Centro Federal de Educação Tecnológica de Minas Gerais
Programa de Pós-Graduação em Modelagem Matemática e Computacional

Aplicação da Metaheurística PSO na Identificação de Pontos Influentes por meio da Função de Sensibilidade de Casos

Adriana Aparecida Batista Costa

Belo Horizonte, Novembro de 2009

Centro Federal de Educação Tecnológica de Minas Gerais
Programa de Pós-Graduação em Modelagem Matemática e Computacional

Aplicação da Metaheurística PSO na Identificação de Pontos Influentes por meio da Função de Sensibilidade de Casos

Dissertação de mestrado apresentada ao Programa de Pós-graduação em Modelagem Matemática e Computacional do Centro Federal de Educação Tecnológica de Minas Gerais, como requisito parcial para obtenção do título de Mestre em Modelagem Matemática e Computacional.

Adriana Aparecida Batista Costa

Orientadora: Prof[a].Dr[a]. Elenice Biazi

Co-Orientador: Prof. Dr. João Francisco de Almeida Vitor

Belo Horizonte, Novembro de 2009

Costa, Adriana Aparecida Batista Costa
C837a Aplicação da metaheurística PSO na identificação de pontos influentes por meio da função de sensibilidade de casos./ Adriana Aparecida Batista Costa. - 2009.
82f.

Orientadora: Elenice Biazi
Co-orientador: João Francisco de Almeida Vítor

Dissertação (mestrado) - Centro Federal de Educação Tecnológica de Minas Gerais

1. Heurística.2. Matemática computacional.
I. Biazi, Elenice. II. Vítor, João Francisco de Almeida
III. Centro Federal de Educação Tecnológica de Minas Gerais.
IV. Título

CDD: 519.6

Agradecimentos

Em primeiro lugar agradeço à minha orientadora, Professora Elenice Biazi, por me dar a chance de participar do curso de mestrado e pelo apoio durante a realização deste trabalho.

Agradeço ao Professor João Francisco de Almeida Vítor, pela disponibilidade e ajuda ao longo da pesquisa.

Meu agradecimento especial à minha irmã Ana Paula, pelo incentivo, motivação e apoio incondicional em todos os momentos.

Agradeço à minha mãe, Maria do Carmo, exemplo de força e sabedoria, pelos ensinamentos e amor.

Agradeço ao meu esposo, Crisantino, pelo amor e compreensão.

Agradeço à minha família, pelo carinho. Em especial ao Ronaldo e ao Amauri, que sempre me incentivaram e me apoiaram.

Agradeço aos colegas do mestrado, em especial ao Vinícius e à Rosiene, pela amizade e apoio nos momentos difíceis.

A todos vocês, meus sinceros agradecimentos.

Dedico este trabalho ao meu irmão, Wanderley.

Resumo

A identificação de pontos influentes é um importante passo na análise de dados. Estes pontos exercem grande influência na determinação dos coeficientes de regressão de um modelo. Existem várias técnicas para detectar pontos influentes, muitas destas técnicas encontram-se implementadas e disponibilizadas em pacotes de programas estatísticos. Para os pontos isolados, o cálculo destas medidas é relativamente simples e de baixo custo computacional. A detecção de grupos de pontos é mais difícil, pois o número elevado de subconjuntos de pontos pode tornar a computação inviável. O presente trabalho apresenta uma aplicação da metaheurística Otimização por Enxame de Partículas (*Particle Swarm Optimization* - PSO) na identificação de pontos influentes em modelos de regressão. Foi utilizada, como função objetivo, a função de sensibilidade de casos $g_{Cook}(\epsilon)$ que tem comportamento multimodal. A eficácia da metodologia proposta foi testada em conjuntos de dados simulados e em um conjunto com dados observados. Os resultados obtidos mostram que esta metodologia apresenta boas soluções na identificação de pontos influentes.

Palavras-Chave: Modelos de regressão, Pontos influentes, Função de Sensibilidade de Casos, Metaheurística PSO.

Abstract

The identification of influential points is an important step in the data analysis. These points have great influence in determining the regression coefficients in the model. There are several techniques to detect influential points, many of them are implemented and available in packages of statistical programs. For the single case, the calculation of these measures is relatively simple and of low computational cost. However, the detection of multiple cases is more difficult because the large number of subsets of points can make the computation unfeasible. In this work, the Particle Swarm Optimization metaheuristic was applied to identify influential points in the regression model. The multimodal Case Sensitivity Function $g_{Cook}(\epsilon)$ was used as target function. The effectiveness of the proposed methodology was tested in simulated data sets and in observed data. The results show that this methodology gives good solutions in the search for influential points.

Keywords: Regression Models, Influential Points, Case Sensitivity Function, PSO Metaheuristic.

Sumário

Lista de Figuras

Lista de Tabelas

Nomenclatura

c_1	Coeficiente cognitivo
c_2	Coeficiente social
C_d	Curvatura máxima
D_i	Distância de Cook
d_{max}	Direção unitária máxima
$\mathbf{e}$	Erros aleatórios
E(e)	Esperança do erro
$gbest$	Melhor posição histórica do enxame
$g(\epsilon)$	Função de influência
$g_T(\epsilon)$	Função de sensibilidade de casos
H	Matriz chapéu
$I(\widehat{\beta})$	Matriz de informação
k	Número de observações influentes
K	Fator de constrição
$l(\beta)$	Logaritmo da verossimilhança
LD	Afastamento da verossimilhança
n	Número de observações
p	Número de variáveis
$pbest$	Melhor valor histórico da partícula
r_1 e r_2	Números aleatórios
r_i	Resíduo studentizado interno
t_i	Resíduo studentizado externo
s^2	Variância amostral
T	Funcional estatístico
Var(e)	Variância do erro
y	Vetor de valores observados
$\hat{y}$	Vetor de valores ajustados

Letras gregas

β	Vetor de parâmetros
$\hat{\beta}$	Vetor de parâmetros estimado
ϵ	Vetor de perturbações

Capítulo 1

Introdução

1.1 Preliminares

Modelos estatísticos, geralmente, são descrições aproximadas de processos bastante complexos, que conseqüentemente podem levar a resultados imprecisos. Portanto, considerações da adequação de um modelo são imprescindíveis.

Uma etapa importante após o ajuste de um modelo de regressão é a análise de diagnóstico, que consiste em verificar possíveis afastamentos das suposições do modelo e em detectar pontos influentes (Paula, 2004). Estes pontos estão quase sempre presentes em qualquer conjunto de dados.

A presença de pontos influentes pode ocorrer por erro na coleta dos dados, na transcrição dos dados ou por realmente existirem alguns pontos diferentes dos demais em um conjunto de dados. A identificação de tais pontos contribui para uma modelagem estatística mais eficiente, já que individualmente ou em conjunto, estes pontos podem produzir grandes alterações em aspectos importantes da análise.

As técnicas para detectar pontos influentes, geralmente, são baseadas em dois princípios: a omissão de pontos e a perturbação. As técnicas baseadas em omissão de pontos consistem em ajustar o modelo proposto, omitindo uma ou mais observações por vez, e comparar estes modelos com o modelo ajustado utilizando todos os pontos. As perturbações são pequenas mudanças que são feitas no

modelo, como exemplo, associando pesos ou probabilidades às observações do conjunto de dados.

A partir da década de 70 surgiram várias propostas relacionadas com a influência das observações nas estimativas dos coeficientes do modelo de regressão linear. Uma técnica muito utilizada é a de omissão de pontos, talvez, esta seja a técnica mais conhecida para avaliar o impacto da retirada de uma observação nas estimativas da regressão. As medidas de omissão de pontos foram propostas por Cook (1977) e Belsley et al. (1980). Um problema que pode ocorrer com a técnica de omissão individual de pontos é o que se denomina "mascaramento", que consiste em deixar de detectar pontos conjuntamente influentes.

Uma metodologia alternativa à técnica de omissão de pontos, denominada Função de influência, foi desenvolvida por Hampel (1974) e explorada por Cook e Weisberg (1982). Nestes trabalhos, são analisadas as perturbações em probabilidades associadas a casos (observações). Uma técnica muito utilizada na análise de diagnóstico em regressão, denominada influência local, foi apresentada por Cook (1986), para avaliar a influência conjunta das observações sob pequenas perturbações nos dados ou no modelo.

Recentemente, são encontrados trabalhos desenvolvidos a partir das técnicas descritas anteriormente ou extensões destas. Por exemplo, em (Imon, 2005) é feita uma generalização da medida de influência DFFITS (Belsley et al., 1980) para casos múltiplos. Em 2005, Peña propôs uma medida de influência, baseada na técnica de omissão de pontos. Esta medida indica o quanto uma observação é influenciada pelas demais observações do conjunto de dados, ou seja, considera o quadrado da norma do vetor de previsões, quando cada um dos pontos é omitido por vez. Em (Lemonte, 2008) é feita uma análise das medidas de influência em modelos lineares normais. Em (Ampanthong e Suwattee, 2009) são aplicadas algumas medidas de influência em modelos de regressão multivariada.

O uso de metaheurísticas para a identificação de pontos influentes não é comum na literatura. Alguns trabalhos referem-se a identificação de *outliers*. *Outliers* são observações atípicas, que podem ser influentes ou não nas estimativas de

um modelo. Aplicações da metaheurística algoritmo genético, utilizando métodos combinatórios para detectar *outliers* em modelos de regressão, podem ser encontradas em (Crawford e Wainwright, 1996) e (Tolvi, 2004). Em (Ye e Chen, 2008) foi utilizada a metaheurística PSO para detectar *outliers*, estudando o comportamento das projeções dos conjuntos de dados.

A função de sensibilidade de casos (*Case Sensitivity Function*), representada por $g_T(\epsilon)$, sendo ϵ um vetor de perturbações e T um funcional estatístico, é uma abordagem para a análise de dados influentes. Esta função foi proposta por Critchley (1989) e explorada por Biazi (1996) e Critchley et al. (2001). Esta metodologia consiste em inserir pequenas perturbações no modelo e maximizar o efeito $|g_T(\epsilon) - g_T(0)|$. Uma vez selecionado o tipo de aspecto da análise abordado, a aplicação desta função determinará agrupamentos de pontos que causam um efeito grande. Nos trabalhos de Biazi (1996) e Critchley et al. (2001), foi utilizada a função de sensibilidade para vários funcionais estatísticos, os quais mostraram-se eficazes para identificar pontos influentes. Entretanto, nestes trabalhos, devido às limitações computacionais, esta metodologia foi testada em pequenos conjuntos de dados.

No presente trabalho é aplicada a metaheurística Otimização por Enxame de Partículas na identificação de pontos influentes em modelos de regressão, utilizando a função de sensibilidade de casos para o funcional estatístico distância de Cook, que apresenta comportamento multimodal. Metaheurísticas evolutivas, dentre elas, Otimização por Enxame de Partículas, têm sido eficientes na otimização de funções multimodais, uma vez que operam simultaneamente com vários indivíduos da população na mesma iteração.

1.2 Objetivos

1.2.1 Objetivo geral

Este trabalho tem como objetivo geral verificar a eficácia da Função de Sensibilidade de Casos na identificação de pontos influentes, quando é empregada a

metaheurística PSO.

1.2.2 Objetivos específicos

- Aplicar a metodologia em conjuntos com múltiplos pontos influentes.
- Realizar o estudo da metaheurística PSO.
- Ajustar os parâmetros do algoritmo PSO visando encontrar a melhor solução possível para o problema.
- Avaliar os resultados obtidos quanto à aplicabilidade do método ao problema.

1.3 Organização da dissertação

O presente trabalho está organizado em seis capítulos. No Capítulo 2, são apresentadas as principais técnicas utilizadas para identificar pontos influentes. Encontram-se descritas a metodologia da função de sensibilidade de casos e a função objetivo.

No capítulo 3, é apresentado o método de Otimização por Enxame de Partículas, empregado na identificação de pontos influentes. Destacam-se as origens e as características específicas do algoritmo PSO.

O algoritmo PSO, adaptado às especificidades do problema tratado, é apresentado no Capítulo 4. São descritos os testes computacionais realizados para o ajuste dos parâmetros de controle do algoritmo.

No capítulo 5, são apresentados os resultados computacionais obtidos.

A conclusão geral e as propostas de trabalhos futuros são apresentadas no capítulo 6.

Capítulo 2

Referencial Teórico

Neste capítulo, são apresentados alguns conceitos relacionados à análise de dados influentes e as principais medidas para detecção de pontos influentes. Um ponto pode ser influente em diversos aspectos da análise estatística. Neste trabalho, é considerada a influência no contexto de regressão linear. Apresenta-se também a função de sensibilidade de casos.

2.1 Análise de dados influentes

2.1.1 Modelo de regressão linear

Um modelo de regressão é um modelo matemático que descreve a relação entre uma variável resposta e uma ou mais variáveis independentes ou regressoras. A regressão linear é usada para uma classe especial de relações, ou seja, aquelas que podem ser descritas por linhas retas, ou por suas generalizações para várias dimensões (Weisberg, 1985).

Um modelo de regressão linear simples descreve a relação entre uma variável dependente y e uma variável independente x. Este modelo é representado pela seguinte equação:

$$y_i = \beta_0 + \beta_1 x_i + e_i \qquad i = 1, ..., n \tag{2.1}$$

sendo β_0 o intercepto, β_1 o coeficiente da variável x e e_i os erros aleatórios supostos, relativos a uma população normal, com média zero e variância constante. Neste sentido, β_0 e β_1 são os parâmetros desconhecidos a serem estimados.

Um modelo de regressão linear múltipla é composto por uma variável dependente y e duas ou mais variáveis regressoras. A forma geral de um modelo de regressão linear múltipla é dada pela seguinte equação linear:

$$\mathbf{y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p + e \tag{2.2}$$

Em modelos de regressão linear múltipla, o conjunto de observações é denotado por um vetor de observações $\mathbf{y} = (y_1, ..., y_n)^T$. O conjunto de variáveis regressoras é denotado por uma matriz X, com n observações e p variáveis independentes. O conjunto de parâmetros é um vetor denotado por $\beta = (\beta_1, ..., \beta_p)^T$ e $\mathbf{e}$ um vetor de erros denotado por $\mathbf{e} = (e_1, ..., e_n)^T$. Desta forma, o modelo (2.2) pode ser escrito matricialmente,

$$\mathbf{y} = X\beta + \mathbf{e} \tag{2.3}$$

Um modelo de regressão linear é obtido a partir dos seguintes pressuspostos:

- A média da distribuição de probabilidade do erro é nula.
- A variância da distribuição de probabilidade do erro é constante, ou seja, o erro é homocedástico.
- A distribuição de probabilidade do erro é normal.
- Os valores do erro $\mathbf{e}$ são independentes entre si.

2.1.1.1 Estimador de mínimos quadrados

O método dos mínimos quadrados, também denominado mínimos quadrados ordinário, fornece uma estimativa para o vetor de parâmetros β que minimiza a soma dos quadrados das distâncias entre cada ponto observado e seu valor estimado pelo modelo. Este método consiste na minimização da seguinte função

$$S = \sum_{i=1}^{n} e_i^2 = \mathbf{e}^T\mathbf{e} = (\mathbf{y} - X\beta)^T(\mathbf{y} - X\beta)$$
$$= \mathbf{y}^T\mathbf{y} - 2\beta^T X^T\mathbf{y} + \beta^T X^T X\beta$$

Para a minimização da função S, o estimador de mínimos quadrados deve, portanto, satisfazer a

$$\frac{\partial S}{\partial \beta} = -2X^T\mathbf{y} + 2X^T X\beta = 0 \tag{2.4}$$

Assim, obtém-se o vetor de parâmetros estimado

$$\hat{\beta} = (X^T X)^{-1} X^T\mathbf{y} \tag{2.5}$$

e o modelo de regressão ajustado apresenta-se na forma

$$\hat{\mathbf{y}} = X\hat{\beta} \tag{2.6}$$

2.1.2 Pontos extremos em um conjunto de dados

A detecção de pontos extremos, isto é, pontos com um "comportamento" diferente dos demais pontos num conjunto de dados é uma das formas de avaliação de modelos ajustados.

No contexto de regressão linear, pontos extremos podem ser classificados, de acordo com suas características, em:

- *Outliers*: são pontos que apresentam grandes resíduos.

- Pontos de alavanca ou de alto *leverage*: são pontos que estão mais afastados dos demais pontos no subespaço gerado pelas colunas da matriz de variáveis explicativas. Estes pontos não influenciam de forma significativa as estimativas dos parâmetros, porém fazem com que as variâncias dos valores ajustados dos pontos próximos a eles sejam maiores do que as variâncias dos valores ajustados correspondentes aos demais pontos.

- Pontos influentes: são pontos que, individualmente ou coletivamente, influenciam a equação de regressão ajustada, quando comparada com outros pontos do conjunto de dados.

Na Fig. 2.1 é apresentada uma ilustração de *outliers*, ponto de alavanca e ponto influente para um modelo de regressão linear simples.

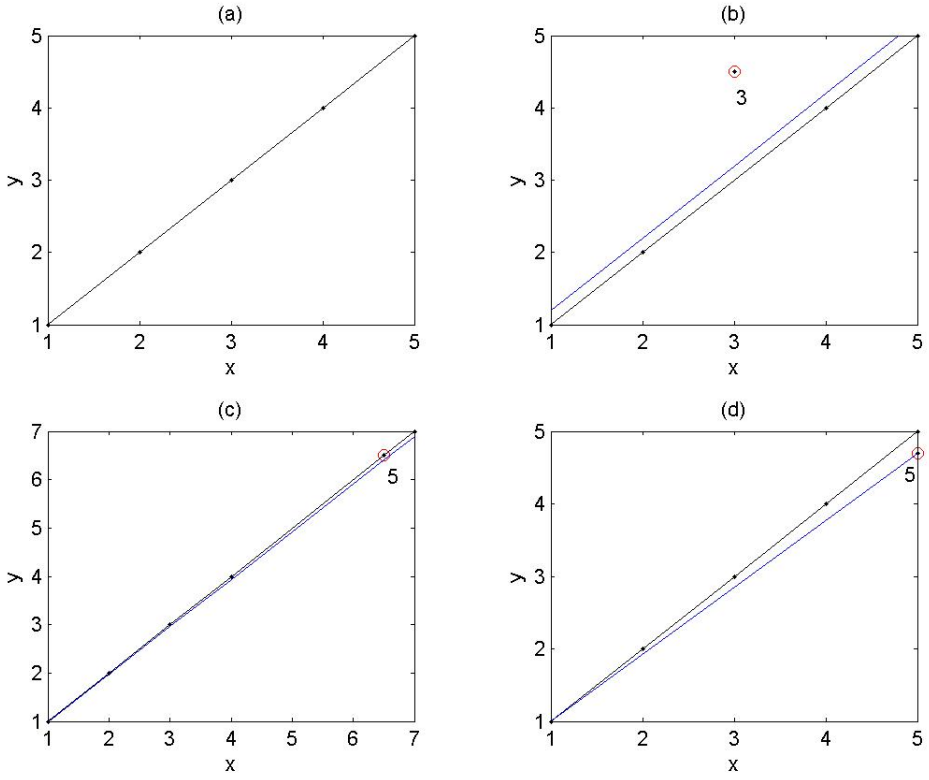

Figura 2.1: Ilustração de pontos extremos (Paula,2004)

Na Fig. 2.1.a os pontos estão bem ajustados sem nenhum ponto extremo. Na Fig. 2.1.b o ponto 3 é um *outlier* (resíduo relativamente alto), sua retirada altera os valores ajustados. Na Fig. 2.1.c o ponto 5 é um ponto de alavanca, sua exclusão não muda praticamente nada nas estimativas dos parâmetros. Na Fig. 2.1.d o ponto 5 é um ponto influente, sua retirada muda a inclinação da reta ajustada.

Quando são detectados pontos extremos num ajuste de regressão estes pontos devem ser examinados cuidadosamente antes de qualquer decisão. Os pontos ex-

tremos podem indicar algumas características importantes sobre o modelo, como modelo incompatível com os dados e omissão de variáveis importantes. Encontrar razões que expliquem o fato dos pontos terem um comportamento atípico, com relação aos demais pontos, pode ajudar a entender melhor a relação entre as variáveis explicativas e o fenômeno sob investigação. A remoção automática de pontos extremos não é um procedimento correto e deve ser o último recurso a ser utilizado. Mudanças no modelo, inclusão, eliminação ou mesmo transformação de variáveis explicativas podem ajudar a atenuar o efeito de pontos extremos. O uso de métodos robustos é outra alternativa à exclusão de pontos (Paula, 2004).

2.1.3 Medidas de diagnóstico para *outliers* e ponto de alavanca

2.1.3.1 Matriz H

A matriz H, conhecida como matriz chapéu ou matriz de projeção da solução de mínimos quadrados, foi introduzida por Hoaglin e Welsch (1978). Os elementos da diagonal principal da matriz H são utilizados para identificar pontos de alavanca ou de alto *leverage*. A matriz H é definida por:

$$H = X(X^T X)^{-1} X^T \tag{2.7}$$

sendo X a matriz de variáveis regressoras.

A inspeção dos elementos da matriz H pode revelar pontos extremos em virtude de sua localização no espaço X. Como,

$$\sum_{i=1}^{n} h_{ii} = p,$$

sendo p o número de parâmetros do modelo e n o número de observações. O tamanho médio de um elemento da diagonal da matriz H é $\overline{h} = p/n$. Desta forma, se $h_{ii} > 2(p/n)$, a observação i é um possível ponto de alavanca.

2.1.3.2 Resíduos

Uma forma de identificar possíveis *outliers* é através da análise dos resíduos, uma vez que pontos bem ajustados possuem resíduos pequenos. Os resíduos podem ser definidos como a diferença entre os valores observados $\mathbf{y}$ e os valores ajustados $\widehat{\mathbf{y}}$ (Weisberg, 1985). O vetor de resíduos é dado por:

$$\mathbf{e} = \mathbf{y} - \widehat{\mathbf{y}} \tag{2.8}$$

Substituindo as Eqs.(2.5) e (2.6) em (2.8), o vetor de resíduos pode ser escrito na forma,

$$\begin{aligned}\mathbf{e} &= \mathbf{y} - X(X^TX)^{-1}X^T\mathbf{y} \\ &= [I - X(X^TX)^{-1}X^T]\mathbf{y} \\ &= [I - H]\mathbf{y}.\end{aligned}$$

Assim, a variância dos resíduos pode ser expressa por:

$$Var(\mathbf{e}) = Var[(I - H)\mathbf{y}] = (I - H)\sigma^2,$$

logo $Var(e_i) = (1 - h_{ii})\sigma^2$. Como os resíduos têm variâncias diferentes, com o intuito de permitir uma comparabilidade entre os mesmos, é conveniente expressá-los em forma padronizada. Em (Cook e Weisberg, 1982) são propostos os resíduos studentizados interno e externo, obtidos pela divisão de cada resíduo pelo seu respectivo desvio padrão.

O resíduo studentizado interno é dado por:

$$r_i = \frac{e_i}{s(1 - h_{ii})^{1/2}}, \qquad i = 1, ..., n. \tag{2.9}$$

sendo $s^2 = \sum_{i=1}^{n} \frac{r_i^2}{n-p}$.

O resíduo studentizado externo é expresso por:

$$t_i = \frac{e_i}{s_{(i)}(1-h_{ii})^{1/2}}, \qquad i = 1, ..., n. \tag{2.10}$$

sendo $s^2_{(i)}$ o erro quadrático médio correspondente ao modelo sem a i-ésima observação. Em (Cook e Weisberg, 1982) é demonstrado que

$$s^2_{(i)} = s^2 \left(\frac{n-p-r_i^2}{n-p-1} \right) \tag{2.11}$$

A relação entre t_i e r_i é encontrada substituindo (2.11) em (2.10). Assim, o resíduo studentizado externo,

$$t_i = r_i \left(\frac{n-p-1}{n-p-r_i^2} \right)^{1/2}, \qquad i = 1, ..., n. \tag{2.12}$$

segue uma distribuição central t_{n-p-1}.

2.1.4 Medidas de influência

As medidas de influência são utilizadas para detectar pontos influentes em um determinado conjunto de dados. Um ponto pode não ter o mesmo impacto em todos os aspectos da regressão do modelo, por isso para detectar pontos influentes é necessário estudar os efeitos que estes pontos produzem no ajuste do modelo.

Uma observação é dita influente se ela produzir alterações relevantes no resultado da análise quando a mesma for omitida ou submetida a uma pequena perturbação. A seguir são apresentadas as principais medidas de influência.

2.1.4.1 Distância de Cook

Em (Cook, 1977) é proposta uma medida de influência, denominada distância de Cook, que considera o quadrado da distância entre as estimativas dos coeficientes de regressão $(\widehat{\beta})$ obtidas com todas as n observações e as estimativas dos coeficientes de regressão obtidas omitindo-se a i-ésima observação $(\widehat{\beta}_{(i)})$. Esta medida de influência é dada por:

$$D_i = \frac{(\widehat{\beta}_{(i)} - \widehat{\beta})^T X^T X (\widehat{\beta}_{(i)} - \widehat{\beta})}{ps^2} \qquad , i = 1, ..., n \tag{2.13}$$

sendo p é o número de variáveis e s^2 a variância amostral.

Considerando $\widehat{\mathbf{y}}_{(i)} = X\widehat{\beta}_{(i)}$ a Eq.(2.13) pode ser escrita na forma

$$D_i = \frac{(\widehat{\mathbf{y}}_{(i)} - \widehat{\mathbf{y}})^T (\widehat{\mathbf{y}}_{(i)} - \widehat{\mathbf{y}})}{ps^2} \qquad i = 1, ..., n \tag{2.14}$$

sendo $\widehat{\mathbf{y}}_{(i)}$ os valores ajustados sem a i-ésima observação.

Um grande valor de D_i indica que a observação i é influente nas estimativas dos coeficientes de regressão. Em (Cook e Weisberg, 1999) é sugerido que os pontos com $D_i > 1$ sejam analisados, pois são possíveis pontos influentes.

2.1.4.2 DFBETA

Em (Belsley et al., 1980) é proposta uma medida que indica o quanto cada coeficiente de regressão $\widehat{\beta}_i$ muda, quando a i-ésima observação for omitida. Esta medida é expressa por:

$$DFBeta_i = \frac{(X'X)^{-1}\mathbf{x}_i^T e_i}{(1 - h_{ii})} \qquad i = 1, ..., n \tag{2.15}$$

sendo h_{ii} elemento da diagonal da matriz H , e_i os resíduos e $\mathbf{x}_i$ um vetor linha.

Um grande valor de $DFBeta_i$ indica que a observação i tem influência no coeficiente de regressão. Uma regra razoável para considerar uma observação influente é analisar observações tais que $|DFBeta_i| > 2/\sqrt{n}$.

2.1.4.3 DFFITS

A medida DFFITS é utilizada para investigar a influência da i-ésima observação nos valores ajustados. Esta medida indica o quanto o valor ajustado muda, em unidades de desvio-padrão, se a i-ésima observação for omitida. Esta medida foi proposta por Belsley et al. (1980), sua equação é dada por:

$$DFFITS_i = \left[\frac{h_{ii}}{1-h_{ii}}\right]^{1/2} \frac{e_i}{s_{(i)}\sqrt{1-h_{ii}}} \quad (2.16)$$

sendo $s^2_{(i)}$ o erro quadrático médio correspondente ao modelo sem a i-ésima observação.

Geralmente, observações em que $DFFITS_i \geq 2\left\{\frac{p}{n-p}\right\}^{1/2}$ são possíveis pontos influentes.

Atkinson (1981) propôs uma outra medida de influência que é um aperfeiçoamento da medida DFFITS. Esta medida é dada por:

$$C_i = \frac{(n-p)h_{ii}}{p(1-h_{ii})}^{1/2} |t_i|, \quad (2.17)$$

quando todos os valores de h_{ii} forem iguais, tem-se que $C_i = |t_i|$.

2.1.4.4 Covratio

A medida Covratio é utilizada para examinar como a precisão da estimação muda quando a i-ésima observação é omitida. Esta medida é definida em (Belsley et al., 1980) como:

$$Covratio_i = \frac{det\left\{s^2_{(i)}[X^T_{(i)}X_{(i)}]^{-1}\right\}}{det[s^2(X^TX)^{-1}]} \quad (2.18)$$

Em (Belsley et al., 1980) é sugerido que se $Covratio_i > 1 + (3p/n)$ ou se $Covratio_i < 1$ - (3p/n), então, a i-ésima observação pode ser influente.

2.1.4.5 Função de influência

A função de influência foi desenvolvida por Hampel (1974). Cook e Weisberg (1982) desenvolveram as principais idéias para avaliar a influência em regressão, utilizando a função de influência. Esta função descreve o efeito de uma observação adicional em qualquer ponto x sobre um funcional estatístico T, para uma amostra com distribuição F. Sua equação é expressa por:

$$IF = \lim_{\epsilon \to 0} \frac{T[(1-\epsilon)F + \epsilon\delta_x] - T(F)}{\epsilon} \tag{2.19}$$

quando este limite existe, sendo δ_x função de densidade que associa massa 1 em $x \in \Re^p$.

Escrevendo $g_T(\epsilon) = T[(1-\epsilon)\widehat{F} + \epsilon\delta_x]$, a função de influência pode ser utilizada de modo a refletir os efeitos dos valores observados por meio de duas abordagens diferentes: local e global. A abordagem local é obtida pela derivada

$$EIF = g'(0) = \lim_{\epsilon \to 0} \frac{g_T(\epsilon) - g_T(0)}{\epsilon} \tag{2.20}$$

EIF mede o impacto de uma perturbação infinitesimal em uma observação qualquer da amostra. Considerando $\epsilon = -1/(n-1)$, a abordagem global é dada pela diferença

$$SIF = -(n-1)\left[g_T\left(-\frac{1}{n-1}\right) - g_T(0)\right] \tag{2.21}$$

SIF mede o efeito da omissão de uma observação x_i em uma amostra com n observações (Cook e Weisberg, 1982).

2.1.4.6 Influência Local

O método de influência local foi proposto por Cook (1986), e consiste em avaliar a influência conjunta das observações sob pequenas perturbações (mudanças) nos dados ou no modelo, ao invés da avaliação pela retirada individual ou conjunta de pontos. Este método propõe o estudo de uma pertubação dos componentes do modelo, ou dos dados, utilizando a função de afastamento pela verossimilhança do modelo.

Dado um conjunto de observações, seja $l(\beta)$ o logaritmo da função de verossimilhança (ver Apêndice A) correspondente ao modelo postulado, sendo β um vetor $(p+1) \times 1$ de parâmetros desconhecidos. Perturbações podem ser introduzidas no modelo através de um vetor $\mathbf{u}^T = (u_1, u_2, ..., u_n)$ pertencente a um subconjunto aberto Ω de $\Re^{n+1}$. Geralmente, $\mathbf{u}$ pode ser usado para introduzir

uma menor modificação nas variáveis explicativas ou para perturbar a matriz de covariância no modelo de regressão linear (Galea et al., 1997).

Sendo $\widehat{\beta}$ o estimador de máxima verossimilhança (ver Apêndice A), obtido através de $l(\beta)$, e $\widehat{\beta}_{\mathbf{u}}$ o estimador de máxima verossimilhança, obtido através de $l(\beta/\boldsymbol{u})$, o objetivo é comparar $\widehat{\beta}$ e $\widehat{\beta}_{\mathbf{u}}$ quando $\mathbf{u}$ varia em Ω. Essa comparação é feita através da função sugerida por Cook (1986):

$$LD(\mathbf{u}) = 2[l(\widehat{\beta}) - l(\widehat{\beta}_{\boldsymbol{u}})]. \tag{2.22}$$

A proposta de Cook é estudar o comportamento da função $LD(\mathbf{u})$ numa vizinhança de u_0, que é o ponto em que as duas verossimilhanças são iguais. Para isso considerou a seguinte superfície geométrica:

$$\alpha(\mathbf{u}) = \begin{pmatrix} \mathbf{u} \\ LD(\mathbf{u}) \end{pmatrix}^T. \tag{2.23}$$

A idéia principal de Cook foi de analisar como $\alpha(\mathbf{u})$ desvia-se de seu plano tangente em u_0 , preocupando-se com o comportamento da função $LD(\mathbf{u})$ em torno de u_0. O procedimento consiste em selecionar uma direção unitária d e, então, considerar o gráfico de LD(u_0 + ad) em função de a, em que $a \in \Re$. Como LD(u_0)= 0, o gráfico de LD(u_0 + ad) tem um mínimo local em $a = 0$. A interseção entre a seção normal da superfície ($\alpha(\mathbf{u})$) e o plano tangente é denominada linha projetada. Cada linha projetada pode ser caracterizada pela curvatura normal C_d em torno de $a = 0$. A direção d_{max} corresponde à maior curvatura $C_{d_{max}}$. Os valores de d_{max} contêm a influência local das observações nessa direção particular. O gráfico de d_{max} contra a ordem das observações pode revelar os pontos que sob pequenas perturbações, exercem grande influência sobre $LD(\mathbf{u})$.

A curvatura normal na direção d pode ser expressa da seguinte forma:

$$C_d = 2\left|d^T F d\right|, \tag{2.24}$$

sendo F $=\Delta^T I(\widehat{\beta})^{-1}\Delta$, I$(\widehat{\beta})$ a matriz de informação do modelo postulado e Δ a matriz $(p+1)\times n$ definida por:

$$\Delta = \frac{\partial^2 l(\beta/\boldsymbol{u})}{\partial\beta\partial\mathbf{u}^T}, \tag{2.25}$$

avaliados em $\beta = \widehat{\beta}$ e $u = u_0$.

2.1.5 Técnicas Gráficas

Uma forma de identificar possíveis pontos extremos é por meio de técnicas gráficas. Para os modelos de regressão linear, os gráficos mais utilizados são:

- gráfico dos resíduos studentizados (t_i) em função da ordem das observações, usado para detectar *outliers*;
- gráfico dos elementos da diagonal da matriz H (h_{ii}) em função da ordem das observações, usado para detectar pontos de alavanca;
- gráficos das medidas distância de Cook (D_i) e DFFITS em função da ordem das observações, podem revelar possíveis pontos influentes;
- gráfico de influência local ($|d_{max}|$) em função da ordem das observações, usado para detectar possíveis pontos influentes.

2.1.6 Métodos Robustos

Os métodos robustos utilizam estimadores de modo que o ajuste não seja tão fortemente afetado por *outliers*. A partir deste ajuste robusto, *outliers* podem ser identificados através dos resíduos.

O método de regressão linear mais utilizado é o método de minímos quadrados, porém este método possui pressupostos que devem ser observados. Na prática, não raramente, estes pressupostos são violados. Os métodos que nao são afetados em relação aos desvios destes pressupostos são chamados de métodos robustos.

Os estimadores robustos foram introduzidos por Huber (1973), desde então inúmeros trabalhos foram publicados no assunto. Referências importantes neste

assunto são, dentre outras, os livros de Rousseeuw e Leroy (1987), Atkinson e Riani (2000), Maronna et al. (2006) e Andersen (2008).

A seguir são apresentados os principais estimadores robustos.

2.1.6.1 M-estimadores

Os M-estimadores foram propostos por Huber (1973). Estes estimadores são uma generalização do estimador de máxima verossimilhança (ver apêndice A) e baseiam-se na minimização da seguinte função de dispersão dos resíduos:

$$min \sum_{i=1}^{n} \rho(e_i), \tag{2.26}$$

sendo ρ uma função de peso e e_i os resíduos. Este método é robusto para *outliers*, porém não é robusto para pontos de alavanca.

2.1.6.2 Estimador LMS(*Least Median Squares*)

Em (Rousseeuw, 1984) foi proposto um estimador com base na menor mediana dos quadrados dos resíduos. As estimativas são obtidas minimizando a função

$$minM(e_i^2), \tag{2.27}$$

em que M denota a mediana. O estimador LMS é obtido substituindo a soma dos quadrados dos resíduos pela mediana dos quadrados dos resíduos, o que resulta em um estimador mais resistente aos *outliers*.

2.1.6.3 Estimador LTS(*Least Trimmed Squares*)

O estimador LTS foi proposto por Rousseeuw e Yohai (1984). Este método consiste na minimização da seguinte função:

$$min \sum_{i=1}^{q} (e_i)^2, \tag{2.28}$$

sendo $q = (n/2) + 1$ e n o número de observações.

Segundo Andersen (2008), este estimador é altamente resistente em relação aos outliers na variável resposta e nas variáveis explicativas, porém tem baixa eficiência.

2.1.6.4 S-estimadores

Os S-estimadores foram introduzido por Rousseeuw e Yohai (1984), com o objetivo de melhorar a eficiência do estimador LTS. Estes métodos baseiam-se na minimização da seguinte função:

$$\frac{1}{n}\sum_{i=1}^{n}\rho\left(\frac{e_i}{\hat{\sigma}_e}\right), \tag{2.29}$$

em que $\hat{\sigma}_e$ é o desvio padrão dos resíduos e ρ uma função de peso.

Este estimador é robusto em relação aos outliers na variável resposta e nas variáveis explicativas.

2.1.6.5 MM-estimadores

Os MM-estimadores foram introduzidos por Yohai (1987). Esta classe de estimadores marcou um grande avanço dos estimadores robustos. O nome MM-estimadores refere-se ao fato de que mais de um M-estimador ser utilizado para calcular as estimativas finais. Estes estimadores são definidos por um procedimento em três estágios:

1. obtém-se as estimativas iniciais para os parâmetros $\beta^{(0)}$ e os correspondentes resíduos, utilizando um estimador com baixa eficiência , por exemplo LTS ;

2. calcula-se um S-estimador a partir dos resíduos obtidos no primeiro estágio;

3. os resíduos obtidos no primeiro e no segundo estágios são usados para determinar um M-estimador de regressão, baseado em uma função ponde-

rada

$$\sum_{i=1}^{n} w_i \left(\frac{e_i^{(1)}}{\hat{\sigma}_e} \right) x_i = 0,$$

em que w_i são os pesos.

2.1.7 Considerações finais

Nesta seção, foram apresentadas as principais técnicas para detectar pontos extremos em um conjunto de dados. Mesmo que estas técnicas possam identificar satisfatoriamente os pontos extremos a maior parte do tempo, existem alguns efeitos em que estas técnicas podem falhar em detectá-los. Dentre os efeitos que podem surgir quando se tem múltiplos pontos influentes em um conjunto de dados, está o "mascaramento".

A identificação de grupos de pontos influentes ou influência conjunta, geralmente, é feita por meio de um procedimento combinatório, o que dependendo do tamanho do conjunto de dados, pode tornar-se inviável.

2.2 Função de Sensibilidade de Casos

Critchley (1989) propôs uma abordagem para a função de influência de Hampel (1974), denominada *Case Sensitivity Function*, em português Função de Sensibilidade de Casos. A abordagem de Critchley é uma generalização da função de influência para casos múltiplos. A função de sensibilidade de casos é representada por $g_T(\epsilon)$, onde ϵ é um vetor de perturbações e T é um funcional estatístico. Nesta abordagem, as perturbações são probabilidades associadas a cada observação do conjunto de dados.

A função de sensibilidade de casos possibilita a identificação de múltiplos pontos influentes, mesmo na presença de "mascaramento"(Biazi, 1996).

Em (Biazi, 1996) é explorada a função $g_T(\epsilon)$ para diferentes funcionais estatísticos, os quais permitem monitorar o efeito das perturbações em diferentes aspectos da análise. Neste trabalho é utilizada a função de sensibilidade de casos para a distância de Cook.

2.2.1 Forma geral da função $g_T(\epsilon)$

A função de influência descrita na Seção 2.4.5, é utilizada para monitorar a influência de observações individuais nas estimativas do modelo. Sua equação é expressa por:

$$g_T(\epsilon) = T[(1-\epsilon)\widehat{F} + \epsilon\delta_{xi}]. \tag{2.30}$$

A função de sensibilidade de casos é obtida substituindo a perturbação ϵ na Eq. (2.30) pelo somatório de n perturbações. Assim, a forma geral da função $g_T(\epsilon)$ é dada por:

$$g_T(\epsilon) = T[(1-\sum_{i=1}^{n}\epsilon_i)\widehat{F} + \sum_{i=1}^{n}\epsilon_i\delta_{xi}], \tag{2.31}$$

sendo:

n - número de observações; ϵ_i - i-ésima perturbação;

δ_{xi} - função de densidade que associa massa 1 em $x \in \Re^n$;

$\widehat{F}$ - função de densidade empírica de uma amostra aleatória $x_1, x_2, ..., x_n$.

Escrevendo $\widehat{F} = \frac{1}{n}\sum_{i=1}^{n}\delta_{xi}$ e efetuando alguns cálculos algébricos, a Eq.(2.31) pode ser escrita na forma,

$$g_T(\epsilon) = T[\sum_{i=1}^{n} p_i(\epsilon)\delta_{xi}], \tag{2.32}$$

sendo $p_i(\epsilon) = \frac{1}{n} + \epsilon_i - \overline{\epsilon}$.

2.2.2 Aspectos metodológicos

Quando um modelo é ajustado a um conjunto de dados, espera-se que as estimativas obtidas a partir do modelo proposto sejam resistentes a pequenas perturbações no modelo ou nos dados. Desta forma, se pequenas perturbações produzem efeitos significativos, pode ser indício de que o modelo não está bem ajustado ou que existem observações com grande influência nas estimativas do

modelo. Com base neste pressuposto foi desenvolvida a metodologia da função de sensibilidade de casos.

Esta metodologia consiste em inserir pequenas perturbações (ϵ) no modelo e avaliar o impacto causado por estas perturbações, comparando $g_T(\epsilon)$ com $g_T(0)$, sendo $g_T(\epsilon)$ o valor da função de sensibilidade com perturbação e $g_T(0)$ o valor da função sem pertubação. Se a diferença entre $g_T(\epsilon)$ e $g_T(0)$ for grande, os dados devem ser analisados, e certamente haverá observações influentes.

O objetivo é maximizar $|g_T(\epsilon) - g_T(0)|$ sujeita a uma região determinada pelas probabilidades:

$$0 \leq p_i \leq \frac{1}{n-k} \qquad para \qquad 1 \leq i \leq n \tag{2.33}$$

sendo $p_i = \frac{1}{n} + \epsilon_i - \bar{\epsilon}$, $\sum_{i=1}^{n} p_i = 1$, n o número de observações e k o número de observações influentes.

O vetor de perturbações $\epsilon = (\epsilon_1, ..., \epsilon_n)^T$ que maximiza a distância $|g_T(\epsilon) - g_T(0)|$ contém os pesos associados a cada ponto do conjunto de dados. Pontos com pesos negativos são considerados influentes no ajuste do modelo.

2.2.3 Função $g_T(\epsilon)$ para a distância de Cook

A função de sensibilidade de casos para a distância de Cook permite identificar pontos influentes, considerando todos os aspectos do ajuste do modelo.

Em (Cook e Weisberg, 1982) é feita uma generalização da distância de Cook para a omissão de subconjuntos de pontos, sendo sua equação expressa por:

$$CD_m = \frac{(\widehat{\beta}_{(m)} - \widehat{\beta})' X' X (\widehat{\beta}_{(m)} - \widehat{\beta})}{ps^2} \tag{2.34}$$

em que m representa o subconjunto de pontos a ser omitido.

A função $g_T(\epsilon)$ para a distância de Cook é definida em (Biazi, 1996) como:

$$g_{Cook}(\epsilon) = \frac{(g_{\widehat{\beta}}(\epsilon) - g_{\widehat{\beta}}(0))' X' X (g_{\widehat{\beta}}(\epsilon) - g_{\widehat{\beta}}(0))}{ps^2} \tag{2.35}$$

sendo:

$g_{\widehat{\beta}}(0) = (X'X)^{-1}X'\mathbf{y}$

$g_{\widehat{\beta}}(\epsilon) = (X'EX)^{-1}X'E\mathbf{y}$

E = diag($\frac{1}{n} + \epsilon_i - \overline{\epsilon}$)

p - número de variáveis

s^2 - variância amostral.

Neste caso, as perturbações ϵ_i associadas a cada observação do conjunto de dados são inseridas no modelo por meio de E (matriz diagonal de probabilidades p_i).

2.2.4 Considerações finais

O problema de identificação de pontos influentes, tratado no presente trabalho, é um problema de otimização, especificamente de maximização da função objetivo. A função de sensibilidade de casos para o funcional estatístico distância de Cook é uma função complexa, com vários máximos locais. A solução ótima, neste caso, é um vetor de perturbações ϵ que maximiza a função $Fo = |g_T(\epsilon) - g_T(0)|$. Tal função tem comportamento multimodal. A metaheurística PSO, por sua característica evolutiva, tem sido adequada no tratamento de problemas multimodais.

Capítulo 3

Otimização por Enxame de Partículas

3.1 Introdução

O método de otimização por enxame de partículas (*Particle Swarm Optimization* -PSO) é uma técnica de computação estocástica baseada em dinâmica de populações. Desenvolvido por Kennedy e Eberhart (1995), este método consiste na otimização de uma função objetivo através da troca de informações entre indivíduos (partículas) de uma população (enxame).

No algoritmo PSO, uma partícula, candidata à solução do problema, corresponde a um ponto no espaço de busca. Cada partícula, representada como um vetor de dimensão igual a dimensão do espaço de busca, é inicializada com posição e velocidade aleatórias, guardando consigo a informação sobre a melhor posição por ela já visitada. O algoritmo preserva também a melhor posição já registrada pelo enxame. A cada iteração os indivíduos da população são avaliados e as melhores posições de cada partícula e do enxame são atualizadas. As posições de cada indivíduo são atualizadas levando em conta a influência da melhor posição já atingida pelo enxame (influência social) e a melhor posição já atingida pelo próprio indivíduo (decisão individual). Desta forma, os indivíduos do enxame movem-se pelo espaço de busca procurando atingir o máximo ou mínimo global

com base na decisão individual e na influência social (Medeiros, 2005).

Apesar de ser relativamente recente, a técnica metaheurística PSO tem sido aplicada em um vasto número de problemas, como por exemplo: otimização de *layouts* industriais (Muller, 2007), engenharia nuclear (Medeiros, 2005), na estimação de parâmetros do modelo de secagem (Gomes, 2008). Em (Poli, 2008) são apresentadas diversas aplicações da otimização por enxame de partículas.

3.2 Origem e Etimologia

O método de otimização por enxame de partículas foi inspirado no comportamento cooperativo de várias espécies de animais: revoadas de pássaros, enxames de abelhas, cardumes de peixes, etc.

Alguns cientistas desenvolveram simulações computacionais baseadas em várias interpretações destes comportamentos. Reynolds (1987) procurou estabelecer regras básicas do comportamento individual que justifiquem o comportamento do grupo, concluindo que o movimento simulado da revoada de pássaros é o resultado da interação de comportamentos relativamente simples do movimento de cada um dos pássaros. Heppner e Grenander (1990), interessados em descobrir as regras que permitem a sincronização de um grande número de pássaros, dispersando e reagrupando novamente, realizaram simulações no plano xy , com bandos de 15 a 30 pássaros. Nas simulações, os pássaros sobrevoavam a área delimitada e com poucas iterações o enxame inteiro aglomera-se dentro de uma região que representava o local ótimo.

Kennedy e Eberhart (1995) inspiraram-se no trabalho de Wilson (1995), no qual defendia que os membros individuais de um bando podem-se beneficiar com as descobertas e experiências de cada um dos membros durante a busca por alimentos. Esta afirmação sugere que o compartilhamento social de informações pode oferecer uma vantagem evolucionária: esta hipótese foi fundamental para o desenvolvimento do método de otimização por enxame de partículas.

O termo *swarm* foi usado em (Millonas, 1994), onde é analisada a inteligência

coletiva dos enxames. Segundo este autor, algumas propriedades da inteligência coletiva são:

- proximidade: necessita de espaço simples e pequeno tempo computacional;
- qualidade: resposta a fatores de qualidade no ambiente;
- resposta diversa: não estar em um subconjunto restrito de soluções;
- estabilidade: pode manter modos de comportamento quando os ambientes mudam;
- adaptabilidade: pode mudar modos de comportamento quando a adaptação for necessária.

O conceito e o paradigma da Otimização por Enxame de Partículas obedecem a todos os cinco princípios. A população responde a fatores de qualidade *pbest* (melhor posição da partícula) e *gbest* (melhor posição do enxame). A atribuição das respostas entre *pbest* e *gbest* garante uma diversidade de respostas. A população muda seu estado (modo de comportamento) apenas quando *gbest* muda, assim respeitando o princípio de estabilidade. A população é adaptativa porque ela faz-se mudar quando *gbest* muda (Kennedy e Eberhart, 1995).

O algoritmo PSO é baseado em um modelo simplificado da teoria de enxames, através do qual cada indivíduo faz uso de sua própria experiência e da experiência do grupo para encontrar o ninho ou alimento. Em analogia, a área sobrevoada é equivalente ao espaço de busca do problema e encontrar o local com comida ou o ninho corresponde a encontrar o ótimo (Prado e Saramago, 2005).

3.3 Algoritmo PSO

No algoritmo PSO, cada partícula, tratada como um ponto no espaço D-dimensional, representa uma solução potencial para um problema, ajustando seu "vôo" com base em sua própria experiência e na experiência do grupo . A cada iteração, a velocidade é atualizada, conforme Eq.(3.1). A nova posição da partícula

é determinada pela soma da sua posição atual e a nova velocidade, de acordo com a Eq.(3.2) (Eberhart et al., 1996).

$$v_i^{k+1} = v_i^k + c_1.r_1(pbest_i - x_i^k) + c_2.r_2(gbest - x_i^k) \tag{3.1}$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{3.2}$$

sendo v_i^k a velocidade atual da partícula i, c_1 e c_2 coeficientes de aceleração, r_1 e r_2 números aleatórios uniformemente distribuídos entre [0 , 1], $pbest_i$ a melhor posição já alcançada pela partícula i e $gbest$ a melhor posição encontrada por uma das n partículas do enxame.

A equação de atualização da velocidade corresponde à soma de três termos distintos: o primeiro refere-se à inércia da partícula; o segundo é um termo cognitivo, relativo à atração da partícula pelo melhor ponto que esta já encontrou; e o terceiro é um termo social , que representa a colaboração entre as partículas.

O algoritmo utiliza vetores de tamanho n, sendo n é a dimensão do problema, para armazenar as velocidades, as posições e as melhores posições já alcançadas por cada uma das partículas. Assim:

- $\mathbf{v}_i = (v_1, v_2, ..., v_n)$: velocidade atual da partícula i;
- $\mathbf{x}_i = (x_1, x_2, ..., x_n)$: posição atual da partícula i;
- $\mathbf{p}_i = (p_1, p_2, ..., p_n)$: melhor posição alcançada pela partícula i, também chamada de $pbest_i$.

Os passos para a implementação do algoritmo PSO básico são os seguintes (Eberhart et al., 1996):

1. Inicializar a população de partículas com posições e velocidades aleatórias no espaço D-dimensional;

2. Avaliar a aptidão de cada uma das partículas de acordo com a função de avaliação;

3. Comparar o valor obtido da partícula i com *pbest*. Se o valor for melhor, atualizar *pbest* com o novo valor;

4. Comparar o valor obtido com o melhor valor global *gbest*. Se for melhor, atualizar *gbest* com o novo valor;

5. Atualizar a velocidade da partícula de acordo com a Eq.(3.1);

6. Atualizar a posição da partícula de acordo com a Eq.(3.2);

7. Repetir os passos 2-6 até que algum critério de parada seja alcançado.

Na Fig. 3.1 está representado um enxame de partículas gerado aleatoriamente dentro do espaço de busca do problema.

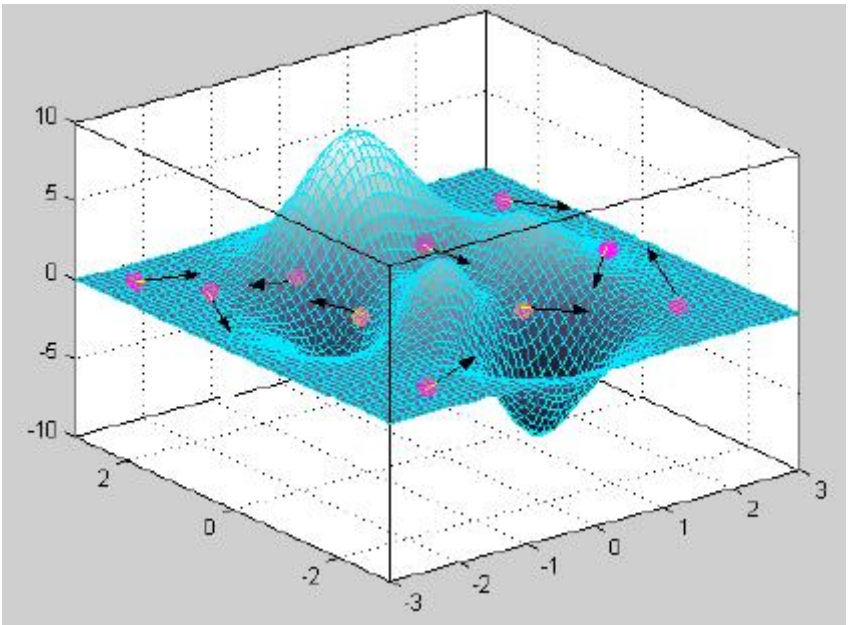

Figura 3.1: Ilustração de um exame de partículas (Medeiros,2005)

Na Fig. 3.2 é apresentado um exemplo de atualização da velocidade e da posição de duas partículas no espaço bidimensional, onde 1 e 2 são posições atuais e o círculo cinza representa a próxima posição. As partículas deslocam-se em direção a melhor posição do enxame (*gbest*) sob a ação de três vetores. Os vetores v_1 e v_2 representam a inércia. Os vetores $(pbest_1 - x_1)$ e $(pbest_2 - x_2)$

representam o aspecto cognitivo, atraindo a partícula para a sua melhor posição ocupada até o momento. Os vetores $(gbest - x_1)$ e $(gbest - x_2)$ representam o aspecto social, que atrai as partículas para a melhor posição do grupo.

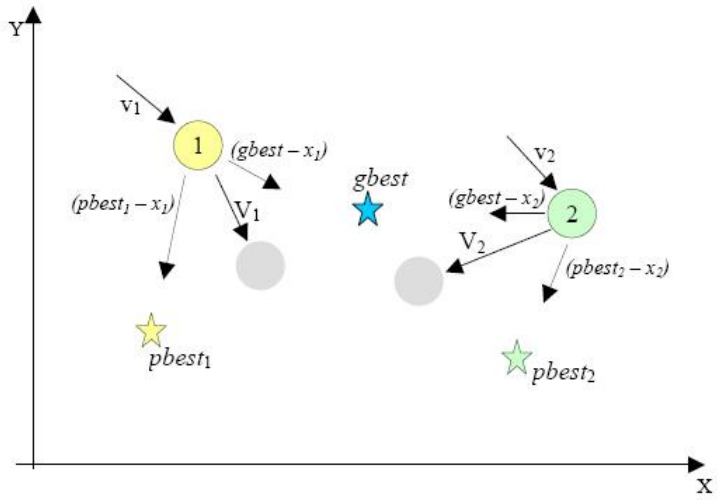

Figura 3.2: Atualização de duas partículas (Müller, 2007)

A Fig. 3.3 ilustra um caso onde o conjunto de partículas convergiu para a região próxima de um ótimo, após sucessivas iterações.

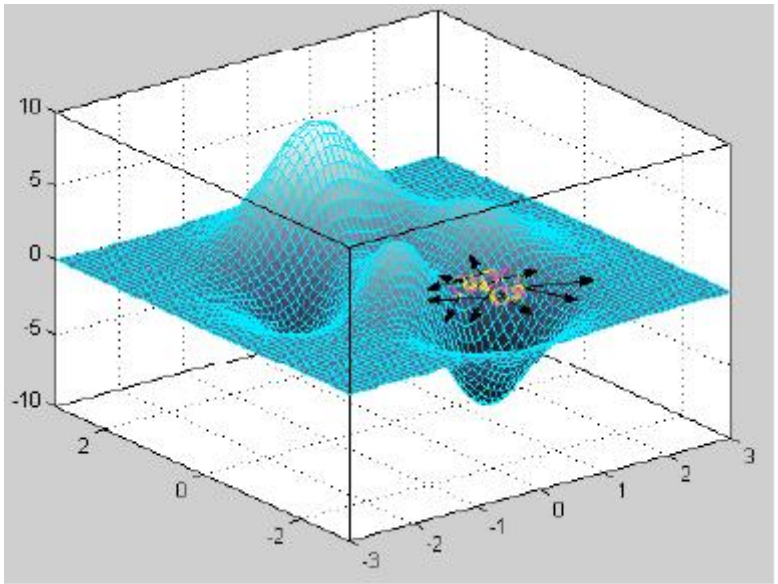

Figura 3.3: Ilustração de convergência do PSO (Medeiros,2005)

3.3.1 Parâmetros de Controle do Algoritmo PSO

O algoritmo PSO depende de parâmetros que devem ser ajustados a cada problema a ser otimizado.

Um desses parâmetros é o tamanho da população. Segundo Poli et al. (2007), o número de partículas presentes na população é determinado empiricamente, com base na dimensionalidade e percepção de dificuldade de um problema. Valores entre 20 e 50 são bastante comuns.

Um número maior de partículas requer menos iterações para encontrar a solução do problema. No entanto, quanto maior a população, mais avaliações da função objetivo são necessárias. Assim, o número de partículas deve ser o menor possível, capaz de garantir a varredura do espaço de busca, sem elevar demasiadamente o custo computacional.

Os parâmetros c_1 e c_2 (Eq.3.1), denominados coeficientes de aceleração, determinam a magnitude das forças em direção a *pbest* e *gbest*, respectivamente. Estes parâmetros interferem significativamente no desempenho do algoritmo. Quando bem determinados, podem reduzir a possibilidade de convergência prematura.

Em (Shi e Eberhart, 1998b) é sugerido que sejam adotados $c_1 = c_2 = 2$, de forma a manter o equilíbrio entre o aprendizado individual de cada partícula e o aprendizado coletivo, no comportamento atual da partícula.

Com o objetivo de diversificar o comportamento da partícula, Ratnaweera et al. (2004) desenvolveram as expressões (3.3) e (3.4) com as quais os parâmetros de confiança c_1 e c_2 podem ser recalculados após cada uma das iterações do algoritmo:

$$c_1^{it} = (c_{1f} - c_{1i})\frac{it}{N} + c_{1i}, \tag{3.3}$$

$$c_2^{it} = (c_{2f} - c_{2i})\frac{it}{N} + c_{2i}, \tag{3.4}$$

nas quais c_{1i} é o valor inicial para o parâmetro de confiança cognitivo, c_{1f} é o valor final para o parâmetro de confiança cognitivo, c_{2i} é o valor inicial para o parâmetro de confiança social, c_{2f} é o valor final para o parâmetro de confiança

social, it é o número da iteração atual e N o número máximo de iterações.

Os parâmetros r_1 e r_2 permitem manter a diversidade da população e são distribuídos uniformemente no intervalo [0 , 1].

3.4 Principais alterações no algoritmo PSO

A partir da formulação original, desenvolvida por Kennedy e Eberhart (1995), foram propostas alterações no método de otimização por enxame de partículas, com o objetivo de melhorar sua performance.

3.4.1 Peso inercial (w)

Em (Shi e Eberhart, 1998b) é introduzido um novo parâmetro no algoritmo PSO, denominado peso inercial (w), com o objetivo de melhorar a sua taxa de convergência. Tal parâmetro tem por objetivo equilibrar a exploração global e local. Valores superiores à unidade irão resultar em uma maior exploração global, ao passo que valores muito menores que a unidade, numa maior exploração local. A partir da Eq.(3.1) obtém-se a nova equação de atualização da velocidade:

$$v_i^{k+1} = w.v_i^k + c_1.r_1(pbest_i - x_i^k) + c_2.r_2(gbest - x_i^k) \tag{3.5}$$

Segundo Shi e Eberhart (1998b), o peso inercial escolhido no intervalo [0,8 , 1,2] tem em média, um melhor desempenho, ou seja, resulta em uma convergência mais rápida.

Em (Shi e Eberhart, 1998a) é proposto que o peso inercial inicie com um valor relativamente alto e vá decrescendo a cada iteração do algoritmo. Desta forma, é possível encontrar soluções mais refinadas com um menor número de iterações. Os autores realizaram experimentos com o peso inercial variando linearmente entre 0,9 e 0,4, obtendo bons resultados.

Segundo Poli et al. (2007) muitos problemas têm sido resolvidos mais satisfatoriamente quando é empregado o peso inercial decrescente.

3.4.2 Fator de constrição

Segundo Clerc e Kennedy (2002), a inclusão de um fator de constrição (K) ajuda a assegurar a convergência do algoritmo, controlando a magnitude das velocidades. O fator K desempenha um papel semelhante à velocidade máxima (v_{max}), limitando a velocidade da partícula, visto que um valor muito alto pode fazer com que a partícula ultrapasse uma posição ótima. A equação da atualização usando este coeficiente é a seguinte:

$$v_i^{k+1} = K[w.v_i^k + c_1.r_1(pbest_i - x_i^k) + c_2.r_2(gbest - x_i^k)] \tag{3.6}$$

sendo:

$$K = \frac{2}{\left|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}\right|} \tag{3.7}$$

sendo $\varphi = c_1 + c_2$, com $\varphi > 4$.

3.5 Considerações Finais

Neste capítulo, é apresentada uma revisão da metaheurística Otimização por Enxame de Partículas. Cabe ressaltar que o algoritmo PSO possui parâmetros intrínsecos a seu funcionamento, que devem ser ajustados a cada problema tratado. Tais parâmetros influenciam diretamente no desempenho do mesmo, especialmente no tratamento de problemas mais complexos. Tal é o caso de otimização de funções multimodais.

Capítulo 4

Aplicação da metaheurística PSO na identificação de pontos influentes

Neste capítulo, é descrita a aplicação da metaheurística PSO ao problema da identificação de pontos influentes, explicitando e justificando as adequações necessárias, de forma a atender às especificidades do problema tratado.

4.1 Descrição do algoritmo PSO adaptado ao problema

No algoritmo PSO adaptado ao problema, cada partícula i representa um vetor de perturbações $\epsilon = (\epsilon_1, ..., \epsilon_n)^T$. O problema consiste em encontrar o vetor de perturbações (ϵ) que maximiza a distância $|g_T(\epsilon) - g_T(0)|$.

A técnica PSO pode ser estruturada a partir das matrizes X, V e P de dimensões $m \times n$, onde m é o número de partículas e n a dimensão do problema, e um vetor G de dimensão n. As posições de cada partícula encontram-se dispostas através das linhas da matriz X e as velocidades, através das linhas da matriz V. As linhas da matriz P registram a melhor posição de cada uma das partículas ($pbest_i$) até a presente iteração. O vetor G registra a melhor posição dentre todas as partículas ($gbest$), até a presente iteração. Estas matrizes e vetores são atualizadas a cada iteração.

Inicialmente, atribuiem-se valores para a posição e velocidade associadas a cada partícula ou perturbação(ϵ), de maneira aleatória. Ao se aplicar a função de avaliação a cada uma das partículas, estes valores irão formar, na primeira iteração o vetor *pbest*. Em seguida, são calculadas as velocidades da próxima iteração. Uma nova matriz posição é então calculada e, através desta, novos cálculos da função de avaliação são realizados. Sempre que *pbest* encontra um valor maior que *gbest*, este é atualizado. Quando algum critério de parada imposto pelo algoritmo é alcançado, G representará a melhor solução do problema, eventualmente, a solução ótima.

Neste trabalho, foram adotados os seguintes critérios de parada: o número máximo de iterações (it_{max}) e o número máximo de iterações sem melhora, que neste caso foi fixado em 10% do número máximo de iterações ($it_{sm} = 0,1 * it_{max}$). Caso o algoritmo atinja o número máximo de iterações sem melhora da solução ou o número máximo de iterações, o programa finaliza sua execução.

Os valores de ϵ_i apresentam restrições de lateralidade que devem ser atendidas pelo algoritmo. Desta forma, as partículas foram geradas aleatoriamente de acordo com a equação:

$$\epsilon_0 = \epsilon_{min} + r(\epsilon_{max} - \epsilon_{min}), \tag{4.1}$$

em que ϵ_{min} e ϵ_{max} são extremos do domínio expresso por,

$$\frac{-1}{n} < \epsilon_i < \frac{k}{(n-k)n}, \tag{4.2}$$

sendo n o número de observações em um conjunto de dados e k o número de observações influentes. Como "a priori" não se sabe qual é o número de pontos influentes presentes no conjunto de dados, considerou-se k igual a 42% do número de observações (n).

Para atender as restrições de lateralidade, no algoritmo PSO são utilizadas duas equações para a atualização da velocidade de cada partícula. Durante a atualização da posição de cada uma das partículas, a cada iteração, se a nova posição

estiver dentro do domínio do problema, a atualização da velocidade será determinada pela Eq.(3.1). Caso a nova posição esteja fora do domínio a atualização da velocidade é dada pela equação:

$$v_i^{k+1} = c_1.r_1(pbest_i - x_i^k) + c_2.r_2(gbest - x_i^k) \tag{4.3}$$

A Equação (4.3) não leva em consideração o valor da velocidade da iteração anterior para o cálculo da nova velocidade. Testes realizados em (Gomes, 2008) mostram que a velocidade, calculada por meio da Eq.(4.3), atualiza a posição corrente dentro do espaço de busca do problema, fazendo com que a partícula volte para a região do domínio do problema.

O algoritmo PSO, adaptado ao problema tratado neste trabalho, pode ser descrito pelos seguintes passos:

Passo 1: Inicializar a população de partículas com posições e velocidades aleatórias, a partir das equações:
$\epsilon_0 = \epsilon_{min} + r_1(\epsilon_{max} - \epsilon_{min})$
$v_0 = \epsilon_{min} + r_2(\epsilon_{max} - \epsilon_{min})$;
Passo 2: Ler os dados de entrada;
Passo 3: Avaliar a aptidão de cada uma das partículas de acordo com a equação:
$Fo = |g_{Cook}(\epsilon) - g_{Cook}(0)|$;
Passo 4: Determinar a melhor posição da partícula, $pbest_i$;
Passo 5: Determinar a melhor posição do enxame, $gbest$;
Passo 6: Se $Fo(\epsilon_i) > Fo(pbest_i)$, atualize $pbest_i$ com a posição corrente;
Passo 7: Se $Fo(pbest_i) > Fo(gbest)$, atualize $gbest$ com $pbest_i$;
Passo 8: Atualizar a velocidade de cada uma das partículas de acordo com as equações:
$v_i^{k+1} = w.v_i^k + c_1.r_1(pbest_i - x_i^k) + c_2.r_2(gbest - x_i^k)$
$v_i^{k+1} = c_1.r_1(pbest_i - x_i^k) + c_2.r_2(gbest - x_i^k)$ (Caso o domínio seja violado);
Passo 9: Atualizar a posição de cada uma das partículas conforme equação:
$x_i^{k+1} = x_i^k + v_i^{k+1}$;

Passo 10: Repetir os passos 2-9 até que um dos critérios de parada, o número de iterações sem melhora ou o número máximo de iterações, seja alcançado.

A implementação do algoritmo PSO foi realizada utilizando o software Matlab versão 7.6, em um computador Core 2 Duo com 2GB de memória RAM, HD 160 GB e sistema operacional Windows XP.

4.2 Ajuste dos parâmetros do PSO

Os parâmetros peso inercial (w), coeficientes de aceleração (c_1 e c_2) e o número de partículas presentes no enxame devem ser ajustados a cada problema a ser otimizado. Estes parâmetros influenciam diretamente no desempenho do algoritmo PSO. Quando estes parâmetros são bem ajustados ao problema, podem fornecer ao algoritmo estabilidade, mantendo o equilíbrio entre a exploração global e a intensificação da busca por soluções mais refinadas.

Os testes para o ajuste dos parâmetros w, c_1, e c_2 foram realizados utilizando um conjunto com quarenta dados simulados, o qual encontra-se apresentado na Tabela 4.1.

Tabela 4.1: Dados simulados com 40 pontos

Nº	Y	X	Nº.	Y	X	Nº.	Y	X
1	4,53535	2,64803	15	3,92878	2,12236	29	4,46251	7,09551
2	6,07065	3,80588	16	3,19128	1,45014	30	4,94675	7,15447
3	3,10827	1,28300	17	3,98016	2,08249	31	2,84652	6,70342
4	4,53078	2,87092	18	5,11618	3,35293	32	2,37480	6,86149
5	3,65160	1,93596	19	4,85078	2,93950	33	2,09322	6,48174
6	3,96597	1,80670	20	3,80726	1,84104	34	2,52804	6,90263
7	4,59489	2,64463	21	5,32465	3,24116	35	1,90945	6,52393
8	4,52457	2,69016	22	3,84507	1,86512	36	1,95644	7,18616
9	6,13131	3,98140	23	4,53330	2,41656	37	1,80580	6,82717
10	3,89639	1,67685	24	5,23911	3,29560	38	2,61904	7,03319
11	2,80804	1,25474	25	4,46586	6,98095	39	1,90778	6,64355
12	5,99618	3,98300	26	4,75674	6,98513	40	1,86854	6,16337
13	3,25335	1,31647	27	5,42457	6,88079	-	-	-
14	4,70990	2,95835	28	5,38646	7,03793	-	-	-

4.2.1 Ajuste do peso inercial

No ajuste do peso inercial considerou-se $c_1 = c_2 = 2$ e o número de partículas igual a 30, para um conjunto com 40 observações. A princípio foi realizada uma bateria de testes, para os seguintes valores de w: 0,6; 0,7; 0,8; 0,9; 1,0; 1,1 e 1,2. Segundo Shi e Eberhart (1998b) o peso inercial escolhido no intervalo [0,8 , 1,2] deve, em geral, permitir uma convergência mais veloz. Verificou-se que para os valores de w iguais a 0,6 e 0,7, houve uma convergência prematura do algoritmo. Para os valores de w iguais a 1,0, 1,1 e 1,2 não houve um refinamento da solução, pois as partículas tendem sempre a buscas globais. Constatou-se que o valor de w igual a 0,8 apresentou melhor resultado, mas em algumas execuções do programa, houve convergência para ótimos locais.

Em (Shi e Eberhart, 1998a) foram realizados experimentos com o peso inercial variando linearmente entre 0,9 e 0,4, obtendo-se bons resultados. Com base nestes autores, foram realizados testes com o peso inercial decrescente. Verificou-se que o algoritmo PSO, para o problema tratado, com peso inercial inicial (w_i) igual a 0,9 e o peso inercial final (w_f) igual a 0,4 apresentou melhor desempenho. Foi possível encontrar soluções refinadas nas execuções do programa.

As tabelas com os resultados obtidos encontram-se apresentadas no Apêndice B.

4.2.2 Ajuste dos parâmetros c_1 e c_2

Inicialmente, os parâmetros c_1 e c_2 foram fixos de acordo com os valores sugeridos por Shi e Eberhart (1998b). Os autores sugerem que sejam adotados $c_1 = c_2 = 2$, de forma a manter o equilíbrio entre as partes cognitiva e social do comportamento da partícula.

Em (Bergh, 2001) é feita uma análise da convergência do algoritmo PSO, e é mostrado que a região de convergência do algoritmo é limitada por $c_1 + c_2 = 4$. Desta forma, após o ajuste do parâmetro w, diversos valores de c_1 e c_2 foram testados, de modo que $c_1 + c_2 = 4$. Por exemplo, iniciar c_1 =0,05 e c_2 =3,95 e incrementar c_1 e decrementar c_2 em 0,05 até que c_1 =3,95 e c_2= 0,05.

Com base nos testes realizados, verificou-se que os valores de $c_1 = 1{,}95$ e $c_2 = 2{,}05$ apresentaram melhores resultados, pois a média das soluções encontrada em 30 execuções do programa aproximou-se da melhor solução encontrada para o problema, com erro de 0,014. Assim, foram adotados neste trabalho os parâmetros $c_1 = 1{,}95$ e $c_2 = 2{,}05$.

4.2.3 Ajuste do número de partículas

Segundo Poli et al. (2007) o número de partículas presentes na população é determinado empiricamente, com base na dimensão do problema. Desta forma, o número de partículas variou de acordo com o tamanho do conjunto de dados. Para o conjunto com 40 observações, foram realizados testes com os seguintes números de partículas: 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55 e 60. Constatou-se que 30 partículas foram suficientes para garantir uma ampla varredura no espaço de busca. Analogamente, foram realizados os testes para os outros conjuntos de dados. Assim, foram utilizadas 30, 100, 150, 230 e 400 partículas para conjuntos com 40, 200, 500, 506 e 1000 observações, respectivamente.

4.3 Testes para a distribuição dos números aleatórios

Os números aleatórios r_1 e r_2, presentes na equação de atualização da velocidade, têm a função de manter a diversidade da população. Estes números são gerados aleatoriamente no intervalo [0 , 1], segundo uma distribuição uniforme.

No presente trabalho, foram testados os números aleatórios gerados por intermédio de uma distribuição normal de Gauss, com o objetivo de verificar a influência dos números aleatórios r_1 e r_2 no desempenho do algoritmo.

Os testes foram realizados, utilizando o conjunto de dados com 40 observações. Os parâmetros de controle do algoritmo PSO foram os seguintes: c_1= 1.95, c_2= 2.05, $w_i = 0{,}9$, w_f= 0,4 e o número de partículas igual a 30.

A Tabela 4.2 apresenta os resultados obtidos com r_1 e r_2 gerados por intermédio de uma distribuição uniforme. A coluna *fGBest* corresponde as soluções

encontradas nas execuções do algoritmo.

A Tabela 4.3 apresenta os resultados encontrados nas execuções do programa, utilizando a distribuição normal de Gauss na geração dos números aleatórios r_1 e r_2, no intervalo [0 , 1].

Com base nos resultados obtidos, observa-se que para os números aleatórios gerados por intermédio de uma distribuição normal foi verificada uma convergência prematura do algoritmo. Comparando os resultados apresentados nas Tabelas 4.2 e 4.3, pode-se concluir que a distribuição uniforme é a mais adequada para o problema de identificação de pontos influentes, uma vez que foram encontradas soluções mais refinadas, quando os parâmetros de controle do algoritmo PSO estão bem determinados.

Tabela 4.2: Resultados computacionais - Distribuição uniforme

Execução	fGBest	Iteração	Tempo(s)
1	12169,7042	200	2,2340
2	12169,9947	200	1,7500
3	12169,1826	200	2,0000
4	12170,0014	200	1,7190
5	12166,7244	200	1,9220
6	12169,3205	200	1,8900
7	12168,9523	200	2,0780
8	12169,9034	200	1,9060
9	12169,6141	200	1,9220
10	12169,9181	200	1,8750
11	12169,9973	200	1,8900
12	12169,9932	200	1,8130
13	12168,8597	200	1,9220
14	12169,9969	200	1,9130
15	12169,9917	200	1,7979
16	12168,7094	200	1,9210
17	12169,8710	200	1,9530
18	12169,8592	200	1,9470
19	12149,8218	200	1,8750
20	12169,5786	200	1,9060
21	12169,4231	200	1,8342
22	12169,9971	200	1,8923
23	12169,9947	200	1,8740
24	12168,8137	200	1,9501
25	12169,5693	200	1,9140
26	12168,9947	200	1,9984
27	12168,4784	200	1,9210
28	12157,8343	200	1,9803
29	12169,6521	200	1,9614
30	12166,8951	200	1,9730
Média	12168,3216	200	1,9177

Tabela 4.3: Resultados computacionais - Distribuição normal

Execução	fGBest	Iteração	Tempo(s)
1	24,0730	61	1,4060
2	18,3487	70	1,5620
3	12170,0662	173	2,1013
4	350,9396	68	1,5230
5	26,4795	54	0,9920
6	408,4903	67	1,3928
7	314,1820	78	1,6180
8	169,9486	74	1,5940
9	489,1479	79	1,7920
10	130,1995	75	1,6325
11	17,4388	65	1,3670
12	317,9310	59	1,3210
13	164,7329	48	0,8832
14	66,9619	72	1,5955
15	17,4495	68	1,4890
16	168,3627	57	1,1220
17	149,2119	70	1,5968
18	179,4685	49	1,1525
19	350,9396	60	1,4110
20	213,8068	55	1,1130
21	18,5978	59	1,4090
22	23,4509	76	1,6070
23	20,7154	69	1,5591
24	456,1788	56	1,2310
25	408,4902	68	1,5420
26	442,8813	73	1,5910
27	191,1620	62	1,4350
28	107,1737	80	1,5990
29	16,1229	65	1,3380
30	21,8488	58	1,3045
Média	581,8267	68,9	1,4427

Capítulo 5

Resultados

Este capítulo apresenta os resultados computacionais obtidos pela aplicação da metaheurística PSO na identificação de pontos influentes. Para testar a metodologia proposta, foram utilizados conjuntos de dados simulados e um conjunto com dados observados.

5.1 Aplicações

5.1.1 Conjuntos de dados simulados

A simulação dos dados foi feita como sugerido em (Atkinson, 1986), onde o número de dados (n) é constituído mantendo uma proporção de 0.6 pontos "bons", bem ajustados, e 0.4 pontos "ruins". Pontos influentes em conjuntos de dados simulados segundo este critério são difíceis de serem identificados por apresentarem forte mascaramento. Neste caso, as técnicas clássicas falham em detectá-los. Em (Atkinson, 1986) é apresentado um exemplo prático, no qual o método de influência local falha em detectar pontos com estas características.

A mesma técnica de geração de dados foi adotada nos trabalhos de Biazi (1996) e Critchley et al. (2001), para conjuntos com até 40 dados. Nestes trabalhos foi utilizado um mesmo procedimento, subdividido em dois estágios. No primeiro estágio, a função $g_{Cook}(\epsilon)$ é maximizada para valores de k (número de observações influentes), tal que $1 \leq k \leq min(n-p, n/2)$, sujeita a restrição $\epsilon^T \mathbf{1}_n = 0$.

Neste estágio, a metade dos pontos do conjunto de dados são identificados como influentes. No segundo estágio são avaliados todos os pontos detectados como influentes, utilizando o teste t (Cook e Weisberg, 1982). Os pontos "bons" identificados erroneamente como influentes são então detectados. Após os dois estágios, os verdadeiros pontos influentes são identificados.

No presente trabalho o procedimento em dois estágios é desnecessário, pois os pontos influentes são identificados quando a distância $|g_{Cook}(\epsilon) - g_{Cook}(0)|$ é maximizada.

Os conjuntos de dados com 40, 200 e 500 pontos foram gerados de forma que, 60% dos pontos seguem o modelo $y_i = x_i + 2 + e_i$, com x_i uniformemente distribuído no intervalo [1 , 4] e e_i normalmente distribuído com desvio padrão igual a 0,2. Os outros 40% dos pontos são normalmente distribuídos com desvio padrão 0,5 e médias $\mu_x = 7$ e $\mu_y = 2$. Para o conjunto com 1000 dados, os 40% dos pontos foram gerados de forma que 240 pontos têm $\mu_x = 7$ e $\mu_y = 2$, e 160 pontos têm $\mu_x = 7$ e $\mu_y = 4$. Os conjuntos de dados são ilustrados graficamente nas Figuras 5.1, 5.2, 5.3 e 5.4. A Tabela 5.1 apresenta o registro dos pontos influentes de cada um dos conjunto de dados, gerados para a aplicação da metodologia proposta no presente trabalho.

Tabela 5.1: Registro dos pontos influentes

Nº de dados	Pontos influentes
40	25 a 40
200	121 a 200
500	301 a 500
1000	601 a 1000

A Tabela 5.2 apresenta os resultados encontrados, objetivando a maximização da função objetivo. Cada linha corresponde aos resultados obtidos para cada conjunto de dados. A coluna melhor solução corresponde ao maior valor encontrado para a função objetivo. Na coluna melhor tempo estão registrados os menores tempos de execução, em segundos, relativos à melhor solução obtida para o problema. A coluna média das soluções corresponde a média aritmética de todas

as soluções encontradas nas execuções do programa. Os números de iterações necessários para encontrar a solução do problema encontram-se na coluna melhor iteração.

Tabela 5.2: Resultados computacionais - Dados simulados

Nº de dados	Nº de exec.	Melhor solução	Média das soluções	Melhor tempo(s)	Tempo médio(s)	Melhor iteração	Nº max. iterações
40	500	$2{,}013\times10^{2}$	$1{,}884\times10^{2}$	1,95	2,41	102	200
200	500	$4{,}675\times10^{2}$	$4{,}527\times10^{2}$	54,28	61,56	692	800
500	50	$2{,}244\times10^{3}$	$2{,}029\times10^{3}$	3098,6	3348,2	1806	2000
1000	20	$2{,}010\times10^{3}$	$1{,}796\times10^{3}$	37642,5	37986,1	2500	2500

A partir dos resultados obtidos, constata-se a capacidade do algoritmo PSO no refinamento das soluções para o presente problema. Ao verificar os tempos necessários para obtenção das soluções do problema, percebe-se que o tempo computacional aumenta significativamente à medida que aumenta o número de dados no conjunto.

A Tabela 5.3 apresenta os valores das perturbações (ϵ_i) obtidos para os conjuntos com 40, 200, 500 e 1000 dados. Em negrito são indicados os pontos influentes.

Tabela 5.3: Valores das perturbações (ϵ_i) - Dados simulados

$n = 40$		$n = 200$		$n = 500$		$n = 1000$	
Pontos	ϵ_i	Pontos	ϵ_i	Pontos	ϵ_i	Pontos	ϵ_i
1 a 24	0,017	1 a 120	0,0035	1 a 300	0,0014	1 a 600	0,0007
25 a 40	**-0,024**	**121 a 200**	**-0,0046**	**301 a 500**	**-0,0019**	**601 a 1000**	**-0,0009**

Os resultados apresentados na Tabela 5.3 indicam os pesos (perturbações) associados a cada ponto quando a função objetivo é maximizada. Observações com pesos negativos são consideradas influentes na determinação dos coeficientes do modelo de regressão. Desta forma, a metodologia identificou corretamente os pontos influentes associando pesos negativos para os 40% dos pontos considerados "ruins".

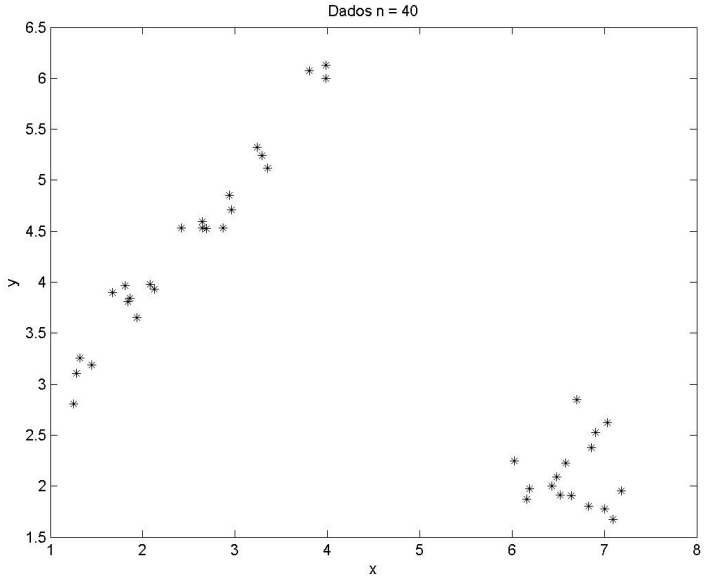

Figura 5.1: Gráfico de dispersão dos dados simulados com 40 pontos

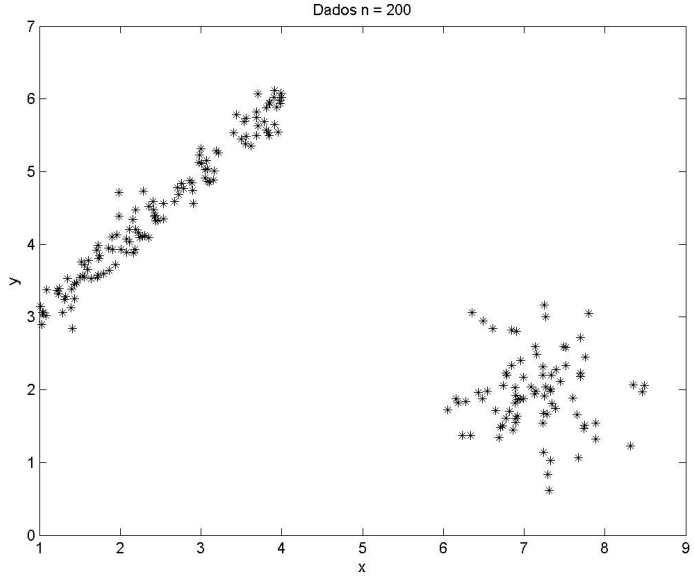

Figura 5.2: Gráfico de dispersão dos dados simulados com 200 pontos

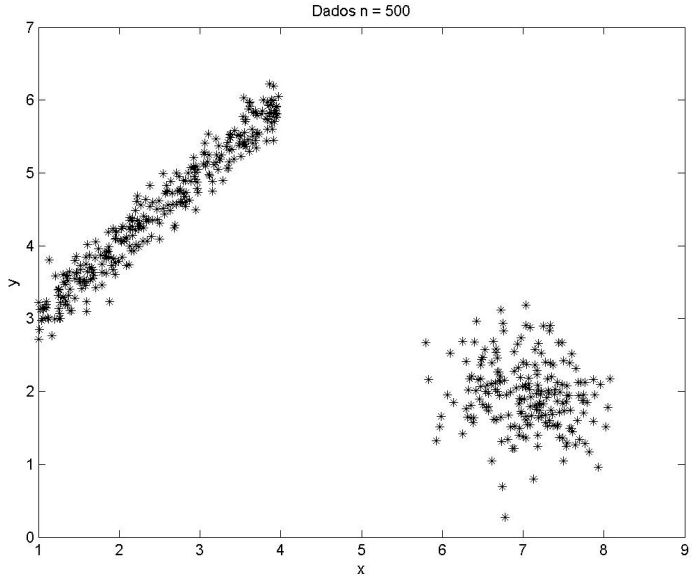

Figura 5.3: Gráfico de dispersão dos dados simulados com 500 pontos

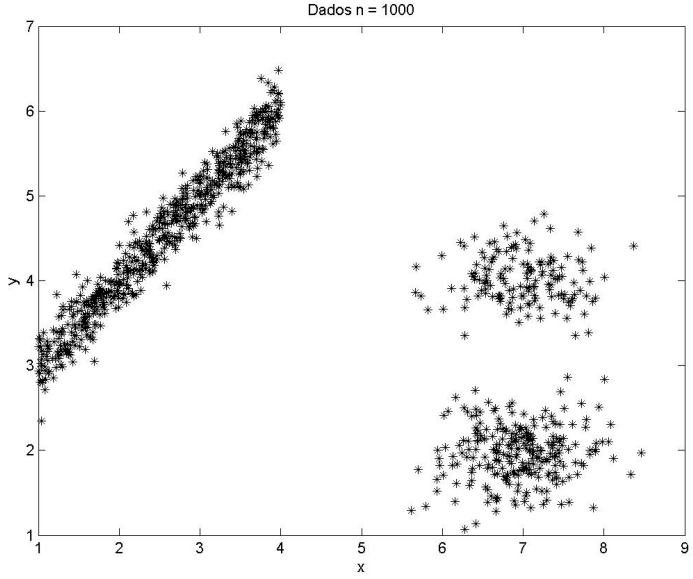

Figura 5.4: Gráfico de dispersão dos dados simulados com 1000 pontos

5.1.2 Conjunto com dados observados

O conjunto de dados utilizado refere-se ao preço médio dos imóveis em Boston. Este conjunto possui 506 observações e 14 variáveis. Os dados encontram-se disponíveis no site *http://lib.stat.cmu.edu* . As variáveis envolvidas no estudo são as seguintes:

- y_i: preço médio dos imóveis em Boston;
- x_{i1}: taxa per capita de criminalidade;
- x_{i2}: proporção de áreas residenciais com lotes acima de 25.000 ft^2;
- x_{i3}: proporção de empresas industriais;
- x_{i4}: variável dummy (1 se o terreno tem fronteira com o rio, 0 caso contrário);
- x_{i5}: concentração de óxido nitríco (poluição);
- x_{i6}: média do número de quartos por habitação;
- x_{i7}: proporção de unidades construídas e ocupadas antes de 1940;
- x_{i8}: distância até o centro de trabalho;
- x_{i9}: índice de acessibilidade às rodovias radiais;
- x_{i10}: valor do imposto sobre a propriedade (em \$10.000);
- x_{i11}: razão entre professor e aluno na cidade;
- x_{i12}: $(B - 0,63)^2$, sendo B a proporção de negros na cidade;
- x_{i13}: proporção da população com baixo poder aquisitivo.

Este conjunto de dados foi analisado por diversos autores, dentre eles, Peña (2005). Peña propôs uma nova medida de influência, baseada na técnica de omissão de pontos, que indica o quanto uma observação é influenciada pelas demais observações do conjunto. Esta medida é uma combinação linear da distância de Cook. Segundo este autor, o conjunto de dados em estudo possui 45 observações influentes no intervalo [366 , 480]. Estas observações correspondem ao distrito central de Boston, e as demais observações correspondem ao subúrbio de Boston.

Os resultados computacionais obtidos com o algoritmo PSO, são apresentados na Tabela 5.4.

Tabela 5.4: Resultados computacionais - Dados observados

Nº de execuções	Melhor solução	Média das soluções	Melhor tempo(s)	Tempo médio(s)	Melhor iteração	Nº max. iterações
50	$7{,}5609\times10^3$	$6{,}9155\times10^3$	6692,36	7389,25	1927	2000

A Tabela 5.5 apresenta os valores das perturbações (ϵ_i) obtidos quando a função objetivo é maximizada.

Tabela 5.5: Valores das perturbações (ϵ_i) - Dados observados

Observações	ϵ_i
366, **374**, **375**, **376**, **377**, **379**, **380**, **381**, **382**, **385**, **386**, **387**, **388**, **389**, **393**, **395**, **399**, **401**, **402**, **404**, **405**, **406**, **407**, **410**, **411**, **412**, **413**, **414**, **415**, **416**, **418**, **419**, **421**, **426**, **428**, **435**, **436**, **437**, **438**, **439** **441**, **445**, **470**, **478**, **480**	**-0,0011**
demais observações	0,001

Os valores das perturbações apresentados na Tabela 5.5, indicam 45 pontos com pesos negativos. Desta forma, as observações influentes foram identificadas corretamente.

5.2 Considerações Finais

A metodologia proposta no presente trabalho, mostrou-se eficaz para detectar pontos influentes em quatro conjuntos de dados simulados e um conjunto com dados observados.

Esta metodologia permite identificar observações influentes em conjuntos com grande número de pontos influentes, o que constitui uma vantagem em relação à técnica de omissão de pontos.

As vantagens desta metodologia em relação ao método utilizado nos trabalhos de Biazi (1996) e Critchley et al. (2001), são: a eliminação do procedimento em dois estágios e a possibilidade de tratar conjuntos com um maior número de dados.

Capítulo 6

Conclusão e Trabalhos Futuros

6.1 Conclusão

Neste trabalho, foi aplicada a metaheurística otimização por enxame de partículas na identificação de pontos influentes em modelos de regressão. A metodologia foi aplicada em conjuntos de dados simulados e um conjunto com dados observados.

Os resultados obtidos, para os conjuntos de dados analisados, mostram que a função de sensibilidade de casos é eficaz para detectar pontos influentes.

A metaheurística PSO depende de fatores aleatórios e/ou probabilísticos. Desta forma, em outras execuções do programa, sob as mesmas condições, podem ser encontrados resultados diferenciados daqueles obtidos neste trabalho. Entretanto, a média das execuções aponta em direção à região da solução ótima. Com base nos resultados computacionais, fica evidenciada a viabilidade da aplicação do método de otimização por enxame de partículas na identificação de pontos influentes.

É importante ressaltar que a distribuição de probabilidade, utilizada para gerar os números aleatórios (r_1 e r_2), tem grande influência no desempenho do algoritmo PSO. Os testes realizados mostram que a distribuição uniforme é mais adequada do que a distribuição normal de Gauss, pois foi possível encontrar soluções mais refinadas para o problema.

Pode-se concluir que a metodologia proposta identifica corretamente os subconjuntos de dados, mesmo quando há "mascaramento" de pontos.

6.2 Trabalhos futuros

Algumas investigações de interesse para ampliar e dar continuidade ao trabalho são:

- A aplicação de outras metaheurísticas na identificação de pontos influentes. Sugere-se, inicialmente, a aplicação de Algoritmos Genéticos. Esta metaheurística pertence à mesma classe do PSO. Desta forma, seria interessante comparar as eficiências destes quanto à qualidade de sua solução e quanto ao tempo necessário de execução;

- Estender a metodologia proposta para modelos lineares generalizados;

- Implementar o algoritmo PSO, utilizando a função de sensibilidade para outros funcionais estatísticos;

- Aperfeiçoar a função de sensibilidade de casos para o traço da matriz de covariância ($g_{tr}[\Omega(\epsilon)]$). Neste trabalho, também foi implementada a função $g_{tr}[\Omega(\epsilon)]$ com o objetivo de identificar observações atípicas (*outliers*) em conjuntos com dados multivariados. Na maioria dos conjuntos testados os *outliers* foram identificados corretamente. No entanto, em alguns conjuntos a metodologia falhou em detectá-los. Assim, a metodologia necessita de algumas melhorias e o tempo foi o fator limitante da pesquisa. A função $g_{tr}[\Omega(\epsilon)]$ e alguns exemplos são apresentados no apêndice C.

Apêndice A

Método da máxima verossimilhança

O método da máxima verossimilhança, assim como o método de mínimos quadrados, permite a estimação dos parâmetros de modelos de regressão. Este método exige que a função de verossimilhança seja conhecida ou pressuposta. O método consiste em maximizar a função de verossimilhança em relação ao parâmetro β.

Uma amostra aleatória $(y_1, y_2, ..., y_n)$, retirada de uma população com uma função de densidade de probalidade $f(y, \theta)$, a qual depende do vetor de parâmetros θ, tem uma função de densidade de probabilidade conjunta dada por

$$\prod_{i=1}^{n} f(y_i, \theta)$$

A função de densidade de probabilidade conjunta é simplesmente o produto das densidades de cada uma das observações

$$f(y_1, \theta) \times f(y_2, \theta) \times ... \times f(y_n, \theta)$$

sendo θ um vetor de parâmetros e y_i uma variável aleatória.

Para uma dada amostra $(y_1, y_2, ..., y_n)$ a função de densidade de probabilidade conjunta vista como função do vetor de parâmetros desconhecidos θ, é denominada função de verossimilhança. Esta função é representada por $L(y, \theta)$.

Uma possibilidade para a resolução do problema de estimação é escolher o

vetor $\hat{\theta}$ que maximize a função de verosssimilhança. Na função $L(y, \theta)$, θ é considerado um vetor de variáveis e o vetor y é fixado (pois ele foi observado). Assim, $L(y, \theta)$ é maximizada em um ponto $\theta = \hat{\theta}$. Em geral, o processo algébrico envolve neste caso o cálculo de derivadas parciais (em relação aos elementos do vetorθ), e igualando-os a zero.

Na maioria dos casos, para a obtenção do máximo de $L(y, \theta)$, é mais simples trabalhar com o logaritmo natural da função de verossimilhança, uma vez que este permite a conversão de produtos em soma.

Como ilustração (Portugal, 1995), considere uma amostra aleatória em que y_i tem distribuição normal. Assim,

$$f(y_i) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left[\frac{-1}{2\sigma^2}(y_i - \beta x_i)^2\right] \tag{A.1}$$

o logaritmo natural da função de verossimilhança é dado por

$$lnL(\beta, \sigma^2, y_i) = \sum_{i=1}^{n} lnf(y_i) \tag{A.2}$$

substituindo (A.1) em (A.2) temos

$$lnL(\beta, \sigma^2, y_i) = \sum_{i=1}^{n}\left[-ln\sqrt{2\pi\sigma^2} - \frac{1}{2\sigma^2}(y_i - \beta x_i)^2\right] \tag{A.3}$$

Para obter o estimador de máxima verossimilhança de β temos que igualar as derivadas primeiras da Eq.(A.3) a zero.

$$\frac{\partial lnL}{\partial \beta} = \frac{-1}{2\sigma^2}\sum_{i=1}^{n} 2(y_i - \beta x_i)(-x_i) = 0 \tag{A.4}$$

A resolução da Eq.(A.4) fornece o estimador de máxima verossimilhança,

$$\hat{\beta} = \frac{\sum_{i=1}^{n}(x_i y_i)}{\sum_{i=1}^{n} x_i^2}$$

Apêndice B

Testes para ajuste do peso inercial

Os testes foram realizados, utilizando o conjunto de dados simulados com 40 observações, e $-0,025 \leq \epsilon_i \leq 0,020$. Foram fixados os seguintes parâmetros: c_1= 2, c_2= 2 e o número de partículas igual a 30.

As Tabelas B.1, B.2, B.3 e B.4 apresentam os resultados encontrados nos testes computacionais. A coluna *fGBest* corresponte a solução encontrada em cada execução do programa.

Os resultados apresentados nos testes evidenciam a forte influência do peso inercial no desempenho do algoritmo PSO.

Tabela B.1: Teste peso inercial - w=0,8

Execução	fGBest	Iteração	Tempo(s)
1	12169,1479	200	1,9690
2	12169,9957	200	1,9530
3	12169,9961	200	1,8780
4	12169,0368	200	1,9220
5	3150,9396	200	1,9840
6	12168,1741	200	1,8750
7	12169,0108	200	1,9530
8	12166,8765	200	1,9370
9	12169,0662	200	1,9220
10	489,4902	200	1,8910
11	12169,0619	200	1,9060
12	12169,9974	200	1,9470
13	4182,1479	200	1,9220
14	350,9396	200	1,8590
15	527,4902	200	1,9060
16	12169,9935	200	1,9110
17	12167,0003	200	1,8810
18	12169,3201	200	1,8750
19	12169,9971	200	1,8560
20	12169,1528	200	1,8910
21	1408,4902	200	1,8734
22	11969,2342	200	1,9106
23	12169,0926	200	1,8951
24	12169,1658	200	1,9123
25	12168,6792	200	1,8993
26	12169,0036	200	1,9060
27	12167,1084	200	1,8496
28	11896,4082	200	1,9025
29	12169,2540	200	1,9752
30	12169,1241	200	1,8732
Média	10056,4132	200	1,9078

Tabela B.2: Teste peso inercial - w=0,9

Execução	fGBest	Iteração	Tempo(s)
1	10114,8011	200	2,1090
2	12152,9636	200	2,0904
3	12168,3658	200	1,9690
4	378,2466	200	1,9370
5	12168,853	200	1,9220
6	7788,6904	200	2,0000
7	12169,6798	200	1,9070
8	8890,0311	200	1,9220
9	12169,9599	200	1,9530
10	11579,8560	200	1,9220
11	11099,6069	200	1,8750
12	12169,9682	200	1,9220
13	579,6331	200	1,8750
14	10913,6642	200	1,9370
15	11912,3862	200	1,9220
16	460,4959	200	2,0630
17	12130,0195	200	1,9060
18	11642,2266	200	1,9530
19	10614,2191	200	1,9370
20	12160,8709	200	1,9530
21	11099,6063	200	1,8750
22	12169,9682	200	1,9220
23	3162,1892	200	1,8750
24	10913,6642	200	1,9370
25	11912,3862	200	1,9220
26	12160,4959	200	2,0630
27	12130,0195	200	1,9060
28	11642,2266	200	1,9530
29	10614,2191	200	1,9370
30	12160,8709	200	1,9530
Média	10041,0061	200	1,947246

Tabela B.3: Teste peso inercial - w= 1,2

Execução	fGBest	Iteração	Tempo(s)
1	24,0730	200	2,6410
2	169,2538	200	2,2970
3	171,3944	200	2,2190
4	179,4684	200	2,6660
5	122,5433	200	2,3440
6	431,2960	200	2,2350
7	93,3444	200	2,3910
8	489,1479	200	2,3670
9	18,5978	200	2,2830
10	408,4902	200	2,4000
11	187,7226	200	2,4750
12	15,5999	200	2,3310
13	130,1995	200	2,6250
14	89,0829	200	2,5120
15	34,0736	200	2,4220
16	17,5543	200	2,2960
17	67,1612	200	2,2190
18	91,0218	200	2,2500
19	317,9310	200	2,3130
20	213,8068	200	2,2960
21	103,5876	200	2,3750
22	16,0172	200	2,2520
23	140,3676	200	2,2180
24	463,6381	200	2,4240
25	314,1820	200	2,4380
26	78,2096	200	2,2950
27	239,2587	200	2,2560
28	32,8615	200	2,3750
29	136,4038	200	2,5260
30	263,0684	200	2,3610
Média	168,6452	200	2,3701

Tabela B.4: Teste peso inercial - w_i=0,9 e w_f=0,4

Execução	fGBest	Iteração	Tempo(s)
1	12169,7042	200	2,2340
2	12169,9947	200	1,7500
3	12169,1826	200	2,0000
4	12170,0014	200	1,7190
5	12166,7244	200	1,9220
6	12169,3205	200	1,8900
7	12168,9523	200	2,0780
8	12169,9034	200	1,9060
9	12169,6141	200	1,9220
10	12169,9181	200	1,8750
11	12169,9973	200	1,8900
12	12169,9932	200	1,8130
13	12168,8597	200	1,9220
14	12169,9969	200	1,9130
15	12169,9917	200	1,7979
16	12168,7094	200	1,9210
17	12169,8710	200	1,9530
18	12169,8592	200	1,9470
19	12149,8218	200	1,8750
20	12169,5786	200	1,9060
21	12169,4231	200	1,8342
22	12169,9971	200	1,8923
23	12169,9947	200	1,8740
24	12168,8137	200	1,9501
25	12169,5693	200	1,9140
26	12168,9947	200	1,9984
27	12168,4784	200	1,9210
28	12157,8343	200	1,9803
29	12169,6521	200	1,9614
30	12166,8951	200	1,9730
Média	12168,3216	200	1,9177

Apêndice C

Função $g_T(\epsilon)$ para o traço da matriz de covariância

A função de sensibilidade de casos para o traço da matriz de covariância permite identificar *outliers* em dados multivariados. Um *outlier* multivariado é uma observação que apresenta grande distanciamento das demais observações no espaço p-dimensional definido por todas as variáveis.

A função $g_T(\epsilon)$ para a matriz de covariância é definida em (Biazi, 1996) como:

$$\Omega(\epsilon) = \sum p_i(\epsilon)[\mathbf{x}_i - \bar{x}(\epsilon)][\mathbf{x}_i - \bar{x}(\epsilon)]^T, \tag{C.1}$$

onde:
$\bar{x}(\epsilon) = X^T P(\epsilon) = X - \mathbf{1}_n \bar{x}(\epsilon)^T = X - \mathbf{1}_n \mathbf{1}_n^T P(\epsilon) X = [I - \mathbf{1}_n \mathbf{1}_n^T P(\epsilon)]X$ e
$P(\epsilon) = diag(p_i) = diag(1/n + \epsilon_i - \bar{\epsilon})$.

Assim, a Eq.(C.1) pode ser escrita na forma,

$$\Omega(\epsilon) = X^T[I - \mathbf{1}_n \mathbf{1}_n^T P(\epsilon)]^T P(\epsilon)[I - \mathbf{1}_n \mathbf{1}_n^T P(\epsilon)]X. \tag{C.2}$$

Escrevendo $\mathbf{q} = \mathrm{P}(\epsilon)\mathbf{1}_n, temos$:

$$[I - \mathbf{1}_n\mathbf{1}_n^T P(\epsilon)]^T P(\epsilon)[I - \mathbf{1}_n\mathbf{1}_n^T P(\epsilon)] = [I - \mathbf{1}_n\mathbf{1}_n^T P(\epsilon)]^T [P(\epsilon) - \mathbf{q}\mathbf{q}^T]$$

$$= (I - \mathbf{q}\mathbf{1}_n^T)(P(\epsilon) - \mathbf{q}\mathbf{q}^T$$

$$= P(\epsilon) - 2\mathbf{q}\mathbf{q}^T + \mathbf{q}\mathbf{q}^T$$

$$= P(\epsilon) - qq^T. \tag{C.3}$$

Substituindo a Eq.(C.3) em (C.2), a função $g_T(\epsilon)$ para a matriz de covariância pode ser expressa por,

$$\Omega(\epsilon) = X^T[P(\epsilon) - \mathbf{q}\mathbf{q}^T]X, \tag{C.4}$$

em que $P(\epsilon) = diag(\frac{1}{n} + \epsilon_i - \overline{\epsilon})$ e $\mathbf{q} = E\mathbf{1}_n$.

O traço da matriz de covariância obtido a partir da Eq.(C.4) é expresso por,

$$tr(\Omega(\epsilon)) = tr(X^T P(\epsilon)X) - tr(X^T\mathbf{q}\mathbf{q}^T X)$$

$$= tr(P(\epsilon)XX^T) - \mathbf{q}^T(XX^T)\mathbf{q},$$

escrevendo $A = XX'$ e $\mathbf{a} = (a_{11}, ...a_{nn})^T$ o vetor de elementos da diagonal de A, a função $g_T(\epsilon)$ para o traço da matriz de covariância pode ser escrita na forma,

$$g_{tr}(\Omega(\epsilon)) = \mathbf{q}^T\mathbf{a} - q^T A\mathbf{q}. \tag{C.5}$$

Para identificar *outliers* multivariados utiliza-se a Eq.(C.5) como função objetivo. O problema consiste em determinar o vetor de perturbações ϵ que minimiza a função $g_{tr}(\Omega(\epsilon))$.

A metodologia foi aplicada em vários conjuntos de dados. Observou-se que na maioria dos conjuntos testados, os *outliers* foram identificados corretamente, mas em alguns conjuntos houve falha em detectar todos os *outliers*.

A seguir são apresentados alguns exemplos, os quais são bastante utilizados em estudos de observações influentes, *outliers* e pontos de alavanca.

Exemplos

Os dados utilizados, no primeiro exemplo, foram simulados por Hawkins et al. (1984). O conjunto de dados é composto por 75 observações, das quais os primeiros 14 pontos são *outliers*, e três variáveis. Os dados são apresentados na Tabela C.2 .

Na Tabela C.1 estão os valores das perturbações (ϵ_i) obtidos quando a função $g_{tr}[\Omega(\epsilon)]$ é minimizada.

Tabela C.1: Valores das perturbações (ϵ_i) - Dados de Hawkins, Bradu e Kass

Observações	ϵ_i
1 a 14	**-0,013**
15 a 75	0,013

Com base nos valores de ϵ_i apresentados na Tabela C.1, observa-se que os pontos de 1 a 14 têm pesos negativos. Assim, os catorze *outliers* foram identificados corretamente.

Tabela C.2: Dados de Hawkins, Bradu e Kass (1984)

Obs.	X_1	X_2	X_3	Obs.	X_1	X_2	X_3
1	10.1	19.6	28.3	39	2.1	0.0	1.2
2	9.5	20.5	28.9	40	0.5	2.0	1.2
3	10.7	20.2	31.0	41	3.4	1.6	2.9
4	9.9	21.5	31.7	42	0.3	1.0	2.7
5	10.3	21.1	31.1	43	0.1	3.3	0.9
6	10.8	20.4	29.2	44	1.8	0.5	3.2
7	10.5	20.9	29.1	45	1.9	0.1	0.6
8	9.9	19.6	28.8	46	1.8	0.5	3.0
9	9.7	20.7	31.0	47	3.0	0.1	0.8
10	9.3	19.7	30.3	48	3.1	1.6	3.0
11	11.0	24.0	35.0	49	3.1	2.5	1.9
12	12.0	23.0	37.0	50	2.1	2.8	2.9
13	2.0	26.0	34.0	51	2.3	1.5	0.4
14	11.0	4.0	34.0	52	3.3	0.6	1.2
15	3.4	2.9	2.1	53	0.3	0.4	3.3
16	3.1	2.2	0.3	54	1.1	3.0	0.3
17	0.0	1.6	0.2	55	0.5	2.4	0.9
18	2.3	1.6	2.0	56	1.8	3.2	0.9
19	0.8	2.9	1.6	57	1.8	0.7	0.7
20	3.1	3.4	2.2	58	2.4	3.4	1.5
21	2.6	2.2	1.9	59	1.6	2.1	3.0
22	0.4	3.2	1.9	60	0.3	1.5	3.3
23	2.0	2.3	0.8	61	0.4	3.4	3.0
24	1.3	2.3	0.5	62	0.9	0.1	0.3
25	1.0	0.0	0.4	63	1.1	2.7	0.2
26	0.9	3.3	2.5	64	2.8	3.0	2.9
27	3.3	2.5	2.9	65	2.0	0.7	2.7
28	1.8	0.8	2.0	66	0.2	1.8	0.8
29	1.2	0.9	0.8	67	1.6	2.0	1.2
30	1.2	0.7	3.4	68	0.1	0.0	1.1
31	3.1	1.4	1.0	69	2.0	0.6	0.3
32	0,5	2.4	0.3	70	1.0	2.2	2.9
33	1.5	3.1	1.5	71	2.2	2.5	2.3
34	0.4	0.0	0.7	72	0.6	2.0	1.5
35	3.1	2.4	3.0	73	0.3	1.7	2.2
36	1.1	2.2	2.7	74	0.0	2.2	1.6
37	0.1	3.0	2.6	75	0.3	0.4	2.6
38	1.5	1.2	0.2	–	–	–	–

O segundo conjunto de dados utilizado refere-se a medições em vasos de cerâmicas pré-históricos encontrados na Tailândia (Manly, 1986). Este conjunto é formado por 25 observações e 6 variáveis. As observções 9, 10, 22,23 e 24 são *outliers*.

Tabela C.3: Dados - Cerâmicas da Tailândia

Obs.	X_1	X_2	X_3	X_4	X_5	X_6
1	13	21	23	14	7	8
2	14	14	24	19	5	9
3	19	23	24	20	6	12
4	17	18	16	16	11	8
5	19	20	16	16	10	7
6	12	20	24	17	6	9
7	12	19	22	16	6	10
8	12	22	25	15	7	7
9	11	15	17	11	6	5
10	11	13	14	11	7	4
11	12	20	25	18	5	12
12	13	21	23	15	9	8
13	12	15	19	12	5	6
14	13	22	26	17	7	10
15	14	22	26	15	7	9
16	14	19	20	17	5	10
17	15	16	15	15	9	7
18	19	21	20	16	9	10
19	12	20	26	16	7	10
20	17	20	27	18	6	14
21	13	20	27	17	6	9
22	9	9	10	7	4	3
23	8	8	7	5	2	2
24	9	9	8	4	2	2
25	12	19	27	18	5	12

Os valores das perturbações (ϵ_i) são apresentados na Tabela C.4.

Tabela C.4: Valores das perturbações (ϵ_i) - Cerâmicas da Tailândia

Observações	ϵ_i
9, 10, 22, 23 e 24	**-0,04**
demais obs.	0,04

Os valores apresentados na Tabela C.4 indicam que os pontos 9, 10, 22, 23 e 24 são *outliers*. Desta forma, todos os *outliers* foram identificados corretamente.

Os dados utilizados, no terceiro exemplo, encontram-se disponíveis em (Hadi, 1992). Este conjunto possui 28 observações e 2 variáveis e está apresentado na Tabela C.5. Os pontos 6, 15, 16 e 25 são *outliers*.

Tabela C.5: Peso cerebral

Obs.	X_1	X_2	Obs.	X_1	X_2
1	1,35	8,1	15	6654	5712
2	465	423	16	9400	70
3	36,33	119,5	17	6,8	179
4	27,66	115	18	35	56
5	1,06	5,5	19	0,12	1
6	11700	50	20	0,023	0,4
7	2547	4603	21	2,5	12,1
8	187	419	22	55,5	175
9	521	655	23	100	157
10	10	115	24	52,16	440
11	3,3	25,6	25	87000	54,5
12	529	680	26	0,28	1,9
13	207	406	27	0,122	3
14	62	1320	28	192	180

A Tabela C.6 apresenta os valores das perturbações (ϵ_i).

Tabela C.6: Valores das perturbações (ϵ_i) - Peso cerebral

Observações	ϵ_i
25	**-0,0357**
demais obs.	0,0357

Os valores apresentados na Tabela C.6 indica somente o ponto 25 como *outlier*. Assim, houve falha em detectar os demais pontos de *outliers*.

O quarto conjunto de dados (*Stackloss*) refere-se ao estudo da oxidação de amônia com ácido nítrico. Este conjunto de dados foi analisado por diversos autores, dentre eles, Peña (2001) e Bagheri et al. (2009). Nestes trabalhos, foram identificadas as seguintes observações como *outliers*: 1, 2, 3, 4, 13, 14, 20 e 21. Os dados são apresentados na Tabela C.7.

Tabela C.7: *Dados - Stackloss*

Obs.	X_1	X_2	X_3	Obs.	X_1	X_2	X_3
1	80	27	89	12	58	17	88
2	80	27	88	13	58	18	82
3	75	25	90	14	58	19	93
4	62	24	87	15	50	18	89
5	62	22	87	16	50	18	86
6	62	23	87	17	50	19	72
7	62	24	93	18	50	19	79
8	62	24	93	19	50	20	80
9	58	23	87	20	56	20	82
10	58	18	80	21	70	20	91
11	58	18	89	–	–	–	–

Na Tabela C.8 estão os valores das perturbações (ϵ_i).

Tabela C.8: Valores das perturbações (ϵ_i) - *Stackloss*

Observações	ϵ_i
1, 2, 3, 17 e 21	**-0,048**
demais obs.	0,048

Os valores das perturbações (ϵ_i), obtidos por meio da minimização da função $g_{tr}(\Omega(\epsilon))$, indicam as seguintes observações como *outliers*: 1, 2, 3, 17 e 21. Os resultados encontrados pela metodologia proposta não estão de acordo com os resultados apresentados por Penã (Peña, 2001) e Bagheri et.al.(Bagheri et al., 2009).

A Tabela C.9 apresenta os valores médios e os melhores resultados encontrados nas execuções do programa. Cada linha corresponde aos resultados obtidos para cada conjunto de dados.

Tabela C.9: Resultados computacionais - Função $g_{tr}(\Omega(\epsilon))$

Nº de dados	Nº de execuções	Melhor solução	Média das soluções	Melhor tempo(s)	Tempo médio(s)	Melhor iteração
75	100	-166,1175	-166,0670	24,73	30,94	377
25	100	-11,9683	-11,9422	5,03	5,18	94
28	100	$-2{,}4103\times10^{8}$	$-2{,}4099\times10^{8}$	4,98	5,04	76
21	100	-0,7312	-0,7311	5,95	6,12	113

Referências Bibliográficas

Ampanthong, P. e Suwattee, P. (2009). *A Comparative Study of Outlier Detection Procedures in Multiple Linear Regression.* Proceedings of the International MultiConference of Engineers and Computer Scientists, Vol I, IMECS. p.704-709.

Andersen, R. (2008). *Modern Methods for Robust Regression.* Sage Publications.

Atkinson, A. e Riani, M. (2000). *Robust Diagnostic Regression Analysis.* New York: Springer-Verlag.

Atkinson, A. C. (1981). *Two graphical displays for outlying and influential observations in regression.* Biometrika, 68. p.13-20.

Atkinson, R. A. (1986). *Masking unmasked.* Biometrika, 73. p.533-541.

Bagheri, A.; Midi, H. e Imon, H.M.R. (2009). *Two-Step Robust Diagnostic Method for Identification of Multiple High Leverage Points.* Journal of Mathematics and Statistics. p.97-106.

Belsley, D.A.; Kuh, E. e Welsch, R.E. (1980). *Regression diagnostics.* John Wiley, New York.

Bergh, F. V. D. (2001). *An Analysis of Particle Swarm Optimizers.* Tese de doutorado, University of Pretoria.

Biazi, Elenice. (1996). *Some Aspects of Influence Analysis and a New Approach.* Tese de doutorado, University of Warwick.

Clerc, M. e Kennedy, J. (2002). *Particle Swarm-Explosion, Stability and Convergence in a Multidimensional Somplex Space.* IEEE Transaction on Evolutionary Computation, 6. p. 58-73.

Cook, R.D. (1977). *Detection of influential observations in linear regression.* Technometrics,19. p.15-18.

Cook, R.D. (1986). *Assessment of local influence.* J.R.Statist.Assoc., 48. p.133-169.

Cook, R.D. e Weisberg, S. (1982). *Residuals and influence in regression.* Chapman and Hall.

Cook, R.D. e Weisberg, S. (1999). *Applied Regression Including Computing and Graphics.* New York: John Wiley.

Crawford, K. D. e Wainwright, R. L. (1996). *Applying genetic algorithms to outlier detection.* Morgan Kaufmann Publishers. p.546-550.

Critchley, F. (1989). *Discussion of leave-k-out diagnostics for time series by A.G.Bruce and R.D.Martin.* J.R.Statist.Soc.,51. p.407-408.

Critchley, F.; Atkinson, R. A.; Lu, G. e Biazi, E. (2001). *Influence Analysis Basead on The Case Sensitivity Function.* Royal Statistical Society, 63. p.307-323.

Eberhart, R.C.; Simpson, P.K. e Dobbins, R.W. (1996). *Computational Intelligence PC Tools.* MA:Academic Press Professional, Boston.

Galea, M.; Paula, G.A. e Bolfarine, H. (1997). *Local Influence in elliptical linear regression models.* The Statistician, 46. p.71-79.

Gomes, M. M. R. C. (2008). *Aplicação das Metaheurísticas PSO e AG na Estimação de Parâmetros do Modelo de Secagem em Leito Fluidizado.* Dissertação de Mestrado, Centro Federal de Educação Tecnólogica de MG.

Hadi, A.S. (1992). *Identifying multiple outliers in multivariate data.* J.R.Statist.Soc.B, 54. p.761-771.

Hampel, F. (1974). *The influence curve and its role in robust estimation.* J.Amer.Statist.Assoc.,69. p.383-393.

Hawkins, D.M.; D.M., and D. Bradu e Kass, G.V. (1984). *Location of several outliers in multiple regression data using elemental sets.* Technometrics, v.26. p.197-208.

Heppner, F. e Grenander, U. (1990). *A Stochastic Nonlinear Model For Coordinate Bird Flocks.* AAAS Publications, Washington, DC.

Hoaglin, D.C. e Welsch, R.E. (1978). *The hat matrix in regression and ANOVA.* The American Statistician, 32. p.17-22.

Huber, P. J. (1973). *Robust Regression: Asymptotics, Conjectures and Monte Carlo.* Annals of Statistics. p.799-821.

Imon, A. H. M. Rahmatullah. (2005). *Identifying Multiple Influential Observations in Linear Regression.* journal of Applied Statistics, 32. p.929-946.

Kennedy, J. e Eberhart, R. (1995). *Particle Swarm Optimization.* Proc.of the IEEE.International Conference on Neural Networks, Piscataway, NJ. p. 1942-1948.

Lemonte, A. J. (2008). *Diagnóstico em Regressão Normal Linear: Princípios e Aplicação.* Rev. Bras. Biom., São Paulo, 26. p.07-26.

Manly, B. F.J. (1986). *Multivariate statistic methods: A primer.* Chapman and Hall Ltd.

Maronna, R. A.; Martin, D. R. e Yohai, V. J. (2006). *Robust Statistics: Theory and Methods.* New York: Willey.

Medeiros, J. A. C. C. (2005). *Exame de Partículas como Ferramenta de Otimização em Problemas de Engenharia Nuclear.* Tese de doutorado, Universidade Federal do Rio de Janeiro.

Millonas, M.M. (1994). *Swarms,phase transitions, and collective intelligence.* In C.G. Langton,Artificial Life III, Addison Wesley, Reading, MA.

Muller, V. (2007). *Otimização de Layouts Industriais Através do Método Exame de Particulas.* Dissertação de Mestrado, Universidade de Santa Cruz do sul.

Paula, G. A. (2004). *Modelos de regressão com apoio computacional.* Instituto de Matemática e Estatística, USP, http://www.ime.usp.br/giapaula/livro.pdf, acessado em 17/02/2008.

Peña, D. (2001). *Multivariate Outlier Detection and Robust Covariance Matrix Estimation.* Technometrics, 43. p.286-310.

Peña, D. (2005). *A New Statistic for Influence in Linear Regression.* Technometrics, 47. p.1-12.

Poli, R. (2008). *Analysis of the publications on the applications of particle swarm optimization.* J. Artif. Evol. App., 1. p. 1-10.

Poli, R.; Kennedy, J. e Blackwell, T. (2007). *Particle Swarm Optimization: An Overview.* Springer New York. p. 33-57.

Portugal, Marcelo S. (1995). *Notas Introdutórias sobre o Princípio de Máxima Verossimilhança: Estimação e Teste de Hipóteses. Textos didáticos.* DECON UFRGS, Porto Alegre.

Prado, J.R. e Saramago, S.F.P. (2005). *Otimização por Colônia de Partículas.* FAMAT em Revista, 4, Nº4. p. 87-103.

Ratnaweera, A.; Halgamuge, S.K. e Watson, H.C. (2004). *Self-Organizing Hierarchical Particle Swarm Optimizer With Time-Varying Acceleration Coefficients.* IEEE Transactions on Evolutionary Computation, 8. p. 240-255.

Reynolds, C.W. (1987). *Flocks, herds, and schools: a distributed behavioral model.* Computer Graphics, 21. p.25-34.

Rousseeuw, P. J. (1984). *Least Median of squares Regression.* Journal of the American Statistical Association. p.871-880.

Rousseeuw, P. J. e Yohai, V. J. (1984). *Robust Regresssion by Means of S-Estimators.* Nonlinear Time Series Analysis: Lecture Notes in Statistics. p.256-272.

Rousseeuw, P.J. e Leroy, A.M. (1987). *Robust regression and outlier detection.* John Wiley, New York.

Shi, Y. e Eberhart, R. (1998)a. *Parameter selection in particle swarm optimization.* Evolutionary Programming VII: Proceedings of the Seventh Annual Conference on Evolutionary Programming, New York. p. 591-600.

Shi, Yuhui e Eberhart, Russel. (1998)b. *A Modified Particle Swarm Optimizer.* Proc.of the IEEE Congress on Evolutionary Computation(CEC 1998), Piscataway, NJ. p. 69-73.

Tolvi, J. (2004). *Genetic algorithms for outlier detection and variable selection in linear regression models.* Springer-Verlag. p.527-533.

Weisberg, S. (1985). *Applied linear regression.* John Wiley Sons.

Wilson, E.O. (1995). *Sociobiology: The new synthesis.* Cambridge, MA: Belknap Press.

Ye, D. e Chen, Z. (2008). *A New Algorithm for High-Dimensional Outlier Detection Based on Constrained Particle Swarm Intelligence.* Springer Berlin, vol.5009,. p.516-523.

Yohai, V. J. (1987). *High Breakdown Point and High Efficiency Robust Estimates for Regression.* Annals of Statistics. p.642-656.

Livros Grátis

(http://www.livrosgratis.com.br)

Milhares de Livros para Download:

Baixar livros de Literatura
Baixar livros de Literatura de Cordel
Baixar livros de Literatura Infantil
Baixar livros de Matemática
Baixar livros de Medicina
Baixar livros de Medicina Veterinária
Baixar livros de Meio Ambiente
Baixar livros de Meteorologia
Baixar Monografias e TCC
Baixar livros Multidisciplinar
Baixar livros de Música
Baixar livros de Psicologia
Baixar livros de Química
Baixar livros de Saúde Coletiva
Baixar livros de Serviço Social
Baixar livros de Sociologia
Baixar livros de Teologia
Baixar livros de Trabalho
Baixar livros de Turismo