

UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE INFORMÁTICA E
ESTATÍSTICA
CURSO DE PÓS-GRADUAÇÃO EM CIÊNCIA DA
COMPUTAÇÃO

5801 100

RUCUPERAÇÃO DE INFORMAÇÕES EM UMA BASE DE
DADOS RELACIONAL COM CONSULTA NEBULOSA

POR

Ilaim Costa Junior

Florianópolis, fevereiro de 1997

Recuperação de Informações em uma Base de Dados Relacional com Consulta Nebulosa

Ilaim Costa Junior

Esta dissertação foi julgada para obtenção de título de

Mestre em Ciência da Computação

e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação



Prof. Dr. Luiz Fernando Jacintho Maia - Orientador



Prof. Dr. Murilo Silva de Camargo - Coordenador do Curso

Banca examinadora



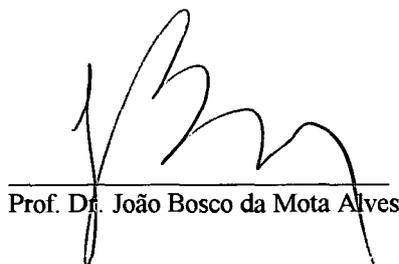
Prof. Dr. Luiz F. Jacintho Maia
Presidente



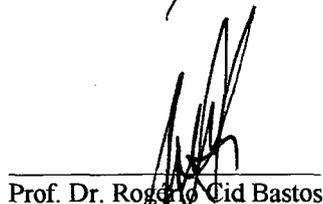
Prof. Dr. Jorge Muniz Barreto



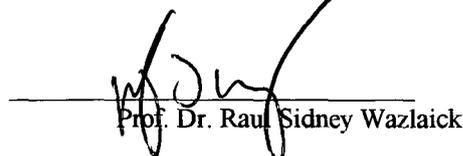
Prof. Dr. Paulo César Bezerra



Prof. Dr. João Bosco da Mota Alves



Prof. Dr. Rogério Cid Bastos



Prof. Dr. Raul Sidney Wazlaick

UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE INFORMÁTICA E
ESTATÍSTICA

CURSO DE PÓS-GRADUAÇÃO EM CIÊNCIA DA
COMPUTAÇÃO

ÁREA DE CONCENTRAÇÃO: SISTEMAS DE
CONHECIMENTO

RUCUPERAÇÃO DE INFORMAÇÕES EM UMA BASE DE
DADOS RELACIONAL COM CONSULTA NEBULOSA

POR

Ilaim Costa Junior

Orientador: Luiz Fernando Jacintho Maia, Dsc.

Dissertação submetida à Universidade Federal de Santa Catarina
como parte dos requisitos para a obtenção do grau de mestre em
Ciência da Computação.

Florianópolis, fevereiro de 1997

RESUMO

Uma das grandes necessidades atuais dos sistemas de informação, em especial os Bancos de Dados, é a representação e o tratamento de informações imprecisas. Existem alguns bons modelos teóricos, tais como o modelo de Kandel, o modelo de Prade; que apresentam soluções para esse problema, utilizando para isso a Teoria dos Conjuntos Nebulosos, em particular os operadores propostos por L. A Zadeh. Esse trabalho procura resolver esse problema fazendo uso da mesma teoria, porém utilizando os operadores de Caracterização e Reconhecimento de Conceitos propostos por L. Maia.

ABSTRACT

A great problem of information systems is its incapacity to manage imprecise information. In our special case, the relational data base systems, can not retrieve this kind of data. In the literature there are several models to solve this problem; like Kandel's model, Prade's model, all use Zadeh's functions and operators. In this document, fuzzy theory is used too, but we work with Maia's operators and functions to model and retrieve information.

Sumário

0) Prefácio.

1) Introdução.

2) Alguns Modelos de Bases de Dados Relacionais com Extensões para o Tratamento de Informações Imprecisas.

3) Tratamento da Informação com a Caracterização e o Reconhecimento de Conceitos.

4) Aspectos Computacionais.

5) Conclusão.

6) Referências Bibliográficas.

ÍNDICE ANALÍTICO

0. <i>Prefácio</i> -----	1
1. <i>Introdução</i> -----	3
1.1 <i>Elementos do modelo.</i> -----	3
1.1.1 <i>Estruturas de Dados.</i> -----	4
1.1.2 <i>Integridade dos Dados</i> -----	4
1.1.3 <i>Manipulação dos Dados.</i> -----	5
1.2 <i>Evolução do Modelo Relacional.</i> -----	5
1.3 <i>Problema do Tratamento da Informação Incompleta.</i> -----	6
1.4 <i>Conjuntos Nebulosos</i> -----	7
1.4.1 <i>Operações Básicas com conjuntos Nebulosos.</i> -----	12
1.5 <i>Princípio da Extensão</i> -----	13
1.6 <i>Números Nebulosos</i> -----	14
1.7 <i>Introdução à Caracterização e ao Reconhecimento de Conceitos.</i> -----	15
1.7.1 <i>Modificação de Conceitos.</i> -----	17
1.7.2 <i>Composição de Conceitos.</i> -----	19
2. <i>Alguns Modelos de Base de Dados Relacional com Extensões para Tratamento de Informações Imprecisas</i> -----	22
2.1 <i>Introdução</i> -----	22
2.2 <i>Tipos de Modelos</i> -----	22
2.3 <i>Modelos de Base de Dados Relacional Nebulosa</i> -----	23
2.3.1 <i>Modelo de Buckles - Petry</i> -----	23
2.3.2 <i>Modelo de Prade - Testemale</i> -----	29
2.3.3 <i>Modelo de Zemankova - Kandel</i> -----	33
3. <i>Tratamento da Informação com Caracterização e Reconhecimento de Conceitos</i> 36	
3.1 <i>Introdução.</i> -----	36
3.2 <i>Problema Proposto.</i> -----	36
3.3 <i>Mapeamento dos Conceitos.</i> -----	36
3.4 <i>Composição Conjuntiva dos Conceitos</i> -----	38
3.5 <i>Exemplos</i> -----	39
4. <i>Aspectos Computacionais</i> -----	45
4.1 <i>Método dos Quadrados Mínimos para o Ajuste de Parâmetros</i> -----	45
4.2 <i>Procedimento para o calibração.</i> -----	51
5. <i>Conclusão</i> -----	61

ÍNDICE DE EQUACÕES

<i>Equação 1-1</i>	16
<i>Equação 1-2</i>	Erro! Indicador não definido.
<i>Equação 1-3</i>	18
<i>Equação 1-4</i>	18
<i>Equação 1-5</i>	18
<i>Equação 1-6</i>	18
<i>Equação 1-7</i>	19
<i>Equação 1-8</i>	20
<i>Equação 1-9</i>	20
<i>Equação 1-10</i>	20
<i>Equação 3-1</i>	39
<i>Equação 3-2</i>	39
<i>Equação 3-3</i>	43
<i>Equação 4-1 Derivada primeira do desvio em relação a p1.</i>	46
<i>Equação 4-2 Derivada primeira do desvio em relação a k1.</i>	47
<i>Equação 4-3 Derivada primeira do desvio em relação a p2.</i>	47
<i>Equação 4-4 Derivada primeira do desvio em relação a k2.</i>	47
<i>Equação 4-5 Componente a13.</i>	47
<i>Equação 4-6 Componente a14.</i>	48
<i>Equação 4-7 Componente a23</i>	48
<i>Equação 4-8 Componente a24</i>	48
<i>Equação 4-9 Componente a31</i>	48
<i>Equação 4-10 Componente a32</i>	49
<i>Equação 4-11 Componente a41.</i>	49
<i>Equação 4-12 Componente a42.</i>	49

ÍNDICE DE FIGURAS

<i>Figura 1-1, conceito jovem</i> -----	9
<i>Figura 1-2</i> -----	9
<i>Figura 1-3 Números próximos de 2</i> -----	10
<i>Figura 1-4 conjunto nebuloso convexo</i> -----	11
<i>Figura 1-5 conjunto nebuloso não convexo</i> -----	12
<i>Figura 1-6 conceito jovem, com $p=24$ e $k=-3$, no eixo das ordenadas estão os graus de pertinência</i> -----	16
<i>Figura 1-7 conceito idoso, com $p=20$ e $k=3$, no eixo das ordenadas estão os graus de pertinência</i> -----	17
<i>Figura 1-8. conceito jovem, com $p=24$ e $k=-1/3$. No eixo das ordenadas estão os graus de pertinência.</i> -	19
<i>Figura 3-1 conceito jovem, $p=20$ (20 anos de idade) e $k=-3$. No eixo x são mostradas as idades</i> -----	37
<i>Figura 3-2 $p=60$(60 anos de idade) e $k=10$. No eixo x são mostradas as idades</i> -----	37
<i>Figura 3-3 conceito bom vendedor, $p=10$(10 mil reais) e $k=5$. No eixo x são mostrados os valores em milhares de reais.</i> -----	38
<i>Figura 3-4</i> -----	40
<i>Figura 3-5</i> -----	41
<i>Figura 3-6</i> -----	42
<i>Figura 3-7</i> -----	43
<i>Figura 3-8</i> -----	44

ÍNDICE DE TABELAS

<i>Tabela 1-1, conjunto nebuloso jovem.</i>	7
<i>Tabela 2-1</i>	24
<i>Tabela 2-2</i>	25
<i>Tabela 2-3</i>	26
<i>Tabela 3-1</i>	39
<i>Tabela 3-2</i>	40
<i>Tabela 3-3</i>	41
<i>Tabela 3-4</i>	42
<i>Tabela 3-5</i>	43
<i>Tabela 3-6</i>	44
<i>Tabela 4-1 Disposição inicial dos dados.</i>	51
<i>Tabela 4-2. Dados com pertinência alterada.</i>	51
<i>Tabela 4-3 Disposição da tabela após 5 iterações.</i>	52
<i>Tabela 4-4 Disposição da tabela após 10 iterações.</i>	52
<i>Tabela 4-5. Configuração da tabela após as 14 iterações</i>	53
<i>Tabela 4-6 Configuração da tabela após 5 iterações.</i>	53
<i>Tabela 4-7. Disposição da tabela após 5 iterações</i>	54
<i>Tabela 4-8. Tabela final.</i>	54
<i>Tabela 4-9 Parâmetros iniciais para o algoritmo.</i>	56
<i>Tabela 4-10. Graus de pertinência calculados a partir da tabela 6-9.</i>	56
<i>Tabela 4-11 Ordenação subjetiva dos candidatos.</i>	57
<i>Tabela 4-12. Parâmetros após a minimização.</i>	57
<i>Tabela 4-13. Resultados obtidos com o primeiro calibração de parâmetros.</i>	57
<i>Tabela 4-14. Especificação de pertinências desejadas.</i>	58
<i>Tabela 4-15. Novos parâmetros após a convergência.</i>	58
<i>Tabela 4-16. Nova classificação após novo ajustamento.</i>	58
<i>Tabela 4-17. Nova especificação de pertinências desejadas.</i>	59
<i>Tabela 4-18. Novos valores de parâmetros.</i>	59
<i>Tabela 4-19. Classificação após as alterações da tabela 6-17.</i>	59
<i>Tabela 4-20</i>	60
<i>Tabela 4-21. Ordenação após a inclusão dos novos exemplares.</i>	60

0. Prefácio

Hoje em dia a informação é um fator muito importante, não só para o desenvolvimento de um país, como também para a competitividade bem sucedida de muitas indústrias e corporações. Portanto, qualquer sociedade ou empresa que pretenda estar na frente deve ficar a par das diversas técnicas de armazenamento, transmissão, recuperação e análise de informações.

Historicamente a informação tem sido tratada de diversas formas, de acordo com sua natureza e objetivos. Dentre as técnicas mais modernas encontram-se os Sistemas Especialistas (SE), que cada vez mais têm sido difundidos e utilizados.

Como todos os sistemas de manipulação e tratamento de informações, os SE têm dificuldade de manipular informações imprecisas, incertas. Uma das formas de tratamento desse tipo de dados que tem sido muito utilizada é a Lógica Nebulosa; com resultados muito satisfatórios.

A meta desse trabalho é mostrar uma outra área que também faz tratamento de informações, e que é de grande importância para a sociedade de um modo geral; que vem a ser Banco de Dados (BD).

A meta dos Bancos de Dados é prover o usuário com habilidades de armazenar dados de maneira conveniente, permitir sua manipulação de forma segura e facilitar o processo de recuperação das informações; as quais são usadas muitas das vezes para a tomada de decisões e soluções de problemas que lidam com imprecisão. Portanto, há a necessidade desses BD manipularem a incerteza de maneira semelhante à humana, ou seja, realizarem raciocínio impreciso quando estão trabalhando com informações incertas, imprecisas, incompletas ou vagas. Como o Modelo Relacional (modelo de Bancos Dados mais difundido e utilizado) não trata tais dados de maneira satisfatória é desejável um modelo que, além de realizar todas as operações do modelo relacional, também represente, recupere, e armazene informações imprecisas. Outro aspecto importante seria a forma de consulta. Existe a necessidade de se fazer consultas utilizando dados imprecisos e com isso receber respostas satisfatórias, sendo essas incertas ou não. Também é interessante se fazer perguntas precisas e receber respostas imprecisas, se isso fosse o ideal (como ideal entende-se a melhor informação que o sistema consegue obter, ou recuperar).

Um ponto importante que não se pode deixar de mencionar seria a necessidade de tais sistemas de Bancos de Dados possuírem a “individualização”¹; o que reflete a percepção que o usuário tem dos dados e da informação requerida.

O trabalho tem como principal objetivo mostrar alternativas para a recuperação de informações em uma Base de Dados Relacional, que satisfazem uma expressão de seleção (query) formulada de maneira nebulosa.

¹ Esse conceito pode ser visto de forma satisfatória em [DAT91] e [ZEK85].

O trabalho divide-se em oito capítulos.

O primeiro trata dos Banco de Dados Relacionais e sua impossibilidade de tratar informações imprecisas. Faz também uma abordagem da Teoria dos Conjuntos Nebulosos proposta por Zadeh e os conjuntos e operadores nebulosos propostos por Maia.

O capítulo segundo apresenta algumas extensões do Modelo Relacional de tal forma que sejam tratados dados imprecisos.

O terceiro capítulo mostra o tratamento das informações imprecisas utilizando as funções apresentadas em [MAI91].

O quarto apresenta os resultados de uma implementação computacional do tratamento teórico mostrado no capítulo quinto.

O quinto é a conclusão.

1. Introdução

Este capítulo tenta dar uma visão clara do modelo de Banco de Dados Relacionais [DAT91], o qual foi introduzido por E. F. Codd em um artigo pioneiro datando de 1970 [COD70]. Serão mostrados os elementos básicos do modelo, e seus problemas no tratamento de informações imprecisas. Isso propiciará subsídios que garantem a necessidade e utilidade das extensões que futuramente serão apresentadas.

Após os conceitos mencionados serão introduzidas algumas noções sobre a Teoria dos Conjuntos Nebulosos [DUP80], [KLY95], [ZIM91], e as notações que serão utilizadas ao longo do trabalho.

Existe na literatura uma enorme quantidade de trabalhos dos mais diversos autores, entretanto, essa abordagem se concentrará na teoria clássica proposta por L. A. Zadeh. Posteriormente serão mostradas as idéias propostas em [MAI91], o que não conflita com a teoria clássica, apenas mostra uma outra representação para os conjuntos difusos e seus operadores.

1.1 Elementos do modelo.

O modelo relacional é aquele no qual os dados são percebidos pelo usuário como tabelas.

Os operadores à disposição do usuário (ex. para recuperação de dados) são operadores que geram novas tabelas a partir das antigas. Poderia existir, por exemplo, um operador para extrair um subconjunto de linhas de uma tabela dada e outro para extrair um subconjunto de colunas. É claro que a combinação de ambos os subconjuntos resultará em uma tabela.

A definição do Modelo Relacional se dará em três aspectos.

i) Estrutura de Dados:

- domínios (valores atômicos);
- relações n-árias (atributos, tuplas, chaves primárias).

ii) Integridade dos Dados:

- Os valores de chave primária não devem ser nulos;
- Os valores de chave externa devem equivaler aos valores de chave primária (ou ser nulos).

iii) Manipulação de Dados:

- Álgebra Relacional (ou os equivalentes em cálculo relacional):

união, interseção, diferença, produto, seleção, projeção, junção, divisão, atribuição relacional.

1.1.1 Estruturas de Dados.

As estruturas de dados vêm sendo caracterizadas pelo conceito de relação.

Uma relação pode ser representada graficamente por uma tabela bidimensional onde se identificam os seguintes elementos:

- CAMPOS: Correspondem aos nomes assinalados a cada coluna.
- CABEÇALHO: Linha contendo os nomes
- REGISTRO: Linha onde se representam dados relativos aos campos. Um registro também pode ser chamado de TUPLA.
- ATRIBUTO: É a variável associada a um determinado campo.
- DOMÍNIO: É o conjunto de todos possíveis valores que o atributo pode tomar.
- CORPO: É o conjunto de todos os registros (preenchidos) de uma relação.
- CHAVE PRIMÁRIA: É um conjunto de atributos que identificam univocamente cada tupla de uma relação. Podem existir vários destes conjuntos para uma relação dada, mas somente se selecionará um destes como Chave Primária, ficando os restantes como Chaves Alternativas.

1.1.1.1 Propriedades das Relações.

- i) Não há tuplas duplicadas. Sempre existirá uma chave primária;
- ii) Os atributos não estão ordenados. Esta propriedade é proveniente do fato de que o cabeçalho de uma relação é um conjunto matemático;
- iii) As tuplas, em si, não se encontram ordenadas;
- iv) Os valores dos atributos são atômicos, no sentido mostrado na definição de domínio. Uma relação que cumpre esta propriedade se diz Normalizada.

1.1.2 Integridade dos Dados

Uma base de dados consiste em uma configuração de dados que se supõe representar uma porção do mundo real. Nenhum modelo de Banco de Dados pode garantir que essa representação corresponda à realidade todo o tempo, pois isto significaria, entre outras coisas, que o Banco de Dados possui um conhecimento não só sobre os dados, mas também

sobre seu significado. O que se pode e deve garantir é que em um Banco de Dados não se introduza informações que não podem ser certificadas, nem que haja referência, em algum lugar do Banco de Dados, à informações que não existam nele mesmo.

1.1.3 Manipulação dos Dados.

A manipulação dos dados em um Banco de Dados é feita através de linguagens formais como Álgebra Relacional² e Cálculo Relacional³. Elas proporcionam operadores, mediante os quais são especificadas as operações a serem realizadas sobre as tabelas; e uma sintaxe, com a qual expressa-se o que se deseja obter das relações sem se ter que especificar o mecanismo para isso.

-**ÁLGEBRA RELACIONAL:** Possui três operadores:

- Operador de Atribuição: Atribui o resultado de outras operações sobre relações em uma nova relação;
- Operações Tradicionais Sobre Conjuntos : União, Interseção, Diferença e Produto Cartesiano;
- Operações Especiais: Seleção, Projeção, Reunião e Divisão.

-**CÁLCULO RELACIONAL:** Baseia-se em um ramo da lógica matemática denominado Cálculo de Predicados. Codd⁴ propõe pela primeira vez o Cálculo Relacional como uma forma específica de Cálculo de Predicados feita especialmente para os Banco de Dados Relacionais

1.2 Evolução do Modelo Relacional.

O Modelo Relacional Clássico tem sofrido muitas revisões e atualizações para se adequar a evolução dos sistemas de tratamento da informação. Algumas carências e inconvenientes presentes na definição original tem sido sanadas em sucessivas revisões. Vários esforços têm sido realizados nesse sentido, tanto por seu criador (E. F. Codd), como por outros autores, (C. J. Date e outros), e dentre todos os pontos revistos o que mais tem gerado polêmica tem sido as extensões dos operadores de manipulação de dados, principalmente no que se refere ao tratamento da informação imprecisa.

² Veja [DAT91]

³ Veja [DAT91]

⁴ Veja [COD70]

Relacionado com este problema está a alteração do conceito de Reunião para dar suporte adequado a esse tipo de informação. O conceito de atomicidade, a forma de manter a integridade dos dados no Banco de Dados, também tem sido objeto de revisão. Idéias que se baseiam no tratamento dos dados utilizando-se lógica de três valores ou quatro têm sido sugeridas.

1.3 Problema do Tratamento da Informação Incompleta.

A necessidade do tratamento da informação incompleta está em se querer representar e tratar situações reais em Banco de Dados. O problema é o enorme risco em se alterar substancialmente algumas das características básicas sobre as quais se constitui o Modelo Relacional. Isso se refere principalmente a representação desse tipo de informação na estrutura de uma relação, e sobre todos os aspectos relacionados com sua manipulação. O maior problema é que todas as propostas que pretendem proporcionar mecanismos para manipular este tipo de informação, se vêm forçadas inevitavelmente a modificar de forma substancial os operadores algébricos do Modelo Relacional, em particular a Seleção e Reunião. Por outro lado, quando se incorpora o tratamento da informação imprecisa, nenhuma proposta garante a continuidade das boas propriedades que apresenta o Modelo Relacional sem esse tipo de informação.

São inegáveis os esforços realizados para incorporar aos Bancos de Dados o tratamento e a representação de informações imprecisas. Alguns desses foram descritos acima, outros omitidos por fugirem ao escopo do trabalho; entretanto todos estão longe dos mecanismos de expressão utilizados pelos seres humanos. Neste sentido, um grande esforço tem sido realizado no campo da Inteligência Artificial [NIL80] para resolver os problemas teóricos e práticos relacionados com a elaboração de Banco de Dados Inteligentes [KLY95]. A idéia é criar mecanismos para armazenar e recuperar informações seguindo um esquema mais próximo do empregado pelo homem. Alguns dos principais aspectos que caracterizam esse tipo de abordagem são:

- - A possibilidade de representar e manipular informação cuja semântica se encontra mais próxima do esquema humano de representação da realidade;
- - A introdução de mecanismos que dotem o sistema da capacidade para inferir informação a partir da que se encontra previamente armazenada no sistema.

No primeiro caso temos a incorporação, nas bases de dados, da capacidade para representar e manipular informação imprecisa e subjetiva em todas suas vertentes. O segundo implica na integração das bases de dados com disciplinas relacionadas com a lógica.

As propostas sugeridas, em primeiro lugar se distinguem dos sistemas de Banco de Dados tradicionais (relacionais) no sentido de que a informação que este possui sobre um atributo existe ou não existe, não permitindo nenhum grau de incerteza ou de imprecisão sobre sua veracidade. Também não se obtém informações imprecisas a partir daquelas representadas de maneira precisa. Desse modo, não possuímos mecanismos para obter, por

exemplo, um atributo que armazene informações sobre as alturas de uma população, por exemplo dos indivíduos que são altos.

Até agora, para tratar o problema da representação e tratamento da informação imprecisa em Bancos de Dados foram usados apenas métodos clássicos, como os baseados na lógica trivalorada. Na seção que se segue será apresentada a Teoria dos Conjuntos Nebulosos, a qual nos possibilitará dar uma solução elegante a este problema. Trataremos da informação ausente como um caso particular do problema da representação e manipulação da informação imprecisa em Banco de Dados.

1.4 Conjuntos Nebulosos

A idéia dos conjuntos nebulosos provem de uma generalização do conceito clássico de conjunto. Um conjunto nebuloso é aquele no qual a pertinência de seus elementos não é absoluta (pertence ou não pertence). O elemento pertence ao conjunto segundo um determinado nível ao qual é dado o nome de grau de pertinência.

Podemos representar numericamente o grau de pertinência de um objeto a um conjunto (ou conceito) por um número entre zero e um, onde o valor um representa a pertinência total(ou absoluta) a o conjunto; o valor zero representa a não pertinência absoluta e os valores intermediários representam os valores graduais de pertinência. Quanto mais próximo de um, mais pertinente o conceito.

A tabela a seguir ilustra o conceito de um conjunto nebuloso.

<u>NOME</u>	<u>IDADE</u>	<u>GRAU DE PERTINÊNCIA</u>
<i>João</i>	20	0.8
<i>Marcela</i>	19	0.9
<i>Paulo</i>	17	1.0
<i>Patrícia</i>	22	0.7
<i>Ulisses</i>	25	0.5
<i>Maurício</i>	32	0.1
<i>Maria</i>	27	0.3

Tabela 1-1, conjunto nebuloso jovem.

Da terceira coluna da tabela pode-se inferir, por exemplo, que Paulo é absolutamente jovem, seguido por Marcela e logo após João, que são respectivamente menos jovens que ele. Logo após vem Patrícia, que é menos jovem que os anteriores; Ulisses por possuir grau de pertinência 0.5 não é jovem nem não jovem. A seguir observa-se Maria e em seguida Maurício que não são jovens (segundo a tabela).

É interessante notar que o conjunto jovem, como outros de natureza nebulosa, respeita critérios subjetivos.

De forma mais formal podemos definir conjuntos nebulosos como se segue.

Definição 1.1

Seja X uma coleção de objetos denotados genericamente por x . Então, um conjunto nebuloso A em X é o conjunto de pares ordenados:

$$A = \{ (x, \mu_A(x)) ; x \in X \}$$

Onde

$\mu_A(x)$ é chamado grau de pertinência ou função de pertinência de x em A .

Exemplo 1-1

A = números reais maiores que 10,

$$A = \{ (x, \mu_A(x)) ; x \in X \}, \text{ onde}$$

$$\mu_A(x) = (1 + (x - 10)^{-2})^{-1}, \text{ com } x > 10$$

$$0, \text{ com } x \leq 10$$

Exemplo 1-2

A = Inteiros próximos de 10

$A = 0.1/7, 0.5/8, 0.8/9, 1/10, 0.8/11, 0.5/12, 0.1/13$, onde os números a esquerda de / são os graus de pertinência. A direita tem-se os números próximos ou longe de 10.

Conceitos como os da tabela 1.1 (jovem), também podem ser representados graficamente:

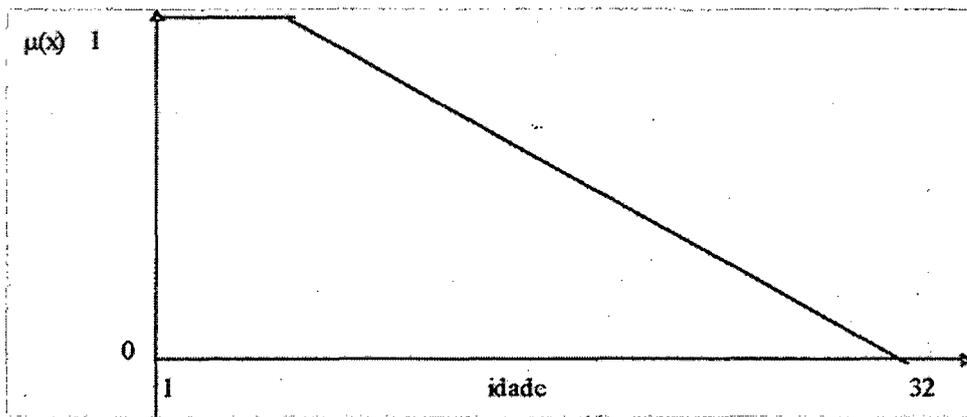


Figura 1-1, *conceito jovem*

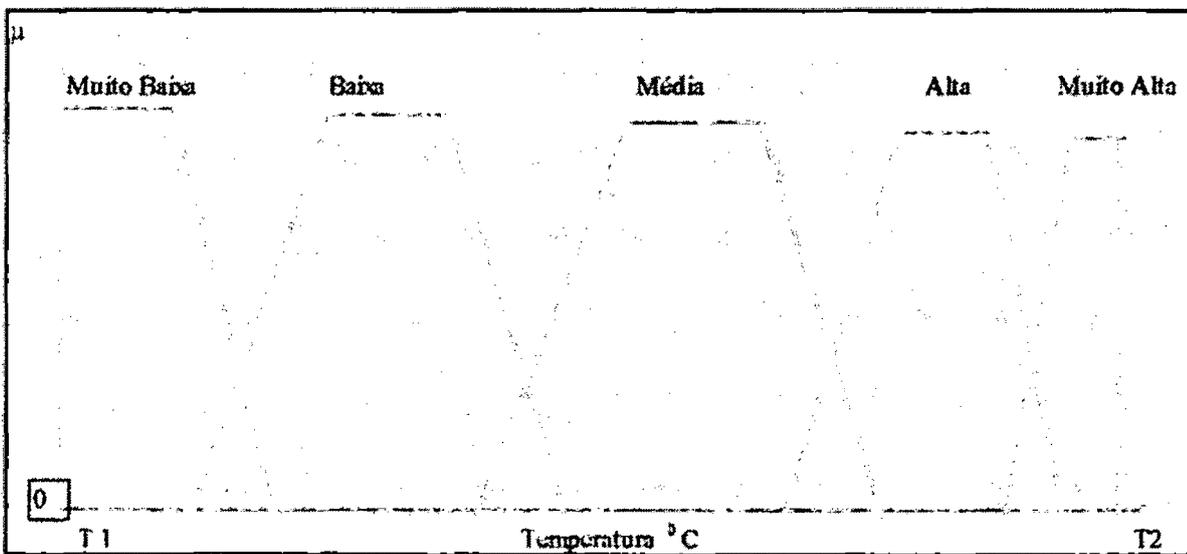


Figura 1-2

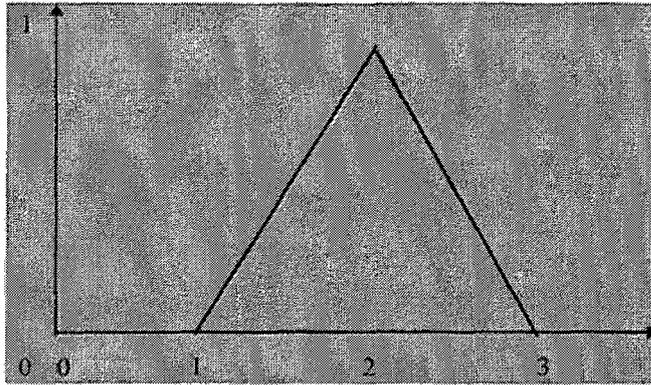


Figura 1-3 Números próximos de 2

Acima (figs. 1-2 e 1-3) tem-se as formas trapezoidais e triangulares para funções de pertinência. Apesar de serem as mais usadas nas aplicações mais comuns, existem diversas outras que podem ser utilizadas levando-se em conta o que se pretende representar. Maiores detalhes acerca desses conceitos podem ser vistos em [KLY95], [DUP80] e [ZIM91].

Definição 1.2

Define-se como suporte de um conjunto nebuloso A , o conjunto clássico $S(A)$ de todos os $x \in X$, tal que $\mu_A(x) > 0$.

Exemplo 1-3

Seja o conjunto nebuloso $A = \{ (1, .2), (2, .5), (3, .8), (4,1), (5, .7), (6, .3) \}$.

O conjunto $S(A) = \{ 1, 2, 3, 4, 5, 6 \}$.

Definição 1.3

Os conjuntos abaixo são definidos respectivamente como *corte α forte (strong α -cut)* e *corte α (α cut)*.

$$A_\alpha = \{ x \in X; \mu_A(x) > \alpha \}$$

$$A_\alpha = \{ x \in X; \mu_A(x) \geq \alpha \}$$

Exemplo 1-4

Seja o conjunto nebuloso $A = \{ (1, .2), (2, .5), (3, .8), (4,1), (5, .7), (6, .3) \}$.

$$A_{.2} = \{ 1, 2, 3, 4, 5, 6 \}$$

$$A_{.5} = \{ 2, 3, 4, 5 \}$$

$$A_{.8} = \{ 3, 4 \}$$

$$A_1 = \{ 4 \}$$

O corte α forte para $\alpha = .8$ é $A_{.8} = \{4\}$.

Definição 1.4

Um conjunto A é convexo se e somente se seus cortes α forem convexos.

Uma definição equivalente pode ser: A é convexo se e só se

$$\begin{aligned} \mu_A(\lambda x_1 + (1 - \lambda)x_2) &\geq \min(\mu_A(x_1), \mu_A(x_2)) \\ &; \\ \forall x_1 \in X \text{ e } \forall x_2 \in X, \quad \forall \lambda \in [0,1] \end{aligned}$$

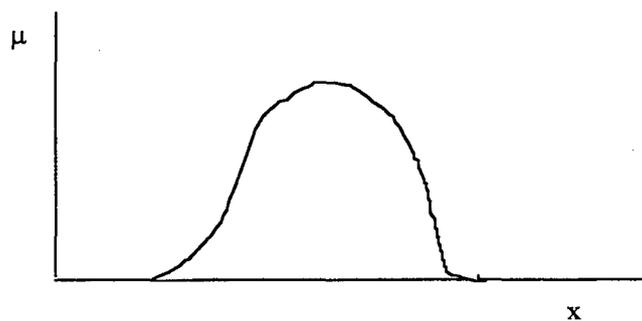


Figura 1-4 conjunto nebuloso convexo

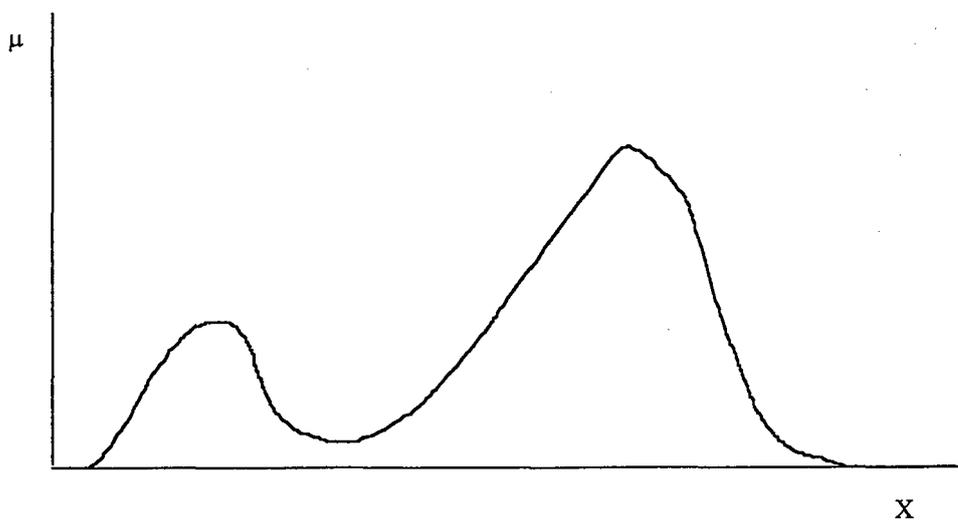


Figura 1-5 conjunto nebuloso não convexo

Definição 1.5

Define-se a cardinalidade de A , $|A|$, como

$$|A| = \sum_{x \in X} \mu_A(x)$$

Exemplo 1-5

Seja o conjunto nebuloso $A = \{ (1, .2), (2, .5), (3, .8), (4,1), (5, .7), (6, .3) \}$.

A cardinalidade do conjunto acima é $|A| = .2 + .5 + .8 + 1 + .7 + .3 = 3.5$.

1.4.1 Operações Básicas com conjuntos Nebulosos.

Apesar de constituírem uma estrutura consistente para os conjuntos nebulosos, não representam a única forma de extensão das operações dos conjuntos clássicos.

Definição 1.6

Define-se a *interseção* $C = A \cap B$ como:

$$\mu_c = \min (\mu_A(x) , \mu_B(x)) , x \in X$$

Definição 1.7

Define-se a *união* $D = A \cup B$ como:

$$\mu_d = \max (\mu_A(x) , \mu_B(x)) , x \in X$$

Definição 1.8

A função de pertinência do *complemento* de um conjunto nebuloso A , $\mu_{cA}(x)$, é definida como:

$$\mu_{cA}(x) = 1 - \mu_A(x), x \in X$$

Exemplo 1-6

Sejam os conjuntos nebulosos $A = \{ (1, .2), (2, .5), (3, .8), (4,1), (5, .7), (6, .3) \}$ e

$B = \{ (3, .2), (4, .4), (5, .6), (6, .8), (7,1), (8,1) \}$. Então $D = A \cap B = \{ (3, .2), (4, .4), (5, .6), (6, .3) \}$,

$E = A \cup B = \{ (1, .2), (2, .5), (3, .8), (4, 1), (5, .7), (6, .8), (7, 1), (8, 1) \}$,

$\mu_{cB} = \{ (1,1), (2,1), (3, .8), (4, .6), (5, .4), (6, .2), (9,1), (10,1) \}$.

1.5 Princípio da Extensão

Esse princípio, proposto por Zadeh, proporciona um método geral que generaliza elementos matemáticos não nebulosos para o tratamento de estruturas nebulosas.

Definição 1.9

Seja X o produto cartesiano dos conjuntos $X = X_1, X_2, \dots, X_n$ e A_1, \dots, A_r os conjuntos nebulosos em X_1, \dots, X_r respectivamente. Se f é um mapeamento de X para o

conjunto Y , $y = f(x_1, \dots, x_r)$, então o princípio da extensão permite definir um conjunto nebuloso B em Y tal que $B = \{(y, \mu_B(y)) \mid y = f(x_1, \dots, x_r); (x_1, \dots, x_r) \in X\}$, onde

$$\left\{ \begin{array}{l} \sup_{(x_1, x_2, \dots, x_r) \in f^{-1}(y)} \min\{\mu_A(x_1), \dots, \mu_A(x_r)\} \text{ se } f^{-1}(y) \neq \emptyset \\ 0 \quad \text{caso contrário} \end{array} \right.$$

onde f^{-1} é o inverso de f .

Para $r = 1$ o princípio da extensão se reduz a

$$B = f(A) = \{(y, \mu_B(y)) \mid y = f(x), x \in X\}, \text{ onde}$$

$$\left\{ \begin{array}{l} \sup_{x \in f^{-1}(y) \neq \emptyset} \mu_A(x) \text{ se } f^{-1}(y) \neq \emptyset \\ 0 \text{ caso contrário} \end{array} \right.$$

Exemplo 1-7

Seja $A = \{(-1, 5), (0, 8), (1, 1), (2, 4)\}$ e $f(x) = x^2$, aplicando o princípio da extensão temos:

$$B = f(A) = \{(1, 5), (0, 8), (1, 1), (4, 4)\}$$

1.6 Números Nebulosos

O conceito de números nebulosos foi introduzido pela primeira vez por Zadeh, com o propósito de analisar e manipular valores numéricos aproximados.

Definição 1.10

Denomina-se *número nebuloso* [DUP80] a um conjunto nebuloso convexo A normalizado em R (*reta real*), tal que:

- $\exists x_0 \in R, \mu_A(x_0) = 1; x_0$ é chamado valor médio de A ;
- $\mu_A(x)$ é seccionalmente contínua em R .

Essa definição tem sido muito modificada. Com a finalidade de melhorar a eficiência computacional e facilitar a aquisição de dados, têm sido usadas com frequência funções de pertinências trapezoidais.

Definição 1.11

Um número nebuloso A é chamado de positivo se sua função de pertinência é tal que $\mu_A(x_0) = 0 \forall x_0 < 0$. E é chamado negativo caso contrário.

Exemplo 1-8

Os conjuntos abaixo são números nebulosos:

aproximadamente 5 = { (3,.2), (4,.6), (5,1), (6,.7), (7,.1) }

aproximadamente 10 = { (8,.3), (9,.7), (10,1), (11,.7), (12,.3) }

{ (3,.8), (4,1), (5,1), (6,.7) } não é número nebuloso, pois $\mu(4)$ e $\mu(5)$ são iguais a 1.

1.7 Introdução à Caracterização e ao Reconhecimento de Conceitos.

As idéias apresentadas abaixo podem ser vistas de maneira detalhada em [Mai91].

Assumiremos aqui, sem perda de generalidade, valores no intervalo [-1, 1] para nossa função de pertinência, onde 1 representa a pertinência absoluta, -1 a não pertinência absoluta e 0 a indiferença de aplicabilidade de um determinado conceito a um objeto.

Definição 1.12

A função de pertinência que mapeia o atributo geométrico⁵ x ($x > 0$) em um conceito elementar z é denotada por:

$$Z = F_0(x;p:k)$$

e definida por

$$Z = (x^k - p^k) / (x^k + p^k)$$

Equação 1-1

$$p \in R^+ \text{ e } k \in R;$$

⁵ Atributo geométrico é aquele medido pela escala de medida geométrica. Essa por sua vez é aquela onde é significativa a razão entre duas medidas, como uma pessoa ter o dobro da idade da outra.

p e k são chamados parâmetros de observação.

Exemplo 1-9

Se o objeto é uma pessoa A, o atributo físico sua altura e o conceito elementar <peessoa alta>, com grau de pertinência $Z = 0$ é interpretado como sendo a pessoa nem alta, nem não alta, ou seja, ela está no limite de discordância ou concordância com o conceito de ser uma pessoa alta.

Os parâmetros p e k definem o ponto e a forma de transição da função que representa o grau de pertinência do conceito elementar ao objeto.

Essa função possui características como: facilidade de cálculo, pequeno número de parâmetros que a descrevem, é contínua, tem derivadas contínuas e admite uma inversa. A existência da inversa nos garante que não há perda de informações no processo de observação.

Sua principal propriedade é ser monotônica e sempre crescente ou decrescente dependendo do valor de k .

Por uma normalização de Z podemos fazê-la assumir valores no intervalo $[0,1]$:

$$Z = (x^k) / (x^k + p^k)$$

Equação 1-2

No entanto optou-se pelo intervalo $[-1, 1]$ por seu significado intuitivo, onde 0 significa a indiferença.

Exemplo 1-10

Pode-se visualizar a função Z e algumas variações na mudança de parâmetros.

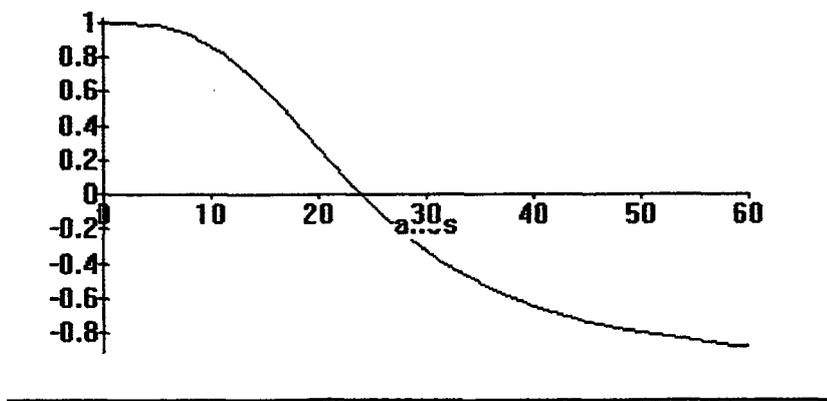


Figura 1-6 conceito jovem, com $p=24$ e $k=-3$, no eixo das ordenadas estão os graus de pertinência

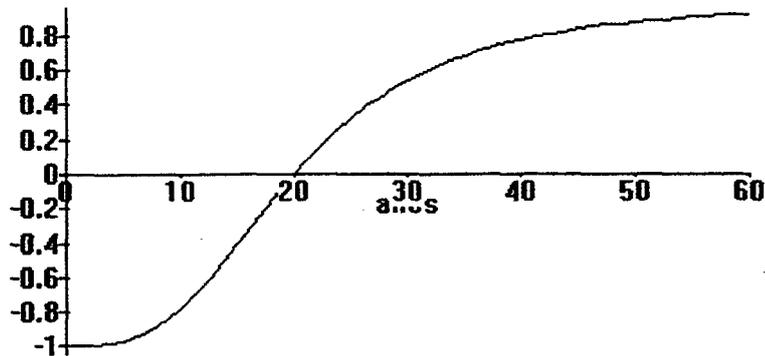


Figura 1-7 conceito idoso, com $p=20$ e $k=3$, no eixo das ordenadas estão os graus de pertinência

Observe que se Z é o grau de pertinência que associa alguém ao conceito < pessoa alta >, sua associação com < pessoa não alta > (negação) é igual a $(-Z)$, o que também pode ser obtido invertendo o sinal de k .

1.7.1 Modificação de Conceitos.

A modificação de conceitos pode ser feita de duas formas, primeiro, modificando os parâmetros p e k de uma função Z . Segundo, por uma função modificadora aplicada sobre um conceito elementar já existente; como se dá abaixo.

Definição 1.13

Define-se como função modificadora (Z') de um conceito elementar Z e denotada por

$$Z' = F_m(Z; q; w)$$

à equação

$$Z' = (y^w - q^w) / (y^w + q^w)$$

Equação 1-1

onde y é o atributo geométrico normalizado associado a Z , dado por

$$y = (1 + Z) / (1 - Z).$$

Equação 1-2

À Z' corresponde o atributo geométrico normalizado dado por :

$$y' = (1 + Z') / (1 - Z')$$

Equação 1-3

sendo

$$y' = (y / q)^w$$

Equação 1-4

com $q \in \mathbb{R}_+$ e $w \in \mathbb{R}$.

Teorema 1.1

***Seja $Z = F_0(x : p : k)$. Então $Z' = F_0(x : p' : k')$
ou $Z' = F_m(Z : q : w)$, com $w = k' / k$ e $q = (p' / p)^k$.***

Do teorema acima tem-se a garantia de que um conceito pode ser modificado segundo uma função modificadora aplicada sobre um conceito elementar já existente.

Maiores detalhes, bem como a demonstração podem ser vistos em [MAI91].

Exemplo 1-11 Conceito jovem.

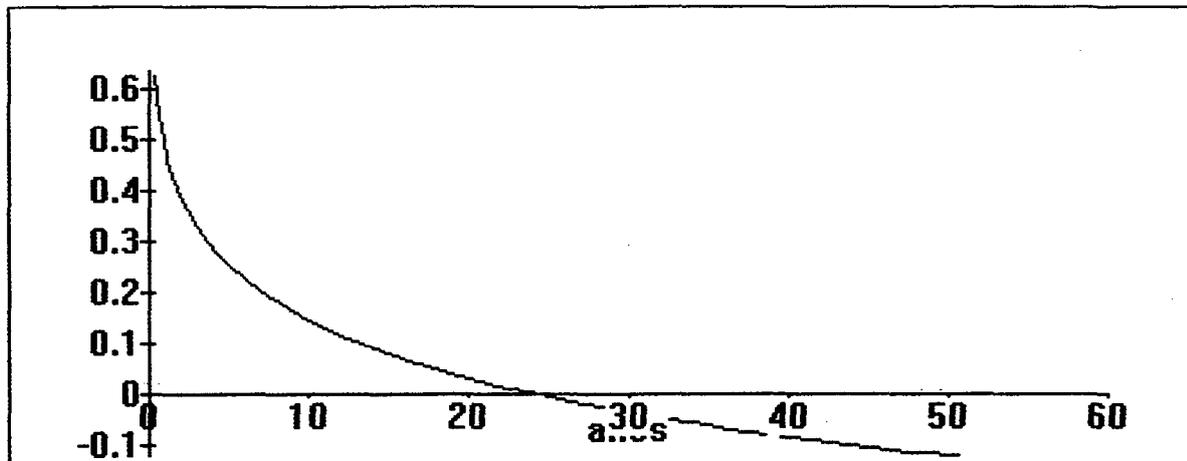


Figura 1-8. conceito jovem, com $p=24$ e $k=-1/3$. No eixo das ordenadas estão os graus de pertinência.

Tem-se conceitos elementares definidos sobre o atributo <juventude>, caracterizados por funções $F_m (Z : q : w)$, onde Z é o conceito primário <jovem> definido por:

$$Z = F_0 (\text{jovem} : 24 : -1/3)$$

Note que q influi no ponto de indiferença enquanto que w altera a forma de transição em torno desse ponto.

1.7.2 Composição de Conceitos.

Foram apresentadas funções para a composição de conceitos de forma disjuntiva e conjuntiva. Através delas pode-se definir conceitos não monotônicos associados a expressões lingüísticas usuais

Definição 1.14

A função conceitual disjuntiva $Z = F_d (z_1 : q_1 : w_1, z_2 : q_2 : w_2, \dots, z_n : q_n : w_n)$ é definida por

$$Z = (y - 1) / (y + 1)$$

Equação 1-5

onde

$$y = (y_1 / q_1)^{w_1} + (y_2 / q_2)^{w_2} + \dots + (y_n / q_n)^{w_n}$$

Equação 1-6

sendo

$$y_i = (1 + z_i) / (1 - z_i)$$

Equação 1-7

Definição 1.15

A função conceitual conjuntiva $Z = F_c(z_1 : q_1 : w_1, z_2 : q_2 : w_2, \dots, z_n : q_n : w_n)$ é definida por

$$Z = (y - 1) / (y + 1)$$

onde

$$1/y = (q_1 / y_1)^{w_1} + (q_2 / y_2)^{w_2} + \dots + (q_n / y_n)^{w_n}$$

Equação 1-8

sendo

$$y_i = (1 + z_i) / (1 - z_i)$$

1.7.2.1 Propriedades das Funções Disjuntivas e Conjuntivas.

As funções disjuntivas gozam de diversas propriedades, tais como:

(i) Propriedade Comutativa:

$$F_c(z_1 : q_1 : w_1, z_2 : q_2 : w_2) = F_c(z_2 : q_2 : w_2, z_1 : q_1 : w_1)$$

$$F_d(z_1 : q_1 : w_1, z_2 : q_2 : w_2) = F_d(z_2 : q_2 : w_2, z_1 : q_1 : w_1)$$

(ii) Propriedade Associativa:

$$F_c(z_1 : q_1 : w_1, z_2 : q_2 : w_2, z_3 : q_3 : w_3) =$$

$$F_c(F_c(z_1 : q_1 : w_1, z_2 : q_2 : w_2) : 1 : 1, z_3 : q_3 : w_3) =$$

$$F_c(z_1 : q_1 : w_1, F(z_2 : q_2 : w_2, z_3 : q_3 : w_3) : 1 : 1).$$

$$F_d(z_1 : q_1 : w_1, z_2 : q_2 : w_2, z_3 : q_3 : w_3) =$$

$$F_d(F_c(z_1 : q_1 : w_1, z_2 : q_2 : w_2) : 1 : 1, z_3 : q_3 : w_3) =$$

$$F_d(z_1 : q_1 : w_1, F(z_2 : q_2 : w_2, z_3 : q_3 : w_3) : 1 : 1).$$

Com essas idéias o capítulo chega ao seu final.

O trabalho utilizou todos os conceitos expostos aqui para representar e recuperar (através de diversos tipos de operações) as informações imprecisas em um Banco de Dados. Nos capítulos 4 e 5 será visto como isso pode ser implementado.

2. Alguns Modelos de Base de Dados Relacional com Extensões para Tratamento de Informações Imprecisas

2.1 Introdução

Neste capítulo trataremos de armazenamento e recuperação de informações imprecisas. Mostraremos os principais modelos de base de dados relacionais nebulosas existentes. Tais modelos tratam informações como “João tem mais de 2 anos”. Esse tipo de informação não pode ser armazenada em uma base de dados relacional. O que podemos fazer nesse caso é representar um total desconhecimento da idade de João. Nos modelos estendidos, que tratam informações imprecisas e que iremos apresentar, será possível armazenar e manipular informações dessa natureza.

2.2 Tipos de Modelos

É interessante a existência de modelos que tratem informações imprecisas da forma que o ser humano trata no seu dia a dia; manipulando a incerteza e a imprecisão de maneira natural, em qualquer situação que ocorra. Seria interessante armazenar e recuperar coisas como cor de um mesmo cabelo de diversas tonalidades, cor da pele das pessoas (levando em conta as variações de tons), conceitos como alto, baixo, jovem, velho, etc.

O termo Base de Dados Relacional Nebulosa tem sido mencionado. Se é relacional, mesmo sendo nebuloso, deve satisfazer requisitos do modelo relacional. É fato também que existem características elementares no modelo relacional que são incompatíveis com a representação imprecisa da informação. Como exemplo temos se uma relação armazena informações nebulosas para todos os seus atributos, poderá ser impossível a definição de chaves primárias e sendo assim não se poderá garantir o acesso a cada tupla da relação. Da mesma forma, também pode ser comprometido o conceito de unicidade.

Serão apresentados alguns modelos consagrados que tratam informações dessa natureza.

Para ser chamado de relacional, um sistema deve trabalhar sobre as características estabelecidas pelo modelo relacional. Como esse modelo apresenta várias variações, aproximações, e os modelos apresentados o satisfazem de diversas maneiras, podemos dizer que esses modelos são relacionais segundo determinado nível.

Apesar desses modelos apresentarem mecanismos adequados para representar informações nebulosas, é preciso que fique claro que estes não têm que estabelecer nenhuma semântica particular à informação. Cabe ao usuário esta tarefa, no momento de criar sua aplicação particular para resolver seus problemas.

Não basta o usuário prover suas informações com semânticas particulares, se o sistema não for capaz de operar sobre elas de maneira razoável. Do que nos serve representarmos a informação “José é jovem”, se na hora que solicitarmos os nomes de todos os indivíduos jovens, na lista não se encontrar o nome José? Para solucionarmos esse problema, os modelos que apresentaremos possuem um conjunto de operadores para recuperar e manipular as informações de natureza nebulosa.

Os modelos de base de dados relacional nebulosos enquadram-se, de maneira geral em três níveis distintos:

- Representação e recuperação de informações nebulosas, com tratamento da mesma, em vistas a inferências de novas informações;
- Representação e recuperação de informações nebulosas;
- Recuperação de informações nebulosas a partir de dados armazenados de forma precisa.

Dentre os três níveis citados acima, o primeiro não será referenciado nesse trabalho. O segundo será tratado de forma a apresentarmos alguns modelos com essa característica. Finalmente o terceiro será tratado com detalhes, por se tratar do objetivo do estudo.

2.3 Modelos de Base de Dados Relacional Nebulosa

Serão descritos alguns modelos característicos do segundo nível.

2.3.1 Modelo de Buckles - Petry⁶

Proposto em artigos publicados em 82 e posteriormente em 84. Inicialmente suportava domínios do tipo: conjunto finito de escalares, conjunto finito de números. Posteriormente foi feita uma extensão para um terceiro tipo de domínio, conjunto de números nebulosos.

Esse modelo possui, como caso especial, o modelo de base de dados relacional clássico. O lado nebuloso desse modelo difere do clássico de duas maneiras: primeiro, os elementos de uma tupla contida em uma relação podem ser subconjuntos categóricos do domínio; e segundo, uma relação de similaridade é definida sobre cada domínio. No primeiro caso é permitido aos elementos das tuplas serem tanto os únicos elementos do domínio (

⁶ A descrição e os exemplos dados desse modelo nesse texto basearam-se em: [BPS89], [BPS89], [BUP82], [BUP83], [BUP83], [BUP84], [KLY95].

modelo relacional clássico), quanto subconjuntos categóricos do domínio. O segundo caso pode ser visto no exemplo [KLY95] abaixo (tabela 2.1), que consiste de uma relação Mercados com os domínios Área, Tamanho e Potencial.

Relação: MERCADOS		
AREA	TAMANHO	POTENCIAL
leste	grande	bom
centro oeste	(grande, médio)	(moderado, bom)
sul	pequeno	(bom, excelente)

Tabela 2-1

No segundo caso, temos o fato de que em modelos de base de dados clássicos, uma relação de equivalência é definida em cada domínio, nos quais grupos são formados por elementos que são equivalentes. Essa equivalência é útil para se eliminar ou ignorar tuplas redundantes. Neste modelo de base de dados nebulosa, essa relação de equivalência é generalizada para uma relação de equivalência nebulosa ou uma relação de similaridade, onde o domínio das relações de equivalência modificam-se apropriadamente de acordo com o usuário, ou seja, de acordo com a semântica que o usuário deseja estabelecer para determinados valores.

A álgebra relacional nebulosa utilizada para acessar informações nesse modelo de base de dados nebulosa, consiste dos mesmos quatro componentes (complemento, união, interseção e junção) da álgebra relacional, adicionalmente permite a especificação de um limiar definindo o grau mínimo aceitável de similaridade entre elementos de domínios especificados. No caso especial de base de dados convencional esse limiar é igual a 1, assim as tuplas ou são aceitas ou são eliminadas. Na base de dados nebulosa, as tuplas podem ser aceitas se são consideradas suficientemente similares.

Como exemplo da utilização desse modelo vamos supor que nossa base de dados contenha a opinião de um grupo de especialistas em três opções. A base de dados contém duas relações: *Especialista* que possui os domínios Nome e Especialidade; *Valor*, com os domínios Opção, Nome e Opinião. As relações podem ser vistas nas tabelas 2.2 e 2.3. Temos também a definição de uma relação de compatibilidade nebulosa para o domínio Opinião da seguinte forma (matriz abaixo): Altamente Favorável(AF), Favorável(F), Pouco Favorável(PF), Pouco Contrário(PC), Contrário(C) e Altamente Contrário(AC).

	<i>AF</i>	<i>F</i>	<i>PF</i>	<i>PC</i>	<i>C</i>	<i>AC</i>
<i>AF</i>	1	.8	.6	.2	0	0
<i>F</i>	.8	1	.8	.6	.2	0
<i>PF</i>	.6	.8	1	.8	.6	.2
<i>PC</i>	.2	.6	.8	1	.8	.6
<i>C</i>	0	.2	.6	.8	1	.8
<i>AC</i>	0	0	.2	.6	.8	1

relação de compatibilidade nebulosa

Assumem-se valores categóricos para os outros domínios.

<u>NOME</u>	<u>RELACÃO :</u>	<u>ESPECIALISTA</u> <u>ESPECIALIDADE</u>
Cohen		Sociólogo
Fadem		Economista
Fee		Advogado
Feldman		Economista
Kass		Físico
Osborn		Sociólogo
Schreiber		Sociólogo
Specterman		Sociólogo

Tabela 2-2

<u>RELACÃO:</u>		<u>VALOR</u>
<u>OPÇÃO</u>	<u>NOME</u>	<u>OPINIÃO</u>
X	Osborn	Favorável
X	Fee	Contrário
X	Fadem	Pouco Favorável
X	Feldman	Altamente Favorável
Y	Cohen	Pouco Favorável
Y	Osborn	Pouco Favorável
Y	Fee	Altamente Favorável
Y	Schreiber	Favorável
Y	Kass	Favorável
Y	Fadem	Contrário
Y	Specterman	Altamente Favorável
Y	Feldman	Pouco Contrário
Z	Osborn	Contrário
Z	Kass	Pouco Contrário
Z	Fee	Pouco Favorável

Tabela 2-3

Supondo que se queira fazer a seguinte consulta ao banco de dados : “ Quais sociólogos estão de acordo com a posição da opção Y de Kass?”

O primeiro passo na recuperação dessa informação é obtermos a opinião da opção Y de Kass. Isso é feito através da seguinte operação algébrica relacional :

(**Project** (**Select** *Valor* **Where** *Nome = Kass and Opção = Y*) **over** *Opinião*) **giving** R1

Essa operação resulta na relação temporária R1 sobre o domínio Opinião

<u>RELACÃO: R1</u>
<u>OPINIÃO</u>
FAVORÁVEL

O segundo passo envolve a seleção de todos os sociólogos da tabela de especialistas. Isso é feito pela operação:

(**Project** (**Select** *Especialista* **Where** *Especialidade = Sociólogo*) **over** *Nome*) **giving** R2

Essa operação resulta na relação temporária R2 sobre o domínio Nome, listando somente sociólogos.

RELAÇÃO R2 - NOME
Cohen
Osborn
Schreiber
Specterman

A próxima relação temporária, R3, deve ser construída sobre os domínios Nome e Opinião, o que fornece as opiniões dos sociólogos de R2 sobre a opção Y. A expressão algébrica que realiza tal recuperação é:

(**Project** (**Select** (**Join** R2 **and** *Opinião* **over** *Nome*) **Where** *Opção = Y*) **over** *Nome, Opinião*) **giving** R3.

O que nos dá a relação R3:

<u>RELAÇÃO</u>	<u>R3</u>
<u>NOME</u>	<u>OPINIÃO</u>
<i>Cohen</i>	Pouco Contrário
<i>Osborn</i>	Pouco Favorável
<i>Schreiber</i>	Favorável
<i>Specterman</i>	Altamente Favorável

Finalmente faremos uma junção da relação R1, que nos dá a opinião de Kass e da relação R3, que nos fornece a opinião dos sociólogos, que especifica um limiar de similaridade da ordem de 0.75 no domínio Opinião, o qual é escolhido nesse exemplo para representar a condição de “Concordância Considerável”. A expressão algébrica para tal operação é:

(**Join** R3 **and** R1 **over** *Opinião*) **with** **Thres**(*Opinião*) ≥ 0.75 **and** **Thres**(*Nome*) ≥ 0

Observe que zero no limiar de similaridade para Nome é necessário para permitir a inclusão dos nomes nos conjuntos.

NOME	OPINIÃO
Osborn	Pouco Favorável
Schreiber	Favorável
Specterman	Altamente Favorável

A incerteza contida na especificação da “Concordância Considerável” e na similaridade definida sobre as opiniões possíveis é propagada para a resposta dada.

O modelo ilustrado e apresentado acima possui características interessantes, tais como o emprego de relações de similaridade usadas como ferramentas adequadas e intuitivas para representar a imprecisão de conceitos no sentido de “proximidade” entre os mesmos. No entanto, existem também alguns problemas que valem a pena serem citados, como o fato de não modelarem bem todos os aspectos nebulosos da informação, por exemplo, conceitos lingüísticos como jovem, alto, baixo, etc; a definição de tuplas viola a atomicidade na representação da informação, já que permite que os atributos que a compõem contenham valores que representam subconjuntos do domínio sobre os quais estão definidos.

Além do mais, como resultado de uma consulta, se agrupam tuplas em classes de equivalência das quais se podem extrair diferentes interpretações, podemos ser conduzidos a valores de atributos errados, o que acarretaria uma degradação irreversível da informação; como consequência disso não podemos definir chaves primárias sobre as relações, o que dificulta garantir a integridade de uma base de dados construída sobre esse modelo.

2.3.2 Modelo de Prade - Testemale ⁷

A estrutura de dados sobre a qual este modelo atua é semelhante a utilizada pelo modelo anterior. Este modelo utiliza medidas de possibilidade e de necessidade para a satisfação das condições estabelecidas na consulta.

Vamos através de um exemplo, tentar entender o mecanismo de representação de informações imprecisas utilizado por este modelo. Considere a relação Estudante (tabela 2.5), juntamente com os domínios Nome, Idade, Notas de Matemática e Notas de Física. Os valores das notas variam num intervalo de 0 a 20.

<u>NOME</u>	<u>IDADE</u>	<u>MATEMÁTICA</u>	<u>FÍSICA</u>
Tom	jovem	16	[14,16]
David	20	pouco mal	não aplicável
Bob	22	mais ou menos	bom
Jane	sobre 21	pouco bom	[10,12]
Jill	jovem	em torno de 10	em torno de 12
Joe	sobre 24	[14,16]	15
Jack	[22,25]	desconhecido	pouco bom

tabela 2.5

Na relação [10, 12] e [14, 16] representam intervalos de valores, “jovem”, “bom”, “pouco bom”, etc, são distribuições de possibilidades previamente definidas.

Suponhamos que desejamos fazer a seguinte consulta: “*Quais são as pessoas que possuem um desempenho muito maior que bom em matemática ?*” Para responde-la, primeiro teríamos que partir da definição da distribuição de possibilidade que representa a classificação “*bom*”. Posteriormente o modelo precisaria da definição do que vem a ser “*muito maior que*”. Nós utilizamos as seguintes definições:

$$bom = \{ 0.3/13, 1/14, 1/15, 1/16, 0.5/16.5 \}_p$$

$$\mu_{muito_maior_que}(u,v) = 0 \text{ se } u - v \leq 2; 0.5 \text{ se } u - v = 3; 1 \text{ se } u - v \geq 4$$

⁷ A descrição desse modelo mostrada no texto foi construída baseada em: [PRT84], [PRD90].

Esse modelo faz uma composição nebulosa de “*muito mais que*” com a distribuição de possibilidade dada por “*bom*”. Para isso utiliza-se da regra de composição contida em [Zadeh 75]. Segundo essa regra, a composição de um subconjunto nebuloso A de um universo de discurso U e um operador relacional R sobre $U \times V$ é dada por:

$$A \circ R = \{ \max_{u \in U} (\mu_A(u) \wedge \mu_R(u, v)) / v : v \in V \}$$

O resultado da aplicação dessa composição é:

$$F = \text{muito_maior_que} \circ \text{bom} = \{ 0.3/16, 0.5/17, 1/18, 1/19, 1/20 \}_p$$

Após a composição, recuperamos da relação as tuplas que satisfazem a consulta agrupadas em dois conjuntos dados pelas medidas de *necessidade* e de *possibilidade*. A *medida de possibilidade* é dada pela expressão:

$$\Pi(F | A) = \sup_{u \in D} \{ \mu_F(u) \wedge \pi_A(u) \}$$

onde D é o domínio de F e de A . Esta equação representa o grau máximo da interseção de F e de A . Já a *medida de necessidade* se calcula pela expressão:

$$N(F | A) = \inf_{u \in D} \{ \mu_F(u) \vee (1 - \pi_A(u)) \}$$

que representa o grau mínimo da união de F e o complemento de A .

O conjuntos respectivamente de possibilidade e necessidade resultantes da consulta são:

$$\Pi = \{ 0.3/Tom, 0.3/Joe, 1/Jack \}$$

$$N = \{ 0.3/Tom \}$$

Esse modelo possui operadores de seleção bem particulares. Para que possa ser feita a seleção as tuplas têm que satisfazer certas condições, pois do contrário não há recuperação. Para tanto elas devem ser atômicas ou compostas. Com relação às compostas, elas constam de várias condições atômicas, unidas pelos conectivos AND(\wedge), OR(\vee) e NOT(\sim), dados pela T-norma, T-conorma e pelo complemento da função de pertinência respectivamente.

Com relação às condições atômicas o modelo apresenta:

$A \Theta a$.

A possibilidade de que o valor do atributo A para o objeto x caia no conjunto de elementos que estão em uma relação Θ com ao menos um elemento de a é dada por:

$$\Pi (a \circ \Theta | A(x)) = \sup_{d \in D} \min(\mu_{a \circ \Theta}(d), \pi_{A(x)}(d))$$

onde $\mu_{a \circ \Theta}(d) = \sup_{d' \in D} \min (\mu_{\Theta} (d, d'), \mu_a (d'))$, com D sendo o domínio do atributo A , μ_{Θ} é a de pertinência (nebulosa ou não) definida sobre $D \times D$, $\pi_{A(x)}$ é a distribuição de possibilidade que restringe para x os possíveis valores do atributo A , a é uma constante (nebulosa ou não), o símbolo $|$ mostra a opção entre duas alternativas, separando a componente consulta dos itens da base de dados.

$A \Theta B$.

De maneira análoga a anterior, a possibilidade (necessidade) de que o valor do atributo A para o objeto x está em relação Θ com o valor do atributo B para o mesmo objeto x é dada por:

$$\Pi(\Theta | (A(x), B(x))) = \sup_{(d, d') \in D \times D} \min(\mu_{\Theta}(d, d'), \pi_{(A(x), B(x))}(d, d'))$$

$$N(\Theta | (A(x), B(x))) = \inf_{(d, d') \in D \times D} \max(\mu_{\Theta}(d, d'), 1 - \pi_{(A(x), B(x))}(d, d'))$$

Para atributos não interativos, ou seja, que não existe dependência em seus respectivos valores a expressão será:

$$\Pi(\Theta | (A(x), B(x))) = \sup_{(d, d') \in D \times D} \min(\mu_{\Theta}(d, d'), \pi_{A(x)}(d), \pi_{B(x)}(d'))$$

$$N(\Theta | (A(x), B(x))) = \inf_{(d, d') \in D \times D} \max(\mu_{\Theta}(d, d'), 1 - \pi_{A(x)}(d), 1 - \pi_{B(x)}(d'))$$

2.3.2.1 Conclusão

Foi descrito uma extensão do modelo relacional para o armazenamento e a recuperação de informações nebulosas. A distribuição de possibilidades é a ferramenta central utilizada nessa abordagem. A linguagem de consulta descrita foi implementada em MACLISP, e os primeiros experimentos produziram resultados que estão em plena concordância com os exemplos e operadores apresentados.

Futuramente pretende-se construir uma interface em linguagem natural para a linguagem de consulta.

2.3.3 Modelo de Zemankova - Kandel⁸

Este modelo se compõem em três partes:

- Uma base de dados de valores, na qual os dados são organizados de forma similar aos outros modelos possibilísticos;
- Uma base de dados explicativa, onde estão armazenadas as definições para os subconjuntos nebulosos e para as relações nebulosas;
- Um conjunto de regras de tradução, que se empenham, por exemplo na manipulação de adjetivos.

A manipulação dos dados está baseada na álgebra relacional, a qual foi implementada de forma estendida (nebulosa) pelo sistema RIM (Relational Information Management) desenvolvido pela Boeing Co.

Tratando da recuperação da informação em si; esse modelo recupera uma consulta de forma semelhante ao modelo de Prade - Testemale, no entanto a medida de possibilidade empregada para determinar a compatibilidade do subconjunto nebuloso F da condição, com o valor do atributo A para cada tupla da relação é dado por:

$$p_A(F) = \sup_{u \in D} \{ \mu_F(u) \cdot \pi_A(u) \}$$

e a medida de certeza por:

$$C_A(F) = \max_{u \in D} \{ 0, \inf \{ \mu_F(u) \cdot \pi_A(u) \} > 0 \}$$

A medida de certeza é usada no lugar da medida de necessidade de Prade - Testemale.

Com relação a seleção, o modelo parte de uma relação de similaridade θ sobre $D \times D$ e a partir dela se constrói outras relações de comparação. Considere os exemplos “aproximadamente igual”, “maior que” e “menor que”. Se s é uma relação de similaridade a partir da qual construímos a relação “maior que”:

$$\mu_{maior_que}(x,y) = \begin{cases} 1 - 0.5 \cdot s(x,y) & \text{se } x \geq y \\ 0.5 \cdot s(x,y) & \text{se } x < y \end{cases}$$

⁸ A descrição apresentada desse modelo nesse texto foi baseada em: [ZEK85], [SCK94].

As medidas de possibilidade correspondentes podem ser construídas utilizando-se a expressão anterior de medida de possibilidade, o que resultaria para condições do tipo $A \in \mathcal{F}$, com f atômico :

$$P_{\Theta \circ f}(A(x)) = \sup_{d \in D} \{ \mu_{\Theta}(d, f) \cdot \pi_{A(x)}(d) \}$$

Para condições do tipo $A \notin \mathcal{F}$, onde F não é atômico, o modelo não apresenta nenhuma regra para se executar os cálculos.

Apresentamos sucintamente três modelos com abordagem nebulosa para tratamento de informações imprecisas. Mostraremos algumas características que os mesmos possuem e que nos parecem bastante interessantes.

- Representar a informação nebulosa mediante o emprego de domínios construídos sobre distribuições de possibilidades, tem mostrado ser uma forma bem apropriada para se obter o significado de uma ampla variedade de informações dessa natureza. Existem outras informações que podem ser tratadas de maneira mais eficaz e natural com o emprego de modelagem por relações de similaridade ou de semelhança.
- No que se refere à manipulação dessas informações, tivemos formas diferentes de tratamentos dessa questão dadas pelos modelos, baseando-se estas na concepção empregada para avaliar a medida com que cada par de dados nebulosos encontram-se em correspondência com relação a uma propriedade dada.

2.3.3.1 Conclusão

Uma vez que o FRDB é um melhoramento dos sistemas de base de dados relacionais clássicos, ele pode ser utilizado em todas as áreas de aplicação dos sistemas convencionais. Além do mais, pode ser empregado com todo o seu potencial em áreas que trabalhem naturalmente com dados imprecisos.

O FRDB pode ser eficientemente usado como a base de informação em sistemas de apoio a decisão.

Uma vez que técnicas de lógica nebulosa têm sido usadas com sucesso em sistemas de diagnóstico médico, pode ser que seja viável representar as condições médicas com seus sintomas no esquema do FRDB, e utilizar seus mecanismos de seleção e recuperação para extrair o conhecimento armazenado.

A estrutura do FRDB e as características de linguagem natural da linguagem de consulta são interessantes para a aplicação em regras de inferência nebulosas. Assim, o FRDB pode auxiliar sistemas especialistas com regras de produção baseadas em dados imprecisos e modelagem aproximada de raciocínio humano.

Podemos concluir que o FRDB possui características excepcionais que o permite ter uma larga aplicabilidade em diversos campos e problemas.

3. Tratamento da Informação com Caracterização e Reconhecimento de Conceitos

3.1 Introdução.

Esse capítulo utiliza a consulta nebulosa (query) para tratar informações armazenadas em uma base de dados relacional. Para isso modela-se e recupera-se os dados através das funções e operadores propostos em [MAI91].

3.2 Problema Proposto.

Uma grande empresa do setor varejista com filiais em todo o País, deseja estabelecer, para seu uso, o conceito de “*bom gerente*”, onde um “*bom gerente*” deve ser caracterizado por uma pessoa “*jovem*” e “*boa de vendas*”. Uma vez que para empresa, uma pessoa *jovem* deve ter entre 22 e 28 anos, e para ser considerada *boa de vendas* deve possuir uma média de vendas anual da ordem de 100 mil reais; ao ser consultado, o banco de dados convencional (BDC) da área de recursos humanos não retornaria como bom candidato o funcionário que tem 30 anos e que possui uma média de vendas anual de mais de 100 mil reais, nem aquele que possui 20 anos e vendeu 10 mil no último ano. Os dois são candidatos interessantes para o cargo de *bom gerente*.

Se os conceitos “*jovem*” e “*boa de vendas*” estiverem disponíveis na Base de Dados da empresa, então “*bom gerente*” deve ser obtido a partir calibração da composição conjuntiva dos conceitos “*jovem*” e “*boa de vendas*”. Caso contrário, “*jovem*” e “*boa de vendas*” devem ser construídos, calibrados convenientemente de acordo com o ponto de vista do observador, e então, a partir daí inicia-se o processo de calibração de “*bom gerente*”.

Tomando a análise a partir do segundo caso, ou seja, os conceitos ainda não estão disponíveis, observa-se que BDC da empresa interpreta o conceito nebuloso *jovem* (extremamente subjetivo e detentor de muitos graus de satisfação), como se fosse um conceito categórico verdadeiro ou falso. Isso gera uma enorme perda de informação. O mesmo acontece com o conceito *bom vendedor*.

3.3 Mapeamento dos Conceitos.

Para contornar o problema da perda de informação foi feito o mapeamento dos valores das idades para os graus de satisfação do conceito *jovem* através de uma função de pertinência chamada de *função de observação*, (equação 1.1). Como já foi dito no primeiro

capítulo, essa função trabalha com graus de pertinência em $[-1,1]$. Para $P=20$ e $k = -3$ obtém-se um gráfico para o conceito *jovem*.

É importante notar que conceitos modelados pela equação 1.1 são ajustados e/ou modificados através dos parâmetros p e k , os quais exprimem o ponto de vista do observador do conceito em questão; permitindo assim, diferenciações nos conceitos de acordo com quem os observa

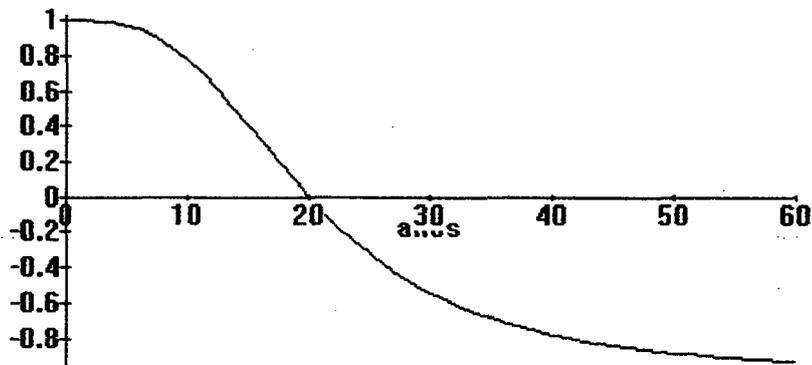


Figura 3-1 conceito *jovem*, $p=20$ (20 anos de idade) e $k=-3$. No eixo x são mostradas as idades

Os parâmetros p e k têm uma importância fundamental nos mapeamentos realizados através dessa função. Por meio de sua calibração, pode-se manipular alterações e modelagens dos conceitos, de maneira a expressar exatamente a subjetividade de cada usuário.

Alterando-se o valor de k e p , observar-se uma mudança na forma da curva, de modo a apresentar um conceito oposto ao anterior. O conceito a seguir representa o *não jovem*.

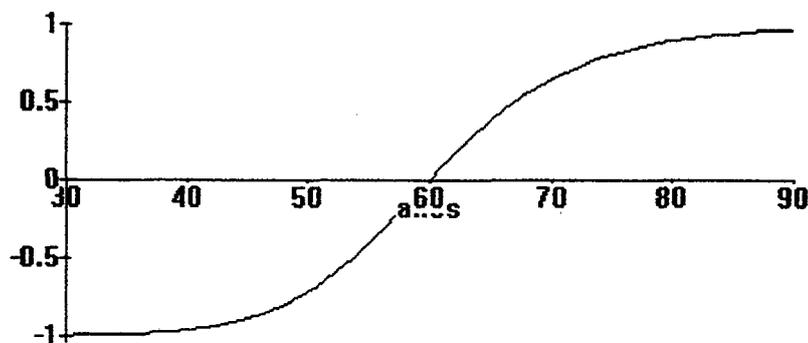


Figura 3-2 $p=60$ (60 anos de idade) e $k=10$. No eixo x são mostradas as idades

Novamente, é importante salientar a importância dos parâmetros p e k . Observe que todos os conceitos mostrados até então foram obtidos graças a uma calibração (ou manipulação dos valores) desses parâmetros.

Voltando ao problema original, facilmente se cria, de forma semelhante às anteriores, o conceito *bom vendedor*, a partir do mapeamento dos valores das vendas para os graus de satisfação do conceito através da equação 1.1, onde os valores do eixo x representam as vendas do último ano em milhares de reais, figura abaixo.

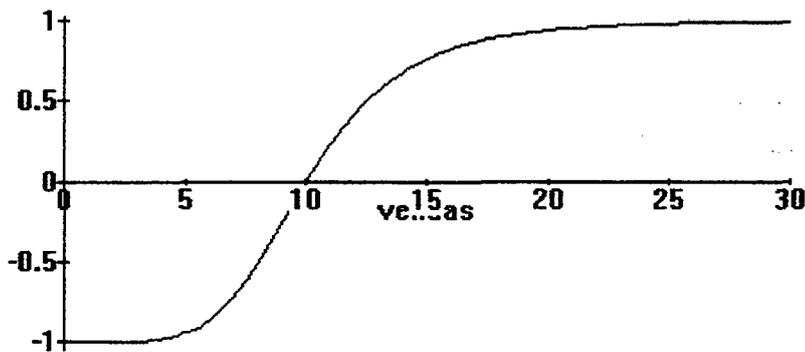


Figura 3-3 conceito *bom vendedor*, $p=10$ (10 mil reais) e $k=5$. No eixo x são mostrados os valores em milhares de reais.

Note que, em contraposição aos exemplos anteriores, aqui o valor de k é positivo, o que produziu uma curva totalmente “inversa” às vistas até agora.

3.4 Composição Conjuntiva dos Conceitos

O conceito composto *jovem e bom vendedor* será representado num espaço homogêneo de graus de pertinências de conceitos simples, expressos em $[-1,1]$. Esse conceito se chamará *bom gerente*, o qual será expresso pela função $Z_M = F_C(Z_J, Z_V)$, onde Z_M , Z_J e Z_V são os graus de satisfação para os conceitos *bom gerente*, *jovem* e *bom vendedor*; sendo F_C a função de combinação conjuntiva adequada, essas funções são modeladas pelas equações 1.7, 1.9, 1.10.

A principal propriedade da função de combinação conjuntiva é que toda variação de Z_J e Z_V se refletirá em Z_M , o que significa que entre dois candidatos com a mesma idade, o melhor é o que possui melhor vendas; ou dentre os que têm a mesma venda, o melhor é o mais jovem.

O conceito *bom gerente* pode ser modelado, inicialmente definindo-se uma variável normalizada Y ligada ao conceito Z ; como na equação 1.5. Posteriormente toma-se as variáveis normalizadas Y_J e Y_V , ligadas aos conceitos Z_J , Z_V , para a variável normalizada Y_M , ligada ao conceito Z_M , através da expressão

$$\frac{1}{Y_M} = \left(\frac{q_J}{Y_J}\right)^{w_J} + \left(\frac{q_V}{Y_V}\right)^{w_V}$$

Equação 3-1

resultando no grau de satisfação do conceito *bom gerente*, expresso por

$$Z_M = \frac{Y_M - 1}{Y_M + 1}$$

Equação 3-2

A equação 3.1 é uma expressão com quatro parâmetros (q_j , q_v , w_j , w_v). Através da mudança de seus valores (calibração), conceitos simples podem ser ajustados a diferentes contextos ou a conceitos subjetivos.

3.5 Exemplos

Considere a tabela a seguir com 13 elementos, na qual cada elemento corresponde a um candidato a gerente. Os graus de pertinência dos conceitos *jovem* e *bom vendedor* foram calculados respectivamente com $P_j=24$, $K_j=-6$, $P_v=10$ e $K_v=3$. A tabela também mostra as variáveis normalizadas correspondentes, limitadas aos conceitos.

	IDADE	VENDAS	Y_j	Y_v	Z_j	Z_v
1	16	14	11.390	2.744	0.839	0.466
2	17	8	7.917	0.512	0.726	-0.123
3	18	15	5.618	3.375	0.698	0.543
4	19	12	4.862	1.728	0.605	0.267
5	20	7	2.986	0.343	0.498	-0.489
6	19	16	4.062	4.096	0.465	0.608
7	22	23	1.685	12.167	0.255	0.848
8	21	11	2.228	1.728	0.380	0.267
9	23	6	1.290	0.216	0.127	-0.645
10	28	20	0.396	8.000	-0.432	0.773
11	27	13	0.493	2.197	-0.399	0.374
12	24	25	0.000	15.625	0.000	0.880
13	32	28	0.178	21.952	-0.698	0.913

Tabela 3-1

A combinação conjuntiva dos conceitos *jovem* e *bom vendedor*, com parâmetros q_j , w_j , q_v e w_v , todos iguais a 1, resulta no valor Z_{M1} para o grau de satisfação do conceito composto *bom gerente*, como mostra a tabela 3.2 abaixo. A superfície de decisão para os parâmetros citados acima pode ser vista na figura 3.4, também abaixo.

	<i>idade</i>	<i>vendas</i>	Y_{est}	Z_{est}
1	16	14	2,211	0,377
3	18	15	2,108	0,357
6	19	16	2,039	0,342
7	22	23	1,480	0,194
4	19	12	1,212	0,096
8	21	12	0,973	-0,014
12	24	25	0,939	-0,031
2	17	8	0,480	-0,381
11	27	13	0,402	-0,426
10	28	20	0,377	-0,452
5	20	7	0,307	-0,529
9	23	6	0,185	-0,688
13	32	28	0,176	-0,700

Tabela 3-2

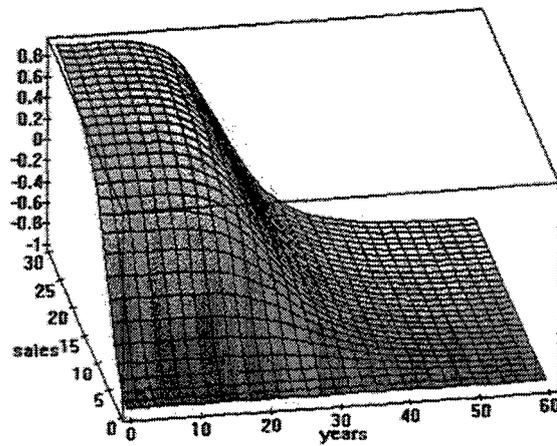


Figura 3-4

Se por ventura houver um desacordo no resultado da seleção, a relevância do fator *idade(years)* sobre as *vendas(sales)* pode ser reduzida. Para isso basta uma mudança no valor de W_j para 2, resultando na tabela 3.3 e na figura 3.5, ambas abaixo.

	idade	vendas	Y_{M2}	Z_{M2}
6	19	16	3,281	0,533
3	18	15	3,049	0,506
1	16	14	2,687	0,458
7	22	23	2,303	0,395
4	19	12	1,564	0,220
8	21	12	1,281	0,124
12	24	25	0,939	-0,031
2	17	8	0,507	-0,326
6	20	7	0,330	-0,503
11	27	13	0,219	-0,641
9	23	6	0,191	-0,679
10	28	20	0,154	-0,733
13	32	28	0,031	-0,939

Tabela 3-3

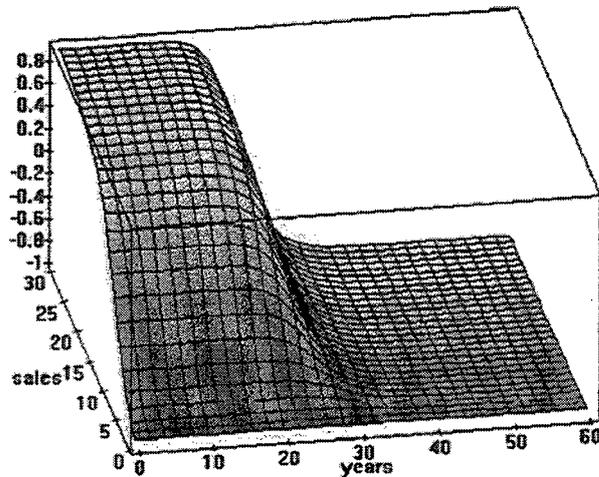


Figura 3-5

É interessante observar na tabela 3.5 que a posição dos três melhores candidatos foi invertida, refletindo o fato de serem as vendas(sales) mais importantes que as idades(years) nesse novo critério de seleção. Caso seja alterado o valor do parâmetro W_j para 4, observa-se na tabela 3.4 que o candidato número 7 é agora o melhor. A figura 3.6 também mostra esse novo contexto.

	IDADE	VENDAS	Y_{PI3}	Z_{PI3}
7	22	23	4,852	0,658
6	19	16	4,035	0,603
3	18	15	3,363	0,542
1	16	14	2,743	0,466
4	19	12	1,717	0,264
8	21	12	1,614	0,235
12	24	25	0,939	-0,031
2	17	8	0,511	-0,323
5	20	7	0,341	-0,491
9	23	6	0,200	-0,666
11	27	13	0,057	-0,891
10	28	20	0,024	-0,952
13	32	28	0,001	-0,998

Tabela 3-4

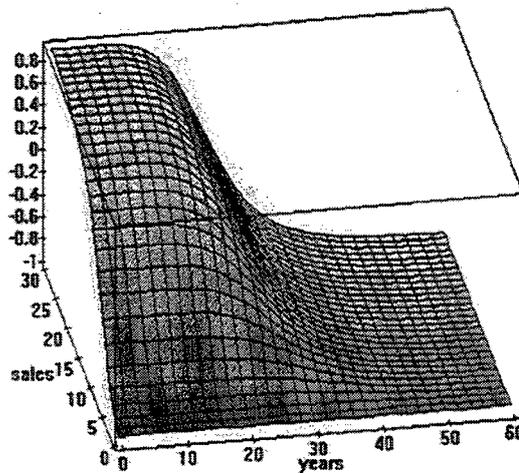


Figura 3-6

Retornando ao primeiro resultado, tabela 3.2, nota-se a indicação do melhor candidato (número 1) com apenas 16 anos de idade(year). Isso não é nada interessante, pois o mesmo não tem condições de assumir as responsabilidades legais necessárias ao cargo. Sendo assim, será criado um novo critério de seleção que inclui o fato do candidato ser obrigado a possuir idade(year) igual ou maior que 18 anos. O novo conceito, *maior*, pode ser definido por uma função de pertinência adequada relacionada com a idade(year) do candidato, ou pode ser definida como uma modificação do conceito *jovem*, expressado linguisticamente como *não tão jovem*.

O novo conceito *bom gerente* pode ser expresso como *jovem* e *bom vendedor* e *maior* ou como *jovem* e *bom vendedor* e *não tão jovem*; modelado da seguinte maneira:

$$\frac{1}{Y_{M_4}} = \left(\frac{q_J}{Y_J}\right)^{w_J} + \left(\frac{q_V}{Y_V}\right)^{w_V} + \left(\frac{q_M}{Y_M}\right)^{w_M}$$

Equação 3-3

Resultando em

$$Z_{M_4} = \frac{Y_{M_4} - 1}{Y_{M_4} + 1}$$

Os resultados da equação 3.3 com $Q_j=1$, $W_j=2$, $Q_v=1$, $W_v=2$, $Q_m=5$ e $W_m=-4$, são mostrados na tabela 3.5 e na figura 3.7 abaixo.

	IDADE	VENIDAS	Y_{M_4}	Z_{M_4}
7	22	23	2,690	0,458
6	19	16	1,799	0,283
8	21	12	1,736	0,269
4	19	12	1,203	0,092
12	24	25	0,994	-0,003
3	18	15	0,583	-0,263
11	27	13	0,231	-0,624
10	28	20	0,156	-0,729
5	20	7	0,114	-0,785
2	17	8	0,098	-0,820
9	23	6	0,045	-0,913
1	16	14	0,036	-0,929
13	32	28	0,031	-0,939

Tabela 3-5

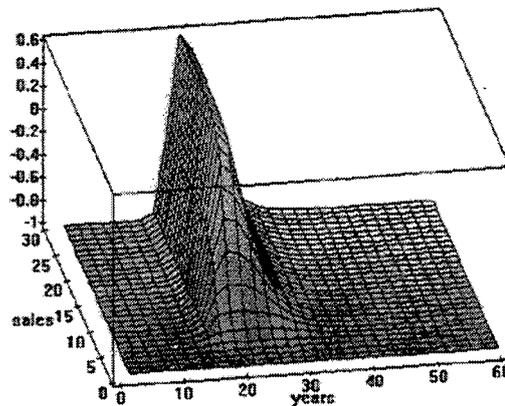


Figura 3-7

Mudando o valor de um único parâmetro Q_m de 5 para 8, gera-se um novo resultado Z_{m5} , que pode ser observado na tabela 3.6 e figura 3.8 abaixo.

	IDADE	VENDAS	Y_{m5}	Z_{m5}
6	19	16	5,356	0,685
7	22	23	2,772	0,470
3	18	15	2,756	0,468
4	19	12	2,164	0,368
8	21	12	1,843	0,297
12	24	25	0,995	-0,002
1	16	14	0,235	-0,619
11	27	13	0,231	-0,624
2	17	8	0,208	-0,655
10	28	20	0,156	-0,729
5	20	7	0,115	-0,792
9	23	6	0,045	-0,913
13	32	28	0,031	-0,939

Tabela 3-6

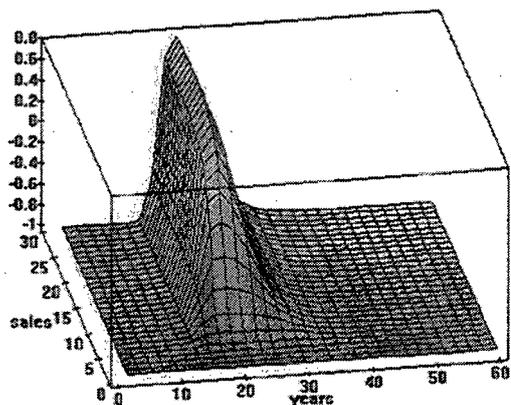


Figura 3-8

Esse último exemplo vem reforçar o expressivo poder de modelagem das expressões utilizadas nesse trabalho para se efetuar a recuperação de informações nebulosas em uma base de dados convencional. É interessante notar o fato delas serem mais simples do que as utilizadas na literatura em geral, vide capítulo 1.

No próximo capítulo será visto um algoritmo que proporciona um calibração interativa e subjetiva de usuários de base de dados relacional com consulta nebulosa.

4. Aspectos Computacionais

Nesse capítulo será abordado o método utilizado para que um usuário qualquer possa realizar, de forma interativa, o calibração e a recuperação da informação de maneira bem subjetiva.

Na seção anterior foram criados três conceitos. Um conceito de *pessoa jovem*, um de *bom vendedor* e um último, que resultou de uma composição disjuntiva dos anteriores que foi o de *bom gerente*. A definição do conceito ideal de *bom gerente* foi feita mediante a troca conveniente dos valores dos parâmetros da equação 1.11. É interessante notar o fato de que, apesar dessa escolha mais adequada dos parâmetros da equação permitir uma definição subjetiva para o conceito *bom gerente*, não se pode negar que ela foi feita de uma maneira muito pouco prática e cômoda. Seria conveniente a existência de uma forma automatizada para fazer tal escolha. Para isso foi utilizado o método dos quadrados mínimos.

4.1 Método dos Quadrados Mínimos para o Ajuste de Parâmetros⁹

É uma técnica de aproximação bastante popular e simples, utilizada, por exemplo, para a aproximação de medidas de entidades físicas obtidas experimentalmente, ou dados que já vêm acompanhados por um certo grau de incerteza. O método a ser apresentado leva em consideração a existência de erros nos valores a serem aproximados, e o critério de aproximação consistem em minimizar os resíduos.

Em virtude da não-linearidade da equação 1.11 foi necessária uma dedução especial do método (Quadrados Mínimos Não-Lineares)¹⁰. Para auxiliar o desenvolvimento do algoritmo foi utilizado o software de computação algébrica MAPLEV.

A seguir são mostrados os passos da dedução no MAPLEV.

⁹ Esse assunto pode ser visto com detalhes em [CUN93] e [PFT88].

¹⁰ Uma dedução genérica e bem didática dos Quadrados Mínimos Não-Lineares pode ser vista em [CUN93].

Em primeiro lugar deve-se definir a equação cujos parâmetros se pretende calibrar. Lembrando que as quatro letras entre parênteses representam os parâmetros a serem ajustados, X1(i) e X2(i) são as variáveis. Isso pode ser visto abaixo.

$$f := (p1, k1, p2, k2) \rightarrow \left(\frac{p1}{x1(i)} \right)^{k1} + \left(\frac{p2}{x2(i)} \right)^{k2}$$

Equação a ter os parâmetros ajustados

O segundo passo é calcular o desvio total, Desvio_total, onde Y(i) representa os graus de pertinência da composição e p1, k1, p2, k2 são os parâmetros da composição a serem ajustados. A operação é mostrada a baixo.

$$\text{Desvio_total} := \sum_{i=0}^n (y(i) - f(p1, k1, p2, k2))^2$$

Desvio a ser minimizado.

A seguir serão calculadas as derivadas primeiras do desvio total em relação aos parâmetros, as quais posteriormente deverão ser igualadas a zero (condições para os pontos críticos). Isso pode ser visto nas quatro equações que se seguem.

$$a := \sum_{i=0}^n \left(-2 \frac{\left(y(i) - \left(\frac{p1}{x1(i)} \right)^{k1} - \left(\frac{p2}{x2(i)} \right)^{k2} \right) \left(\frac{p1}{x1(i)} \right)^{k1}}{p1} \right)$$

Equação 4-1 *Derivada primeira do desvio em relação a p1.*

$$a1 := \sum_{i=0}^n \left(-2 \left(y(i) - \left(\frac{p1}{x1(i)} \right)^{k1} - \left(\frac{p2}{x2(i)} \right)^{k2} \right) \left(\frac{p1}{x1(i)} \right)^{k1} \ln \left(\frac{p1}{x1(i)} \right) \right)$$

Equação 4-2 Derivada primeira do desvio em relação a k1.

$$b := \sum_{i=0}^n \left(-2 \frac{\left(y(i) - \left(\frac{p1}{x1(i)} \right)^{k1} - \left(\frac{p2}{x2(i)} \right)^{k2} \right) \left(\frac{p2}{x2(i)} \right)^{k2} k2}{p2} \right)$$

Equação 4-3 Derivada primeira do desvio em relação a p2.

$$b1 := \sum_{i=0}^n \left(-2 \left(y(i) - \left(\frac{p1}{x1(i)} \right)^{k1} - \left(\frac{p2}{x2(i)} \right)^{k2} \right) \left(\frac{p2}{x2(i)} \right)^{k2} \ln \left(\frac{p2}{x2(i)} \right) \right)$$

Equação 4-4 Derivada primeira do desvio em relação a k2.

O próximo passo consiste no cálculo da matriz Jacobiana. Uma vez que devemos calcular as derivas de a, a1, b e b1 em relação aos quatro parâmetros, a Jacobiana será uma matriz com 16 elementos 4 X 4. Algumas derivadas e conseqüentemente algumas componentes da matriz podem ser vistas nas figuras que seguem.

$$a13 := \sum_i \left(2 \frac{\left(\frac{p2}{x2(i)} \right)^{k2} k2 \left(\frac{p1}{x1(i)} \right)^{k1} k1}{p2 p1} \right)$$

Equação 4-5 Componente a13.

$$a_{14} := \sum_i \left(2 \frac{\left(\frac{p_2}{x_2(i)} \right)^{k_2} \ln \left(\frac{p_2}{x_2(i)} \right) \left(\frac{p_1}{x_1(i)} \right)^{k_1} k_1}{p_1} \right)$$

Equação 4-6 *Componente a14.*

$$a_{23} := \sum_i \left(2 \frac{\left(\frac{p_2}{x_2(i)} \right)^{k_2} k_2 \left(\frac{p_1}{x_1(i)} \right)^{k_1} \ln \left(\frac{p_1}{x_1(i)} \right)}{p_2} \right)$$

Equação 4-7 *Componente a23*

$$a_{24} := \sum_i \left(2 \left(\frac{p_2}{x_2(i)} \right)^{k_2} \ln \left(\frac{p_2}{x_2(i)} \right) \left(\frac{p_1}{x_1(i)} \right)^{k_1} \ln \left(\frac{p_1}{x_1(i)} \right) \right)$$

Equação 4-8 *Componente a24*

$$a_{31} := \sum_i \left(2 \frac{\left(\frac{p_2}{x_2(i)} \right)^{k_2} k_2 \left(\frac{p_1}{x_1(i)} \right)^{k_1} k_1}{p_2 p_1} \right)$$

Equação 4-9 *Componente a31*

$$a_{32} := \sum_i \left(2 \frac{\left(\frac{p_2}{x_2(i)}\right)^{k_2} k_2 \left(\frac{p_1}{x_1(i)}\right)^{k_1} \ln\left(\frac{p_1}{x_1(i)}\right)}{p_2} \right)$$

Equação 4-10 *Componente a32*

$$a_{41} := \sum_i \left(2 \frac{\left(\frac{p_2}{x_2(i)}\right)^{k_2} \ln\left(\frac{p_2}{x_2(i)}\right) \left(\frac{p_1}{x_1(i)}\right)^{k_1} k_1}{p_1} \right)$$

Equação 4-11 *Componente a41.*

$$a_{42} := \sum_i \left(2 \left(\frac{p_2}{x_2(i)}\right)^{k_2} \ln\left(\frac{p_2}{x_2(i)}\right) \left(\frac{p_1}{x_1(i)}\right)^{k_1} \ln\left(\frac{p_1}{x_1(i)}\right) \right)$$

Equação 4-12 *Componente a42.*

Os elementos a15, a25, a35 e 45 são, respectivamente a, a1, b, b1.

Gerados esses elementos pode ser descrito o algoritmo para o calibração dos parâmetros da equação 1.11.

Algoritmo:

$p1^0$, $p2^0$; $k1^0$; $k2^0$, são valores iniciais.

Repita

Montar a matriz Jacobiana.

Igualar a Jacobiana à matriz dos elementos a , $a1$, b , $b1$. Isso resultará em um sistema linear.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = - \begin{bmatrix} a_{15} \\ a_{25} \\ a_{35} \\ a_{45} \end{bmatrix}$$

Resolver o sistema (foi utilizado o método da eliminação de Gauss).

O vetor solução do sistema contém os resíduos que faltam para o calibração completo dos parâmetro, os quais chamaremos de $\Delta p1$, $\Delta p2$, $\Delta k1$, $\Delta k2$.

$$\text{Fazemos } p1 = p1^0 + \Delta p1;$$

$$p2 = p2^0 + \Delta p2;$$

$$k2 = k2^0 + \Delta k2;$$

$$k1 = k2^0 + \Delta k1;$$

Até $|\Delta p|$ e $|\Delta k|$ serem menores de ε .

Fim do Algoritmo.

4.2 Procedimento para o calibração.

Para efetuar o calibração de forma iterativa vamos retomar o exemplo dado no capítulo anterior. Considere então a tabela abaixo.

	<i>Idade</i>	<i>Vendas</i>	<i>Pertinência</i>
1	16	14	0.395
3	18	15	0.373
6	19	16	0.359
7	22	21	0.228
4	19	12	0.113
12	24	25	0.026
8	21	12	0.011
2	17	8	-0.355
11	27	13	-0.358
10	28	20	-0.369
5	20	7	-0.538
13	32	28	-0.622
9	23	6	-0.697

Tabela 4-1 Disposição inicial dos dados.

A coluna *Pertinência* corresponde à composição dos conceitos *jovem e bom vendedor*, resultando no conceito *bom gerente*. Os parâmetros p_i , k_i , p_v e k_v são todos iguais a 1.

Caso haja discordância em relação ao resultado, por exemplo o elemento 6 que está em terceiro lugar parecer ser melhor que o elemento 1 pois, é um pouco mais velho, maior de dezoito anos e vendeu mais; pode-se tentar obter um resultado melhor procedendo-se da seguinte forma. Aumentamos a pertinência do candidato número 6, por exemplo para 0.8. Depois, o programa é executado com essa pertinência alterada, como mostra a tabela abaixo:

	<i>Idade</i>	<i>Vendas</i>	<i>Pertinência</i>
1	16	14	0.395
3	18	15	0.373
6	19	16	0.888
7	22	23	0.228
4	19	12	0.113
12	24	25	0.026
8	21	12	0.011
2	17	8	-0.355
11	27	13	-0.358
10	28	20	-0.369
5	20	7	-0.538
13	32	28	-0.622
9	23	6	-0.697

Tabela 4-2. Dados com pertinência alterada.

Se ainda houver insatisfação quanto ao posicionamento dos elementos da tabela acima, pode-se proceder uma calibração onde o satisfatório seja a obtenção do elemento 6 em primeiro. Para isso são necessárias 14 iterações do algoritmo, as quais são mostradas em parciais de 5, 5 e 4 nas próximas 3 tabelas, juntamente com as respectivas variações nos parâmetros.

	<i>Idade</i>	<i>Vendas</i>	<i>Pertinência</i>
1	16	14	0.365
5	18	15	0.353
6	19	16	0.347
7	22	23	0.241
4	19	12	0.090
12	24	25	0.044
8	21	12	-0.003
11	27	13	-0.352
10	28	20	-0.355
2	17	8	-0.576
5	20	7	-0.550
13	32	28	-0.610
9	23	6	-0.702

Tabela 4-3 Disposição da tabela após 5 iterações.

Os parâmetros iniciais eram $PI = 1$, $PV = 1$, $KI = 1$, $KV = 1$ (tabela 6-1). Ao fim dessas primeiras 5 iterações seus valores são $PI = 0.953$, $PV = 1.08$, $KI = 1.006$, $KV = 0.971$.

É interessante notar que houve uma mudança em algumas posições de alguns elementos, e que a diferença do elemento 6 em relação ao elemento 1 diminuiu. As próximas 5 iterações são vistas na tabela abaixo.

	<i>Idade</i>	<i>Vendas</i>	<i>Pertinência</i>
1	16	14	0.310
6	19	16	0.3061
3	18	15	0.3064
7	22	23	0.228
4	19	12	0.041
12	24	25	0.040
8	21	12	-0.043
10	28	20	-0.350
11	27	13	-0.359
2	17	8	-0.409
5	20	7	-0.571
13	32	28	-0.600
9	23	6	-0.712

Tabela 4-4 Disposição da tabela após 10 iterações.

Ao fim dessas 10 iterações os valores dos parâmetros são: $PI = 0.941$, $PV = 1.214$, $KI = 0.989$, $KV = 0.934$. Novamente, houve uma mudança em algumas posições. Nosso elemento alvo (6) pulou para o segundo lugar, diminuindo mais ainda a diferença em relação a (1). As

últimas iterações são vistas a seguir:

	<i>idade</i>	<i>Vendas</i>	<i>Pertinência</i>
6	19	16	0.236
3	18	15	0.225
1	16	14	0.216
7	22	23	0.209
12	24	25	0.041
4	19	12	-0.038
8	21	12	-0.108
10	28	20	-0.334
11	27	13	-0.367
2	17	8	-0.464
13	32	28	-0.572
5	20	7	-0.607
9	23	6	-0.731

Tabela 4-5. Configuração da tabela após as 14 iterações

Ao fim dessa 14 iterações os valores dos parâmetros são: PI = 8.898, PV = 1.505, KI = 0.955, KV = 0.881.

Houve uma drástica mudança na configuração da tabela. O elemento 6 assumiu a primeira posição e o que estava em primeiro agora está em terceiro. Os parâmetros também sofreram uma substancial alteração.

Apesar do programa ter apresentado uma resposta dentro do esperado, ninguém é obrigado a achar que essa solução é a melhor. Pode, por exemplo, haver uma discordância em relação à posição do elemento 7, uma vez que ele é apenas 3 anos mais velho que o atual primeiro colocado e possui um índice de vendas de aproximadamente 45% a mais. Aumentado a pertinência do elemento 7 para 0.888 e ajustando os parâmetros tem-se após 5 iterações a tabela abaixo.

	<i>idade</i>	<i>Vendas</i>	<i>Pertinência</i>
6	19	16	0.189
3	18	15	0.177
1	16	14	0.167
7	22	23	0.166
12	24	25	-0.002
4	19	12	-0.082
8	21	12	-0.151
10	28	20	-0.376
11	27	13	-0.407
2	17	8	-0.487
13	32	28	-0.606
5	20	7	-0.622
9	23	6	-0.740

Tabela 4-6 Configuração da tabela após 5 iterações.

Ainda não houve trocas nas posições dos elementos, no entanto já pode ser observada uma variação nos valores das pertinências dos mesmos em relação à tabela anterior. Os valores dos parâmetros que eram $PI = 8.898$, $PV = 1.505$, $KI = 0.955$, $KV = 0.881$, agora são $PI = 0.975$, $PV = 1.675$, $KI = 0.971$, $KV = 0.851$. A próxima tabela apresenta mais cinco iterações do algoritmo.

	<i>Idade</i>	<i>Vendas</i>	<i>Pertinência</i>
6	19	16	0.172
7	22	23	0.165
3	18	15	0.155
1	16	14	0.138
12	24	25	-0.004
4	19	12	-0.104
8	21	12	-0.170
10	23	20	-0.389
11	27	13	-0.421
2	17	8	-0.502
5	20	7	-0.631
9	23	6	-0.745
13	32	28	-0.622

Tabela 4-7. *Disposição da tabela após 5 iterações*

Nessa tabela já se nota mudanças nas posições dos elementos, o próprio elemento 7 já ocupa a segunda posição, sua diferença em relação ao primeiro colocado é bem pequena. Os valores dos parâmetros agora são: $PI = 0.968$, $PV = 1.826$, $KI = 1.011$, $KV = 0.830$. As 8 últimas iterações são vista a seguir:

	<i>Idade</i>	<i>Vendas</i>	<i>Pertinência</i>
7	22	23	0.155
6	19	16	0.139
3	18	15	0.117
1	16	14	0.093
12	24	25	-0.014
4	19	12	-0.140
8	21	12	-0.200
10	28	20	-0.409
11	27	13	-0.443
2	17	8	-0.525
13	32	28	-0.6459
5	20	7	-0.6456
9	23	6	-0.753

Tabela 4-8. *Tabela final.*

Com essa última tabela chega ao fim o calibração dos parâmetros. Seus valores são $PI = 0.968$, $PV = 2.08$, $KI = 1.063$, $KV = 0.802$. O interessante é que como o critério para dizer

quem é o melhor elemento é subjetivo, o calibração poderia ter se encerrado na tabela 4.5. Porém, nada impede que essa última configuração seja tida como a melhor. O importante nisso tudo é que a subjetividade foi modelada com sucesso.

Pode ser observado na tabela acima uma mudança substancial nas posições de todos os elementos.

Na tabela tem-se o arranjo dos candidatos que parece mais adequado. Com isso, o conceito *bom gerente* é expresso como *não tão jovem e bom vendedor*.

Com esse simples exemplo é mais uma vez reforçado não só o expressivo poder de modelagem das expressões utilizadas nesse trabalho, como também a grande capacidade de recuperação de informações dos operadores usados.

O Algoritmo de Quadrados Mínimos Não-Lineares apresentado anteriormente foi implementado em linguagem C, e a partir daí é que foram feitos os ajustes dos parâmetros e geradas as tabelas acima.

Esse capítulo preocupou-se exclusivamente com a modelagem da subjetividade do ponto de vista computacional. O algoritmo descrito, deduzido com a ajuda do software MAPLEV, pode ser utilizado para o calibração dos parâmetros de qualquer composição conjuntiva de dois conceitos. Apesar de apresentar razoável sucesso, esse algoritmo tem o inconveniente de ter que ser refeito a cada inclusão de novos conceitos na composição, acarretando não só mais trabalho, como também sua implementação pode vir a ter problemas de memória e tempo de execução sérios. Só para se ter uma idéia, as iterações realizadas anteriormente trabalham com uma matriz da ordem de 4 x 5, ou seja 20 elementos. Caso seja necessário uma composição de 3 conceitos a ordem da matriz sobe para 6 x 7, ou seja 42 elementos; e 4 conceitos seriam 72 elementos; isso sem contar a trabalhosa implementação das 72 equações que vão gerar as componentes da matriz..

O algoritmo também não possui uma convergência absoluta. Isto é, querer melhorar a posição de um elemento totalmente avesso a tendência da composição, por exemplo um elemento com pertinência -0.888, não funciona. Os testes mostraram uma divergência para os valores dos parâmetros que geram resultados absurdos de composição. Porém, fora isso o algoritmo funciona bem.

A limitação devido á complexidade do algoritmo anterior, particularmente o cálculo da matriz jacobiana (em forma simbólica) e a codificação das expressões em C, levaram ao desenvolvimento e teste de um algoritmo mais prático, em um ambiente mais amigável, sem maiores preocupações com eficiência ou precisão analítica.

O modelo permite a definição de conceitos caracterizados por componentes definidas sobre os atributos de base de dados.

No caso do exemplo em estudo, sobre os atributos *idade* e *vendas* foram definidos os conceitos primários *velho* e *bom-vendedor*, e o conceito *gerente* como:

$$\langle \textit{Gerente} \rangle = \langle \textit{não velho} \rangle \mathbf{E} \langle \textit{bom vendedor} \rangle \mathbf{E} \langle \textit{não muito não velho} \rangle$$

O estabelecimento inicial de parâmetros para esse algoritmo, que seguem os valores mostrados na tabela 4-9 abaixo, produziu os graus de pertinência calculados da tabela 4-10, também abaixo, ordenados pelos graus de satisfação ao conceito “Gerente”.

(os parâmetros Q e W que aparecerão daqui para frente equivalem aos respectivos P e K anteriores).

COMPONENTE	ATRIBUTO	Q	W
não muito não velho	idade	1	1
não velho	idade	1	-1
bom vendedor	vendas	1	1

Tabela 4-9 Parâmetros iniciais para o algoritmo.

Exemplares	Idade	Vendas	Gerente
7	22	23	-0.380
12	24	25	-0.392
8	21	12	-0.552
6	19	6	-0.593
11	27	13	-0.628
10	28	20	-0.630
4	19	12	-0.643
3	18	7	-0.673
5	20	7	-0.788
13	32	28	-0.804
1	16	14	-0.816
2	17	8	-0.822
9	23	6	-0.833

Tabela 4-10. Graus de pertinência calculados a partir da tabela 4-9.

O usuário tem todo o direito de não concordar com a classificação obtida, nesse caso deverá informar a classificação (subjativa) desejada, pela especificação dos novos graus de pertinência que espera ser alcançado por um ou mais exemplares.

No teste, foi especificada subjativamente uma ordenação dos candidatos e, arbitrariamente, valores para os graus de pertinência desejados, constantes da tabela 4-11.

Exemplar	Pertinência Desejada	Idade	Vendas
7	0.95	22	23
12	0.85	24	25
6	0.8	19	16
4	0.7	19	12
8	0.5	21	12
10	0.3	28	20
11	0.15	27	13
3	-0.1	18	15
13	-0.3	32	28
5	-0.5	20	7
9	-0.65	23	6
2	-0.75	17	8
1	-0.95	16	14

Tabela 4-11 Ordenação subjetiva dos candidatos.

O desvio entre o reconhecimento proporcionado pelo modelo e o desejado é expresso pela soma dos quadrados das diferenças entre esses reconhecimentos.

O algoritmo de calibração altera os parâmetros do modelo procurando minimizar esse desvio, convergindo para uma situação onde os parâmetros resultaram nos valores da tabela 4-12 e na ordenação da tabela 4-13.

COMPONENTE	ATRIBUTO	Q	W
Não muito não velho	idade	0.234	8.30
não velho	idade	5.73	-1.54
bom vendedor	vendas	0.315	1.59

Tabela 4-12. Parâmetros após a minimização.

Exemplar	Pertinência obtida	Idade	Vendas
7	0.88	22	23
12	0.77	24	25
6	0.76	19	16
8	0.58	21	12
4	0.54	19	12
11	0.29	27	13
10	0.27	28	20
3	0.1	18	15
13	-0.33	32	28
5	-0.44	20	7
9	-0.68	23	6
2	-0.91	17	8
1	-0.99	16	14

Tabela 4-13. Resultados obtidos com o primeiro calibração de parâmetros.

Observando que na ordenação obtida os exemplares 4 e 8 bem como 10 e 11 estão invertidos, novo ciclo de calibração foi realizado alterando-se os valores de reconhecimento desejados para esses exemplares para as constantes da tabela 4 -14:

<i>Exemplar</i>	<i>Pertinência</i>
4	0.9
8	0.3
10	0.5
11	0

Tabela 4-14. *Especificação de pertinências desejadas.*

O novo calibração convergiu para os valores de parâmetros da tabela 4-15, produzindo a classificação da tabela 4-16.

COMPONENTE	ATRIBUTO	Q	W
Não muito não velho	idade	0.234	21.9
não velho	idade	6.32	-1.53
bom vendedor	vendas	0.347	1.43

Tabela 4-15. *Novos parâmetros após a convergência.*

<i>Exemplar</i>	<i>Pertinência Desejada</i>	<i>Idade</i>	<i>Vendas</i>
7	0.87	22	23
6	0.81	19	16
12	0.78	24	25
4	0.52	19	12
8	0.50	21	12
10	0.32	28	20
11	0.29	27	13
3	-0.05	18	15
13	-0.26	32	28
5	-0.46	20	7
9	-0.67	23	6
2	-0.99	17	8
1	-1	16	14

Tabela 4-16. *Nova classificação após novo ajustamento.*

Nessa nova classificação observa-se que as distorções anteriores foram corrigidas mas houve a inversão de posição dos exemplares 12 e 6. Para corrigir a situação, os

reconhecimentos desejados para esses elementos foram especificados como na tabela 4-17 e, novamente, ajustados os parâmetros.

<i>Exemplar</i>	<i>Pertinência</i>
12	0.9
6	0.75

Tabela 4-17. Nova especificação de pertinências desejadas.

O processo convergiu com valores para os parâmetros da tabela 4-18, produzindo a classificação da tabela 4-19.

COMPONENTE	ATRIBUTO	O	W
não muito não velho	idade	0.234	13.03
não velho	idade	6.322	-1.61
bom vendedor	vendas	0.347	1.43

Tabela 4-18. Novos valores de parâmetros.

Exemplar	Idade	Vendas	Gerente Calculado	Gerente Desejado
7	22	23	0.88	0.95
12	24	25	0.80	0.85
6	19	16	0.79	0.8
4	19	12	0.51	0.7
8	27	13	0.50	0.5
10	28	20	0.34	0.3
11	19	12	0.30	0.15
3	18	15	-0.50	-0.1
13	20	7	-0.27	-0.3
5	32	28	-0.46	-0.5
9	16	14	-0.67	-0.65
2	17	8	-0.98	-0.75
1	23	6	-0.99	-0.95

Tabela 4-19. Classificação após as alterações da tabela 4-17.

Apesar de não se ter obtido exatamente os valores de pertinência desejados, o modelo proporcionou a mesma ordenação pretendida, sendo portanto, eficaz no discernimento de quem é melhor que quem.

Introduzindo mais três exemplares de vendedores, com atributos da tabela 4-20, o reconhecimento ao conceito, com os mesmos valores dos parâmetros, é mostrado na tabela 4-21, em uma ordenação que foi considerada coerente com a preferência subjetiva explicitada.

Exemplar	Idade	Vendas
14	20	20
15	21	15
16	24	30

Tabela 4-20

Exemplar	Idade	Vendas	<i>Gerente</i>	
			Calculado	Desejado
14	20	20	0.90	-
7	22	23	0.88	0.95
16	24	30	0.81	-
12	24	25	0.80	0.85
15	21	15	0.74	-
6	19	16	0.79	0.8
4	19	12	0.51	0.7
8	27	13	0.50	0.5
10	28	20	0.34	0.3
11	19	12	0.30	0.15
3	18	15	-0.50	-0.1
13	20	7	-0.27	-0.3
5	32	28	-0.46	-0.5
9	16	14	-0.67	-0.65
2	17	8	-0.98	-0.75
1	23	6	-0.99	-0.95

Tabela 4-21. *Ordenação após a inclusão dos novos exemplares.*

5. Conclusão

Com os dois últimos capítulos ficou claro o expressivo poder de modelagem da subjetividade das funções de [MAI91]. Com a implementação computacional do calibração, mostrou-se que é possível extrair na prática, informações (dessa subjetividade modelada) de forma precisa e subjetiva, a partir de conceitos modelados com curvas suaves. Isso contradiz, com a enorme riqueza de um contra exemplo a frase do Dr. Henri Prade contida em [PRT84] : “ (...) *Trapezoidal membership functions are generally sufficient for practical applications: indeed, slight modifications of the shape of membership functions (which cannot be always identified with a great precision) do not affect the evaluation of queries sensibly (...)*”.

Esse trabalho tem uma enorme aplicação. Uma das maiores carências dos sistemas de informações atualmente é a falta da modelagem, representação e recuperação de informações imprecisas, o que não é nem um pouco cômodo e viável, uma vez que nós, seres humanos, trabalhamos muito bem esse tipo de conhecimento; e pelo fato disso fazer parte de nós, de nosso dia a dia, é que não só os Sistemas Especialistas, Sistemas de Apoio à Decisão, mas também os Bancos de Dados; tão presentes nas indústrias, escolas, universidades, bancos, hospitais, bibliotecas... etc, necessitam de tratar dados dessa natureza. Do contrário, como serão respondidas perguntas do tipo “*Quais são os homens muito mais altos do que Pedro?, Quais os amigos íntimos de Pedro que não são baixos e são aproximadamente tão velhos quanto Laura? Quais são as cidades do Brasil onde é bem possível se encontrar um clima agradável?*” Tais questionamentos necessitam, e muito, receberem respostas, não só pelas pessoas, mas também por computadores. É óbvio que tudo isso deve ser feito segundo um determinado ponto de vista, fixado pelas funções de pertinência.

Apesar do trabalho ter sido concluído da forma prevista, muita coisa ainda existe para ser feita nessa direção. Essa dissertação não passou de uma simples amostra do que pode ser construído no futuro. Por exemplo, seria interessante uma comparação, em todos os aspectos; com os modelos consagrados de recuperação, representação e modelagem de informações imprecisas em base de dados relacionais.

Outro ponto que fica em aberto é o desempenho de tais funções em um modelo de base de dados orientado a objetos.

O presente trabalho justifica pesquisa adicional na implementação (aplicação prática) de um banco de dados nebuloso, todo baseada na linha [MAI91], e que tivesse por caso particular as situações e informações convencionais (não nebulosas) que são hoje o único objetivo dos bancos de dados; possivelmente enriquecido com consultas em linguagem natural. A final, ao longo de toda essa escritura, em todos os modelos vistos, ficou claro o quanto isso pode ter o uso facilitado pelo emprego da Teoria dos Conjuntos Nebulosos.

Como trabalho futuro de extrema importância seria interessante se pensar em algoritmos eficazes de calibração de conceitos, o que permitiria não só o desenvolvimento de aplicações eficientes em BD como também em qualquer área em que as funções e operadores de [MAI91] forem aplicadas.

A área que se ocupa da representação, modelagem e recuperação de informações nebulosas tem sido bastante investigada. Aqui está mais uma contribuição a ela, que apesar de não resolver complicados problemas em aberto, vai ficando por aqui, com a certeza de ser uma importante contribuição; podendo também auxiliar com bastante utilidade aqueles que, porventura possam vir a se preocupar com essas questões.

6. Referências Bibliográficas

- [AKB88] Akdag, H.; Bouchon, B. *Using Fuzzy Set Theory in the Analysis of Structures of Information*. Elsevier Science Publishing, Holland, v.28, p.263-271, 1988.
- [AMM89] Ammar, S. *Determining the 'Best' Decision in the Presence of Imprecise Information*. Elsevier Science Publishing, Holland, v.29, p.293-302, 1989.
- [ANR87] Anvari, J. M.; Rose, G.F. *Fuzzy Relational Data*. Analysis of Fuzzy Information. Vol 2 CRC Press, 1987.
- [BGH88] Bosc, P.; Galibourg, M.; Hamon, G.; *Fuzzy Querying with SQL: Extensions and Implementation Aspects*. in Fuzzy Sets and Systems, vol. 28, p. 333-349.1988.
- [BGH88] Bosc, P.; Galibourg, M.; Hamon, G. *Fuzzy Querying with SQL: Extensions and Implementation Aspects*. Elsevier Science Publishing, Holland, v.28, p. 333-349, 1978.
- [BKK85] Bloc, L.; Kozłowska, B.; Strazłkowski, T.; Kowalski, A.; *A Natural Language Information Retrieval System with Extensions Towards Fuzzy Reasoning*, Int. J. Man-Machine Studies, 1985.
- [BOP95] Bose, P.; Pivert, O. *Imprecise Data Management and Flexible Querying in Databases*. In: Yager, R.R.; Zadeh, L.A. Fuzzy Sets, Neural Networks, and Soft Computing. New York: Van Nostrand Reinhold, 1995. p. 368-395.
- [BOS95] Boss, B. *Handling Vague Design Constraints in a Design Environment*. In: Proc. Int Workshop on Concurrent/Simultaneous Engineering Frameworks and Applications, 1995. P. 321-332.
- [BPS89] Buckles, B.P.; Petry, F.E.; Sachar, H.S.; *A Domain Calculus for Fuzzy Relational Databases*. in Fuzzy Sets and Systems, Elsevier Science Publishing, Holland, vol.29, p.327-340.1989.
- [BUP82] Buckles, B.P.; Petry, F.E.; *A Fuzzy Representation of Data for Relational Databases*. In Fuzzy Sets and Systems, vol 7, p.213-226.1982.
- [BUP83] Buckles, B.P.; Petry, F. E; *Extension of the Fuzzy Databases with Fuzzy Arithmetic*. In proc. of IFAC Symposium, Fuzzy Information, Knowledge Representation and Decision Processes, 1983.
- [BUP83] Buckles, P.B.; Petry, F.E. *Information-Theoretical Characterization of Fuzzy Relational Databases*. IEEE Transactions on Systems, Man, and Cybernetics, v.13, n.1, p.74-77.1983.
- [BUP84] Buckles, B.P; Petry, F.E.; *Extending the Fuzzy Database with Fuzzy*

- Numbers*. in Information Sciences, vol.34, p.145-155.1984.
- [COD70] Codd, E.F. *A Relational Model of Data for Large Shared Data Banks*. in Commun. of ACM, vol.13, p.377-387.1970.
- [COD79] Cod, E. F.; *Extending the database relational model to capture more meaning*. Trans. Database Systems, v.4, p.397-434, 1979.
- [COD87] Codd, E.F. *More Commentary on Missing Information in Relational Databases*. in ACM SIGMOD Record, vol.16, 1987.
- [CUN93] Cunha, C. *Métodos Numéricos, para as engenharias e ciências aplicadas*. Campinas, Editora da Unicamp, 1993.
- [DAT91] Date, C.J. *Introdução a Sistemas de Bancos de Dados*. 4.ed. Rio de Janeiro: Campus, 1991.
- [DAV92] Danforth S.; Valduriez, P. *The Data Model of FAD, a Database Programming Language*. Elsevier Science Publishing, New York, v.60, p.51-75, 1992.
- [DAV92] Danforth S.; Valduriez, P. *Functional SQL (FSQL), a n SQL Upward-Compatible Database Programming Language*. Elsevier Science Publishing, New York, v.62, p.183-203, 1992.
- [DUP80] Dubois, D.; Prade, H. *Fuzzy Sets and Systems: Theory and Applications*. New York: Academic Press, 1980.
- [GBP93] George, R.; Buckles, B.P.; Petry, E. *Modelling Class Hierarchies in the Fuzzy Object-Oriented Data Model*. Elsevier Science Publishing, Holland, v. 60, p.259-272, 1993.
- [KAZ86] Kacprzyk,J.; Ziolkowski, A. *Retrieval from Data Bases Using Queries with Fuzzy Linguistic Quantifiers*, Polish Academy of Science, 1986.
- [KLY95] Klir, G.J.; Yuan, B. *Fuzzy Sets and Fuzzy Logic: Theory and Applications* New York: Prentice Hall, 1995.
- [KWN90] Kirimoto, T.; Wakayama, S.; Nomoto, K.; Ohashi, Y.; Kondo, M. *A Document Retrieval System Based on Citations Using Fuzzy Graphs*. Elsevier Science Publishing, Holland, v.38, p.207-222, 1990.
- [LEN91] Lenzerini, M. *Type Data Base with Incomplete Information*. Information Sciences, New York, v.53, p. 61-87, 1991.
- [LIU94] LIU,W.Y. *Constraints on Fuzzy Values and Fuzzy Functional Dependencies*. Information Sciences, New York, v.78, p. 303-309, 1984.
- [LMS90] Sheno, S. Melton, A; *Equivalence Class Model of Fuzzy Relational Databases*. Elsevier Science Publishing, Holland, v. 38, p.153-170, 1990.
- [MAI91] Maia, L.F.J. *Caracterização e Reconhecimento de Conceitos*. Campinas, 1991. Tese, (Doutorado em Engenharia Elétrica) FEE - DCA, Unicamp.

- [MCM90] Mántaras, L.R.; Cortés, U.; Manero, J. *Knowledge Engineering for a Document Retrieval System*. Fuzzy Sets and Systems, Elsevier Science Publishing, Holland, v.38, p. 223-240, 1990.
- [MIY90] Miyamoto, S. *Information Retrieval Based on Fuzzy Associations*. Fuzzy Sets and Systems, Elsevier Science Publishing, Holland, v. 38, p.191-205, 1990
- [MPV94] Medina, J.M.; Pons, O; Vila, M.A; *GEFRED: A Generalized Model of Fuzzy Relational Databases*. Information Sciences, New York, v.76, p.87-109, 1994.
- [NIL80] Nilsson, J. N.; *Principles of Artificial Intelligence*. Tioga Publishing Company, 1980.
- [PFT88] Press, W. H., Flannery, B. P. Teukolsky, S. A., Vetterling, W. T. *Numerical Recipes: The Art of Scientific Computing*. New York, Cambridge Univ. Press, 1998.
- [PPB90] Buckles, P.B.; Petry, F.E.; Pillai, J. *Network Data Model for Representation of Uncertainty*. Fuzzy Sets and Systems, Elsevier Science Publishing, Holland, v.38, p.171-190, 1990.
- [PRD90] Prade, H.; Dubois, D. *Measuring Properties of Fuzzy Sets: A General Technique and its Use in Fuzzy Query Evaluation*. Fuzzy Sets and Systems, Elsevier Science Publishing, Holland, v.38, p.137-152, 1990.
- [PRT84] Prade, H.; Testemale, C. *Generalizing Database Relational Algebra for the Treatment of Incomplete or Uncertain Information and Vague Queries*. Information Sciences, New York, v. 34, p. 115-143, 1984.
- [SAS92] Saade, J.J; Schwarzlander, H. *Ordering Fuzzy Sets Over the Real Line: Approach Based on Decision Making Under Uncertainty*. Fuzzy Sets and Systems, Elsevier Science Publishing, Holland, v.50, p.237-246, 1992.
- [SCK94] Schneider, M.; Kandel, A; *On Uncertainty Management in Fuzzy Inference Procedures*. Information Sciences, New York, v.79, p.181-190, 1994.
- [SHM89] Sheno, S.; Melton, A. *Proximity Relations in the Fuzzy Relational Database Model*. Fuzzy Sets and Systems, Elsevier Science, Holland, v.31, p.285-296, 1989.
- [SHM90] Sheno, S.; Melton, A. *An Extended Version of the Fuzzy Relational Database Model*. Fuzzy Sets and Systems, Elsevier Science Publishing, New York, v.52, p. 35-52, 1990.
- [SHM92] Sheno, S.; Melton, A. *Functional Dependencies and Normal Forms in the Fuzzy Relational Database Model*. Fuzzy Sets and Systems, Elsevier Science Publishing, New York, v.60, p. 1-28, 1992.
- [SME95] Smelcer, J.B. *User Errors in Database Query Composition*. International

Journal Human-Computer Studies, v.42, p.353-381, 1995.

- [TRS90] Tripathy, R.C.; Saxena, P.C. *Multivalued Dependencies in Fuzzy Relational Databases*. Fuzzy Sets and Systems, Elsevier Science Publishing, Holland, v.38, p.267-279, 1990.
- [UMA83] Umamo, M. *Retrieval from Fuzzy Databases by Fuzzy Relational Algebra*. In: Proc. IFAC Conf. On Fuzzy Information, Knowledge Representation and Decision Processes, p. 175-182, 1983.
- [UMF82] Umamo, M. *Freedom-O: A Fuzzy Database System*. In: Gupta, M.; Sanchez, E. Fuzzy Information and Decision Processes, North-Holland, 1982.
- [USZ86] Uszynski, M. *Fuzzy Queries with Linguistic Quantifiers for Information Retrieval from Data Bases*. In: Report No.UCB/CSD/87/333, Computer Science Division (EECS) University of California, Berkeley, 1986.
- [ZAA78] Zadeh, L. A ; *A meaning representation language for natural languages*. Internat. J. Man-Machine Studies, v.10, p.395-460, 1978.
- [ZAD71] Zadeh, L.A; *Similarity Relations and Fuzzy Ordering*. Information Sciences, v.3, p.177-200, 1971.
- [ZAD75] Zadeh, L. A ; *The concept of a linguistic variable and its application to approximate reasoning*. Part1. Inform. Sci. V.8, p.199-249, Part2. V.8, p. 301-357; Part3. V.9, p. 43-80, 1975.
- [ZAD78] Zadeh, L.A. *Fuzzy Sets as a Basis for Theory of Possibility*, Fuzzy Sets and Systems, p. 3-28, 1978.
- [ZAD81] Zadeh, L. A ; *Test-score semantics for natural languages and meaning representation via PRUF*. Empirical Semantics, v.1, p.281-349, 1981.
- [ZEK85] Zemankova, M.; Kandel, A; *Implementing Imprecision in Information Systems*. Information Sciences, New York, v.37, p. 107-141, 1985.
- [ZIM82] Zimmerman, A. *Some Experiments Concerning the Fuzzy Meaning of Logical Quantifiers*. General Surveys of Systems Methodology, Editado por L. Troncoli, p. 435-441, 1982.
- [ZIM91] Zimmermann, H.J. *Fuzzy Set Theory and its Application*. 2ed. Boston: Kluwer Academic Publishers, 1991.