JOSÉ DOMINGOS ALBUQUERQUE AGUIAR

MCAC - *MONTE CARLO ANT COLONY*:

UM NOVO ALGORITMO ESTOCÁSTICO DE AGRUPAMENTO DE DADOS

RECIFE-PE – Fevereiro/2008

UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO

PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO

PROGRAMA DE PÓS-GRADUAÇÃO EM BIOMETRIA E ESTATÍSTICA APLICADA

MCAC - *MONTE CARLO ANT COLONY*:

UM NOVO ALGORITMO ESTOCÁSTICO DE AGRUPAMENTO DE DADOS

Dissertação apresentada ao Programa de Pós-Graduação em Biometria e Estatística Aplicada como exigência parcial à obtenção do título de Mestre.

Área de Concentração: Desenvolvimento de Métodos Estatísticos e Computacionais.

Orientador: Prof. Dr. Adauto José Ferreira de Souza.

Co-orientador: Prof. Dr. Borko D. Stosic.

RECIFE-PE – Fevereiro/2008

FICHA CATALOGRÁFICA

A282m Aguiar, José Domingos Albuquerque
MCAC – Monte Carlo Ant Colony : um novo algoritmo es – tocástico de agrupamentos de dados / José Domingos Albu - querque Aguiar. -- 2008.
88 f. : i.

Orientador : Adauto José Ferreira de Souza
Dissertação (Mestrado em Biometria e Estatistica Aplicada) - Universidade Federal Rural de Pernambuco. Departamento de Estatística e Informática.
Inclui anexo e bibliografia.

CDD 574. 018 2

1. Dados estatisticos
2. Método de Monte Carlo
3. Otimização matemática
I. Souza, Adauto José Ferreira de
II. Título

UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO

PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO

PROGRAMA DE PÓS-GRADUAÇÃO EM BIOMETRIA E ESTATÍSTICA APLICADA

MCAC - *MONTE CARLO ANT COLONY*: UM NOVO ALGORITMO ESTOCÁSTICO DE AGRUPAMENTO DE DADOS

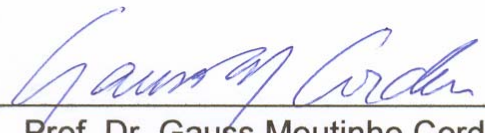
JOSÉ DOMINGOS ALBUQUERQUE AGUIAR

Dissertação julgada adequada para obtenção do título de mestre em Biometria e Estatística Aplicada, defendida e aprovada por unanimidade em 29/02/2008 pela Comissão Examinadora.

Orientador:

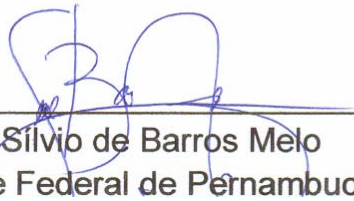
Prof. Dr. Adauto José Ferreira de Souza
Universidade Federal Rural de Pernambuco

Banca Examinadora:



Prof. Dr. Gauss Moutinho Cordeiro
Universidade Federal Rural de Pernambuco

Prof. Dr. Borko Stosic
Universidade Federal Rural de Pernambuco



Prof. Dr. Sílvio de Barros Melo
Universidade Federal de Pernambuco

Dedico este trabalho:

Aos meus pais Domingos Maciel de Aguiar (*in memoriam*) e Maria do Socorro Bezerra de Albuquerque, que foram os pilares da minha educação.

A minha amada esposa Polyana de Cassia Cavalcanti Aguiar, minha eterna flor, cuja essência me inspira e impulsiona dando-me forças todos os dias, e que sem ela minha vida não teria sentido.

A minha querida irmã Amanda Maria Albuquerque Aguiar, grande amiga, por toda sua torcida pela minha felicidade.

Agradecimentos

A Deus, pelo dom da vida, pela saúde e pela oportunidade de concretizar esse sonho.

Aos meus pais Domingos Maciel de Aguiar (*in memoriam*) e Maria do Socorro Bezerra de Albuquerque por todo amor, dedicação e educação que me ofertaram.

A minha esposa Polyana de Cássia Cavalcanti Aguiar que acredita em mim e sempre me incentiva na luta pela vida, e por todo amor que ela me dedica.

Agradeço aos meus amigos integrais, Maria Antônia, coronel Alfredo e Luciane Carvalho por todo incentivo e apoio que me proporcionaram na minha jornada profissional.

Agradeço aos amigos e companheiros de curso que certamente contribuíram direta ou indiretamente para a conclusão deste trabalho.

Agradeço ao professor Doutor Adauto José Ferreira de Souza por todos os ensinamentos e paciência.

“Além da mente humana e como um impulso livre, cria-se a ciência. Esta se renova, assim como as gerações, frente a uma atividade que constitui o melhor jogo do homo ludens: a ciência é, no mais estrito e melhor dos sentidos, uma gloriosa diversão”.

Jacques Barzu.

Resumo

Esta dissertação apresenta um algoritmo inédito de agrupamento de dados que têm como fundamentos o método de Monte Carlo e uma heurística que se baseia no comportamento social das formigas, conhecida como Otimização por Colônias de Formigas. Neste trabalho realizou-se um estudo comparativo do novo algoritmo com outros dois algoritmos de agrupamentos de dados. O primeiro algoritmo é o K-Médias que é muito conhecido entre os pesquisadores. O segundo é um algoritmo que utiliza a Otimização por Colônias de Formigas juntamente com um híbrido de outros métodos de otimização. Para implementação desse estudo comparativo utilizaram-se oito conjuntos de dados sendo três conjuntos de dados reais, dois artificiais gerados deterministicamente e três artificiais gerados aleatoriamente. Os resultados do estudo comparativo demonstram que o novo algoritmo identifica padrões nas massas de dados, com desempenho igual ou superior aos outros dois algoritmos avaliados. Neste trabalho investigou-se também a capacidade do novo algoritmo em identificar o número de grupos existentes nos conjuntos dados. Os resultados dessa investigação mostram que o novo algoritmo é capaz de identificar o de número provável de grupos existentes dentro do conjunto de dados.

Palavras-chave: Agrupamentos de dados, Otimização por Colônias de Formigas, método de Monte Carlo.

Abstract

In this work we present a new data cluster algorithm based on social behavior of ants which applies Monte Carlo simulations in selecting the maximum path length of the ants. We compare the performance of the new method with the popular k-means and another algorithm also inspired by the social ant behavior. For the comparative study we employed three data sets from the real world, three deterministic artificial data sets and two random generated data sets, yielding a total of eight data sets. We find that the new algorithm outperforms the others in all studied cases but one. We also address the issue concerning about the right number of groups in a particular data set. Our results show that the proposed algorithm yields a good estimate for the right number of groups present in the data set.

Keywords: Data clustering, Ant Colony Optimization, method Monte Carlo.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

Capítulo 1

Introdução.

Os humanos realizam a tarefa de reconhecer padrões naturalmente e de maneira eficiente, no entanto, apresentam limitações na realização de cálculos. Por outro lado, os computadores são excelentes ferramentas de cálculos, porém, são muito limitados na tarefa de reconhecimento de padrões. Esta limitação dos computadores justifica a procura por novos algoritmos que lhes permitam melhorar a capacidade de reconhecimento de padrões.

A união do poder de cálculo já existente nos computadores com um método eficiente de reconhecimento de padrões proporcionaria uma excelente ferramenta capaz de realizar automaticamente a tarefa de reconhecer padrões em várias áreas das ciências puras e aplicadas. Por exemplo: na medicina, separando pessoas saudáveis de pessoas doentes; na biologia, classificando espécies distintas de animais ou plantas; na química, separando substâncias diferentes baseadas em características como concentração, pH, temperatura e assim por diante; na economia, identificando consumidores de baixa ou alta renda com baixo ou alto consumo, para fornecimento de crédito.

Uma forma de reconhecimento de padrões é realizada através de algoritmos de agrupamento de dados. Esses algoritmos separam um conjunto de dados, sem qualquer tipo de classificação anterior, em grupos homogêneos. Nesse trabalho será proposto um algoritmo inédito de agrupamento de dados que se inspira no comportamento social de formigas. A utilização de algoritmos que têm como fundamento o comportamento social de formigas tem início com o trabalho de Colorni, Dorigo e Maniezzo (1991), que aplicaram esses fundamentos no conhecido problema do caixeiro viajante. Após essa aplicação inicial, outros algoritmos baseados no comportamento de formigas foram surgindo e sendo aplicados a outros problemas inclusive ao problema de agrupamento de dados, como no caso do algoritmo ACBHO (Sinha et al. 1996) que serviu de referência teórica para a nova proposta apresentada nesse trabalho.

A parte computacional deste trabalho foi realizada utilizando-se duas linguagens de programação: Microsoft ® Visual C e Borland ® C Builder. Um total de

seis programas foram implementados. Desses seis programas, quatro em C padrão ANSI utilizando-se o Microsoft ® Visual C, sendo um deles para implementação do primeiro algoritmo baseado em formigas e três para implementação de três algoritmos de agrupamento de dados. Em Borland ® C Builder foram implementados dois programas que foram responsáveis por gerar saídas gráficas dos agrupamentos, e realizar a avaliação dos resultados gerados pelos algoritmos de agrupamentos.

Abaixo se encontra uma breve descrição da estrutura da dissertação que está dividida em 6 capítulos.

No capítulo 2, encontra-se a descrição de alguns algoritmos baseados em formigas, e suas aplicações em problemas de otimização, como por exemplo, o problema do caixeiro viajante. O capítulo inicia-se com uma breve explanação sobre a inspiração biológica que deu origem a todos os algoritmos baseados em inteligência coletiva, batizada por alguns de inteligência de enxame. Após, seguem-se algumas definições e características dos algoritmos baseados em formigas juntamente com algumas aplicações.

No capítulo 3, descreve-se a definição, as aplicações e os componentes da tarefa de agrupar dados. Posteriormente, discutem-se formas de pré-tratamento dos dados e algumas medidas de similaridade e dissimilaridade. Neste mesmo capítulo descreve-se uma forma de classificação dos algoritmos de agrupamentos juntamente com dois métodos de avaliação de resultados gerados pelos algoritmos.

O capítulo 4 inicia-se com uma breve descrição do método de Monte Carlo que será necessário para os algoritmos de agrupamento. Depois se segue uma exposição de três algoritmos de agrupamentos. O primeiro é o K-Médias que é muito conhecido e utilizado por pesquisadores em suas análises exploratórias de dados. O segundo é um algoritmo baseado no comportamento social de formigas juntamente com outros processos de otimização. O terceiro é uma nova proposta, também baseada no comportamento social das formigas, semelhante ao segundo algoritmo na forma de verificação dos agrupamentos, mas diferente na forma de construção dos grupos.

No quinto capítulo realiza-se a comparação dos resultados gerados pelos três algoritmos expostos no capítulo anterior, utilizando-se como referência alguns conjuntos de dados artificiais e dados reais que possuem classificação conhecida. Em algumas situações, realiza-se, também neste capítulo, a comparação entre os

resultados publicados na literatura e os gerados pela nova proposta descrita no capítulo 4.

Por fim, no capítulo 6, apresentam-se às conclusões.

Capítulo 2

Algoritmos Inspirados em Formigas

Nos últimos anos, cientistas e engenheiros de várias áreas das ciências têm encontrado na observação de fenômenos naturais biológicos e não biológicos, a inspiração para criar novas soluções para problemas diversos. Como exemplo de aplicação que surgiu da observação de fenômenos não biológicos tem-se o recozimento simulado que tem como fundamento o comportamento termodinâmico de um sistema magnético. Por outro lado, são exemplos de aplicações que surgiram da observação de fenômenos biológicos as redes neurais artificiais, algoritmos genéticos e inteligência de enxame.

A inteligência de enxame, que vêm do termo em inglês *swarm intelligence*, é uma técnica de inteligência computacional. Ela é definida por Dorigo e colaboradores (2006) como método para resolver problemas que tem como inspiração o comportamento social de insetos e outros animais. Bonabeau e Meyer (2001) citam como exemplos de animais que são objetos de estudos da inteligência de enxame como as formigas, as abelhas, as vespas, os peixes e os pássaros entre outros.

Segundo Bonabeau e Meyer (2001), a inteligência de enxame tem três vantagens importantes:

i. Flexibilidade: o grupo ou enxame pode se adaptar rapidamente a mudanças de ambientes;
ii. Robustez: até mesmo quando um ou mais indivíduos falham, o grupo ou enxame continua a executar suas tarefas;
iii. Auto-organização: o grupo ou enxame exige relativamente pouca supervisão ou controle.

Uma das primeiras pesquisas realizadas com o comportamento de insetos foi feita pelo francês Pierre Paul Grassé (Dorigo e Socha 2005), entre os anos 40 e 50, com duas espécies de cupins, a *Bellicositermes natalensi* e a *Cubitermes species*, onde ele descobriu que esses insetos ativam certas reações geneticamente

gravadas através de um “estímulo significante”. Esse tipo de comunicação indireta foi chamado por Grassé de “*stigmergy*”.

A *stigmergy* tem duas diferenças básicas em relação a outros tipos de comunicação:

i. A forma da comunicação é feita de maneira não simbólica;
ii. A informação só pode ser recebida localmente, ou seja, só pode ser acessada por insetos que visitem o local ou que estejam próximos de onde a informação foi deixada.

As formigas quando estão em busca de comida (forrageando) deixam no solo uma trilha de uma substância química denominada de feromônio (Dorigo e Gambardella 1997). Desta forma, as formigas estão utilizando a *stigmergy* como forma de comunicação.

O pesquisador Deneubourg e seus colaboradores (1989) afirmam que uma colônia de insetos é um super-organismo sem um cérebro e que cada inseto da colônia só é capaz de obter informação através da *stigmergy*. Deneubourg pesquisou o complexo comportamento social das formigas através de uma experiência que inicialmente ele chamou de ponte em formato de diamante e que depois ficou conhecida como ponte dupla (Dorigo e Stützle 2004). No experimento, ele utilizou uma colônia de formigas da espécie *Iridomyrmex humilis*, conhecida como formiga Argentina.

Deneubourg separou a colônia de formigas de uma fonte de comida por duas pontes de mesmo comprimento que formavam um ângulo de 60 graus entre si, como mostra a figura 2.1. Como inicialmente não havia feromônio, as formigas escolhiam aleatoriamente uma das duas pontes com a mesma probabilidade. Porém quando as formigas passaram marcando o caminho com feromônio essa probabilidade de escolha foi modificada. Deneubourg observou que depois de certo tempo, após uma flutuação inicial, as formiga tendiam a escolher uma das duas pontes para percorrer o caminho entre a colônia e a fonte de alimento.

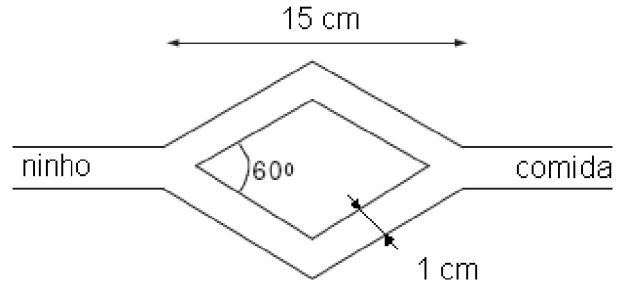


Figura 2.1 Representação do experimento realizado por Deneubourg.

Um outro experimento foi conduzido por Goss (Goss et al. 1989) utilizando também as formigas argentinas. Nessa experiência ele utilizou dois módulos idênticos onde cada módulo era formado por duas pontes, sendo que uma ponte era significantemente maior que a outra como mostra a figura 3.2. Ele observou que depois de uma flutuação aleatória inicial as formigas tendiam a escolher o menor caminho entre o ninho e a fonte de comida. Isto se deu porque as formigas que escolhem o caminho mais curto realizam o percurso de ida e volta em um tempo menor, o que resulta em uma quantidade maior de feromônio depositada na trilha de caminho mínimo. Nas viagens subseqüentes, as formigas tenderam a escolher com uma probabilidade maior a trilha com maior quantidade de feromônio, reforçando-a ainda mais com feromônio, até que em determinado momento praticamente todas as formigas estavam seguindo pelo caminho de menor comprimento.

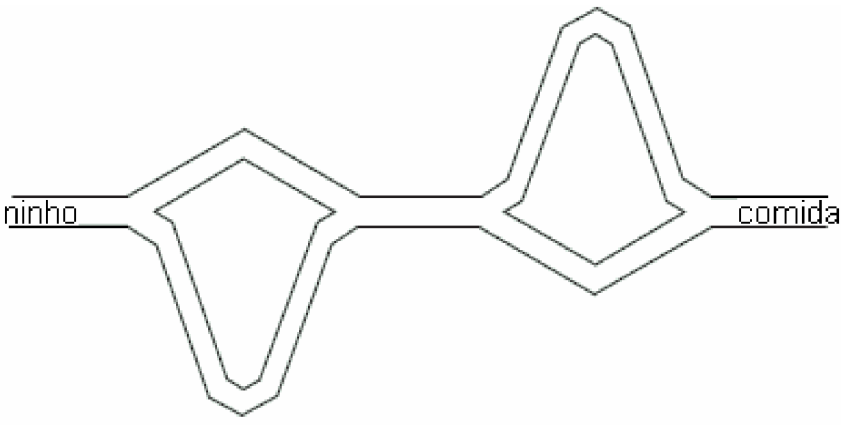


Figura 2.2 Representação do experimento realizado por Goss.

2.1 Otimização da Colônia de Formigas.

O complexo comportamento social da colônia de formigas, em especial, a capacidade da colônia encontrar um caminho mínimo entre o ninho a fonte de comida, inspirou alguns pesquisadores a criar algoritmos baseados nesse comportamento como proposta para resolução de diversos tipos de problemas de otimização. Essa classe de algoritmos baseados no comportamento de formigas ficou conhecida como Otimização da Colônia de Formigas, do inglês *Ant Colony Optimization* (ACO).

O primeiro algoritmo da classe do ACO foi proposto por Colorni, Dorigo e Maniezzo (1991) e apresentado em uma conferência de vida artificial em Paris e recebeu o nome de Sistema de Formigas, do inglês *Ant System* (AS). Esse algoritmo foi inicialmente aplicado ao conhecido problema de otimização do caixeiro viajante que será descrito no próximo item.

2.2 O Problema do Caixeiro Viajante.

O Problema do Caixeiro Viajante (PCV), conhecido em inglês como *Traveling Salesman Problem* (TSP), é um dos mais famosos e fascinantes problemas da matemática e da computação, que gerou uma grande quantidade de pesquisas e de publicações referentes a ele e suas aplicações. Este problema é uma generalização e comparação de muitos outros problemas matemáticos, físicos, computacionais, biológicos, entre outros. Vale destacar que é através de sua solução que se dá a resolução de muitos outros problemas, nas áreas citadas acima, tornando-o de grande importância de forma geral.

Uma definição formal para o PCV pode ser encontrada na literatura (Gambardella e Dorigo 1997), porém, o PCV pode ser formulado de uma maneira simples da seguinte forma: dado um número N de cidades que devem ser visitadas por um caixeiro, qual a seqüência de cidades que torna o comprimento do percurso o menor possível, considerando o início e o término na mesma cidade? Destaca-se ainda, que todas as cidades são interligadas umas as outras e que cada cidade deve ser visitada uma única vez.

Existem algumas variantes do PCV, como:

i. Podem não existir algumas rotas que liguem um dado par de cidades;
ii. A distância pode ser trocada por outra medida como custo, tempo e etc.
iii. O custo entre duas cidades quaisquer pode ser igual na ida como na volta (que determina o PCV simétrico), ou o custo de ida e de volta podem ser diferentes (que determina o PCV assimétrico);

Como descrito em Gutin e Punnen (1976) o PCV é um problema de otimização que pertence a uma classe de problemas conhecidos como NP-difíceis, que implica em um alto custo computacional para obter-se uma solução.

O número de rotas R do PCV assimétrico, com N sendo o número de cidades é dados por:

$$R = (N-1)! \tag{2.1}$$

Devido ao fato da distância ou custo de ida ser o mesmo da volta, o número de rotas do PCV simétrico é a metade do assimétrico o qual é dado por:

$$R = \frac{(N-1)!}{2} \tag{2.2}$$

Para exemplificar o alto custo computacional, tomaremos por base um computador capaz de executar um bilhão (10^9) de operações com ponto flutuante por segundo (1GigaFlop) para resolver um PCV simétrico. Para cada rota o computador necessita realizar (N – 1) operações de soma. Dessa forma o número de rotas por segundo que o computador poderá processar é dado por:

$$Rs = \frac{10^9}{N-1} \tag{2.3}$$

TABELA 2.1: N é o número de cidades a serem visitadas. R é o valor (aproximado no caso de N igual a 15, 20 e 25) do número de rotas dados por R = (N – 1)! / 2. Rs é o número (aproximado nos casos para N igual 10, 15, 20 e 25) de rotas por segundo que o computador em questão consegue processar e é dado por Rs = 10^9 / (N-1). T é o tempo aproximado gasto no processamento das rotas e é dado por T = R / Rs.

N	R	Rs	T
5	12	250 milhões	insignificante
10	181.440	111,11 milhões	0,0016 segundos
15	43,59 bilhões	71,42 milhões	10 minutos
20	$6,08 \times 10^{16}$	52,63 milhões	36 anos
25	$3,10 \times 10^{23}$	41,66 milhões	236 milhões de anos

Pela análise da tabela 2.1, observamos que pequenos aumentos em N gera um extraordinário aumento no tempo de processamento, devido à presença do fatorial no cálculo do número de rotas, em função do número de cidades. Este fato nos mostra a impraticabilidade de solução do problema, para números relativamente pequenos, através do cálculo de todas as possíveis combinações de rotas.

2.3 Semelhanças e Diferenças Entre Formigas Reais e Artificiais.

Considerando-se as formigas reais como sendo as formigas da espécie *Iridomyrmex humilis,* e as formigas artificiais como sendo a implementação de alguns comportamentos das formigas reais em um modelo computacional, Dorigo et al. (1999) afirma que muitas das características dos algoritmos baseados em formigas acabam sendo semelhantes às características das formigas reais como, por exemplo:

i. Tanto as formigas reais como as formigas artificiais são compostas de população ou colônia de agentes que cooperam para encontrar soluções globalmente;
ii. As formigas reais ao caminharem alteram o ambiente depositando no solo o feromônio que pode ser sentida por outras formigas reais. As formigas artificiais possuem uma informação numérica que funciona como um feromônio artificial. Quando as formigas artificiais

se movimentam, alteram essa informação numérica guardada localmente de maneira que outras formigas artificiais possam ler essa informação;

iii. O feromônio depositado pelas formigas reais evapora ao passar do tempo. Já a informação numérica que funciona como um feromônio artificial tem uma taxa que varia entre zero e um, e que simula a evaporação das formigas reais. Essa evaporação permite para ambas as colônias (reais e artificiais) esqueçam sua história de caminhada lentamente de forma a dirigir sua procura para novas direções;

iv. Tanto as formigas reais como as artificiais possuem uma tarefa comum de encontrar o caminho mínimo entre a origem e o destino;

v. Como as formigas reais, as artificiais, se movimentam de maneira estocástica, ou seja, elas se movimentam aplicando uma política de decisão que depende de uma probabilidade.

Ainda segundo Dorigo et al. (1999), as formigas artificiais possuem algumas diferenças das formigas reais. Essas diferenças se encontram em algumas características que as formigas artificiais possuem e que não são encontradas nas formigas reais, como:

i. As formigas artificiais operam em um mundo discreto;

ii. As formigas artificiais possuem memória das ações passadas;

iii. A quantidade de feromônio depositada pelas formigas artificiais depende de uma função que varia de acordo com o tipo de algoritmo implementado;

iv. Para melhorar a eficiência das formigas artificiais, podem ser adicionadas novas capacidades como, por exemplo, uma otimização local.

2.4 Definições do Sistema de Formiga.

Para aplicar o Sistema de Formigas ao problema do caixeiro viajante Dorigo et al. (1996) definiu que cada formiga é um agente simples que possui as seguintes características:

i. Realiza a escolha da cidade para onde deve ir com uma probabilidade que é função da distância e do valor da trilha de feromônio;
ii. Utiliza uma lista que funciona como memória para forçá-la a visitar todas as cidades. A lista também evita que a mesma visite uma cidade já visitada em uma mesma viagem.
iii. Deixa uma trilha de feromônio pelo caminho realizado, quando a viagem estiver completa.

No Algoritmo, o número total de formigas será m, a distância euclidiana entre as cidades i e j será d_{ij} e a visibilidade $\eta_{ij} = 1 / d_{ij}$.

A intensidade da trilha de feromônio entre as cidades i e j no tempo t será dada por $\tau_{ij}(t)$.

A atualização da intensidade da trilha de feromônio será dada pela fórmula (2.4) onde ρ é uma taxa de fixação e $(1 - \rho)$ representa a evaporação da trilha entre os tempos t e t+1.

$$\tau_{ij}(t+1) = \rho \cdot \tau_{ij}(t) + \Delta \tau_{ij} \qquad (2.4)$$

$\Delta\tau_{ij}$ é dada pela fórmula (2.5).

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \qquad (2.5)$$

$\Delta\tau_{ij}^{k}$ é quantidade da trilha de feromônio por unidade de comprimento deixada pela formiga k entre os tempos t e t+1 e é dada por:

$$\Delta\tau_{ij}^{k} = \begin{cases} \dfrac{Q}{L_k} & se\ a\ formiga\ k\ passou\ entre\ as\ cidades\ i\ e\ j\ \ entre\ os\ tempos\ t\ e\ t+1 \\ 0 & em\ outros\ casos \end{cases} \quad (2.6)$$

Onde Q é uma constante e L_k é o comprimento da viagem da formiga k.

Segundo Dorigo et al (1996) o valor de ρ deve ser um valor entre zero e um para evitar uma acumulação ilimitada da trilha de feromônio, e a intensidade inicial da trilha $\tau_{ij}(0)$, deve ser um pequeno valor positivo.

Cada formiga possui uma lista que contém a seqüência de cidades por onde esta formiga já passou. Uma cidade será considerada “permitida” se não estiver contida nessa lista. Quando as m formigas concluírem suas viagens, ou seja, passarem por todo o conjunto de cidades será definido que foi concluído um ciclo. Quando um ciclo for finalizado, a lista será utilizada para calcular o valor do comprimento da viagem de cada formiga e verificar se este valor é o menor encontrado até então. A cada início de um novo ciclo as listas serão limpas e as formigas livres para realizarem uma nova viagem.

A probabilidade de transição da cidade *i* para a cidade *j* para uma formiga *k* é dada pela seguinte equação:

$$P_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}]^{\beta}}{\sum_{k \in\ permitida(k)} [\tau_{ik}(t)]^{\alpha} \cdot [\eta_{ij}]^{\beta}} & se\ j \in permitida(k) \\ 0\ \ em\ \ outro\ \ caso \end{cases} \quad (2.7)$$

Os parâmetros α e β são de controles relativos à importância da trilha e da visibilidade respectivamente.

2.5 O Algoritmo do Sistema de Formigas.

O algoritmo básico do Sistema de Formigas está descrito na figura 2.3, onde NC é o contador do número de ciclos e NCmax é um parâmetro livre que determina a quantidade máxima de ciclos.

Dorigo (Dorigo et al. 1996) observou empiricamente que o melhor valor para o número de formigas é o mesmo do número de cidades.

NC = 0.
Atribua o valor inicial da trilha com uma constante $\tau_{ij}(t) = c$.
Repita enquanto o NC < NCmax.
{
    Para todas as trilhas (ij) faça $\Delta\tau_{ij} = 0$.
    Para todas as formigas faça:
        Preencha o primeiro elemento da lista de cada formiga com a cidade inicial.
    Repita enquanto a lista não está cheia.
    {
        Para todas as formigas faça:
        {
            Escolha a cidade j de acordo com a equação (2.7).
            Mova a formiga k para a cidade j colocando a cidade j na lista(k).
        }
    }
    Para todas as formigas faça:
    {
        Calcule o comprimento L_k da viagem da formiga.
        Teste se esse caminho L_k é o menor encontrado.
    }
    Para todas as trilhas (ij) faça:
    {
        Para todas as formigas faça:
        {
            Calcule $\Delta\tau_{ij}^{k}$ de acordo com a equação (2.6).
            Calcule $\Delta\tau_{ij} = \Delta\tau_{ij} + \Delta\tau_{ij}^{k}$.
        }
    }
    Para todas as trilhas (ij) atualize as trilhas de acordo com a equação (2.6).
    NC = NC + 1.
    Limpe a lista de cada formiga.
}
Mostre o valor do menor caminho encontrado.

Figura 2.3: Algoritmo básico do Sistema de Formigas.

O algoritmo funciona da seguinte forma: inicialmente atribui um valor constante a todas as trilhas. Enquanto não concluir o número de ciclos estabelecido, que é um parâmetro livre, o algoritmo irá:

i. Fazer que todas as formigas percorram todas as cidades, realizando assim uma viagem, escolhendo as cidades não visitadas de acordo com a equação (2.7);
ii. Calcular o comprimento total da viagem de cada formiga e verificar se esse comprimento é o menor encontrado até então;
iii. No final da viagem, calcular a quantidade de feromônio depositada pelas formigas em cada trilha de acordo com as equações (2.5) e (2.6);
iv. Atualizar o valor da trilha de feromônio de acordo com a equação (2.4).

Concluído o número de ciclos, o algoritmo irá apresentar o menor caminho encontrado durante a sua execução.

Existem ainda mais duas variações do algoritmo básico do AS, a formiga-densidade e a formiga-quantidade. A diferença dessas duas variações para o algoritmo básico que trabalha por ciclo conhecido por formiga-ciclo é no cálculo de $\Delta\tau_{ij}^{k}$. Nesses dois modelos cada novo passo que a formiga dá ela já deixa a trilha, sem ter que esperar o fim do ciclo para fazê-lo. O valor de $\Delta\tau_{ij}^{k}$ na formiga-densidade é um valor fixo, enquanto na formiga-quantidade é um valor inversamente proporcional à distância.

No modelo de formiga-densidade o valor de $\Delta\tau_{ij}^{k}$ é dada pela expressão:

$$\Delta\tau_{ij}^{k} = \begin{cases} Q \text{ se a formiga } k \text{ passou entre as cidades } i \text{ e } j \text{ entre os tempos } t \text{ e } t+1 \\ 0 \text{ em outro caso} \end{cases} \quad (2.8)$$

No modelo de formiga-quantidade o valor de $\Delta\tau_{ij}^{k}$ é dado pela equação:

$$\Delta\tau_{ij}^{k} = \begin{cases} \dfrac{Q}{d_{ij}} \text{ se a formiga } k \text{ passou entre as cidades } i \text{ e } j \text{ entre os tempos } t \text{ e } t+1 \\ 0 \text{ em outro caso} \end{cases} \quad (2.9)$$

Segundo Dorigo (Dorigo et al. 1996) dos três modelos do AS, o modelo da formiga-ciclo apresentou melhores resultados empiricamente.

2.6 Exemplo de Aplicação do Sistema da formiga.

Para exemplificar uma aplicação do AS, foi utilizado um conjunto de quatorze pontos (cidades) aleatoriamente distribuídos como mostra a figura 2.4.

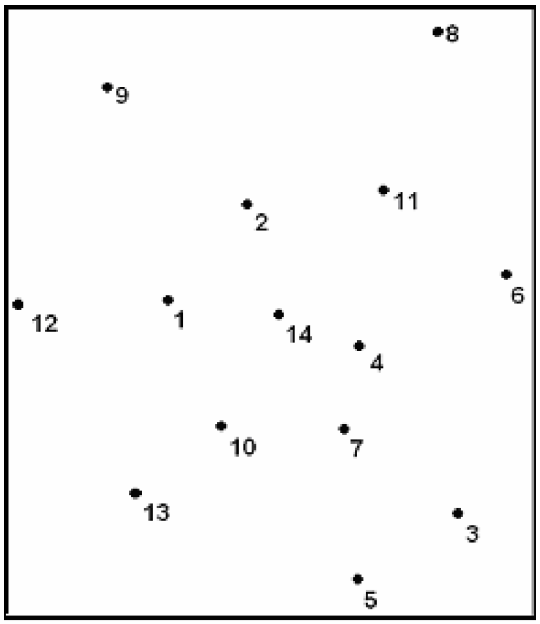


Figura 2.4: Representação de 14 pontos (cidades) em um plano

Inicialmente foram calculadas todas as permutações de caminhos de sete até quatorze cidades em um computador com um processador Atlon XP de 1,67 GHz e com 512 Mb de memória RAM. Os resultados obtidos estão expostos na tabela 2.2 que contem o valor do menor caminho, o número de permutações calculado e o tempo gasto para calcular todas as permutações.

Tabela 2.2: Resultados do cálculo de todas as permutações.

Nrº de cidades	Menor caminho	Nrº de permutações	Tempo
7	280	5.040	Inferior a 1 segundo
8	351	40.320	Inferior a 1 segundo
9	424	362.880	Inferior a 1 segundo
10	440	3.628.800	Inferior a 1 segundo
11	450	39.916.800	3 segundos
12	596	479.001.600	47 segundos
13	605	6.227.020.800	10 min 35 s
14	629	87.178.291.200	2h 29min 29s

Na tabela 2.2 quando o número de cidade é sete, significa que foi considerada a cidade de um até sete e assim sucessivamente para os outros números de cidades.

Calculando-se todas as permutações encontra-se com exatidão o menor caminho possível que é mostrado na tabela 2.2, porém o tempo cresce exponencialmente. Caso o número de cidades fosse igual a 15 o tempo seria de aproximadamente de 37,5 horas.

O AS modelo de ciclo foi aplicado ao mesmo conjunto de 14 cidades utilizando o mesmo computador. O número de ciclos aplicado foi de 500, a constante atribuída ao valor inicial da trilha de feromônio igual a dez, o valor da taxa de evaporação igual a 0,5 e os parâmetros livres de importância da trilha e importância da visibilidade foram iguais a um. O AS modelo ciclo encontrou os mesmos valores de menor caminho que os encontrados realizando todas as permutações, apresentados na tabela 2.2, sendo que o tempo gasto sempre foi inferior a um segundo.

A Figura 2.5 apresenta um gráfico que expõe o número mínimo de ciclos necessários para encontrar o menor caminho.

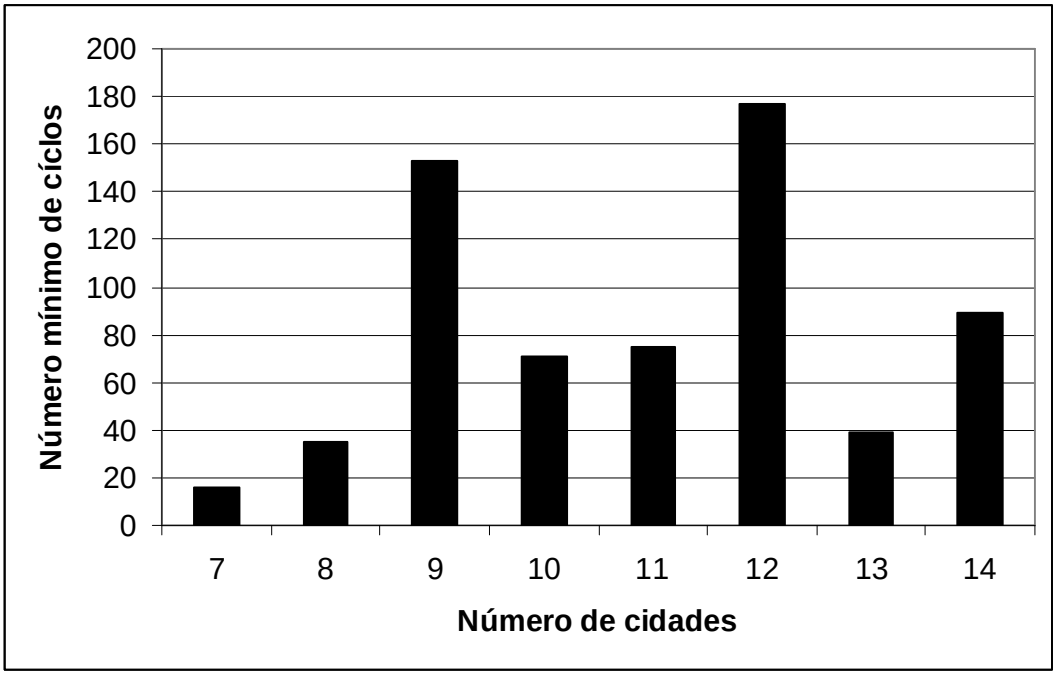


Figura 2.5: Gráfico com o número mínimo de ciclos.

Pela análise do gráfico, observa-se que com o aumento do número de cidades, o número mínimo de ciclos não aumenta obrigatoriamente. Isto se deve ao fato do processo ser estocástico.

A figura 2.6 apresenta o efeito sobre a trilha de feromônio enquanto o AS é aplicado ao conjunto de quatorze cidades. Na figura, a distância entre os pontos é proporcional à distância real e a espessura da linha é proporcional a quantidade de feromônio sobre a trilha. Inicialmente, (figura 2.6a) as trilhas de feromônio estão com uma distribuição uniforme, e a escolha do próximo ponto é guiada principalmente pela visibilidade. Após um certo número de simulações (figura 2.6b) as trilhas de feromônio começam a definir a melhor viagem, evaporando as trilhas que formariam uma viagem com resultados insatisfatórios. No final (figura 2.6c) o resultado com a construção da melhor viagem que representa o caminho com o menor comprimento.

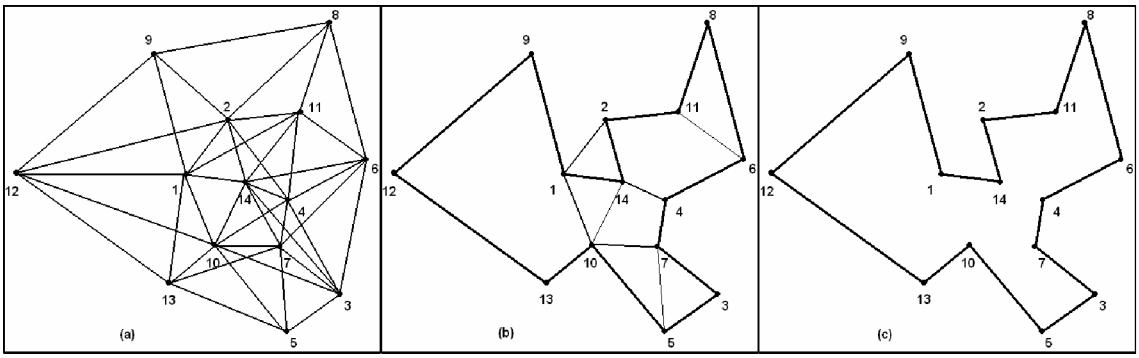


Figura 2.6: Evolução da trilha de feromônio aplicado ao grupo de 14 cidades. a) Nível das trilhas de feromônio no início (com 10 ciclos). b) Nível das trilhas de feromônio com 50 ciclos. c) Nível da trilhas de feromônio com 100 ciclos.

Aplicou-se também o mesmo conjunto de dados a uma alteração do algoritmo básico do AS. A alteração realizada no algoritmo foi a retirada de toda a parte que trabalha com a trilha de feromônio deixando apenas a parte que trabalha com a visibilidade. O objetivo era verificar se a trilha influencia diretamente no resultado do algoritmo. Resultado dessa alteração é mostrado na tabela 2.4.

Tabela 2.3: Resultados da aplicação do AS alterado.

Nrº de cidades	Menor caminho	Tempo	Nrº mínimo de cíclos
7	280	Inferior a 1 segundo	16
8	351	Inferior a 1 segundo	37
9	424	Inferior a 1 segundo	210
10	440	Inferior a 1 segundo	1.390
11	450	Inferior a 1 segundo	696
12	596	3 segundos	21.416
13	605	2 segundos	11.085
14	629	2 min 16 s	533.299

Observa-se pela tabela 2.4 que os resultados do menor caminho são os mesmos encontrados pelas duas maneiras anteriores e que os tempos mostram que o processo apesar de ser bem eficiente é mais lento que quando considerada a trilha de feromônio para número de cidades a partir de 12.

Baseado nos resultados apresentados na tabela 2.4 observa-se que utilização da trilha de feromônio no algoritmo do AS influencia positivamente na solução do problema.

2.7 Outros Algoritmos Baseados em Formigas.

Além do AS (Corlorni et al. 1991), existem outros algoritmos baseados em formigas que são aplicados ao PCV como o Sistema de Colônia de Formigas (Dorigo e Gambardella 1997), do inglês *Ant Colony System* (ACS) e o Sistema de Formigas Max-Min (Stützle e Hoos 2000), do inglês Max-Min *Ant System* (MMAS). A característica principal do ACS é que somente a formiga com o melhor desempenho atualiza a trilha de feromônio. Já no MMAS é introduzido um novo conceito de atualização local da trilha de feromônio, onde a equação (2.4) é substituída pela equação (2.10).

$$\tau_{ij}(t+1) = \rho \cdot \tau_{ij}(t) + (1-\rho) \cdot \Delta \tau_{ij} \qquad (2.10)$$

Além do PCV, existem outros problemas de otimização onde algoritmos baseados em formigas são aplicados, como o problema de roteamento de redes de comunicação, onde existem dois algoritmos chamados de AntNet (Di Caro e Dorigo 1998) e de ABC (Schoonderwoerd et al. 1997).

O problema de atribuição quadrática, do inglês *Quadratic Assignment Problem* (QAP), é outro que possui várias versões de algoritmos baseados em formigas como o HAS-QAP (Gambardella et al. 1999), o AS-QAP (Maniezzo e Colorni 1999) e o PAC-QAP (Talbi et al. 2001).

O roteamento de veículos é outro problema que possui adaptações de algoritmos baseados em formigas como o AS-VRP (Bullnheimer et al. 1999) e o ACS-DVRP (Montemanni et al. 2005).

Costa e Herz (1997) propuseram uma extensão do AS com aplicação no problema de coloração de grafos obtendo bons resultados comparados a outras heurísticas.

2.8 Características dos Algoritmos Inspirados em Formigas.

Os algoritmos inspirados no comportamento social das formigas possuem algumas características desejáveis como:

i. É versátil (Dorigo et al. 1996), pois pode ser aplicada para versões similares do mesmo problema, como por exemplo, as duas versões, simétrica e assimétrica, do problema do caixeiro viajante.
ii. É robusto (Dorigo et al. 1996), pois pode ser aplicado com poucas mudanças a outros problemas de otimização como o de atribuição quadrática e o roteamento de redes de comunicação.
iii. Pode ser transformado em um sistema paralelo (Manfrin et al. 2006). Com a tendência atual de aumentar o poder de processamento através de processamento em grupo de computadores e o aumento do número de núcleos em um só processador, essa característica pode aumentar ainda mais a velocidade da resolução de problemas.

iv. É invariante (Birattari et al. 2005). Birattari e seus colegas mostraram que as soluções produzidas pelos algoritmos de otimização inspirados em formigas não dependem da instância do problema sob o qual é analisado.

Capítulo 3

Agrupamento de Dados.

Quando crianças brincam de separar objetos redondos de objetos retangulares, elas estão realizando um agrupamento automático de objetos. Desta forma, o agrupamento de objetos faz parte do processo de aprendizagem da criança. Naturalmente os seres humanos vivem realizando um constante agrupamento de objetos de duas ou três dimensões com o auxílio da visão. Porém a natureza freqüentemente proporciona o aparecimento de problemas que envolvam mais de três dimensões, gerando assim a necessidade da procura por métodos que possam trabalhar com valores altos de dimensões.

A análise multivariada é o ramo da estatística que consiste em um conjunto de técnicas que podem ser usadas em situações onde cada objeto de um conjunto de dados possui várias dimensões, onde cada dimensão representa uma variável (Rencher 2002). Para fins de melhor compreensão considera-se nesta dissertação que objeto é um elemento, constituído de duas ou mais variáveis, pertencentes a um conjunto ou banco de dados.

Uma dessas técnicas multivariadas é a análise de grupos que consiste em um conjunto de ferramentas para separação de objetos de um banco de dados em grupos ou classes. O objetivo principal é formar grupos onde os elementos que constituem o grupo sejam os mais homogêneos possíveis entre si, e que os grupos sejam o mais heterogêneos entre si (Härdlee e Simar 2007).

A análise de agrupamento é considerada uma forma de classificação não supervisionada, porque os objetos são classificados (agrupados) sem nenhum conhecimento anterior sobre a existência dos grupos. Já a analise discriminante, que também é uma técnica multivariada, é considerada uma forma de classificação supervisionada, pois classifica os objetos em grupos pré-existentes. (Timm 2002).

3.1 Definição Formal do Problema de Agrupamento.

Formalmente o problema de agrupamento de dados pode ser definido como um problema de otimização como mostra a seqüência abaixo (Bilmes et al 1997).

- Exemplo: Um conjunto finito X, uma medida de distância d (i,j) $\in$ R^+ para todo i, j $\in$ X, dois inteiros positivos K e B, e uma função J (C, d(·,·)) definida sobre um K-partição, C = {$C_{1,}$, C_2, ..., C_k}, de X e da medida d(·,·).
- Pergunta: Há uma partição C_i de X dentro dos conjuntos disjuntos $C_{1,}$, C_2, ..., C_k tal que J (C_i, d(·,·)) ≤ B?
- Otimização: Encontrar a partição de X nos conjuntos disjuntos $C_{1,}$, C_2, ..., C_k que minimize a expressão J(C, d(·,·)).

Segundo Abraham e companheiros (2007) o número R de possíveis soluções para a divisão de um conjunto de dados formado por N objetos em K partições (grupos) é dado pela equação (3.1).

$$R(K,N)=\frac{1}{K!}\sum_{i=1}^{K}(-1)^{K-i}\binom{K}{i}(i)^N \qquad (3.1)$$

A tabela 3.1 expõe a quantidade de possíveis soluções considerando um exemplo com trinta objetos (N = 30) e com o número de partições K variando de dois até cinco. Observa-se pela tabela que o número de possíveis soluções cresce rapidamente com pequenos acréscimos no valor de K. Por esse motivo o problema de agrupamento de dados é conhecido na literatura como um problema NP completo (Bilmes et al. 1997).

Tabela 3.1: Quantidade de possíveis soluções para um conjunto com 30 objetos.

K=2	K=3	K=4	K=5
536.870.911	34.314.651.811.530	48.004.081.105.038.305	7.713.000.216.608.565.075

3.2 Aplicações.

Jain et al. (1999) afirmam que os algoritmos de agrupamentos de dados podem ser usados em uma grande variedade de aplicações como as seguintes:

i. Segmentação de imagens, que é definida como uma exaustiva partição de uma imagem em algumas regiões homogêneas com respeito a alguma propriedade de interesse como cor, intensidade ou textura. A segmentação de imagens é um componente fundamental em muitas aplicações de visão digital;
ii. Reconhecimento de objetos, que é o uso de agrupamento de dados de imagens para classificar visões de objetos em três dimensões;
iii. Recuperação de informações, cujo objetivo é automática classificação e guarda de documentos para recuperação futura;
iv. Mineração de dados. A cada dia há um grande aumento de dados coletados de todos os tipos. A mineração de dados é responsável por procurar informações importantes no meio dessa grande quantidade de dados.

3.3 Componentes da Tarefa de Agrupamento.

A típica tarefa de agrupamento de dados envolve os seguintes passos: (Jain e Dubes 1999).

- Escolha das variáveis e obtenção dos dados;
- Tratamento dos dados;
- Definição de um critério de similaridade ou de dissimilaridade;
- Escolha e aplicação de um algoritmo de agrupamento de dados;
- Avaliação dos resultados.

A escolha das variáveis e a obtenção dos dados estão por natureza inseparavelmente ligados ao campo da aplicação do problema em estudo e depende da experiência e bom senso do pesquisador sobre a importância das variáveis e a

forma de obtenção dos dados. A escolha das variáveis é um dos fatores que mais influencia no resultado de um agrupamento de dados. Se o pesquisador optar por escolher variáveis que assumem praticamente o mesmo valor para todos os objetos, sua inclusão pouco contribuirá para a determinação dos agrupamentos, pois são pouco discriminatórias. Se o pesquisador optar por escolher variáveis altamente discriminantes, mas irrelevantes ao problema, poderá mascarar os grupos e obter resultados equivocados (Bussab et al.1990).

O tratamento de dados e a escolha de um critério de similaridade ou de dissimilaridade serão estudados respectivamente nos itens 3.4 e 3.5, dada a necessidade de tratar as variáveis que podem aparecer em várias escalas diferentes, e de ter que reuni-las em um único índice de similaridade ou dissimilaridade.

No item 3.6 será vista a classificação de alguns algoritmos de agrupamentos e no quarto capítulo serão estudados alguns algoritmos juntamente com algumas de suas características inerentes.

Dois métodos de avaliação de resultados gerados por algoritmos de agrupamento dados serão discutidos no item 3.8.

3.4 Estandardização e Normalização dos Dados.

A estandardização é um pré-tratamento dos dados originais que realiza uma transformação desses dados em função das variações inerentes ao sistema.

Na tabela 3.2 temos um exemplo (Neto e Moita, 1997) que apresenta intervalos de algumas variáveis químicas, que explicitará a importância desse pré-tratamento dos dados originais.

Tabela 3.2: Exemplo de intervalos de variáveis químicas.

Variável química	Intervalo
Densidade relativa	0,919 – 0,925
Índice de refração	1,466 – 1,470
Índice de saponificação	189 – 195
Índice de iodo	120 - 143

Observando-se a tabela 3.2, nota-se que a variação de amplitude da variável do índice de refração é de 0,004 enquanto que a variação de amplitude do índice de iodo é de 23. Uma diferença de 0,002 no índice de refração representa uma variação de 50% enquanto que uma diferença 0,002 no índice de iodo representa uma variação inferior a 0,01% tornando-se praticamente desprezível.

Uma forma de resolver este problema é apresentada por Jain e Dubes (1988) de acordo com a equação (3.2) que transforma os valores originais em novos valores com média zero e variância igual a um. Na equação (3.2), x' é novo valor transformado da variável, x é o valor original, $\bar{x}$ é a média da variável x e S_x é o desvio padrão da variável x.

$$x' = \frac{x - \bar{x}}{S_x} \tag{3.2}$$

Outra forma de transformação de dados é utilizada por Güngör e Ünler (2007). Essa transformação é realizada de acordo com a equação (3.3), onde x' é novo valor transformado da variável, x é o valor original, x_{min} é o valor mínimo da variável x e o x_{max} é o valor máximo. A mesma, consiste na normalização dos dados deixando os valores de cada variável entre zero e um.

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{3.3}$$

Existe uma terceira forma de transformação que toma a média como fator normalizador e é definida pela expressão (3.4) onde, x' é novo valor transformado da variável, x é o valor original, $\bar{x}$ é a média da variável x (Bussab et al. 1990).

$$x' = \frac{x}{\bar{x}} \tag{3.4}$$

Com respeito a quantidade de propostas de transformações, Bussab et al. (1990) recomenda que a escala de variáveis seja definida utilizando-se o bom-senso e o conhecimento da área de aplicação.

3.5 Medidas de Similaridade e Dissimilaridade.

A maioria das técnicas de agrupamento de dados utiliza um critério que quantifica o quanto dois objetos são similares ou dissimilares. Nas medidas de similaridade, quanto maior o valor, mais parecidos são os objetos. Já nas medidas dissimilaridade, quanto maior o valor, menos parecidos são os objetos. Um exemplo de medida de similaridade é o coeficiente de correlação. A distância euclidiana é um bom exemplo de medida de dissimilaridade (Bussab et al. 1990).

Pedrycz (2005) cita várias formas de calcular distâncias entre dois objetos X e Y, sendo cada objeto formado de N variáveis ou dimensões. Dentre elas algumas estão descritas nos itens subseqüentes.

3.5.1 Requisitos das Funções de Distâncias.

As funções de cálculo de distâncias que serão descritas do item 3.5.2 ao item 3.5.7 devem satisfazer às seguintes condições:

i. $d(X,Y) \geq 0$;

ii. $d(X,X) = d(Y,Y) = 0$;

iii. $d(X,Y) = d(Y,X)$;

iv. $d(X,Y) \leq d(X,Z) + d(Z,Y)$, com Z sendo um terceiro objeto.

O primeiro requisito afirma que a distância não pode ser um valor negativo. O segundo requisito diz que a distância de um objeto a ele mesmo é zero. O terceiro requisito afirma que as distâncias são sempre simétricas. O quarto e último requisito que deve ser satisfeito é a desigualdade triangular.

3.5.2 Distância Euclidiana.

A distância euclidiana é a mais conhecida das distâncias e a mais utilizada devido a este fato. Ela é calculada pela equação (3.5).

$$d(X,Y)=\sqrt{\sum_{i=1}^{N}(X_i-Y_i)^2} \tag{3.5}$$

3.5.3 Distância Euclidiana Média.

Segundo Bussab et al. (1990) a distância euclidiana média é um escalonamento da distância euclidiana, possuindo apenas uma propriedade a mais. Essa propriedade adicional é capacidade de poder ser utilizada quando o conjunto de dados apresentar ausência de algumas coordenadas. A distância euclidiana média é calculada pela equação (3.6).

$$d(X,Y)=\sqrt{\frac{\sum_{i=1}^{N}(X_i-Y_i)^2}{N}} \tag{3.6}$$

3.5.4 Distância Absoluta.

A distância absoluta é também conhecida como distância Manhattan (Kaufman e Rousseeuw, 1990) é definida pela expressão (3.7).

$$d(X,Y)=\sum_{i=1}^{N}\left|X_i-Y_i\right| \tag{3.7}$$

3.5.5 Distância Minkowski.

A distância de Minkowski é dada pela equação (3.8), onde P é um valor positivo. Essa distância é na realidade uma família de distâncias. Se P for igual a

um, a distância de Minkowski torna-se a distância de Manhattan. Se P for igual a dois ela se torna a distância euclidiana.

$$d(X,Y)=\sqrt[P]{\sum_{i=1}^{N}(X_i-Y_i)^P},\ p>0 \tag{3.8}$$

3.5.6 Distância Tchebychev.

A distância de Tchebychev é dada pela equação (3.9), onde representa o valor máximo do módulo das diferenças das coordenadas X_i e Y_i com i variando de 1 até N.

$$d(X,Y)=\max_{i=1,2,\ldots,N}\left|X_i-Y_i\right| \tag{3.9}$$

3.5.7 Distância Binária de Sokal.

As distâncias discutidas do item 3.6.1 ao item 3.6.5 são destinadas a variáveis do tipo quantitativa. Já distância binária de Sokal é utilizada para calcular a distância de variáveis binárias do tipo sim ou não (um ou zero). Para ilustrar o uso da distância binária de Sokal será utilizado o exemplo fictício apresentado na tabela 3.3 que representa dois objetos (X e Y) composto de dez variáveis binárias (Bussab et al. 1990).

Tabela 3.3: Exemplo ilustrativo para calculo da distância de Sokal.

Variável	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
Objeto X	1	0	1	1	0	1	1	0	0	0
Objeto Y	1	1	0	1	0	1	0	1	1	1

A tabela 3.4 apresenta o número de pares (1,1), (0,0), (1,0) e (0,1) observados na tabela 3.3.

Tabela 3.4: Número de pares observados na tabela 3.3.

Pares	(1,1)	(0,0)	(1,0)	(0,1)
Quantidade	a = 3	b = 1	c = 2	d = 4

A distância de Sokal é dada pela expressão (3.10), onde a é o número de pares (1,1), b é o número de pares (0,0), c é o número de pares (1,0) e c é o número de pares (0,1).

$$d(X,Y)=\sqrt{\frac{c+d}{a+b+c+d}} \tag{3.10}$$

A expressão (3.10) quantifica um valor que é proporcional ao número de atributos não coincidentes nos dois objetos, ou seja, quanto mais os objetos forem distintos, maior será o valor da distância. Aplicando os valores da tabela 3.4 na expressão (3.10) obtemos como resposta aproximadamente 0,77. Vale salientar também que a expressão (3.10) só retorna valores no intervalo de zero até um.

3.6 Classificação dos Algoritmos de Agrupamento.

A execução de algoritmos (ou técnicas) de agrupamento é o fundamental componente da tarefa de agrupar dados, pois são os responsáveis em cumprir o objetivo principal que é encontrar e separar os objetos em grupos similares. É interessante destacar que os algoritmos de agrupamento também podem ser agrupados, ou classificados, das formas seguintes: técnica hierárquica, técnica de partição, técnica baseada em densidade e técnicas diversas de agrupamento.

3.6.1 Técnicas Hierárquicas.

A técnica hierárquica é um método tradicional de agrupamento de dados que estabelece uma ordem ou subordinação entre os grupos e pode ser subdividida em dois tipos: de aglomeração e de divisão. A de aglomeração começa considerando cada objeto como um grupo e realiza sucessivas uniões até que todos os objetos sejam englobados por um único grupo. Já a de divisão, realiza o inverso, ou seja,

parte de um único grupo que contém todos os objetos e aplica sucessivas divisões até que cada objeto seja um grupo (Afifi e Clarck 1997).

Alguns exemplos de técnicas hierárquicas são: método das médias das distâncias, método da ligação simples também conhecida como método do vizinho mais próximo, e método da ligação completa também conhecida como método do vizinho mais longe (Bussab et. al). Normalmente os resultados da utilização de alguma técnica hierárquica são representados em um dendrograma que é um diagrama bidimensional em forma de árvore, como mostra a figura 3.1, onde cada ramo representa um objeto e a raiz representa o agrupamento de todos os objetos(Albuquerque 2005).

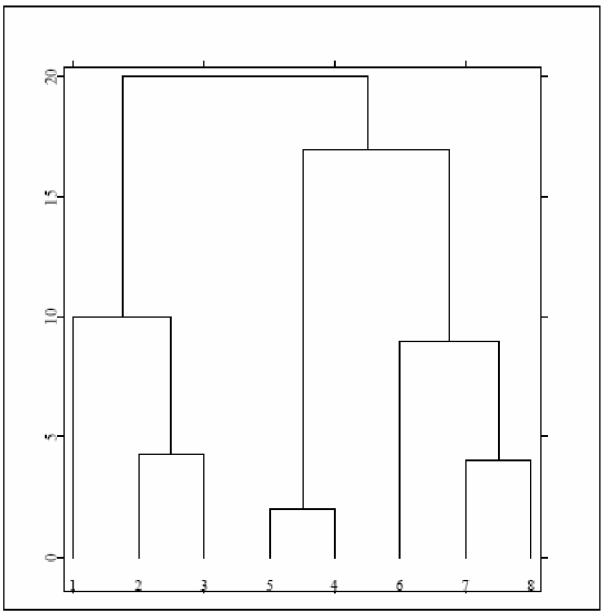


Figura 3.1: Exemplo de dendrograma.

A maior parte das aplicações de agrupamento de dados nos campos da biologia e zoologia utiliza técnicas hierárquicas de aglomeração. Essa técnica é particularmente útil para animais e plantas que são hierarquicamente agrupados com respeito a características genéticas (Everitt 1977).

Kogan et al. (1998) descrevem algumas vantagens e desvantagens das técnicas hierárquicas. Entre as vantagens tem-se: fácil manejo de qualquer medida de similaridade ou de dissimilaridade, aplicabilidade de qualquer tipo de atributo e fácil visualização dos objetos através de dendrogramas. Já as desvantagens são: a dificuldade de escolha do correto critério de parada e a não relocação de objetos

entre grupos já divididos ou aglomerados, durante o processo de hierarquização dos objetos.

3.6.2 Técnicas de Partição.

A técnica de partição é outro método tradicional de agrupamento de dados que tenta dividir os objetos em certa quantidade *k* de grupos, obedecendo algum critério de adequação do agrupamento dos objetos. O valor de *k* deve ser conhecido previamente antes da execução do algoritmo. (Abonyi e Feil 2007).

Existem duas diferenças básicas entre as técnicas hierárquicas e as técnicas de partição: a primeira é que a técnica hierárquica não precisa conhecer o número de grupos antes da execução da técnica, isto é importante quando não se tem idéia sobre quantidade de grupos existentes; a segunda é que as técnicas de partição vão realizando relocações dos objetos entre os grupos, pondo em prática uma reavaliação dos grupos obtidos até o cumprimento de um critério de parada.

As técnicas de partição divergem umas das outras nos seguintes procedimentos: no método de iniciar os agrupamentos; no método de alocar os objetos aos agrupamentos iniciais; e no método de relocar um ou mais objetos já agrupados para outros agrupamentos. Algumas das técnicas por partição mais conhecidas são K-Médias e K-Medóides (Bussab et al 1990).

As técnicas de partição têm uma vantagem em relação às técnicas hierárquicas, quando a aplicação envolve um conjunto de dados com um grande número de objetos, onde a construção de dendrogramas é computacionalmente proibitiva. A desvantagem das técnicas de partição está exatamente na escolha do correto número de grupos em conjunto de dados onde esta informação é desconhecida (Jain et al. 1999).

3.6.3 Técnicas Diversas de Agrupamento.

Englobam um conjunto de técnicas de agrupamento de dados não tão tradicionais como as hierárquicas ou as de partição. Uma dessas técnicas forma grupos procurando por regiões que contenham uma alta concentração de objetos e é conhecida como técnica de densidade. Outra técnica trabalha com objetos onde os grupos estão sobrepostos. Existem ainda técnicas que não se enquadram em

nenhuma outra ou que são combinações de outras técnicas, aproveitando assim as melhores características de cada técnica (Kogan et al. 1998).

Nos últimos anos tem surgido uma nova classe de técnicas de agrupamento de dados que são baseadas no comportamento social de insetos.

O primeiro algoritmo de agrupamento de dados baseado no comportamento de formigas foi proposto por Deneubourg, em 1990, inspirado na conduta de uma espécie de formiga conhecida como *Pheidole pallidula* que organiza corpos de formigas mortas em grupos como se fosse um cemitério. A figura 3.2 mostra um exemplo dessa conduta onde formigas trabalhadoras agrupam 1500 corpos inicialmente espalhados aleatoriamente durante um período de 36 horas (Dorigo et al. 2000).

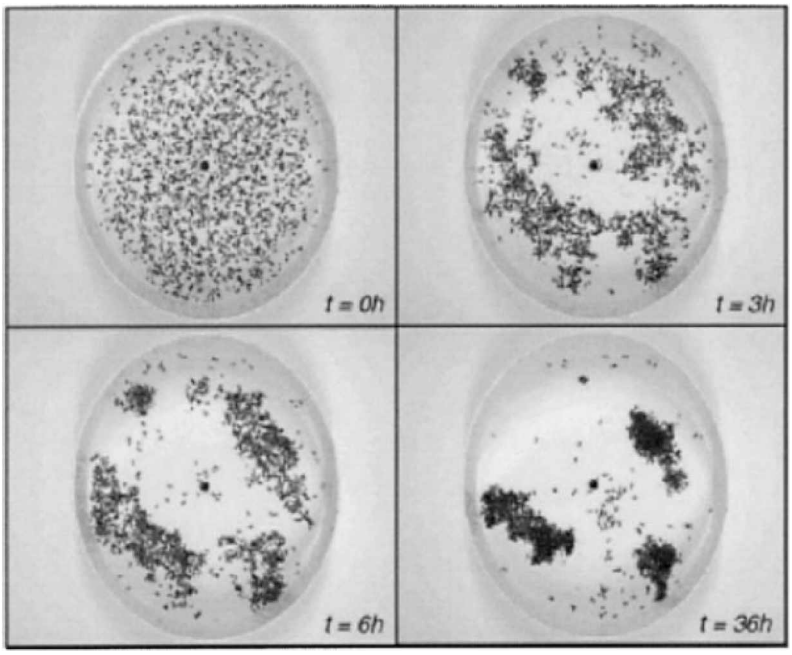


Figura 3.2 Formigas reais agrupando corpos de formigas mortas.

O algoritmo de agrupamento de dados proposto por Deneubourg foi utilizado em um programa de computação para análise de dados bancários (Bonabeau e Théraulaz 2000).

Outras propostas de algoritmos baseados no comportamento de formigas surgiram após o algoritmo de Deneubourg, como por exemplo: o AntClass (Monmarché et al. 1999) que é um algoritmo híbrido entre o algoritmo de Deneubourg e o K-Médias, o AntClust (Labroche et al. 2003), o AntTree (Azzag et al.

2004) e o ACODF (Tsai et al. 2004) onde a sigla significa otimização por colônia de formigas com favorecimento diferenciado. O AntClass, o AntClust e o AntTree são semelhantes ao algoritmo de Deneubourg pois têm os mesmos fundamentos baseados no comportamento de agrupamentos de corpos, e não utilizam a trilha de feromônios. Já o ACODF é diferente dos demais não utilizando o comportamento de agrupar corpos e empregando a trilha de feromônio para formar os grupos.

3.7 Avaliação dos Resultados.

Após a execução de algum algoritmo de agrupamento de dados, existe a necessidade de avaliar se o resultado do agrupamento gerado tem uma boa qualidade, e se esse resultado é melhor do que do resultado gerado por outro algoritmo. Uma forma de resolver esse problema é realizar comparação entre um conjunto de dados cuja classificação seja conhecida previamente e o resultado produzido pelos algoritmos de agrupamento de dados. Abaixo serão descritas duas formas de implementar essa comparação.

3.7.1 Medida F.

O valor da medida F dá uma idéia da precisão da recuperação de informação gerada pelo algoritmo. Partindo-se dos grupos i, que são os grupos cuja classificação é conhecida e que serão utilizadas como referência, e dos grupos j, que são os grupos gerados pelo algoritmo, o valor da medida F será a média harmônica de dois valores: P (i, j) e R (i, j). P (i, j) = n_{ij} / n_j, onde n_{ij} é o número de elementos do grupo i que também pertencem ao grupo j e n_j é o número de elementos do grupo j. R (i, j) = n_{ij} / n_i, onde n_i é o número de elementos do grupo i. Para cada grupo i e j, o valor de F (i, j) é dado pela equação (3.11) (Oca et al. 2005).

$$F(i,j)=\frac{2\cdot P(i,j)\cdot R(i,j)}{P(i,j)+R(i,j)} \tag{3.11}$$

O valor final da medida F para o agrupamento gerado pelo algoritmo é dado pela expressão (3.12).

$$F = \sum_i \frac{n_i}{n} \cdot \max_j \{F(i, j)\} \tag{3.12}$$

O valor de n é o número total de objetos pertencentes ao banco de dados. O valor de F varia no intervalo fechado de zero a um. Quanto mais próximo da unidade melhor o resultado do agrupamento gerado pelo algoritmo.

3.8.2 Índice de Rand.

O índice de Rand (1971) determina o grau de similaridade entre agrupamento cuja classificação é conhecida e que será utilizado como referência e o agrupamento gerado pelo algoritmo de agrupamento de dados. O valor do índice de Rand é dado por:

$$R = \frac{a+d}{a+b+c+d} \tag{3.13}$$

Dado A o conjunto de todos os objetos sem qualquer tipo de classificação e B o conjunto onde os elementos são todas as combinações dois a dois dos elementos de A, têm-se que:

- a é o número de elementos do conjunto B que estão em um mesmo grupo nos dados que servem de referência, e que também estão em um mesmo grupo no resultado gerado pelo algoritmo;
- b é o número de elementos do conjunto B que estão em um mesmo grupo nos dados que servem de referência, e que não estão em um mesmo grupo no resultado gerado pelo algoritmo;
- c é o número de elementos do conjunto B que não estão em um mesmo grupo nos dados que servem de referência, mas que estão em um mesmo grupo no resultado gerado pelo algoritmo;

- d é o número de elementos do conjunto B que não estão em um mesmo grupo nos dados que servem de referência, e que também não estão em um mesmo grupo no resultado gerado pelo algoritmo;

Observa-se que a + b + c + d = n (n - 1)/2, onde n é o número total de objetos do conjunto de dados.

Como a Medida F, o índice de Rand também varia no intervalo fechado de zero até um e quanto maior seu valor melhor o agrupamento gerado pelo algoritmo.

Para melhor esclarecer o cálculo do índice de Rand será utilizado um exemplo fictício. Dado um conjunto A = {a, b, c, d, e, f}, composto de dois grupos conhecidos {a, b, c} e {d, e, f} que servirão de referência. Os grupos gerados pelo algoritmo são {a, b, d} e {c, e, f}. Partindo-se do conjunto A constrói-se o conjunto B ={(a,b), (a,c), (a,d), (a,e), (a,f), (b,c), (b,d), (b,e), (b,f), (c,d), (c,e), (c,f), (d,e), (d,f), (e,f)}. De posse dessas informações monta-se a tabela 3.5.

Tabela 3.5: Determinação dos valores necessários para o cálculo do valor do Índice de Rand.

	a,b	a,c	a,d	a,e	a,f	b,c	b,d	b,e	b,f	c,d	c,e	c,f	d,e	d,f	e,f	
a	*														*	a=2
b		*				*							*	*		b=4
c			*				*				*	*				c=4
d				*	*			*	*	*						d=5

De posse dos dados da tabela 3.5 obtém-se aproximadamente o valor de R=0,467.

Capítulo 4

Algoritmos para Agrupamento de Dados

Existe uma grande variedade de algoritmos de agrupamentos de dados. Kaufman e Rousseeuw (1990) citam três motivos principais que podem justificar essa grande variedade. O primeiro motivo é que o estudo de agrupamentos de dados é uma área muito jovem e em grande crescimento. Esse crescimento pode ser notado pela grande quantidade de artigos espalhados em muitos periódicos (principalmente em jornais de estatística, biologia, ciência da computação e comércio). O segundo motivo para a grande diversidade de algoritmos é que não existe uma definição geral de grupos, e na realidade existem vários tipos de grupos como os lineares e os esféricos. O terceiro motivo é que diferentes algoritmos trabalham usando diferentes tipos de dados, como variáveis contínuas, variáveis discretas, medidas de similaridades e medidas de dissimilaridades.

Neste capítulo serão estudados três algoritmos de agrupamentos: o algoritmo K-Médias que é um dos mais conhecidos algoritmos de agrupamentos; o ACBHO que é um algoritmo baseado em diferentes métodos de otimização; e uma nova proposta de algoritmo que utiliza o Método de Monte Carlo (MMC) juntamente com a teoria dos algoritmos baseados em formigas.

Antes de iniciar o estudo dos três algoritmos citados, será feito uma breve explanação sobre o Método de Monte Carlo que é uma técnica necessária aos três algoritmos e também será utilizada para geração de valores que servirão de referência para testar os três algoritmos no próximo capítulo.

4.1 Método de Monte Carlo.

Segundo Sobol (1972) o MMC é um método de solução numérica cuja base é essencialmente a simulação de variáveis aleatórias. A origem de seu nome é devido às roletas dos cassinos da famosa cidade de Monte Carlo no principado de Mônaco,

pelo fato das roletas serem um dos dispositivos mecânicos mais simples de geração de números aleatórios que são extremamente necessários para o MMC.

Uma experiência que utiliza a idéia do MMC foi realizada por volta do ano 1777 por George Louis-Leclerc conhecido como o conde de Buffon. Ele utilizou papel com linhas e agulhas para estimar um valor aproximado para o número π (Beichl e Sullivan 2006).

O MMC tem oficialmente sua origem em 1949 com a publicação do artigo "The Monte Carlo Method" (Metropolis e Ulan 1949), e vem se ampliando e ganhando novas aplicações com o auxílio dos modernos computadores que geram grandes quantidades de números aleatórios e realizam o processamento de muitos cálculos rapidamente.

Atualmente o MMC tem uma vasta gama de áreas de aplicações como física, química, medicina, economia entre outras, e se baseia na Lei dos Grandes Números e no Teorema Central do Limite.

Uma forma de ilustrar a utilização do MMC para a estimação do valor de π pode ser vista na figura 4.1. Esta figura possui um círculo de raio 0,5 inteiramente contido em um quadrado de lado um. De posse do raio e da fórmula da área do círculo temos que o valor de $\pi=4 \cdot A$, onde A é a área do círculo.

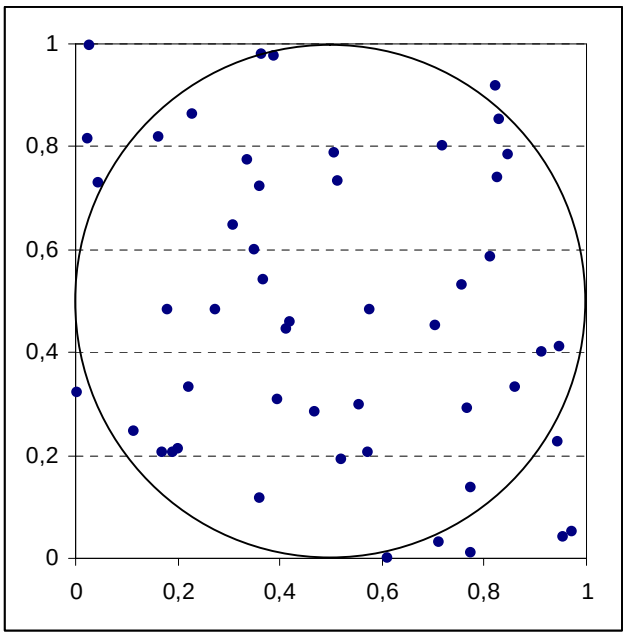


Figura 4.1: Quadrado de área um com circunferência inscrita e 50 pontos aleatórios.

Tendo-se N como o número total de pontos aleatórios no interior do quadrado e N' o número de pontos aleatórios pertencentes ao círculo, estima-se o valor da área A pelo quociente N'/N. No caso da figura 4.1, tem-se N = 50 e N' = 39, obtendo-se o valor A = 0,78. De posse de A calcula-se π como 3,12 que é uma boa estimativa para o valor de pi.

Sobol (1972) também afirma que o a precisão do resultado aumenta com o aumento do número de provas e que o erro inerente ao MMC é proporcional a $\sqrt{D/N}$ onde D é uma certa constante e N é o número de provas. Desta forma para diminuir o erro em dez vezes, e por conseqüência obter um algarismo a mais de precisão no resultado, deve-se multiplicar o número de provas N por cem, aumentando-se assim o tempo computacional em cem vezes também.

4.2 K-Médias.

O algoritmo K-Médias (Hartigan, 1975) é um dos mais conhecidos e utilizados algoritmos de agrupamento de dados. Por ser um método de partição, ele divide o conjunto de dados em k grupos. O K-Médias recebe este nome porque ele calcula a centróide de cada um dos k grupos pela média ponderada dos seus objetos.

4.2.1 Algoritmo do K-Médias.

O algoritmo básico do K-Médias é relativamente simples e é dado pelos seguintes passos:

i. Distribua aleatoriamente todos os objetos aos k grupos;
ii. Calcule a centróide de cada grupo;
iii. Para cada objeto calcule a distância entre ele e os centros de todos os grupos, atribuindo esse objeto ao grupo mais próximo;
iv. Caso haja alguma movimentação de objeto de um grupo para o outro no passo iii, volte para o passo ii;
v. Saída com o resultado do agrupamento.

4.2.2 Exemplo de Aplicação do K-Médias.

Na figura 4.2 encontra-se uma pequena seqüência de interações para exemplificar a utilização do K-Médias no agrupamento de dez objetos.

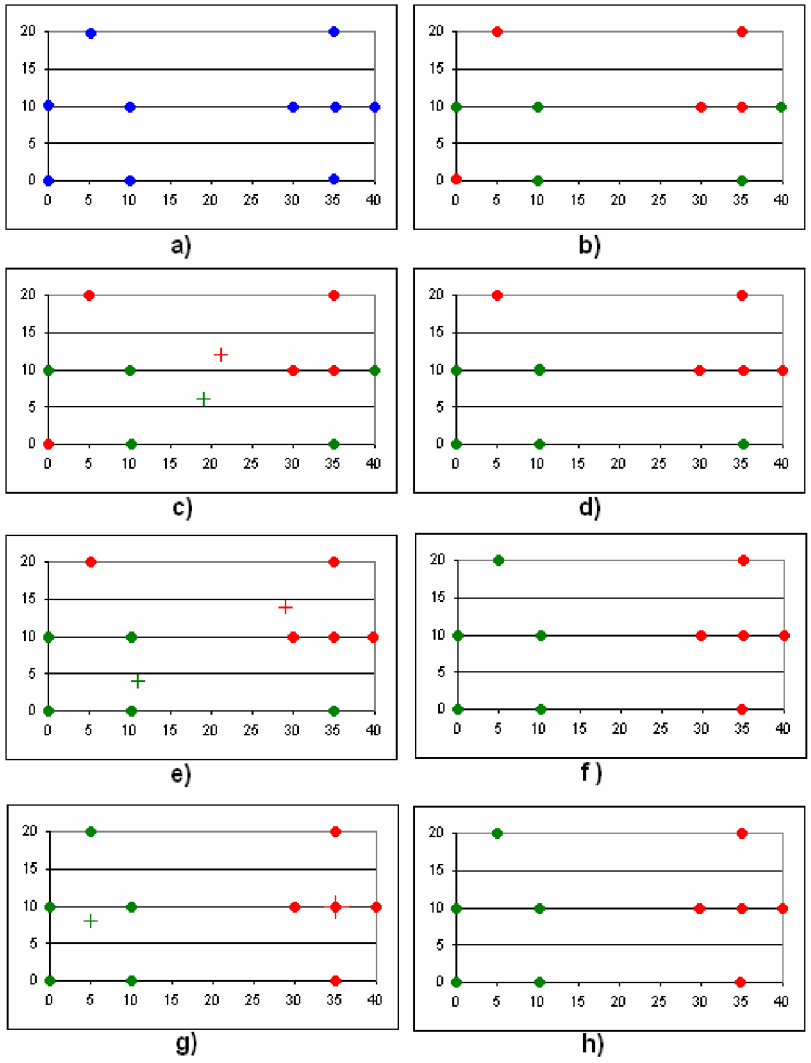


Figura 4.2: Exemplo de aplicação do K-Médias. a) Objetos inicias. b) Partição aleatória. c) Cálculo dos centros dos grupos. d) Agrupamento baseado nos centros. e) Cálculo dos novos centros. f) Novo agrupamento baseado nos novos centros. g) Re-cálculo dos novos centros. h) Agrupamento baseado nos últimos cálculos de centros, onde não há mudança de objetos.

Na figura 4.2 a) têm-se dez objetos no plano que serão divididos em dois grupos. Na figura 4.2 b), a partição aleatória dos objetos em dois grupos. Na figura 4.2 c) encontram-se destacados os centróides de cada grupo. Na figura 4.2 d)

encontra-se a partição dos objetos após a verificação da proximidade aos centróides de cada grupo. Como houve mudança de objetos entre os grupos da figura b para figura d, a figura 4.2 e) destaca os novos centros. Na figura 4.2 f) a nova partição dos objetos após a verificação da proximidade aos centróides de cada grupo. Como novamente houve mudança de objetos entre grupos, a figura 4.2 g) mostra destacados os novos centros re-calculados. A figura 4.2 h) mostra o agrupamento baseado no último re-cálculo onde se observa que não houve mudança de objetos entre os grupos, finalizando assim a execução do algoritmo.

4.2.3 Limitações do K-Médias.

Handl (2003) descreve algumas limitações que o algoritmo K-Médias possui:

- O resultado final é altamente dependente da partição inicial, o que torna o K-Médias propenso a convergir para soluções locais;
- O número de grupos K deve ser informado inicialmente como um parâmetro de entrada, o que pode ser um problema em aplicações onde o correto número de grupos não é conhecido antecipadamente;
- O K-Médias gera freqüentemente grupos vazios. Se grupos vazios são gerados o algoritmo é reinicializado para forçar a geração de exatamente K grupos;
- Devido ao uso do critério de mínima variância, o K-Médias apresenta melhor desempenho na identificação de grupos na forma esférica. Em grupos com outras formas o K-Médias pode ser completamente ineficaz;
- O K-Médias requer o cálculo do centro dos grupos, isto somente pode ser aplicado em conjunto de dados numéricos;
- O algoritmo não é eficiente para conjunto de dados com altas dimensões, uma vez que a distância entre os centros de cada grupo e todos os objetos do conjunto de dados tem que ser recalculada em cada interação.

4.3 O ACBHO.

O ACBHO (Sinha et. al, 2007) é um algoritmo que utiliza a otimização por colônia de formigas juntamente com outros processos de otimização para realizar a tarefa de agrupar dados. Além da otimização por colônia de formigas o ACBHO utiliza os seguintes processos de otimização: seleção por torneio, recozimento simulado e busca proibitiva.

O *tournament selection*, ou em português, seleção por torneio, é uma parte integrante dos algoritmos genéticos. A seleção por torneio funciona basicamente da seguinte maneira: dentre toda população de indivíduos escolhe-se aleatoriamente *n* indivíduos, e o melhor dentre esses *n* indivíduos escolhidos é o selecionado. O ACBHO utiliza a seleção por torneio para selecionar qual o caminho que a formiga irá seguir, e funciona da seguinte forma: dadas todas as trilhas que a formiga poderá seguir, ela escolherá algumas possibilidades aleatoriamente, e dentre essas, a trilha que tiver maior nível de feromônio será a escolhida para a formiga artificial seguir.

O *simulated annealing*, ou em português, recozimento simulado, é um processo de otimização que tem seu fundamento na termodinâmica. O conceito está baseado na maneira de congelar líquido ou de re-cristalizar metais no processo de recozer. Nesse processo os objetos são aquecidos a altas temperaturas e depois resfriados lentamente até atingirem aproximadamente o equilíbrio termodinâmico. Segundo Sinha et al. (1997), o conceito da simulação de recozimento é utilizado no ACBHO para encontrar a solução rapidamente através de um decréscimo do número de cidades a ser visitadas pelas formigas artificiais a cada nova interação.

O *tabu search*, ou em português busca proibitiva, é um processo que proíbe ou penaliza certos movimentos, evitando que o algoritmo procure soluções em locais já investigados. De acordo com Sinha et al. (1997), o conceito da procura proibitiva é utilizada no ACBHO para restringir o movimento das formigas artificiais evitando que elas retornem aos pontos previamente visitados.

4.3.1 O Algoritmo do ACBHO.

Segundo Sinha et al. (1997), o algoritmo do ACBHO é composto de quatro procedimentos: inicialização, primeira viagem, viagem seletiva e formação de grupos.

No procedimento de inicialização todos os parâmetros do ACBHO são iniciados realizando as seguintes tarefas: as coordenadas dos N objetos são lidas, o número de formigas artificiais é calculado como sendo metade do número de objetos, os valores inicias de feromônio de todas trilhas são zerados e as N/2 formigas são postas aleatoriamente em diferentes objetos sendo que duas ou mais formigas não podem estar no mesmo objeto.

O procedimento chamado de primeira viagem é usado para iniciar o movimento das formigas. Sinha et al. (1997) afirma que não há a necessidade das formigas visitarem todos os objetos e sim apenas metade deles para se obter resultados satisfatórios. Os objetos a serem visitados pelas formigas são escolhidos aleatoriamente e a busca proibitiva é utilizada para evitar que uma formiga visite um objeto já anteriormente visitado pela mesma. Depois da visita da formiga a um determinado objeto é adicionada a trilha, que liga os dois objetos, uma quantidade de feromônio que é o inverso da distância percorrida entre os objetos. Após todas as formigas completarem as suas viagens o número de objetos a serem visitados pelas formigas no próximo procedimento é calculado pelo produto de 0,95 pelo número de objetos anteriormente visitados que é N/2, aplicando assim o conceito de simulação de recozimento (Sinha et al. 1997).

O próximo procedimento, chamado de viagem seletiva, é onde as formigas selecionam os objetos a serem visitados utilizando a seleção por torneio. Cada formiga irá escolher aleatoriamente dez ou quinze trilhas dentre as permitidas, e a que tiver maior valor de feromônio será a escolhida para a formiga seguir viagem. Logo após, será adicionada a trilha escolhida, e uma quantidade de feromônio que é o inverso da distância percorrida entre os objetos. Cada vez que o procedimento de viagem seletiva é executado o novo número de objetos a ser visitado é a metade do anterior aplicando assim novamente o conceito de simulação de recozimento. Sinha et al. (1997) afirma que precisão dos agrupamentos depende do número de vezes

que o procedimento de viagem seletiva é executado e que vários estudos o levou a fixar esse número em dois.

O último procedimento é o responsável pela definição dos agrupamentos. Neste procedimento determina-se um valor satisfatório da trilha de feromônio, onde trilhas que possuem quantidade de feromônios maior ou iguai ao valor arbitrado são consideradas visíveis e trilhas com valores menores que este, são consideradas não visíveis. Grupos formados por pequenas quantidades de objetos são unidos a grupos adjacentes que possuem um maior número de objetos.

4.4 Um Novo Algoritmo para Agrupamento de Dados Baseado em Formigas.

A nova proposta consiste em construir agrupamento de dados com o uso de trilhas de feromônios de forma semelhante aos algoritmos ACODF e ACBHO. Neles, os objetos são conectados com o requisito a seguir: objetos que pertencem ao mesmo grupo devem estar conectados por trilhas de feromônio mais intensas que as trilhas de feromônio entre objetos de grupos distintos. A diferença básica entre a nova proposta das propostas anteriores está na forma em que essas trilhas de feromônio são construídas.

Na nova proposta, o comprimento máximo do percurso que cada formiga poderá realizar em cada ciclo de procedimento é escolhido de acordo com uma certa distribuição de probabilidade. Empregou-se o método de Monte Carlo para simular a distribuição, isto é, selecionar o comprimento do percurso. Nesta implementação, o comprimento máximo do percurso é exponencialmente distribuído, de maneira que percursos muito longos são exponencialmente suprimidos. O nível de feromônios será, posteriormente, utilizado para identificar os grupos. Especificamente, caso a intensidade de feromônios em uma trilha que conecta dois objetos esteja acima de um determinado limiar, consideramos os dois objetos conectados ou pertencentes ao mesmo grupo. Caso contrário, a trilha é apagada e os dois objetos considerados pertencem a grupos distintos.

A nova proposta está esquematizada na figura 4.3, e consiste em quatro procedimentos básicos: Inicialização, Livre Exploração, Simulação e Finalização.

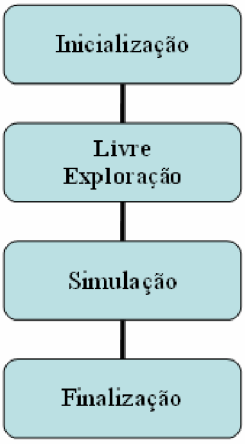


Figura 4.3: Seqüência do algoritmo da nova proposta.

O procedimento de Inicialização tem cinco funções básicas:

i. Leitura das coordenadas dos N objetos.

ii. Cálculo das distâncias geométricas entre todos os objetos, que resulta numa matriz N x N. Não são permitidos objetos com as mesmas coordenadas no conjunto de dados para evitar divisão por zero na rotina de atualização da intensidade da trilha.

iii. Faz o valor inicial de todas as trilhas iguais a zero;

iv. Faz o número de formigas igual a N/3;

v. Cada formiga é colocada aleatoriamente em um dos objetos.

Dois ou mais objetos com as mesmas coordenadas serão considerados como um único objeto e receberão todos eles a mesma classificação.

Durante o procedimento de livre exploração, cada formiga visita aleatoriamente (distribuição uniforme) um terço dos objetos. Os objetos já visitados não poderão ser novamente visitados pela mesma formiga. Esta breve exploração aleatória fornece a escala de comprimento inerente à massa de dados. Esta informação será utilizada na etapa seguinte do algoritmo.

No procedimento de Simulação, as trilhas de feromônios são construídas. Isto é, cada formiga visita um número de objetos que depende do comprimento máximo de seu percurso e atualiza a intensidade de feromônios. O número de simulações é um parâmetro livre. Cada formiga é livre para visitar de zero até N-1 objetos distintos aleatoriamente, em cada simulação. Como dito anteriormente, a cada ciclo, o comprimento máximo do percurso de cada formiga é determinado a partir de uma distribuição exponencial. A formiga só visitará o próximo objeto, se cumprir dois pré-requisitos: i) o número de objetos já visitados, incluindo o objeto inicial, deve ser menor que o total de objetos; ii) o comprimento do caminho já percorrido pela formiga somado à distância que ela pretende percorrer, deve ser menor que o comprimento máximo do percurso selecionado para aquele ciclo. Caso o novo objeto passe pelos dois pré-requisitos, este objeto é visitado e o nível de feromônio da trilha é atualizado somando-se ao valor anterior, o quociente entre uma constante Q e a distância entre o objeto anterior e o novo objeto.

No procedimento de Finalização o valor médio das intensidades de feromônios nas trilhas é calculado. Toma-se como limiar de ativação de uma conexão entre dois objetos, o nível médio das trilhas de feromônios vezes certo fator alfa. Todas as trilhas são analisadas, e aquelas que possuem um nível de feromônios menor que o limiar são consideradas não visíveis ou apagadas. Enquanto que, aquelas trilhas que possuem um nível de feromônios maior ou igual que o limiar são consideradas visíveis. Todos os objetos ligados por trilhas visíveis estão no mesmo grupo. Pequenos grupos, ou seja, grupos menores que 10% do total de objetos, são unidos a grupos maiores e mais próximos. Esse valor de 10% pode ser adaptado para cada caso.

Capítulo 5

Resultados dos Algoritmos de Agrupamento

Handl et al. (2003) afirma que diferentes algoritmos de agrupamento de dados geram diferentes resultados. Uma maneira para avaliar os diferentes resultados dos algoritmos é comparar a eficácia desses resultados através de conjuntos de dados que já possuem classificação.

Neste capítulo serão utilizados vários conjuntos de dados artificiais e reais para comparar os métodos K-Médias e ACBHO com a nova proposta de agrupamento. Neste trabalho a nova proposta será identificada como Monte Carlo *Ant Colony* ou pela abreviatura MCAC.

Vale salientar alguns aspectos da aplicação dos algoritmos: número de simulações do MCAC sempre será de 500 para todos os conjuntos de dados; Os resultados gerados pelo K-Médias que apresentarem grupos vazios serão descartados; A distância utilizada será a distância euclidiana.

Em relação às tabelas de comparação de resultados, serão apresentados os valores médios de 20 experiências para cada algoritmo de agrupamento. O desvio padrão estará ao lado da média entre parênteses. Os melhores valores serão destacados em negrito. Os índices de comparação utilizados serão a medida F, o índice de Rand e a taxa de acerto, onde esta última é simplesmente o número de objetos corretamente classificados sobre o número total de objetos.

No ACBHO e no MCAC toma-se como limiar de ativação de conexão entre dois objetos o nível médio das trilhas de feromônios vezes um certo fator percentual, ou seja, o limiar de ativação será um valor percentual do valor da média de todas as trilhas de feromônio. Nas tabelas onde houver a comparação desse limiar entre o ACBHO e MCAC, o valor mínimo e o valor máximo significam respectivamente o menor e o maior percentual da média para formar a determinada quantidade de grupos existentes no conjunto de dados entre as 20 experiências realizadas. O valor da média e o valor desvio padrão são respectivamente a média e o desvio padrão populacional do limiar de ativação das 20 experiências realizadas.

5.1 Ruspini.

O primeiro conjunto de dados é uma base conhecida como Ruspini (1970), como mostra a figura 5.1, que foi gerada para testes de agrupamentos e é composta de 75 objetos bidimensionais, ou seja, cada objeto é composto de duas variáveis.

O conjunto de dados Ruspini é formado por quatro grupos com 20, 23, 17 e 15 elementos em cada grupo.

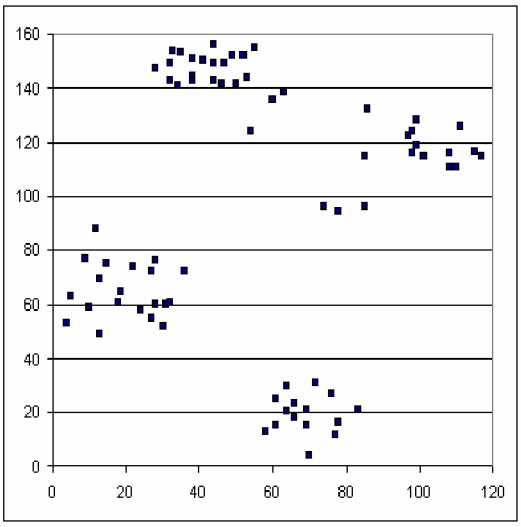


Figura 5.1: Representação da base de dados Ruspini.

A tabela 5.1 apresenta avaliação dos resultados dos três algoritmos. Todos algoritmos apresentaram bons resultados sendo que o MCAC se destacou apresentado os melhores resultados com 100% de acerto nas três formas de avaliação.

Tabela 5.1: Tabela de comparação dos resultados do conjunto de dados Ruspini.

	Taxa de acerto	Medida F	Índice de Rand
K-Médias	0,94400 (0,11318)	0,95875 (0,09105)	0,97180 (0,05685)
ACBHO	0,99933 (0,00291)	0,99933 (0,00292)	0,99930 (0,00306)
MCAC	**1,00000** (0)	**1,00000** (0)	**1,00000** (0)

Os resultados do ACBHO e do MCAC são muito próximos. Porém quando se verificam os limiares de ativação das trilhas de feromônio em relação á media de

todas as trilhas têm-se os seguintes resultados apresentados na tabela 5.2. Pela observação da tabela 5.2 nota-se que o MCAC é mais estável na classificação dos objetos em relação ao ACBHO.

Tabela 5.2: Tabela de valores dos limiares de ativação das trilhas de feromônio.

	Mínimo	Máximo	Média	Desvio Padrão
ACBHO	305 %	535 %	396,75000 %	(70,62710)
MCAC	125 %	175 %	144,50000 %	**(12,93252)**

A figura 5.2 mostra as conexões entre os objetos, no algoritmo MCAC, em quatro situações diferentes.

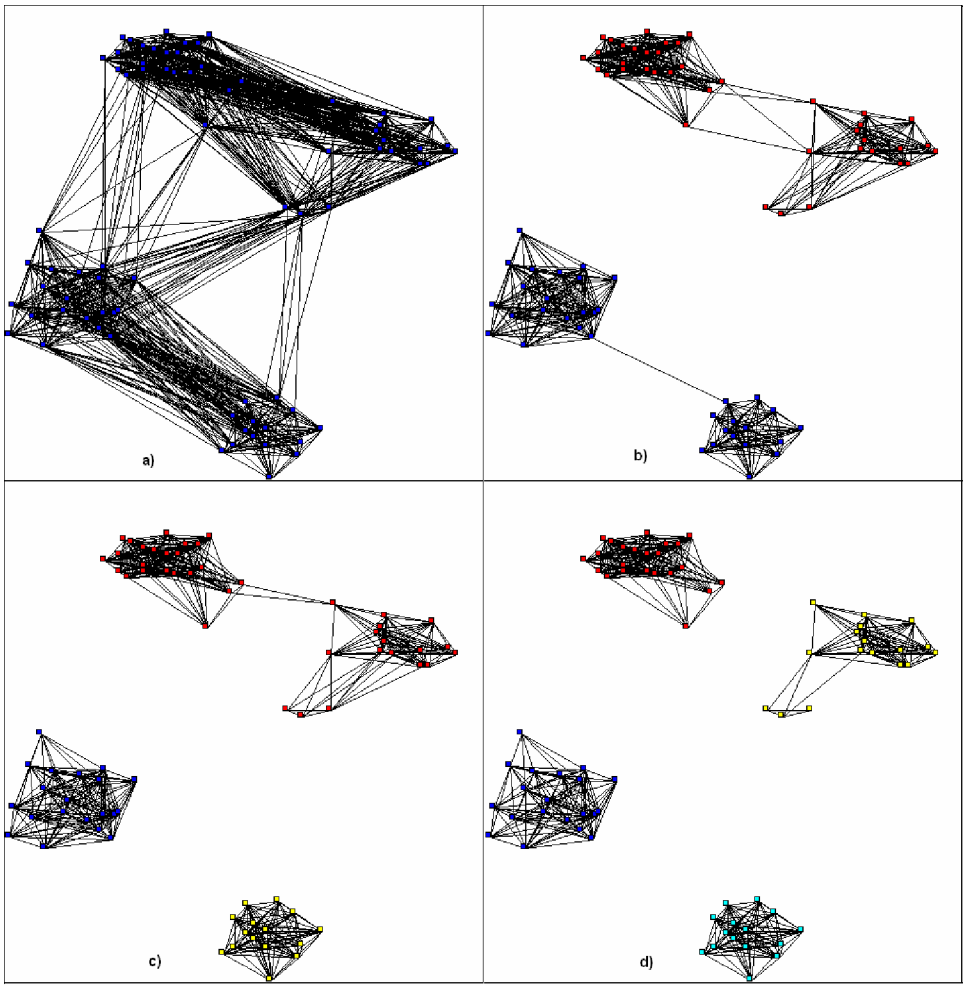


Figura 5.2: Conexões entre os objetos geradas pelo algoritmo MCAC. a) Formando um grupo. b) Formando dois grupos. c) Formando três grupos. d) Formando quatro grupos.

A figura 5.2 a) mostra as conexões entre os objetos com o limiar de ativação igual a 50% da média de todas as trilhas onde encontra-se apenas um grupo. A

figura 5.2 b) mostra as conexões com o limiar de ativação igual a média onde encontra-se formado dois grupos. A figura 5.2 c) mostra as conexões com o limiar de ativação igual 120% da média onde encontra-se formados três grupos. A figura 5.2 d) mostra as conexões com o limiar de ativação igual 135% da média onde encontra-se formados quatro grupos.

5.2 Espirais.

O segundo conjunto de dados é composto por 60 objetos bidimensionais que formam duas espirais com 30 objetos cada uma como mostra a figura 5.3.

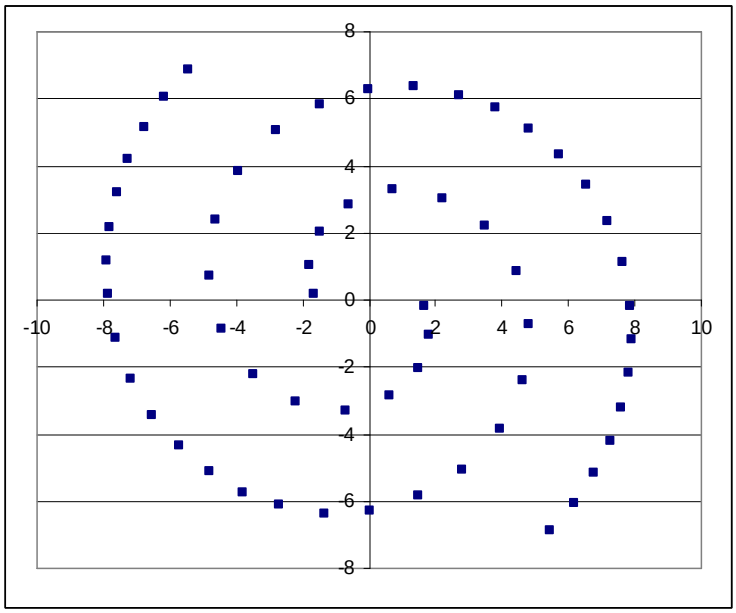


Figura 5.3: Representação do conjunto de dados em forma de espiral.

A tabela 5.3 apresenta avaliação dos resultados dos três algoritmos. O K-Médias e o ACBHO apresentaram resultados muito semelhantes, porém com valores que apresentam relativamente desempenhos pouco significativos. O MCAC se destacou com 100% de acerto nas 20 experiências.

Tabela 5.3: Tabela de comparação dos resultados do conjunto de dados em forma de espirais.

	Taxa de acerto	Medida F	Índice de Rand
K-Médias	0,65167 (0,03723)	0,65167 (0,03723)	0,54113 (0,01666)
ACBHO	0,65667 (0,04295)	0,65213 (0,04564)	0,54520 (0,02885)
MCAC	**1,00000** (0)	**1,00000** (0)	**1,00000** (0)

A figura 5.4 mostra as conexões entre os objetos das espirais, no algoritmo MCAC, em duas situações diferentes. A figura 5.4 a) mostra as conexões entre os objetos com o limiar de ativação igual a média de todas as trilhas onde encontra-se apenas um grupo. A figura 5.4 b) mostra as conexões com o limiar de ativação igual a 230% da média onde encontra-se formado dois grupos, um na cor azul e outro na cor vermelha, mostrando também que o MCAC realiza um correto encadeamento entre os objetos pertencentes a cada espiral.

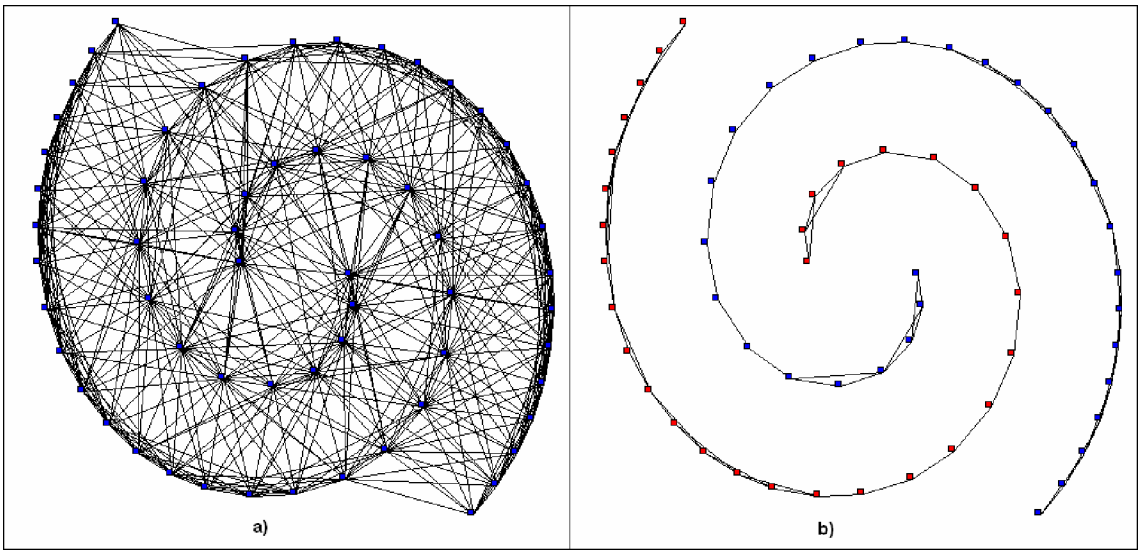


Figura 5.4: Conexões entre os objetos gerados pelo algoritmo MCAC nos dados em espiral. a) Formando um grupo. b) Formando dois grupos.

A figura 5.5 exibe uma má classificação realizada pelo algoritmo K-Médias nos dados em forma de espiral. O K-Médias forma dois grupos, um na cor azul e outro na cor vermelha, onde ambos os grupos possuem metade do número de elementos. Isto se deve a dificuldade do K-Médias em identificar grupos que não sejam de forma esférica.

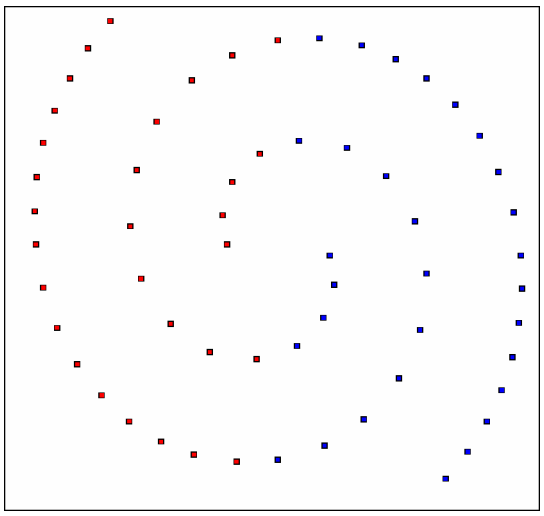


Figura 5.5: Classificação realizada pelo K-Médias aos dados em forma de espiral.

5.3 2D-4C.

O conjunto de dados 2D-4C é formado por 200 objetos bidimensionais gerados aleatoriamente com distribuição gaussiana e composto por quatro grupos de 50 objetos cada como mostra a figura 5.6. O número do lado esquerdo da letra D significa o número de dimensões e o número do lado esquerdo da letra C significa o número de grupos existentes no conjunto de dados.

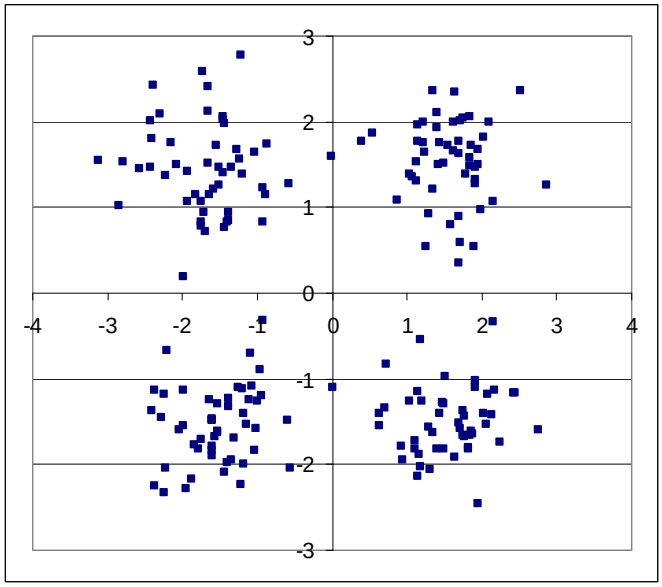


Figura 5.6: Representação do conjunto de dados 2D-4C.

A construção das variáveis x e y de cada grupo é dada por G(μ; σ) onde μ é a média e σ é desvio padrão da distribuição gaussiana, como mostra a tabela 5.4.

Tabela 5.4: Valores de média e desvio padrão para a construção das variáveis x e y de quatro grupos distintos.

	x	y
1º grupo	G(1,5; 0,5)	G(1,5; 0,5)
2º grupo	G(1,5; 0,5)	G(-1,5; 0,5)
3º grupo	G(-1,5; 0,5)	G(1,5; 0,5)
4º grupo	G(-1,5; 0,5)	G(-1,5; 0,5)

A tabela 5.5 apresenta avaliação dos resultados dos três algoritmos com relação ao conjunto de dados 2D-4C. Os resultados dos três algoritmos foram muito próximos, porém o MCAC apresentou o melhor resultado.

Tabela 5.5: Tabela de comparação dos resultados do conjunto de dados 2D-4C.

	Taxa de acerto	Medida F	Índice de Rand
K-Médias	0,99200 (0,00245)	0,99200 (0,00245)	0,99204 (0,00244)
ACBHO	0,99200 (0,00367)	0,99200 (0,00368)	0,99206 (0,00363)
MCAC	**0,99250 (0,00296)**	**0,99250 (0,00296)**	**0,99255 (0,00293)**

Como os resultados do ACBHO e do MCAC são muito próximos, foram verificados os limiares de ativação das trilhas de feromônio em relação á media de todas as trilhas obtendo-se os resultados apresentados na tabela 5.6, onde se observa que o MCAC é mais estável na classificação dos objetos em relação ao ACBHO.

Tabela 5.6: Tabela de valores dos limiares de ativação das trilhas de feromônio no conjunto de dados 2D-4C.

	Mínimo	Máximo	Média	Desvio Padrão
ACBHO	510 %	710 %	605,50000 %	(62,12689)
MCAC	210 %	275 %	243,00000 %	**(18,26198)**

5.4 Hélices Cilíndricas.

O quarto conjunto de dados é composto por 120 objetos tridimensionais que formam duas hélices cilíndricas com 60 objetos cada uma como mostra a figura 5.7.

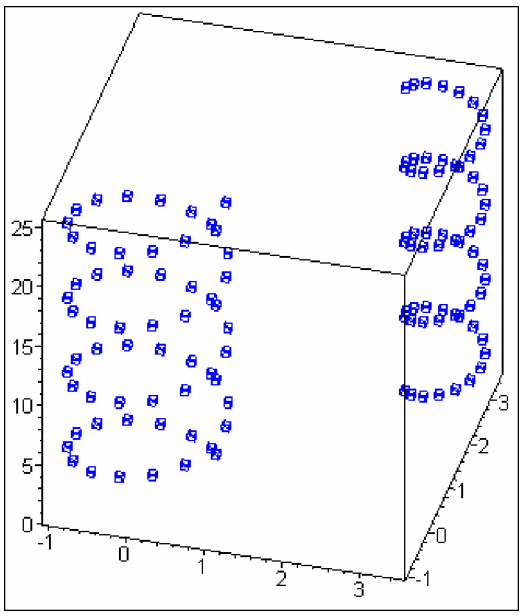


Figura 5.7: Conjunto de dados em forma de hélices cilíndricas.

A tabela 5.7 apresenta avaliação dos resultados dos três algoritmos. O K-Médias só apresentou 50% de acerto em todas as experiências. O ACBHO apresentou resultados melhores que o K-Médias. O MCAC se destacou novamente com 100% de acerto em todas as experiências.

Tabela 5.7: Tabela de comparação dos resultados do conjunto de dados em forma hélices cilíndricas.

	Taxa de acerto	Medida F	Índice de Rand
K-Médias	0,50000 (0)	0,50000 (0)	0,49153 (0)
ACBHO	0,66625 (0,10851)	0,64624 (0,11634)	0,60870 (0,13482)
MCAC	**1,00000** (0)	**1,00000** (0)	**1,00000** (0)

5.5 3D-2C.

O conjunto de dados 3D-2C é formado por 450 objetos tridimensionais gerados aleatoriamente com distribuição gaussiana e é composto por dois grupos de 400 objetos e 50 objetos como mostra a figura 5.8.

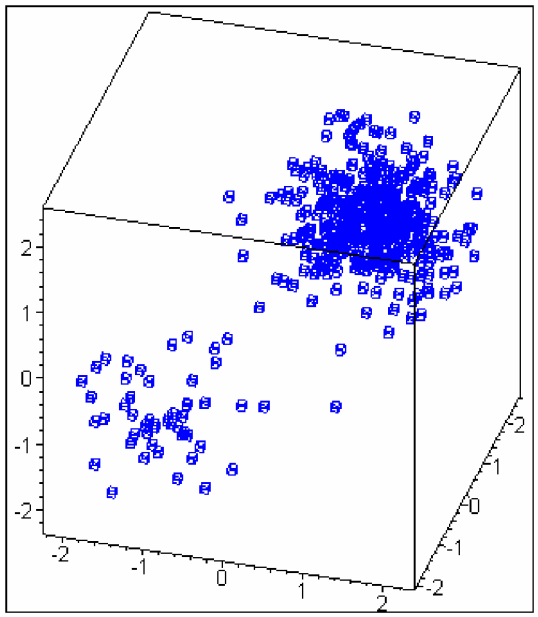


Figura 5.8: Representação do conjunto de dados 3D-2C.

A construção das variáveis x, y e z de cada grupo é dada por G(μ; σ) onde μ é a média e σ é desvio padrão da distribuição gaussiana, como mostra a tabela 5.8.

Tabela 5.8: Valores de média e desvio padrão para a construção das variáveis x, y e z de dois grupos distintos.

	x	y	z
1º grupo	G(1; 0,5)	G(1; 0,5)	G(1; 0,5)
2º grupo	G(-1; 0,5)	G(-1; 0,5)	G(-1; 0,5)

A tabela 5.9 apresenta avaliação dos resultados dos três algoritmos com relação ao conjunto de dados 2D-4C. Os resultados dos três algoritmos foram muito próximos.O MCAC apresentou o melhor resultado em média, mas o K-Médias se mostrou menos variante.

Tabela 5.9: Tabela de comparação dos resultados do conjunto de dados 2D-4C.

	Taxa de acerto	Medida F	Índice de Rand
K-Médias	0,99778 (0)	0,99779 (0)	0,99556 (0)
ACBHO	0,99800 (0,00185)	0,99799 (0,00187)	0,99601 (0,00368)
MCAC	**0,99811** (0,00202)	**0,99810** (0,00204)	**0,99623** (0,00403)

Como os resultados do ACBHO e do MCAC são muito próximos, foram verificados os limiares de ativação das trilha de feromônio em relação á media de todas as trilhas obtendo-se os resultados apresentados na tabela 5.10, onde se observa que o MCAC é mais estável na classificação dos objetos em relação ao ACBHO.

Tabela 5.10: Tabela de valores dos limiares de ativação das trilhas de feromônio do conjunto de dados 3D-2C.

	Mínimo	Máximo	Média	Desvio Padrão
ACBHO	165 %	440 %	357,75000 %	(58,85310)
MCAC	140 %	205 %	165,25000 %	**(14,44602)**

5.6 Dados Craniofaciais de Gorilas.

Esse é um conjunto de dados real e é formado por dados craniais e faciais de 59 gorilas (O'Higgins e Dryden 1993), sendo 29 machos e 30 fêmeas. A dimensão desse conjunto de dados é 13, ou seja, foram feitas 13 medidas craniofaciais de cada gorila.

A tabela 5.11 apresenta avaliação dos resultados dos três algoritmos. Tando o K-Médias como o MCAC apresentaram excelentes resultados com 100% de acerto em todas experiências. O ACBHO apresentou um bom resultado, porém inferior ao K-Médias e ao MCAC.

Tabela 5.11: Tabela de comparação dos resultados do conjunto de dados craniofaciais dos gorilas.

	Taxa de acerto	Medida F	Índice de Rand
K-Médias	**1,00000** (0,0)	**1,00000** (0,0)	**1,00000** (0,0)
ACBHO	0,89576 (0,11373)	0,89407 (0,11597)	0,83635 (0,14262)
MCAC	**1,00000** (0,0)	**1,00000** (0,0)	**1,00000** (0,0)

A figura 5.9 mostra a representação de duas das treze variáveis do conjunto de dados crânios-faciais de gorilas sem qualquer tipo de classificação.

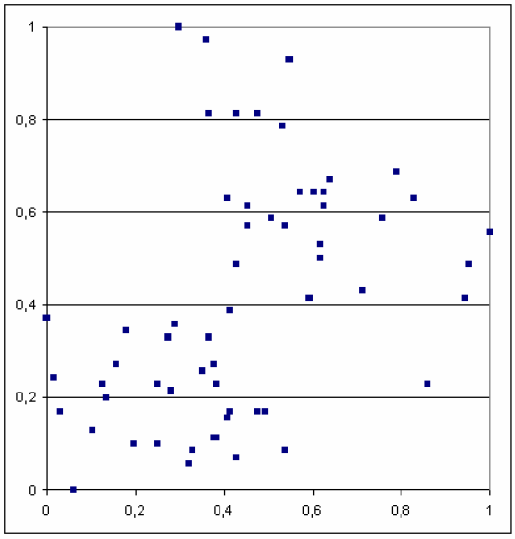


Figura 5.9: Representação de duas das treze variáveis dos dados craniofaciais de gorilas.

A figura 5.10 mostra uma classificação realizada pelo MCAC no conjunto de dados crânios-faciais de gorilas com o limiar de ativação das trilhas de 155% da média de todas as trilhas. Na figura 5.10, os pontos azuis representam os machos e os pontos vermelhos representam as fêmeas.

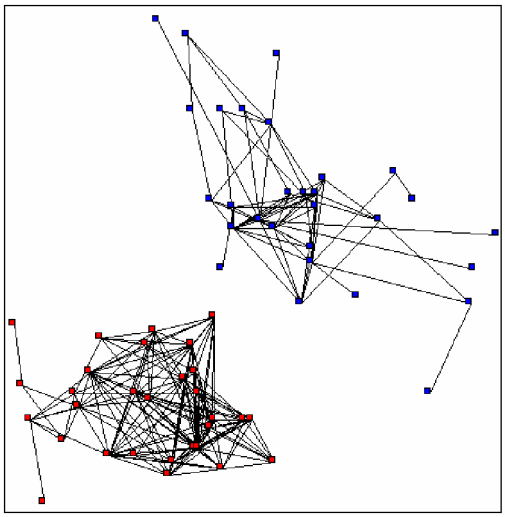


Figura 5.10: Classificação realizada pelo MCAC no conjunto de dados crânios-faciais de gorilas.

5.7 Dados da Planta Íris.

O conjunto de dados da planta Íris (Fisher 1936) é um conjunto de dados reais, fornecido pela Universidade da Califórnia em Irvine. O conjunto de dados Íris consiste de 150 objetos caracterizados por quatro medidas numéricas que descrevem respectivamente o comprimento da sépala, a largura da sépala, o comprimento da pétala e a largura da pétala. Esse conjunto é composto de três grupos cada um com 50 objetos. Cada grupo representa um tipo diferente de planta Íris que são chamadas de Íris Setosa, Íris Versicolor e Íris Virginica. O grupo da Íris Setosa é linearmente separado dos grupos das outras duas. Porém os grupos das Íris Versicolor e Íris Virginica não são separados linearmente.

Uma representação de três das quatro variáveis que possui o bando de dados Íris pode ser vista na figura 5.11.

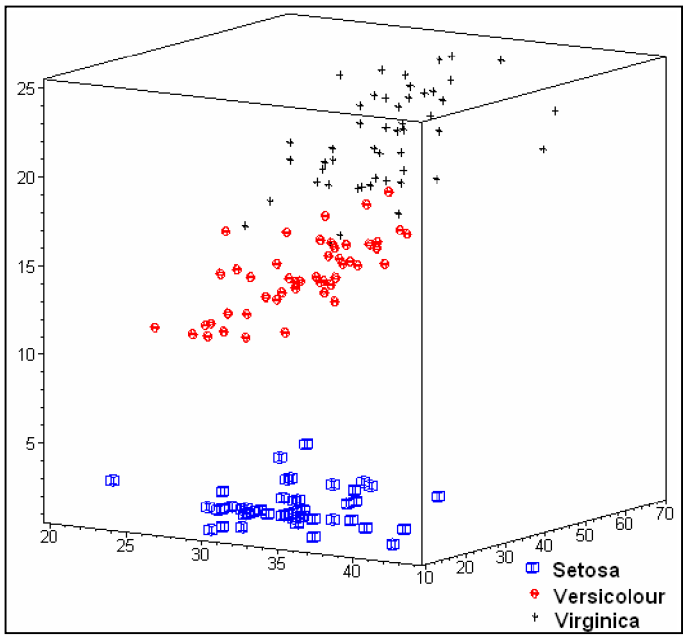


Figura 5.11: Representação de três das quatro variáveis do conjunto de dados Íris.

A tabela 5.12 apresenta avaliação dos resultados dos três algoritmos. Os três algoritmos apresentaram bons resultados, porém o MCAC apresentou os melhores resultados em relação aos outros dois algoritmos.

Tabela 5.12: Tabela de comparação dos resultados do conjunto de dados Íris.

	Taxa de acerto	Medida F	Índice de Rand
K-Médias	0,82400 (0,12533)	0,81332 (0,14391)	0,84294 (0,06160)
ACBHO	0,89467 (0,03103)	0,89214 (0,03355)	0,88289 (0,02733)
MCAC	**0,94833** (0,02764)	**0,94795** (0,02858)	**0,93837** (0,03025)

O conjunto de dados Íris também foi utilizado para verificação e comparação de resultados de muitos outros algoritmos de agrupamentos de dados. Um desses algoritmos que utilizou o conjunto Íris foi proposto por Domany (1999) cujo algoritmo é baseado em propriedades da física em especial o supermagnetismo. O modelo de Domany teve uma taxa de acerto de aproximadamente 0,83333. O MCAC teve uma taxa média de acerto no conjunto Íris de 0,94833 com o melhor resultado sendo 0,97333.

Handl, Knowles e Dorigo (2003) realizaram um estudo comparativo entre o modelo de Deneubourg com os seguintes algoritmos de agrupamentos: K-Médias, Mapas Auto-Organizáveis de uma dimensão (1D-SOM) e o processo hierárquico chamado de método das médias das distâncias (MMD). Handl, Knowles e Dorigo realizaram 50 experiências para cada um dos quatro algoritmos no conjunto de dados Íris e os resultados estão na tabela 5.13 juntamente com o resultado do MCAC também com 50 experiências.

Tabela 5.13: Resultados do estudo comparativo de Handl, Knowles e Dorigo juntamente com os resultados do MCAC.

	Medida F	Índice de Rand
Modelo de Deneubourg	0,81681 (0,01484)	0,82542 (0,00804)
K-Médias	0,82452 (0,08486)	0,81659 (0,10128)
1D-SOM	0,86141 (0,00773)	0,85734 (0,00554)
MMD	0,80985 (0,0)	0,82231 (0,0)
MCAC	**0,94248** (0,03253)	**0,93249** (0,03305)

Pela observação da tabela 5.13 nota-se que o MCAC teve o melhor resultado com relação aos quatro modelos que foram estudados por Handl, Knowles e Dorigo.

5.8 Dados de Câncer de Mama.

O conjunto de dados de câncer de mama (Wolberg et. al) é um conjunto de dados reais, fornecido também pela Universidade da Califórnia em Irvine. Esse conjunto de dados é originalmente composto de 699 objetos, sendo que 16 desses objetos estão incompletos apresentando algum valor faltando em uma de suas variáveis. Para fins deste trabalho os 16 objetos incompletos foram descartados, sendo considerados apenas os objetos 683 objetos completos restantes. Desses 683 objetos, 444 são classificados como tumor benigno e 239 são classificados como tumor maligno. Cada objeto é composto de nove variáveis que apresentam valores variando de um até dez. As nove variáveis são: espessura do grupo, homogeneidade do tamanho das células, homogeneidade do formato das células, aderência marginal, tamanho das células epiteliais, núcleo reduzido, cromatina branda, nucléolo normal e mitose.

A tabela 5.13 apresenta avaliação dos resultados dos três algoritmos com relação ao conjunto de dados do câncer de mama. Os três algoritmos apresentaram resultados semelhantes com uma pequena vantagem do MCAC em relação aos outros dois algoritmos.

Tabela 5.14: Tabela de comparação dos resultados do conjunto de dados câncer de mama.

	Taxa de acerto	Medida F	Índice de Rand
K-Médias	0,96188 (0,0)	0,96168 (0,0)	0,92655 (0,0)
ACBHO	0,95425 (0,00619)	0,95374 (0,00642)	0,91263 (0,01112)
MCAC	**0,96947** (0,00618)	**0,96924** (0,00624)	**0,94080** (0,01168)

5.9 Estudo do Número de Grupos.

O limiar de ativação da trilha é um parâmetro ajustável do algoritmo de agrupamento proposto neste trabalho e está associado à resolução com que os objetos são observados. Um limiar de ativação muito baixo provoca a coalescência dos grupos. Por outro lado, um limiar alto demais pode distinguir objetos similares.

Como em geral, não se conhece o número de grupos a priori, seria interessante introduzir-se um critério para a fixação do limiar de ativação das trilhas de feromônio.

Um gráfico do número de grupos em função do fator de ativação da trilha fornece uma indicação da estrutura inerente aos dados.

Na figura 5.12 mostra-se o número de grupos identificados k em função do limiar de ativação α das trilhas para o conjunto de dados Ruspini.

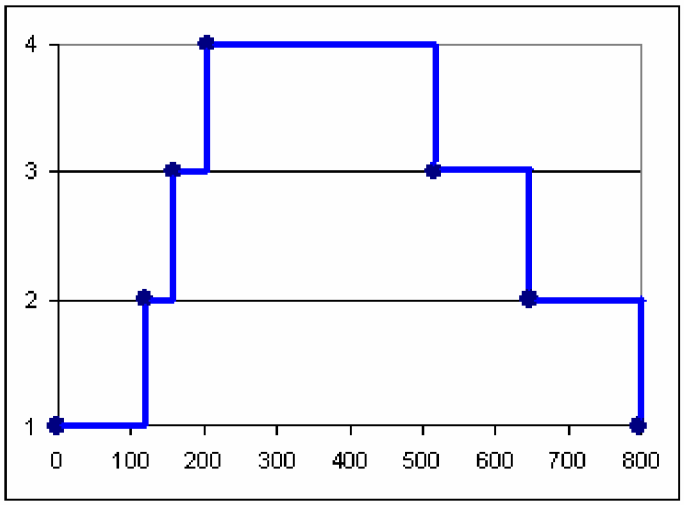


Figura 5.12: Gráfico do número de grupos identificados em função do limiar de ativação das trilhas para o conjunto de dados Ruspini.

Observa-se na figura 5.12 apenas um único grupo para α<120%. A medida que α cresce, o número de grupos aumenta gradativamente até atingir k=4 para α em torno de 200%. Acima deste valor, o número de grupos atinge um platô para uma larga faixa de valores de α. A diminuição do número de grupos na figura 5.12 para α>515% ocorre porque o algoritmo proíbe a manutenção de grupos muito pequenos.

A análise do resultado apresentado na figura 5.12 sugere que o número de grupos mais provável é aquela que menos apresenta sensibilidade ao valor de α, ou seja, o platô mais largo.

A figura 5.13 mostra o gráfico do número de grupos identificados k em função do limiar de ativação α das trilhas para o conjunto de dados em forma de espiral. Pela análise da figura 5.13 observa-se que o número de grupos k varia até quatro. O platô mais largo sugere que o número de grupos existentes no conjunto de dados em forma de espiral é dois.

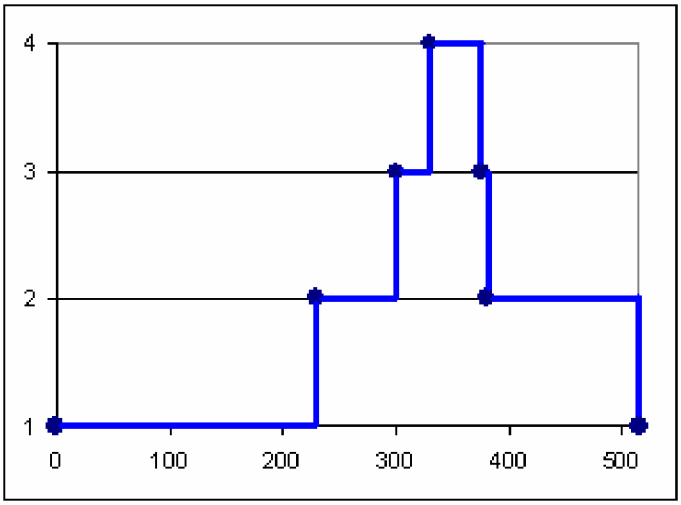


Figura 5.13: Gráfico do número de grupos identificados em função do limiar de ativação das trilhas para o conjunto de dados em forma de espiral.

A figura 5.14 apresenta o gráfico do número de grupos identificados k em função do limiar de ativação α das trilhas para o conjunto de dados 2D-4C. Pela análise da figura 5.14 observa-se que o número de grupos k varia até quatro. O platô mais largo sugere que o número de grupos existentes no conjunto de dados em forma de espiral é quatro.

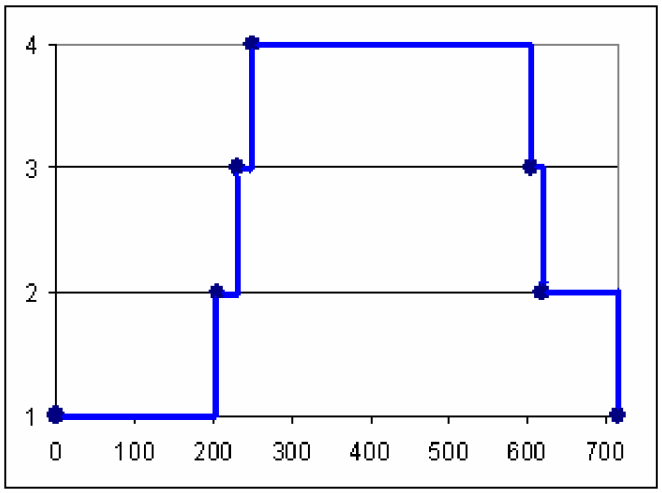


Figura 5.14: Gráfico do número de grupos identificados em função do limiar de ativação das trilhas para o conjunto de dados 2D-4C.

A figura 5.15 apresenta o gráfico do número de grupos identificados k em função do limiar de ativação α das trilhas para o conjunto de dados em forma de

hélices cilíndricas. Pela análise da figura 5.15 observa-se que o número de grupos k varia até três. O platô mais largo sugere que o número de grupos existentes no conjunto de dados em forma de espiral é dois.

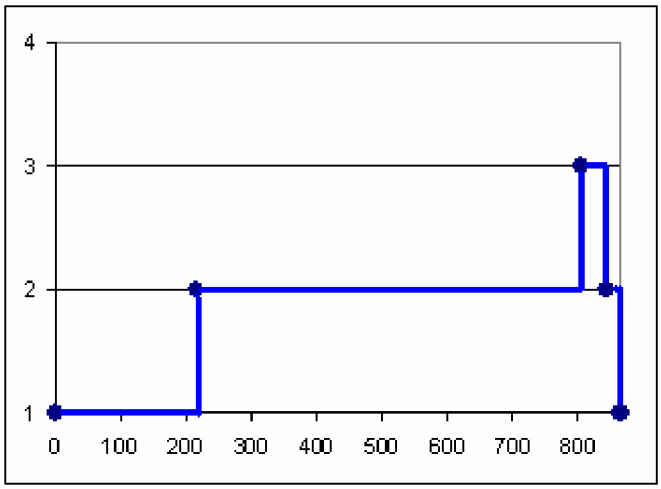


Figura 5.15: Gráfico do número de grupos identificados em função do limiar de ativação das trilhas para o conjunto de dados em forma de hélices cilíndricas.

A figura 5.16 apresenta o gráfico do número de grupos identificados k em função do limiar de ativação α das trilhas para o conjunto de dados 3D-2C. Pela análise da figura 5.16 observa-se que o número de grupos k é dois, por ser o único platô com número de grupos diferente de um.

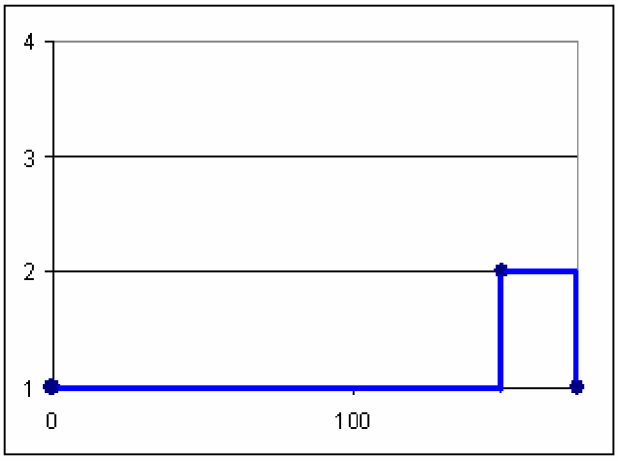


Figura 5.16: Gráfico do número de grupos identificados em função do limiar de ativação das trilhas para o conjunto de dados 3D-2C.

A figura 5.17 apresenta o gráfico do número de grupos identificados k em função do limiar de ativação α das trilhas para o conjunto craniofaciais de gorilas. .

Pela análise da figura 5.17 observa-se que o número de grupos k é dois, por ser o único platô com número de grupos diferente de um.

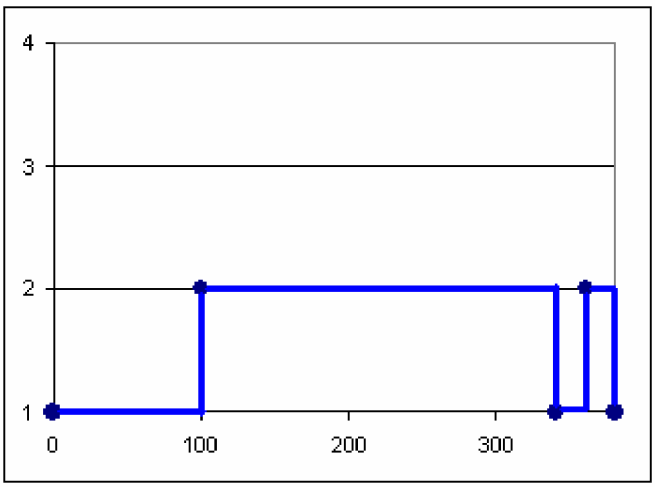


Figura 5.17: Gráfico do número de grupos identificados em função do limiar de ativação das trilhas para o conjunto de dados craniofaciais de gorilas.

A figura 5.18 apresenta o gráfico do número de grupos identificados k em função do limiar de ativação α das trilhas para o conjunto de dados da planta Iris. Pela análise da figura 5.18 observa-se que o número de grupos k varia até três. O platô mais largo sugere que o número de grupos existentes no conjunto de dados em forma de espiral é dois. Porém o número de grupos existentes nesse conjunto de dados é três. Isto se deve ao fato de dois desses grupos não serem linearmente separados.

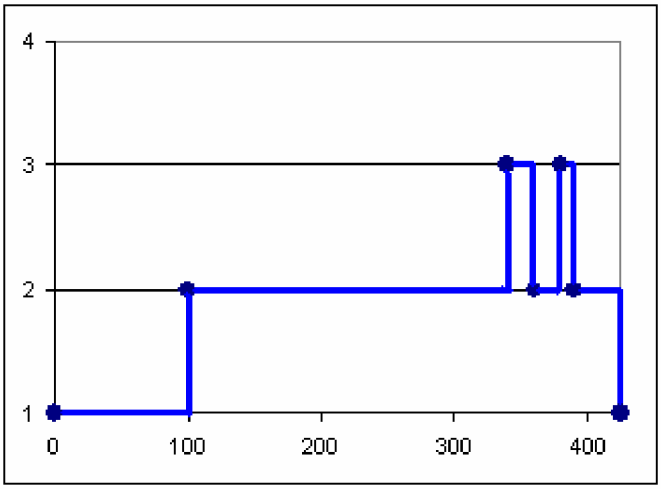


Figura 5.18: Gráfico do número de grupos identificados em função do limiar de ativação das trilhas para o conjunto de dados da planta Iris.

A figura 5.19 apresenta o gráfico do número de grupos identificados k em função do limiar de ativação α das trilhas para o conjunto dados do câncer de mama. Pela análise da figura 5.19 observa-se que o número de grupos k é dois, por ser o único platô com número de grupos diferente de um.

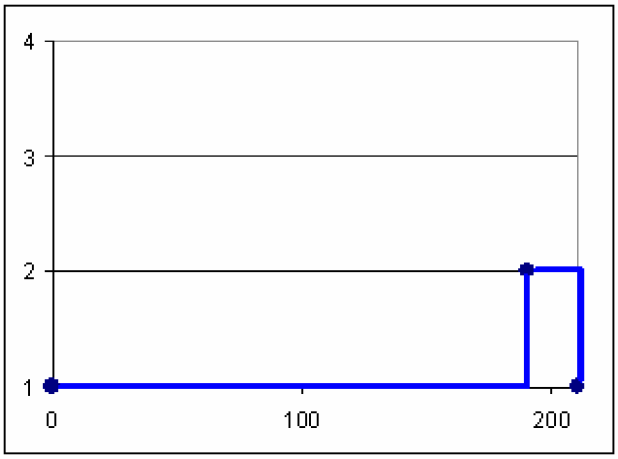


Figura 5.19: Gráfico do número de grupos identificados em função do limiar de ativação das trilhas para o conjunto de dados do câncer de mama.

A analise dos gráficos do número de grupos identificados em função do limiar de ativação das trilhas fornece também a informação do número máximo possível de grupos existentes no conjunto de dados em estudo.

Capítulo 6

Conclusões.

Neste trabalho foi proposto um novo algoritmo de agrupamento de dados baseado no comportamento social de formigas e foi realizado um estudo comparativo dessa nova proposta com outros dois algoritmos. O primeiro algoritmo que serviu de comparação foi o K-Médias que é muito conhecido e utilizado por pesquisadores de várias áreas de conhecimento. O segundo é o algoritmo ACBHO que também se baseia no comportamento social de formigas, combinado com um híbrido de várias técnicas de otimização, como recozimento simulado, busca proibitiva e seleção por torneio. A nova proposta, que foi batizada pela sigla MCAC, tem a forma de verificação dos agrupamentos semelhante ao do ACBHO, porém a forma de construção das trilhas que definem os agrupamentos é diferente. No MCAC é introduzido um comprimento máximo aleatório para o percurso de cada formiga. Os comprimentos são exponencialmente distribuídos, de forma a privilegiar percursos curtos.

Os resultados apresentados no quinto capítulo mostraram que o MCAC tem um desempenho, na identificação dos grupos, igual ou superior aos algoritmos avaliados. Os resultados mostraram também algumas características do MCAC como: eficiência em agrupar padrões reais, como o conjunto de dados Íris e o conjunto de dados do câncer de mama; eficiência em agrupar padrões artificiais determinísticos, como os dados em forma de espirais e os dados em forma de hélices cilíndricas; eficiência em agrupar padrões artificiais estocásticos, como os conjuntos Ruspini, 2D-4C e 3D-2C; possibilidade de trabalhar com dados com altas dimensões, como o conjunto de dados craniofaciais de gorilas; vantagem de não ser necessário conhecer o número de grupos antes da execução do algoritmo; possibilidade de trabalhar com grupos onde o número de elementos de um grupo é várias vezes superior ao do outro grupos, como no conjunto de dados 3D-2C. Além disso, o MCAC captura correlações espaciais de curto alcance entre os objetos do conjunto, como verificado nos exemplos dos dados em forma de espirais e hélices cilíndricas. Uma dificuldade do MCAC está na determinação do valor do limiar de

ativação das trilhas de feromônio para que a quantidade de grupos seja a ideal. Este ponto foi abordado no final do quinto capítulo, onde foi mostrado como o número de grupos varia com o limiar de ativação da trilha. Em todos os casos estudados, foi observado um número máximo de grupos. Além disso, um forte indício do número correto de grupos na massa de dados pôde ser inferido.

Referências Bibliográficas

ABONYI, J.; FEIL, B. **Cluster Analysis for Data Mining and System Identification.** New York: Springer, 2007.

ABRAHAM, A.; DAS, S.; ROY, S. Swarm Intelligence Algorithms for Data Clustering. In: MAIMON, O.; ROKACH, L. **Soft Computing for Knowledge Discovery and Data Mining**. New York: Springer, p. 279-313, 2007.

AFIFI, A. A.; CLARK, V. **Computer-Aided Multivariate Analysis**. New York: Chapman & Hall, 1997.

ALBUQUERUQE, M. A. **Estabilidade em Análise de Agrupamento (Cluster Analysis).** 2005. 62f. Dissertação (Mestrado em Biometria e Estatística Aplicada) Universidade Federal Rural de Pernambuco, Recife, 2005.

AZZAG, H.; VENTURINI, G. A Clustering Model Using Artificial Ants**.** In: **Proceeding of 16th European Conference on Artificial Intelligence**. Valencia: IOS Press, p. 480-484, 2004.

BEICHL, I.; SULLIVAN, F. Monte Carlo Methods**. IEEE CS 8**, p. 7-8, 2006.

BILMES, J.; VAHDATY, A.; HSU, W. Empirical Observations of Probabilistic Heuristics for the Clustering Problem. **International Computer Science Institute - Technical Report TR-97-018**, 1997.

BIRATTARI, M.; PELLEGRINI, P.; DORIGO, M. On the Invariance of Ant Colony Optimization. **IRIDIA – Technical Report Series 4**, p. 1-21, 2005.

BONABEAU, E.; MEYER, C. Swarm Intelligence: A Whole New Way to Think About Business. **Harvard Business Review**, p. 106-114, 2001.

BONABEAU, E.; THÉRAULAZ, G. Swarm Smarts. **Scientific American,** p. 73-79, 2000.

BULLNHEIMER, B.; HARTL, R. F.; STRAUSS, C. An improved Ant System algorithm for the Vehicle Routing Problem. **Annals of Operations Research 89**, p. 319–328, 1999.

BUSSAB, W. O.; MIAZAKI, E. S.; ANDRADE, D. F. **Introdução à Análise de Agrupamentos**. São Paulo: Associação Brasileira de Estatística, 1990.

COLORNI, A.; DORIGO, M.; MANIEZZO, V. Distributed Optimization by Ant Colonies. In: **Proceedings of the First European Conference on Artificial Life**, Amsterdam, Elsevier Publishing, p. 134-142, 1991.

COSTA, D.; HERTZ, A. Ants Can Colour Graphs. **The Journal of the Operational Research Society 48,** p. 295-305, 1997.

DENEUBOURG, L. et al. The Self-Organizing Exploratory Pattern of the Argentine Ant, **Journal of Insect Behavior 3**, p. 159-168, 1989.

DI CARO, G.; DORIGO, M. AntNet: Distributed Stigmergetic Control for Communications Networks. **Journal of Artificial Intelligence Research 9**, p. 317-365, 1998.

DOMANY, E. Superparamagnetic Clustering of Data – The Definitive Solution of an Ill-Posed Problem. **Physica A 263**, p.158-169, 1999.

DORIGO, M.; BONABEAU, E.; THÉRAULAZ, G. Ant algorithms and stigmergy. **Future Generation Computer Systems 16**, p. 851–871, 2000.

DORIGO, M.; DI CARO, G.; GAMBARDELLA, L. M. Ant Algorithms for Discrete Optimization. **Artificial Life 5**, p. 137–172, 1999.

DORIGO, M.; SOCHA, K. An Introduction to Ant Colony Optimization. **IRIDIA - Technical Report Series 10**, p. 1-24, 2005.

DORIGO, M.; GAMBARDELLA, L. M. Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem. **IEEE Transactions on Evolutionary Computation 1**, p. 53-66, 1997.

DORIGO, M.; BIRATTARI, M.; STÜTZLE, T. Ant Colony Optimization Artificial Ants as a Computational Intelligence Technique. **IEEE Computational Intelligence Magazine**, p. 28-39, 2006.

DORIGO, M.; STÜTZLE, T. Ant Colony Optimization. Londres: **The MIT Press**, 2004.

DORIGO, M.; MANIEZZO, V.; COLORNI, A. The Ant System: Optimization by a Colony of Cooperating Agents. **IEEE Transactions on Systems, Man, and Cybernetics-Part B 26**, p. 1-13, 1996.

EVERITT, B. **Cluster Analysis. Londres:** Heinemann Educacional Books, 1977.

FISHER, R. A. **Iris Data Set**. 1936, Disponível em: <http://archive.ics.uci.edu/ml/datasets/Iris>. Acesso em: 22 jun. 2007.

GAMBARDELLA, L. M.; TAILLARD, É. D.; DORIGO, M. Ant Colonies for the Quadratic Assignment Problem. **The Journal of the Operational Research Society 50**, p. 167-176, 1999.

GAMBARDELLA, L. M.; DORIGO, M. Solving Symmetric and Asymmetric TSPs by Ant Colonies. In: **IEEE Conference on Evolutionary Computation**, Nayoga, IEEE, p. 1-6, 1996.

GOSS, S. et al. Self-Organized Shortcuts in the Argentine Ant, **Naturwissenschaften 76**, p. 579–581, 1989.

GÜNGÖR, Z.; ÜNLER, A. K-harmonic Means Data Clustering With Simulated Annealing Heuristic. **Applied Mathematics and Computation 184**, p. 199–209, 2007.

GUTIN, G.; PUNNEN, A. **The Traveling Salesman Problem and its Variations**. Boston: Kluwer Academic Publishers, 1976.

HANDL, J**. Ant-Based Methods for Task of Clustering and Topographic Mapping: Improvements, Evaluation and Comparison with Alternative Methods.** 2003.130f. Tese (Doutorado em informática) Universidade Erlangen-Nürnberg, 2003.

HANDL, J.; KNOWLES, J.; DORIGO, M. On the performance of ant-based clustering. In: Abraham A., M. Koppen, K. Franke. **Design And Application of Hybrid Intelligent System**. IOS Press, p. 204-213, 2003.

HANDL J.; KNOWLES, J.; DORIGO, M. Ant-Based Clustering: A Comparative Study of its relative performance with respect to k-means, average link and 1D-SOM. **IRIDIA - Technical Report Series 24**, p. 1-16, 2003.

HÄRDLE, W.; SIMAR, L. **Applied Multivariate Statistical Analysis**. New York: Springer, 2007.

HARTIGAN, J. A. **Clustering Algorithms**. New Jersey: Wiley, 1975.

JAIN, A. K.; MURTY, M. N.; FLYNN, P. J. Data Clustering: A Review. **ACM Computing Surveys 31**, p. 264-323, 1999.

JAIN, A. K.; DUBES, R. C. **Algorithms for Clustering Data**. New Jersey: Prentice Hall, 1988.

KAUFMAN L.; ROUSSEEUW, P. J. **Finding Groups in Data, An Intruduction to Cluster Analysis.** New Jersey: Wiley Interscience, 1990.

KOGAN, J.; NICHOLAS, C. ; TEBOULLE, M. **Grouping Multidimensional Data, Recent Advances in Clustering.** New York: Springer, 1998.

LABROCHE, N.; MONMARCHÉ, N.; VENTURINI, G. A New Clustering Algorithm Based on Chemical Recognition System of Ants. In: **Proceedings of the 15th European Conference on Artificial Intelligence**. Amasterdam: IOS Press, 2003.

MANFRIN, M. et al. Parallel Ant Colony Optimization for the Traveling Salesman Problem. **IRIDIA – Technical Report Series 7**, p. 1-18, 2006.

MANIEZZO, V.; COLORNI, A. The Ant System Applied to the Quadratic Assignment Problem. **IEEE Transactions on Knowledge and Data Engineering 11**, p. 769-778, 1999.

METROPOLIS , N.; ULAM, S. The Monte Carlo Method. **Journal of the American Statistical Association 44**, p. 335-341, 1949.

MONMARCHÉ, N.; SLIMANE, M.; VENTURINI, G. Ant Class: Discovery of Clusters in Numeric Data by an Hybridization of an Ant Colony with K-Means Algorithm. **Laboratorie d'Informatique, University of Tours - Technical Report 213**, p. 1-21, 1999.

MONTEMANNI, R. et al. Ant Colony System for a Dynamic Vehicle Routing Problem. **Journal of Combinatorial Optimization 10**, p. 327–343, 2005.

NETO, J. M. M.; MOITA, G. C.. Uma Introdução à Análise Exploratória de Dados Multivariados. **Química Nova 21**, p. 467-469, 1997.

OCA, M. A. M.; GARRIDO, L.; AGUIRRE, J. L. An hybridization of an antbased clustering algorithm with growing neural gas networks for classification tasks. **ACM Symposium on Applied Computing**, Santa Fe, 2005.

O'HIGGINS P.; DRYDEN, I. L. Sexual dimorphism in hominoids: further studies of craniofacial shape differences in Pan, Gorilla, Pongo. **Journal of Human Evolution 24**, p. 183-205, 1993.

PEDRYCZ, W. **Knowledge-Based Clustering, From Data to Information Granules**. New Jersey: Wiley Interscience, 2005.

RAND, W. M. Objective Criteria for the Evaluation of Clustering Methods**. Journal of the American Statistical Association 336**, p. 846-850, 1971.

RENCHER, A. C. **Methods of Multivariate Analysis**. New Jersey: Wiley Interscience, 2002.

RUSPINI, E. H. Numerical methods for fuzzy clustering. **Information Sciences 2**, p. 319-350, 1970.

SCHOONDERWOERD, R. et al. Ant-based load balancing in telecommunications networks. **Adaptive Behavior 5**, p. 169-207, 1996.

SINHA, A. N.; DAS, N. ; SAHOO, G. Ant Colony Based Hybrid Optimizatin for Data Clustering. **Kybernetes 36**, p. 175-191, 2006.

SOBOL, I. **O Método de monte Carlo**. Moscou: MIR, 1972.

STÜTZLE, T.; HOOS, H. H. MAX–MIN Ant System, **Future Generation Computer Systems 16**, p. 889–914, 2000.

TALBI, E. G. et al. Parallel Ant Colonies for the quadratic assignment problem. **Future Generation Computer Systems 17**, p. 441–449, 2001.

TIMM, N. **Applied Multivariate Analysis**. New York: Springer, 2002.

TSAI, C.-F. et al. ACODF: a Novel Data Clustering Approach for Data Mining in Large Databases. **The Journal of Systems and Software 73**, p. 133–145, 2004**.**

WOLBERG, W. H.; STREET, W. N.; MANGASARIAN, O. L. **Breast Cancer Wisconsin (Diagnostic) Data Set.** Disponível em: <http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29>. Acesso em: 22 jun. 2007

Anexo A

Neste apêndice encontra-se o pseudo-algoritmo da nova proposta apresentada nesta dissertação.

```
Início ()
{
        Inicialização ()
        Livre_exploração ()
        Simulação ()
        Finalização ()
}

Inicialização ()
{
        Entre com as coordenadas dos N objetos.
        Faça i de 1 até N
            Faça j de 1 até N
            {
                Calcule a distância entre os objetos i e j.
                Determine τ ij = 0      //Fixa o valor inicial das trilhas
            }
        n_formigas = N/3                //Número de formigas = número de objetos / 3
        Coloque todas as formigas em um objeto aleatoriamente
}

Livre_exploração ()
{
        n_objetos = N/3         // O número de objetos para serem visitados em um viagem
        Faça k de 1 até n_formigas
        {
             D[k]=0             // Distância da viagem da formiga k.
             Faça i de 2 até n_objetos
             {
                 Aleatoriamente escolha um objeto para a formiga k visitar. A formiga não pode
voltar a um objeto já visitado.
                 A formiga visita o objeto escolhido
                 D[k] = D[k] + distância entre os objetos  i e  i - 1.
             }
        }
}
```

```
Simulação ()
{
        Faça s de 1 até número_de_simulações.
          Faça k de 1 até n_formigas.
          {
             max_distância_da_formiga[k] = - ln (γ) * D[k]     // Monte Carlo simula a distância máxima que a formiga pode percorrer. γ é um número aleatório entre zero e um. ln é o logaritmo neperiano.
            Distancia = 0.
            condição = 1.
            J = 2.
            Enquanto (condição==1)
                Se j >= N então condição = 0.
                Se não
                {
                    Aleatoriamente escolha um objeto (j) para a formiga k visitar. A formiga não pode voltar a um objeto já visitado.
                    Distancia = Distancia + distância entre os objetos j e j–1.
                    Se Distancia > max_distancia_da_formiga[k]  então condição = 0.
                    Se não
                    {
                                Visite o objeto escolhido.
                                τ ij = τ ij + Q / distância entre os objetos j e j–1.  // Q é uma constante.
                        j = j + 1.
                    }
                }

        }
}

Finalização() // Em português
{
        Calcule a média de feromônio das trilhas.
        Calcule o limiar de ativação pelo produto da média das trilhas por um fator alfa. Trilhas com valores maiores ou iguais que o limiar serão consideradas visíveis e trilhas com valores menores que o limiar são consideradas não visíveis.
        Use as trilhas visíveis para encontrar todos os grupos.
        Pequenos grupos serão unidos a grupos maiores que estão mais próximos.
}
```