáveis, (c) Relação RMSEP por Norma do Vetor de Coeficientes, (d) Relação Número de Variáveis por Norma do Vetor de Coeficientes.*

A Figura 6.6 apresenta a concentração de proteínas real versus a concentração predita pela solução do AEMT-MLR. Em uma predição perfeita, os pontos estariam dispostos sobre a reta. O erro de predição pode ser medido pela distância entre cada ponto e a reta. Como pode ser visto, a predição dos valores são próximos dos valores reais, o que mostra que a solução produzida pelo algoritmo proposto é capaz de realizar predições com baixo erro.

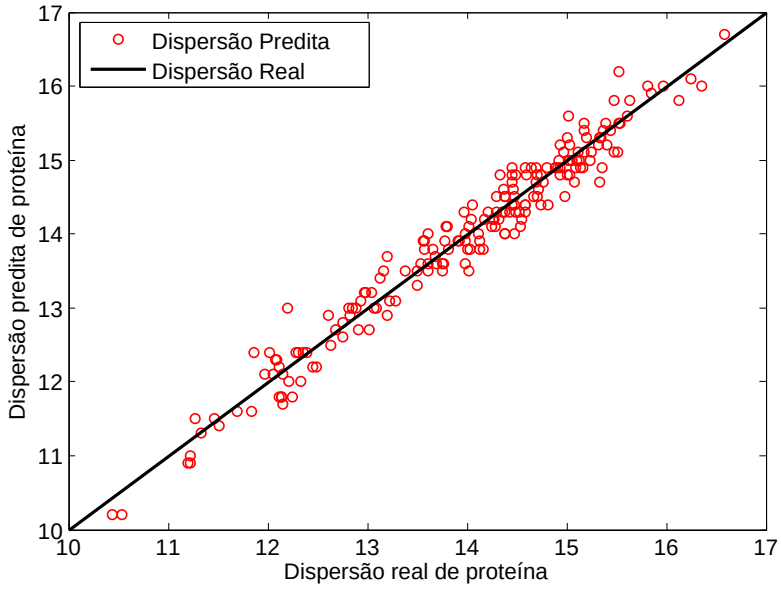

Figura 6.6: *Amostras do grupo de predição.*

6.3 Considerações Finais

Analisando os resultados obtidos pelos algoritmos mono-objetivos, NSGA-II-MLR e AEMT-MLR, pode-se observar que o AEMT-MLR obteve soluções melhores do que o MONO-GA-MLR em todos os objetivos considerados. Em relação ao NSGA-II-MLR, o AEMT-MLR obteve resultados satisfatórios quando considerado também a sensibilidade à ruído, apesar de mais variáveis selecionadas e pequena diferença no erro de predição (RMSEP), o AEMT-MLR mostrou menor sensibilidade à ruídos do que o NSGA-II-MLR, o que mostra que este algoritmo tem uma melhor robustez.

CAPÍTULO 7

Conclusões

Neste trabalho foi proposto uma formulação do algoritmo genético multiobjetivo para o problema de seleção de variáveis em problemas de calibração. Um estudo de caso baseado na concentração de proteína de trigo foi apresentado. Tal conjunto possui inicialmente 690 variáveis, 389 amostras no conjunto de calibração, 193 amostras no conjunto de predição e 193 amostras no conjunto de teste. A natureza deste conjunto de dados implica em variáveis colineares entre si levando o sistema de equações à apresentar problemas de multicolinearidade. Foi possível mostrar que a formulação mono-objetivo leva o algoritmo genético a um número maior de variáveis selecionadas quando comparado ao APS-MLR, além de possuir uma alta sensibilidade à ruído.

Os resultados obtidos demonstram que enquanto a formulação com algoritmo genético mono-objetivo usa um grande número de variáveis com um erro de predição similar aos algoritmos clássicos, os algoritmos multi-objetivos propostos usam menos variáveis com um menor erro de predição.

A principal contribuição do trabalho inclui uma modelagem mais adequada para o problema de seleção de variáveis no domínio original dos dados em sistemas lineares considerando além do erro de predição e do número de variáveis, a sensibilidade à ruído. Neste cenário, foi proposto o uso de um algoritmo multi-objetivo em tabelas para selecionar as variáveis com baixa colinearidade, maior capacidade preditiva do composto de interesse, menor número de variáveis e uma baixa sensibilidade à ruído instrumental.

Quando comparado com os algoritmos clássicos como PLS, APS-MLR e MONO-GA-MLR, os resultados obtidos foram melhores em todos os casos analisados. No estudo de caso, o AEMT-MLR melhorou a predição da concentração de proteína em grãos de trigo em 67% comparado com PLS e MONO-GA-MLR e 65% comparado ao APS-MLR.

Os resultados de erro de predição obtidos com o AEMT-MLR foram próximos aos obtidos com o NSGA-II-MLR. Já em relação ao número de variáveis selecionadas o NSGA-II-MLR selecionou um subconjunto menor que o AEMT-MLR, porém a sensibilidade a ruído da solução obtida pelo NSGA-II-MLR foi maior que a da solução obtida pelo AEMT-MLR.

7.1 Trabalhos Futuros

Como estudos futuros, sugere-se adicionar tabelas de dominância para todos os objetivos em questão e um estudo de métricos de desempenho como hipervolume, uma vez que o algoritmo proposto trabalha pontualmente cada um dos objetivos e essa pontualidade mapeiam somente os extremos de cada um dos objetivos.

Sugere-se ainda o desenvolvimento de mecanismos de reprodução que levam em consideração as características particulares dos problemas de calibração, tais como a multicolinearidade, pois como o conjunto de variáveis no conjunto de predição possui variáveis redundantes e/ou irrelevantes para o problema, essas acabam interferindo na acurácia do modelo.

E sugere-se ainda um estudo de um algoritmo que utilize técnicas de coevolução com mecanismos competitivos e cooperativos, em que as características das variáveis selecionadas no conjunto de predição interfiram nas características para seleção das amostras do conjunto de calibração, o que implica em diferentes resultados de RMSEP, número de variáveis e sensibilidade de todo o conjunto de entrada.

Referências Bibliográficas

[1] ABDI, H. **Partial least squares (pls) regression**. *Encyclopedia of Social Sciences Research Methods*, 3(6):1–7, 2003.

[2] ARAKAWA, M.; YAMASHITA, Y.; FUNATSU, K. **Genetic algorithm-based wavelength selection method for spectral calibration**. *Journal of Chemometrics*, 25(1):10–19, 2011.

[3] ARAÚJO, M. C. U.; SALDANHA, T. C. B.; GALVÃO, R. K. H.; YONEYAMA, T.; CHAME, H. C.; VISANI, V. **The successive projections algorithm for variable selection in spectroscopic multicomponent analysis**. *Chemometrics and Intelligent Laboratory Systems*, 57(2):65–73, 2001.

[4] BARRICO, C. M. C. S. **Optimização evolucionária multi-objectivo em ambientes incertos**, 2007.

[5] BEEBE, K. R.; PELL, R. J.; SEASHOLTZ, M. B. **Chemometrics: a practical guide**. 1998.

[6] BRASIL, C. R. S. **Algoritmo evolutivo de muitos objetivos para predição ab initio de estrutura de proteínas**, May 2012.

[7] CHONG, I.-G.; JUN, C.-H. **Performance of some variable selection methods when multicollinearity is present**. *Chemometrics and Intelligent Laboratory Systems*, 78(1):103–112, 2005.

[8] COELLO, C.; PULIDO, G. **Multiobjective optimization using a micro-genetic algorithm**. In: Spector, L.; Goodman, E.; Wu, A.; Langdon, W.; Voigt, H.; Gen, M.; Sen, S.; Dorigo, M.; Pezeshk, S.; Garzon, M.; Burke, E., editors, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2001)*, p. 274–281. Morgan Kaufmann Publishers, 2001.

[9] CORNE, D.; JERRAM, N.; KNOWLES, J.; OATES, M. **Pesa-ii: Region-based selection in evolutionary multiobjective optimization**. In: Spector, L.; Goodman, E.; Wu, A.; Langdon, W.; Voigt, H.; Gen, M.; Sen, S.; Dorigo, M.; Pezeshk, S.; Garzon, M.;

Burke, E., editors, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2001)*, p. 283–290. Morgan Kaufmann Publishers, 2001.

[10] CORNE, D.; KNOWLES, J.; OATES, M. **The pareto envelope-based selection algorithm for multiobjective optimization**. In: Deb, K.; G. Rudolph, X. Y.; Lutton, E.; Merelo, J. J.; Schwefel, H. P., editors, *Proceedings of the Parallel Problem Solving from Nature VI Conference,*, p. 839–848. Springer. Lecture Notes in Computer Science No. 1917, 2000.

[11] CORTINA, J. M. **Interaction, nonlinearity, and multicollinearity: Implications for multiple regression**. *Journal of Management*, 19(4):915–922, 1994.

[12] DE LUCENA, D. V.; DE LIMA, T. W.; DA SILVA SOARES, A.; FILHO, A. R. G.; COELHO, C. J. **Multiobjective evolutionary algorithm for variables selection in calibration problems: a case study for protein concentration prediction**. *Proccedings of the IEEE Congress of Evolutionary Computation*, 2013.

[13] DEB, K.; PRATAP, A. A. S. . M. T. **A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II**. IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, VOL. 6, NO. 2, 2002.

[14] DEB, K. **Multi-Objective Optimization using Evolutionary Algorithms**. John Wiley and Sons, LTD, 2001.

[15] DEB, K.; AGRAWAL, S.; PRATAB, A.; MEYARIVAN, T. **A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II**. KanGAL report 200001, Indian Institute of Technology, Kanpur, India, 2000.

[16] DEB, K.; PRATAP, A.; AGARWAL, S.; MEYARIVAN, T. **A fast elitist multi-objective genetic algorithm: Nsga-ii**. *IEEE Transactions on Evolutionary Computation*, 6:182–197, 2000.

[17] DOS SANTOS, A. C. **Algoritmo evolutivo computacionalmente eficiente para reconfiguração de sistemas de distribuição**, 2009.

[18] FERREIRA, M. A. M. C.; ANTUNES, A. M.; MELGO, M. S.; VOLPE, P. L. O. **Quimiometria I: calibração multivariada, um tutorial**. *Química Nova*, 22:724 – 731, 09 1999.

[19] FILHO, A. R. G.; ARAÚJO, M.; GALVÃO, R. K. H. **Effect of the subsampling ratio in the application of subagging for multivariate calibration with the successive projections algorithm**. *Journal of the Brazilian Chemical Society*, 22:2225–2233, 2011.

[20] FILHO, P. A. C.; POPPI, R. J. **Aplicação de algoritmos genéticos na seleção de variáveis em espectroscopia no infravermelho médio. determinação simultânea de glicose, maltose e frutose**. *Quim. Nova*, 25(1):46–52, 2002.

[21] FONSECA, C.; FLEMING, P. **Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization**. In: Forrest, S., editor, *Proceedings of the Fifth International Conference on Genetic Algorithms*, p. 416–423, San Mateo, California, 1993. University of Illinois at Urbana-Champaign, Morgan Kauffman Publishers.

[22] GALVAO, R. K. H.; ARAUJO, M. C. U.; FRAGOSO, W. D.; SILVA, E. C.; JOSE, G. E.; SOARES, S. F. C.; PAIVA, H. M. **A variable elimination method to improve the parsimony of {MLR} models using the successive projections algorithm**. *Chemometrics and Intelligent Laboratory Systems*, 92(1):83–91, 2008.

[23] GELADI, P.; KOWALSKI, B. R. **Partial least-squares regression: a tutorial**. *Analytica Chimica Acta*, 185(0):1 – 17, 1986.

[24] GEORGE, E. I. **The variable selection problem**. *Journal of the American Statistical Association*, 95(452):1304–1308, 2000.

[25] GUY, R. H.; HOSTYNEK, J. J.; HINZ, R. S.; LORENCE, C. R. **Metals and the Skin**. Marcel Dekker Incorporated, 1999.

[26] HAALAND, D. M. & THOMAS, E. V. **Partial Least-Squares Methods for Spectral Analysis 1**. Relation to Other Quantitative Calibration Methods and the Extraction of Quantitative Information, Anal. Chem, 60, (pp 1193), 1988.

[27] HAJELA, P.; LIN, C. Y. **Genetic search strategies in multicriterion optimal design**. *Structural Optimization*, 4:99–107, 1992.

[28] HOLLAND, J. **Adaptation in natural and artificial systems**. University of Michigan Press, 1975.

[29] HORN, J.; NAFPLIOTIS, N.; GOLDBERG, D. **A Niched Pareto Genetic Algorithm for Multiobjective Optimization**. In: *Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*, volume 1, p. 82–87, Piscataway, New Jersey, June 1994. IEEE Service Center.

[30] JOLLIFFE, I. T. **Principal Component Analysis**. Springer, second edition, Oct. 2002.

[31] JOLLIFFE, I. T. **A Note on the Use of Principal Components in Regression**. *Applied Statistics*, 31(3):300+, 1982.

[32] KALIVAS, J. H. **Pareto calibration with built-in wavelength selection**. *Analytica Chimica Acta*, (505):9–14, Dezembro 2004.

[33] KENNARD, R. . S. L. A. **Computer aided design of experiments**. Technometrics, 11, (pp 137-148), 1969.

[34] KITA, H.; YABUMOTO, Y.; MORI, N.; NISHIKAWA, Y. **Multi-Objective Optimization by Means of the Thermodynamical Genetic Algorithm**. In: Voigt, H.-M.; Ebeling, W.; Rechenberg, I.; Schwefel, H.-P., editors, *Parallel Problem Solving from Nature—PPSN IV*, Lecture Notes in Computer Science, p. 504–512, Berlin, Germany, September 1996. Springer-Verlag.

[35] KNOWLES, J.; CORNE, D. **The Pareto Archived Evolution Strategy: A New Baseline Algorithm for Multiobjective Optimisation**. In: *1999 Congress on Evolutionary Computation*, p. 98–105, Washington, D.C., July 1999. IEEE Service Center.

[36] LAUMANNS, M.; RUDOLPH, G.; SCHWEFEL, H.-P. **A Spatial Predator-Prey Approach to Multi-Objective Optimization: A Preliminary Study**. In: Eiben, A. E.; Schoenauer, M.; Schwefel, H.-P., editors, *Parallel Problem Solving From Nature — PPSN V*, p. 241–249, Amsterdam, Holland, 1998. Springer-Verlag.

[37] LINDEN, R. **Algoritmos Genéticos**. Brasport, Rio de Janeiro, Brasil, 2008.

[38] LOPES, A. M. **Uma abordagem multiobjetivo para o problema de corte de estoque unidimensional**. Master's thesis, Universidade Estadual Paulista Júlio de Mesquita Filho, Jan. 2009.

[39] LUCASIUS, C. B.; KATEMAN, G. **Genetic algorithms for large-scale optimization in chemometrics - an application, trends in analytical chemistry**. 10:p.254–261, 1991.

[40] MILLER, A. J. **Selection of subsets of regression variables**. *Journal of the Royal Statistical Society. Series A (General)*, p. 389–425, 1984.

[41] MONTGOMERY, D. C.; PECK, E. A. **Introduction to Linear Regression Analysis**. John Wiley and Sons, New York, 1992.

[42] MOREIRA, E. D.; PONTES, M. J.; GALVÃO, R. K.; ARAÚJO, M. C. **Near infrared reflectance spectrometry classification of cigarettes using the successive projections algorithm for variable selection**. *Talanta*, 79(5):1260–1264, 2009.

[43] NAES, T.; MEVIK, B. H. **Understanding the collinearity problem in regression and discriminant analysis**. *Journal of Chemometrics*, 15(4):413–426, 2001.

[44] NEBRO, A. J.; DURILLO, J. J.; LUNA, F.; DORRONSORO, B.; ALBA, E. **Mocell: A cellular genetic algorithm for multiobjective optimization**. *Int. J. Intell. Syst.*, 24(7):726–746, July 2009.

[45] REIS, M. M. **Conceitos elementares de Estatística**, 2013.

[46] RENCHER, A. C. **Methods of Multivariate Analysis**. Willey-Interscience, 2002.

[47] RUDOLPH, G. **Evolutionary Search under Partially Ordered Fitness Sets**. In: *Proceedings of the International NAISO Congress on Information Science Innovations (ISI 2001)*, p. 818–822. ICSC Academic Press: Millet/Sliedrecht, 2001.

[48] SANCHES, D. S. **Algoritmos evolutivos multi-objetivo para reconfiguração de redes em sistemas de distribuição de energia elétrica**, Dec. 2012.

[49] SCHAFFER, J. **Multiple objective optimization with vector evaluated genetic algorithms**. In: *Genetic Algorithms and their Applications: Proceedings of the First International Conference on Genetic Algorithms*, p. 93–100. Lawrence Erlbaum, 1985.

[50] SHIMADA, M.; MASUDA, Y.; YAMADA, Y.; ITOH, M.; TAKAHASHI, M.; YATAGAI, T. **Explanation of human skin color by multiple linear regression analysis based on the modified lambert-beer law**. *Optical Review*, 7(4):348–352, 2000.

[51] SKOOG, D. A. **Princípios de análise instrumental**. Bookman, 2002.

[52] SOARES, S. F.; PEREIRA, A. F. C.; G.NETO, F. F.; PONTES, M. J. C.; .SILVA, E. C.; ARAÚJO, M. C.; FRAGOSO, W. D.; SANTOS, S. R. B.; GALVÃO, R. K. H. **Um novo algoritmo de seleção de variáveis aplicado à determinação de viscosidade de Óleos vegetais por espectrometria nir**. *25ª Reunião Anual da Sociedade Brasileira de Química - SBQ*, 2002.

[53] SOARES, S. F.; GOMES, A. A.; ARAUJO, M. C.; FILHO, A. R. G.; GALVAO, R. K. H. **The successive projections algorithm**. *TrAC Trends in Analytical Chemistry*, 42:84–98, 2013.

[54] SOARES(A), A. S.; GALVÃO FILHO, A. R. G. A. R. K. H. . A. M. C. U. **Improving the computational efficiency of the successive projections algorithm by using a sequential regression implementation: a case study involving nir spectrometric analysis of wheat samples**. J. Braz. Chem. Soc., v. 21, n. 4, São Paulo, 2010.

[55] SOARES(B), A. S.; GALVÃO FILHO, A. R. G. A. R. K. H. . A. M. C. U. **Multi-core computation in chemometrics: case studies of voltammetric and NIR spectrometric analyses**. J. Braz. Chem. Soc., v. 21, n. 9, São Paulo, 2010.

[56] SRINIVAS, N.; DEB, K. **Multiobjective optimization using nondominated sorting in genetic algorithms**. *Evolutionary Computation*, 2:221–248, 1994.

[57] TOBIAS, R. **An Introduction to Partial Least Squares Regression**. TS-509. SAS Institute Inc., Cary, NC., 1997.

[58] TOMASSINI, M. **Spatially Structured Evolutionary Algorithms: Artificial Evolution in Space and Time (Natural Computing Series)**. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.

[59] V, A. M. S. T. G. R. Y. T. C. H. V. **The successive projections algorithm for variable selection in spectroscopic multicomponent analysis**. *Chemometrics and Intelligent Laboratory Systems*, 57(2):65–73, 2001.

[60] VARGAS, D. V.; MURATA, J.; TAKANO, H.; DELBEM, A. C. B. **General subpopulation framework and taming the conflict inside populations.** *Evol Comput*, 2014.

[61] VELDHUIZEN, D. **Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations**. PhD thesis, Department of Electrical and Computer Engineering. Graduate School of Engineering. Air Force Institute of Technology, Wright-Patterson AFB, Ohio, May 1999.

[62] ZITZLER, E.; LAUMANNS, M.; THIELE, L. **SPEA2: Improving the Strength Pareto Evolutionary Algorithm**. Technical Report 103, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Zurich, Gloriastrasse 35, CH-8092 Zurich, Switzerland, May 2001.

[63] ZITZLER, E.; THIELE, L. **An Evolutionary Algorithm for Multiobjective Optimization: The Strength Pareto Approach**. Technical Report 43, Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, May 1998.

[64] ZITZLER, E. **Evolutionary algorithms for multiobjective optimization: Methods and applications**, 1999.

[65] ZITZLER, E.; LAUMANNS, M.; THIELE, L. **Spea2: Improving the strength pareto evolutionary algorithm**. Technical report, 2001.

APÊNDICE A

Norma-2

Em problemas de calibração multivariada para análises espectrofotométricas, os valores das variáveis independentes correspondem a medidas instrumentais contaminadas por ruído. Desse modo é necessário ter uma medida da sensibilidade das predições do modelo com respeito a tal ruído instrumental[19]. Neste apêndice, adaptado de Galvão Filho[19], mostra-se que tal medida de sensibilidade pode ser dada pela norma-2 do vetor de coeficientes da regressão. Sendo assim, considere que um modelo da forma

$$\hat{y} = b_0 + \sum_{k=1}^{K} b_k x_k \tag{A-1}$$

seja utilizado para calcular valores $\hat{y}$ previstos a partir de valores medidos $x = [x_1, x_2, ..., x_k]^T$. Além disso, suponha que tais valores medidos estejam contaminados por ruído, de modo que

$$x = \mu + \eta \tag{A-2}$$

sendo $\mu = E\{x\}, E\{\eta\} = 0$ e $E\{\eta\eta^T\} = \Sigma_\eta$, onde $E\{\cdot\}$ denota o valor esperado de uma variável aleatória. Tem-se portanto

$$\begin{aligned} E\{\hat{y}\} &= b_0 + E\left\{\sum_{k=1}^{k} b_k x_k\right\} \\ &= b_0 + \sum_{k=1}^{k} b_k E x_k \\ &= b_0 + \sum_{k=1}^{k} b_k \mu_k 0 + \sum_{k=1}^{k} b_k \mu_k \end{aligned} \tag{A-3}$$

Assim, a variância de $\hat{y}$ pode ser obtida por

$$\sigma_{\hat{y}}^2 = E\left\{[\hat{y} - E(\hat{y})]^2\right\}$$

$$
\begin{aligned}
&= E\left\{\left[\sum_{k=1}^{k} b_k(x_k-\mu_k)\right]^2\right\} \\
&= E\left\{\left[\sum_{k=1}^{k} b_k\eta_k\right]^2\right\} \\
&= E\{(b^T\eta)^2\} = E\{b^T\eta\eta^T b\} \\
&= b^T E\{\eta\eta^T\}b = b^T\Sigma_\eta b \qquad \text{(A-4)}
\end{aligned}
$$

Se o ruído η for branco e homoscedástico, a matriz de variância-covariância Σ_η é da forma $\sigma_\eta^2 I$, onde I e uma matriz identidade (KxK). Neste caso, a Equação (A-4) torna-se

$$\sigma_{\hat{y}}^2 = \sigma_{\hat{y}}^2 b^T b = \sigma_{\hat{y}}^2 \|b\|_2^2 \qquad \text{(A-5)}$$

onde $\|b\|_2$ é a norma-2 do vetor de coeficientes de regressão. A Equação (A-5) mostra que vetores de regressão maiores (no sentido da norma-2) tendem a gerar previsões que são mais sensíveis ao ruído de medição nas variáveis x.

Na Equação (5-1), tem-se a formulação simplificada da Equação (A-5), na qual foi utilizada como uma das funções objetivo desse trabalho.