

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA
COMPUTAÇÃO**

Gabriel Barreto Moura

REDES PROBABILÍSTICAS FUZZY NAÏVE BAYES

Florianópolis

2016

Gabriel Barreto Moura

REDES PROBABILÍSTICAS FUZZY NAÏVE BAYES

Dissertação submetida ao Programa
de Pós-Graduação em Ciência da Com-
putação para a obtenção do Grau de
Mestre.

Orientador: Prof. Dr. Mauro Roisen-
berg

Florianópolis

2016

AGRADECIMENTOS

Agradeço primeiramente a meus pais, pelo apoio incondicional não só nesta pós-graduação, mas durante toda a vida. Aos meus amigos e familiares que sempre ofereceram motivação e pediram calma e paciência quando eu sentia que não conseguiria concluir este trabalho. Em especial agradeço aos amigos Altieres del Sent e Douglas Schroeder pelo companheirismo durante o mestrado. Ao professor Mauro Roisenberg pela orientação oferecida durante a realização deste trabalho, de maneira a executá-lo da melhor forma possível. E, principalmente, pela paciência durante os períodos mais difíceis. A UFSC, por ter sido a minha segunda casa durante esses anos de pós-graduação, oferecendo bons professores e a estrutura necessária para o meu aprendizado, além das inúmeras novas amizades proporcionadas pelo ambiente acadêmico. E, por fim, a Fundação Capes pela bolsa recebida durante os períodos de estudo que me permitiram realizar este trabalho com dedicação exclusiva.

RESUMO

Redes bayesianas são modelos gráficos probabilísticos capazes de modelar a incerteza estocástica e são vastamente aplicados em vários problemas de classificação. Especificamente, a estrutura *naïve bayes* são largamente utilizadas devido aos bons resultados atingidos, apesar da sua estrutura simples e “*ingênua*”. Sistemas *fuzzy*, por outro lado, são modelos bem conhecidos para lidar com a imprecisão linguística por representar o conhecimento através de regras e funções de pertinências simples e interpretáveis. Porém, sistemas *fuzzy* tradicionais não são capazes de modelar a incerteza estocástica e, por isso, sistemas *fuzzy* probabilísticos foram desenvolvidos de forma a levar em consideração ambas as incertezas. Neste trabalho é proposta a rede probabilística *fuzzy naïve bayes*, uma combinação de sistemas *fuzzy* probabilísticos e redes bayesianas *naïve*, também capaz de modelar ambas as incertezas simultaneamente. O modelo proposto é primeiramente aplicado a um problema simples e didático de classificação para demonstrar o seu potencial e vantagem em relação a classificadores *naïve bayes*, enquanto mantém-se a interpretabilidade do modelo tradicional. Para testar o modelo, experimentos foram realizados em conjuntos de dados de *benchmark* e seus resultados são comparados a modelos de outros autores que utilizam os mesmos conjuntos de dados. A rede proposta apresenta resultados levemente inferiores a outros modelos. Porém, é observada uma vantagem qualitativa da rede proposta devido a seu alto nível de interpretabilidade.

Palavras-chave: Sistemas Fuzzy Probabilísticos; Redes Bayesianas; Modelagem de Incerteza; Aprendizado de Máquina

ABSTRACT

Bayesian networks are probabilistic graphical models capable of modeling statistical uncertainty and are widely applied in many classification problems. Specifically, Naive Bayesian networks are largely used due to their simple, naive structure, while still producing precise results. Fuzzy systems, on the other hand, are a well known technique capable of dealing with linguistic vagueness by representing knowledge with simple and interpretable rules and membership functions. As traditional fuzzy systems are unable to model statistical uncertainty, Probabilistic Fuzzy Systems were developed in order to account for both kinds of uncertainties. In this work we propose the Probabilistic Fuzzy Naive Bayes classifier as a combination of both probabilistic fuzzy systems and naive bayesian networks, also capable of simultaneously modeling both kinds of uncertainties. The proposed model is firstly applied in a very simple classification problem in order to show its potential and advantage over traditional naive bayes classifiers, while maintaining their interpretability. For validation, experiments were done using benchmark classification data sets from the UCI machine learning repository and the results are then compared with other similar alternate methods. **Keywords:** Probabilistic Fuzzy Systems; Bayesian Networks; Uncertainty Modeling; Machine Learning

LISTA DE FIGURAS

Figura 1	FPs trapezoidais modelando conjuntos <i>fuzzy</i> para a variável <i>comida</i>	21
Figura 2	Partes principais de um sistema <i>fuzzy</i> , adaptado de Rezaee et al. (2009)	22
Figura 3	Fuzzificação e inferência de um sistema do tipo Mamdani, adaptado de Rezaee et al. (2009)	23
Figura 4	Exemplos de <i>clusters crisp</i> e <i>clusters fuzzy</i>	26
Figura 5	Tipos de incerteza e seus métodos, adaptado de Katarzyna e Walaszek-Babiszewska (2011)	28
Figura 6	Conjuntos <i>fuzzy</i> tipo-2 e probabilístico, retirados de Mendel e John (2002), Zhang e Li (2012)	28
Figura 7	Exemplo de relação entrada-saída de um sistema <i>fuzzy</i> probabilístico, adaptado de Zhang e Li (2012)	29
Figura 8	Exemplo de estimação de uma função de densidade, através do produto $f(x)\mu(x)$	31
Figura 9	Espaço amostral <i>fuzzy</i> bem definido	31
Figura 10	Exemplo de histograma com 10 <i>bins</i> de tamanho 50	33
Figura 11	Exemplo de função de pertinência equivalente a um <i>bin crisp</i> , ambos com área 3	34
Figura 12	Comparação entre histogramas tradicionais e funções de densidade <i>fuzzy</i> de dados normalmente distribuídos	35
Figura 13	Exemplo de rede bayesiana	35
Figura 14	Exemplo de tabelas de probabilidade incondicional $P(A)$ e $P(B)$ e condicional $P(C A, B)$. A variável B foi discretizada em duas classes.	36
Figura 15	Tipos de inferências bayesianas. (KORB; NICHOLSON, 2011)	37
Figura 16	Exemplo de rede bayesiana do tipo <i>naïve bayes</i>	38
Figura 17	Estrutura geral do modelo proposto	42
Figura 18	Comparação entre a <i>naïve bayes</i> tradicional e a rede PFNB proposta	46
Figura 19	Conjunto de dados <i>Simple Class</i>	47
Figura 20	Funções de pertinência de x e y obtidas no treinamento da rede.	48

Figura 21 Histogramas tradicionais e funções de densidade estimadas do eixo x e y através de histogramas <i>fuzzy</i> com 12 <i>bins fuzzy</i> .	49
Figura 22 Funções de densidade <i>fuzzy</i> calculadas com a equação 3.1 para dois <i>clusters</i> do eixo x	49
Figura 23 $P(x C)$ estimadas através da multiplicação das $\hat{f}(x \in C)$ pelas $\mu(x)$	50
Figura 24 Rede <i>naïve bayes</i> aprendida no software <i>NETICA</i> [®] para o <i>Simple Class</i>	51
Figura 25 Representação <i>naïve bayes</i> da rede PFNB treinada para o <i>Simple Class</i>	51
Figura 26 Funções de pertinência extraídas com o FCM do <i>Chemical</i>	57
Figura 27 Histogramas das variáveis do conjunto <i>Chemical</i>	58
Figura 28 Histogramas das variáveis do conjunto <i>Iris</i>	59
Figura 29 Funções de pertinência extraídas com o FCM do <i>Iris</i>	60
Figura 30 Histogramas das variáveis do conjunto <i>Glass</i>	61
Figura 31 Funções de pertinência extraídas com o FCM do <i>Glass</i>	62
Figura 32 Histogramas das variáveis do conjunto <i>Breast Cancer</i>	63
Figura 33 Funções de pertinência extraídas com o FCM do <i>Breast Cancer</i>	64
Figura 34 Funções de pertinência extraídas com o FCM do <i>Wine</i>	65
Figura 35 Histogramas das variáveis do conjunto <i>Wine</i>	66
Figura 36 Relação entre acurácia e os parâmetros de treinamento da rede para cada conjunto de dados	68
Figura 37 Comparação entre as funções de densidade aproximadas por histogramas <i>fuzzy</i> com $b = 4$ e $b = 20$	69

LISTA DE TABELAS

Tabela 1	Matriz de pesos da camada de saída	50
Tabela 2	Saída da rede PFNB para os pontos da Figura 19.....	52
Tabela 3	Saída de uma rede <i>naïve bayes</i> para os pontos da Figura 19.....	52
Tabela 4	Visão geral dos conjuntos de dados aplicados na validação.....	55
Tabela 5	Estudo comparativo do <i>Chemical</i>	56
Tabela 6	Valores de saída reais e probabilidades estimadas pela rede PFNB.....	57
Tabela 7	Estudo comparativo do <i>Iris</i>	60
Tabela 8	Estudo comparativo do <i>Glass</i>	62
Tabela 9	Estudo comparativo do <i>Breast Cancer</i>	64
Tabela 10	Estudo comparativo do <i>Wine</i>	65
Tabela 11	Média da acurácia em por cento da rede PFNB para cada conjunto de dados e variações dos valores b e C	67

SUMÁRIO

1 INTRODUÇÃO	15
1.1 CONTEXTUALIZAÇÃO	15
1.2 JUSTIFICATIVA	18
1.3 OBJETIVOS	18
1.4 ORGANIZAÇÃO DA DISSERTAÇÃO	19
2 FUNDAMENTOS TEÓRICOS	21
2.1 SISTEMAS <i>FUZZY</i>	21
2.1.1 Introdução	21
2.1.2 Mecanismo de Inferência	22
2.1.3 Fuzzificador e Defuzzificador	23
2.1.4 Classificação com Sistemas <i>Fuzzy</i>	24
2.1.5 Clusterização <i>Fuzzy C-Means</i>	25
2.2 SISTEMAS <i>FUZZY</i> PROBABILÍSTICOS	27
2.2.1 Introdução	27
2.2.2 Probabilidade de Eventos <i>Fuzzy</i>	30
2.2.3 Espaço Amostral <i>Fuzzy</i> Bem Definido	30
2.2.4 Estatística de Eventos <i>Fuzzy</i>	32
2.2.5 Histogramas <i>Fuzzy</i>	33
2.3 REDES BAYESIANAS	34
2.3.1 Introdução	34
2.3.2 Inferência	37
2.3.3 Classificador Naïve Bayes	38
2.3.4 Redes <i>Fuzzy</i> Bayesianas	38
3 REDE PROBABILÍSTICA <i>FUZZY NAÏVE-BAYES</i> .	41
3.1 INTRODUÇÃO	41
3.2 FUNÇÕES DE PERTINÊNCIA	41
3.3 FUNÇÕES DE DENSIDADE	43
3.4 PROBABILIDADES A PRIORI E CONDICIONAIS	43
3.5 MECANISMO DE INFERÊNCIA BAYESIANA	44
3.6 CAMADA DE SAÍDA	44
3.7 TREINAMENTO	45
3.8 INTERPRETABILIDADE	46
3.9 CONJUNTO DE DADOS <i>SIMPLE CLASS</i>	47
3.9.1 Funções de pertinência	48
3.9.2 Funções de densidade	48
3.9.3 Probabilidades condicionais e a priori	48
3.9.4 Camada de saída	49

3.9.5 Discussão	50
4 EXPERIMENTOS E RESULTADOS	55
4.1 <i>CHEMICAL</i>	56
4.2 <i>IRIS</i>	59
4.3 <i>GLASS</i>	61
4.4 <i>BREAST CANCER</i>	63
4.5 <i>WINE</i>	65
4.6 DISCUSSÃO	67
5 CONCLUSÃO	71
REFERÊNCIAS	73

1 INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO

Para modelar a maioria dos problemas do mundo real, é necessário lidar com algum tipo de incerteza. Não levando em consideração a ignorância, incerteza derivada da ausência de informações acerca do problema em questão, há dois tipos de incerteza mais presentes na maioria dos cenários: imprecisão linguística e a incerteza estocástica (KORB; NICHOLSON, 2011).

Imprecisão linguística se refere a incerteza inerente a linguagem natural. Conceitos como “morno” ou “rápido” podem ser definidos de maneira diferente por cada indivíduo. Vários trabalhos discutem que a ideia de pertinência (*fuzzy*) e de probabilidade não se referem ao mesmo fenômeno, mas sim, incertezas complementares (MEGHDADI; AKBARZADEH-T, 2001). Por exemplo, suponha a frase “A probabilidade de fazer frio na segunda-feira é de 80%”. “Frio” é um conceito *fuzzy*, pois apresenta imprecisão ou vagueza linguística, ou seja, não há definição exata e precisa da temperatura referente ao conceito de “frio”. Por outro lado, “80%” se refere a incerteza estocástica inerente ao clima.

Sistemas *fuzzy* são vastamente conhecidos e aplicados na modelagem de imprecisão, mantendo um alto nível de interpretabilidade por humanos. O seu conhecimento é armazenado na forma de regras e funções de pertinência, ambas mantendo um certo relacionamento com a linguagem natural. Isto torna os sistemas *fuzzy* facilmente interpretáveis, possibilitando auxiliar na compreensão sobre o problema a ser modelado após a utilização de algoritmos de aprendizagem. Desta forma, surge a possibilidade de extração de conhecimento e uma maior interpretabilidade do que em modelos de caixa preta, onde a modelagem está implícita. Uma desvantagem dos sistemas *fuzzy* é a sua incapacidade de modelar a incerteza estocástica. (WALTMAN, 2005; KAYMAK; BERG; BERGH, 2002).

A incerteza estocástica diz respeito a fenômenos não determinísticos, pois partem de eventos aleatórios. Esta é relacionada a uma variabilidade inerente ao processo físico ou contexto a ser modelado. A incerteza estocástica é a incerteza mais comumente estudada, e geralmente é representada matematicamente através de distribuições de probabilidade, que podem ser estimadas através do estudo de dados amostrais (BARRICO, 2007).

Ao utilizar sistemas *fuzzy* para modelar problemas onde existe

incerteza estocástica, este poderá apresentar menor acurácia em relação a outros sistemas, pois perde-se parte da informação disponível acerca do problema. Cria-se, então, a necessidade de aliar as duas informações para obter resultados melhores e mais descritivos. Usando como partida os conceitos de probabilidade e estatística *fuzzy* introduzidos por Zadeh (1968), vários autores desenvolveram sistemas que modelam os dois tipos de incerteza simultaneamente, chamados de sistemas *fuzzy* probabilísticos (KATARZYNA; WALASZEK-BABISZEWSKA, 2011; KAYMAK; BERG, 2013; ZHANG; LI, 2012).

Há exemplos que comprovam que sistemas *fuzzy* probabilísticos são eficientes em modelar os dois tipos de incerteza, linguística e estocástica, demonstrando também uma maior facilidade de interpretação do sistema e sua confiabilidade (LI; LIU, 2009; ZHANG; LI, 2012; KAYMAK; BERG, 2013). Waltman (2005) faz uma análise teórica dos sistemas *fuzzy* probabilísticos em problemas de classificação e regressão, além de mostrar uma equivalência destes a redes neurais de base radial para tarefas de classificação. Katarzyna e Walaszek-Babiszewska (2011) utilizam um sistema *fuzzy* probabilístico para predição da velocidade do vento, um processo geológico estocástico, e auxílio na tomada de decisão de sistemas gerenciadores de usinas de energia eólica. Kaymak e Berg (2013) aplicam este conceito na predição de mercados financeiros. Almeida et al. (2014) demonstra equivalências funcionais entre sistemas *fuzzy* aditivos e sistemas *fuzzy* probabilísticos.

Outro modelo vastamente aplicado à modelagem de incerteza estocástica, são as redes bayesianas (RB). Estes são modelos probabilísticos representados por grafos direcionados acíclicos, capazes de modelar as relações probabilísticas entre variáveis estocásticas. As redes bayesianas são estudadas desde a década de 50 e continuam sendo ferramentas eficientes na modelagem probabilística (PEARL, 1985). Nodos denotam variáveis aleatórias e suas arestas direcionadas representam o relacionamento entre estas variáveis. Nodos que influenciam outros nodos são chamados de pais e nodos influenciados são chamados filhos. Cada nodo possui uma distribuição de probabilidade, que descreve e quantifica a relação entre as variáveis.

Cada nodo possui um conjunto finito de possíveis estados. Por isto, para modelar RBs tradicionais, é necessário que seja feita a discretização de variáveis contínuas (LIMA, 2014). Desta forma, a distribuição de probabilidade associada a cada nodo toma a forma de tabelas de probabilidades condicionais. Esta tabela especifica a probabilidade de cada estado de um nodo, dada todas as possibilidades de estados dos seus nodos pais. Isto implica em um crescimento exponencial da quantidade

de registros nas tabelas de probabilidade condicional, dependendo da estrutura da rede (DAS, 2004).

RBs são modelos conhecidos por também apresentarem uma estrutura intuitiva em relação a linguagem humana. Além disso, seus parâmetros podem ser estimados através de algoritmos de aprendizado, incentivando a geração de conhecimento através da análise e interpretação das redes após o treinamento (KORB; NICHOLSON, 2011).

A inferência das redes bayesianas é um processo conhecido por inferência probabilística, onde a distribuição posterior de variáveis de consulta em relação a variáveis de evidência é computada. Vários algoritmos foram desenvolvidos e podem ser utilizados para este fim, de forma a adequá-los às diferentes estruturas possíveis das RBs. O método tradicional de realizar a inferência probabilística é baseada no Teorema de Bayes (PEARL, 1985).

Por conta disto, as redes bayesianas *naïve* são uma estrutura particular das redes bayesianas, onde há um nodo referente a classes do problema em questão, e um conjunto de nodos de atributos. O nodo de classes é sempre o pai de todos os nodos, e não há relação entre os próprios nodos de atributos. Esta suposição *naïve* (ingênua) do problema torna o modelo computacionalmente eficiente, reduzindo grandemente a quantidade de registros nas distribuições de probabilidade. Isto também permite que os parâmetros de probabilidade para cada atributo sejam estimados separadamente, uma vez que estes não exercem influência nos outros atributos. Como sugere o seu nome, a suposição de independência entre os nodos de atributos é uma suposição *ingênua* em relação a processos modeláveis do mundo real. No entanto, é possível obter níveis altos de acurácia (MCCALLUM; NIGAM, 1998; BUSTAMANTE; GARRIDO; SOTO, 2007).

Da mesma forma que os sistemas *fuzzy* probabilísticos foram desenvolvidos a partir da necessidade de inserir a incerteza estocástica na modelagem do sistema, há também abordagens originadas da necessidade de inserir a imprecisão nas redes bayesianas tradicionais. Estas são conhecidas por redes *fuzzy* bayesianas (RYHAJLO, 2013; BRIGNOLI et al., 2015; PAN, 1999; TIBIRICA, 2005). Ryhajlo (2013) utiliza redes *fuzzy* bayesianas para a prevenção de falhas em componentes elétricos e mecânicos. Bustamante, Garrido e Soto (2007) utilizam conjuntos *fuzzy* em redes *naïve* para dar inteligência a um jogo de futebol, possibilitando a modelagem de variáveis contínuas, de forma que não haja perda na discretização tradicional *crisp* (valores “puros”, ou seja, não *fuzzy*). Em geral, elas consistem na conversão dos nodos discretos de uma rede tradicional em nodos *fuzzy*.

1.2 JUSTIFICATIVA

Como mencionado anteriormente, sistemas *fuzzy* clássicos são capazes de modelar a imprecisão linguística e redes bayesianas modelam a incerteza estocástica. Ao combinar os dois modelos, obteve-se um sistema capaz de modelar ambos os tipos de incerteza. Este tipo de modelagem viu um ganho de popularidade, considerando que a maioria dos processos modeláveis apresentam pelo menos um dos tipos de incerteza, ou ambos (BARRICO, 2007).

Além disso, o sistema resultante é simples e permite maior interpretabilidade do processo modelado e da incerteza representada pela saída do mesmo. Através de um algoritmo de aprendizado supervisionado e levando em consideração o alto nível de interpretabilidade, herdado dos modelos tradicionais, será possível gerar conhecimentos acerca da relação entre as variáveis do problema após o treinamento.

1.3 OBJETIVOS

O objetivo geral deste trabalho é propor um modelo capaz de modelar a incerteza linguística e probabilística baseando-se em conceitos de sistemas *fuzzy* e redes bayesianas, utilizando os trabalhos de Kaymak e Berg (2013), principalmente no que diz respeito aos conceitos de probabilidade e estatística *fuzzy*. O modelo implementado é um modelo híbrido entre sistemas *fuzzy* probabilísticos e redes *naïve bayes*: a rede probabilística *fuzzy naïve-bayes* (PFNB). Este modelo é diferente do modelo proposto por Kaymak e Berg (2013), pois este não utiliza conceitos de redes bayesianas, e diferente das redes *fuzzy* bayesianas ao utilizar como base os sistemas *fuzzy* probabilísticos. Além disso, o modelo proposto foi implementado de forma que é possível interpretá-lo como um sistema *fuzzy* e uma rede *naïve bayes* simultaneamente. Para alcançar este objetivo geral, pretende-se atingir os seguintes objetivos específicos:

- Entender o funcionamento de sistemas *fuzzy* e redes bayesianas;
- Entender os conceitos relacionados a probabilidade e estatística *fuzzy*;
- Analisar as principais propostas de modelagem de sistemas *fuzzy* probabilísticos e suas aplicações;

- Testar o modelo proposto através de experimentos com conjuntos de dados;
- Comparar os resultados do modelo desenvolvido com os resultados de outros modelos semelhantes;

1.4 ORGANIZAÇÃO DA DISSERTAÇÃO

Esta dissertação está organizada da seguinte maneira:

- **Capítulo 2 - Fundamentos Teóricos:** Este capítulo apresenta a revisão bibliográfica dos principais conceitos utilizados: Sistemas *fuzzy*, sistemas *fuzzy* probabilísticos, redes bayesianas e redes bayesianas *fuzzy*.
- **Capítulo 3 - Rede Probabilística *Fuzzy Naïve-Bayes*:** Este capítulo detalha a rede probabilística *fuzzy naïve-bayes* proposta nesta dissertação, mostrando sua estrutura geral e seus módulos e demonstrando o seu funcionamento utilizando o conjunto de dados *Simple Class* disponível no software MATLAB[®].
- **Capítulo 4 - Experimentos e Resultados:** Este capítulo descreve e discute os experimentos realizados na validação do modelo proposto em quatro conjuntos de benchmark: *Iris*, *Glass*, *Breast Cancer* e *Wine*, retirados do repositório de aprendizado de máquina UCI.
- **Capítulo 5 - Conclusão:** Por fim, este capítulo formula uma breve conclusão sobre este trabalho e apresenta algumas propostas para trabalhos futuros

2 FUNDAMENTOS TEÓRICOS

2.1 SISTEMAS FUZZY

2.1.1 Introdução

Diferente da lógica clássica ou binária, onde uma afirmação pode ser apenas falsa ou verdadeira (representados por 0 e 1 respectivamente), na lógica *fuzzy* esse valor pode variar entre todos os reais contidos no intervalo $[0, 1]$. Zadeh (1965) foi o primeiro a introduzir o conceito de conjuntos *fuzzy*. Cada elemento pertence a um conjunto *fuzzy* de acordo com um grau de pertinência μ obtido através de uma função de pertinência $\mu(x)$ (FP). Estas funções podem adotar várias formas como triangular, gaussiana, trapezoidal, sigmoidal, entre outras (SAGGAF; NEBRIJA, 2003; REZAEI et al., 2009).

Os conjuntos *fuzzy* podem se sobrepor, gerando valores de verdade de acordo com o grau de pertinência μ de cada conjunto. Por exemplo, na Figura 1, em um dado valor x da variável *comida*, observando os valores $\mu(x)$, podemos dizer que ela pertence ao conjunto das comidas sem gosto com grau de pertinência 0,26 e ao mesmo tempo pertence ao conjunto das comidas razoáveis com grau de pertinência 0,74. O grau de pertinência para o conjunto *saborosa* é igual a zero, ou seja, a *comida* não está nada saborosa.

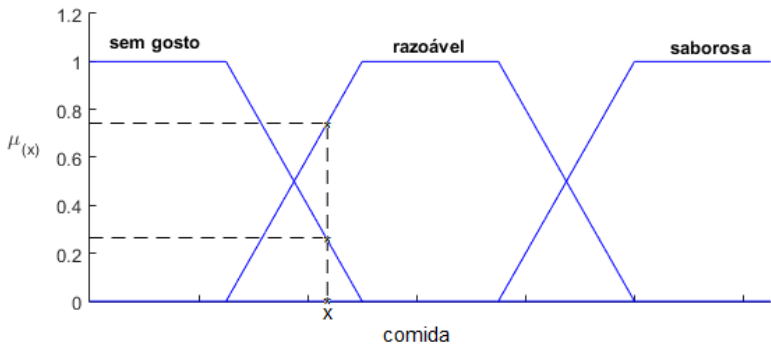


Figura 1 – FPs trapezoidais modelando conjuntos *fuzzy* para a variável *comida*.

Um sistema especialista *fuzzy* clássico consiste de três partes principais: Um fuzzificador, um mecanismo de inferência (acompanhado de um conjunto de regras *fuzzy*) e o defuzzificador, como mostra a Figura 2. Como estes componentes funcionam dependerá especificamente da implementação utilizada. Algumas implementações largamente utilizadas são, por exemplo, a *Mamdani* e a *Takagi-Sugeno*.

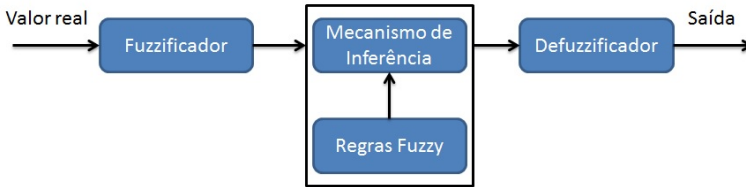


Figura 2 – Partes principais de um sistema *fuzzy*, adaptado de Rezaee et al. (2009)

Em situações práticas, a definição dos conjuntos *fuzzy*, suas FPs e de suas regras é feita, geralmente, coletando informações através de um especialista, utilizando algoritmos de aprendizagem (*machine learning*), ou uma combinação destes métodos (SAGGAF; NEBRIJA, 2003). Dessa forma, a lógica *fuzzy* é uma ferramenta poderosa para modelar conhecimentos incertos, relacionados à imprecisão da linguagem, e integrar informações obtidas através de dados e conhecimentos provenientes da experiência de um especialista (ZHANG; LI, 2012).

2.1.2 Mecanismo de Inferência

O mecanismo de inferência é a parte principal do sistema, que consiste no conjunto de regras que modelam os dados. As regras *fuzzy* assumem o formato *SE-ENTÃO (IF-THEN)* e em geral utilizam linguagem simples, próxima a do ser humano. Num exemplo para determinar o valor de uma gorjeta, uma possível regra se apresentaria como:

Se comida é saborosa e serviço é excelente, então gorjeta é generosa.

Denomina-se a parte anterior ao *ENTÃO* de antecedente e, após, de consequente. Observe que “*comida*” e “*serviço*” são variáveis de entrada, enquanto “*saborosa*” e “*excelente*” são conjuntos *fuzzy* especifica-

dos na definição do sistema, com suas respectivas FPs.

No caso de regras onde o antecedente apresenta duas ou mais variáveis, conectores como o *E* (*AND*) ou o *OU* (*OR*), entre outros, são utilizados para relacioná-las. Neste exemplo, o conector “e” é utilizado para relacionar duas variáveis de entrada. “*Gorjeta*” é uma variável de saída do sistema e “*generosa*” é um dos conjuntos *fuzzy* desta variável.

A diferença principal entre os diferentes mecanismos de inferência como *Mamdani* ou *Takagi-Sugeno* é como a saída (ou seja, o consequente) é manipulado. Sistemas *Mamdani* apresentam conjuntos *fuzzy* na saída, enquanto, na maioria dos casos, sistemas *Takagi-Sugeno* definem seus consequentes por uma função polinomial $f(x)$ (KAUR; KAUR, 2012).

2.1.3 Fuzzificador e Defuzzificador

A fuzzificação utiliza os valores “puros” (*crisp*) recebidos como entrada e os relacionam com as FPs definidas no sistema, resultando nos seus graus de pertinência em relação aos diferentes conjuntos *fuzzy*. Na Figura 3, supomos que temos dois valores de entrada x e y que se relacionam com o sistema *fuzzy* através de FPs triangulares A_i e B_i e ativam duas regras de um sistema do tipo Mamdani:

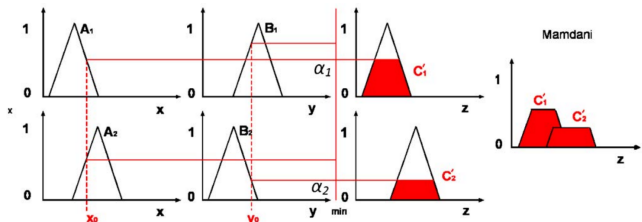


Figura 3 – Fuzzificação e inferência de um sistema do tipo Mamdani, adaptado de Rezaee et al. (2009)

Observe que os conjuntos A_i e B_i mapeiam os valores de x e y de forma que seja possível extrair graus de pertinência contidos no intervalo $[0, 1]$. No exemplo da Figura 3, é utilizado o método *min* para relacionar os antecedentes da regra, geralmente associado ao conector “e”, obtendo os graus de ativação das regras, C'_1 e C'_2 . Logo após, através de algum método, estas duas funções são agregadas, geralmente utilizando-se a união (PEDRYCZ; GOMIDE, 2007).

A saída da inferência *fuzzy Mamdani* ainda é *fuzzy*, ou seja, os resultados são graus de pertinência de conjuntos *fuzzy*. Como mostrado na Figura 3 temos como resultado, neste exemplo, as funções C'_1 e C'_2 . O próximo e último passo do sistema *fuzzy* é a defuzzificação, processo contrário da fuzzificação. Existem muitos métodos para realizar a defuzzificação, como maior ativação, que usa apenas a informação da regra com maior grau de ativação, assim perdendo a informação das outras regras; menor, maior, e média do máximo, que podem ser valores diferentes caso o maior grau de pertinência se repita pela variável de saída; biseção, que determina o ponto onde ocorre a divisão da área da função pela metade, entre outros, cada um apresentando aplicações onde produzem melhor resultado. Porém, o método mais largamente utilizado é o do centróide, que consiste em encontrar o centro de gravidade da função agregada, de acordo com a seguinte equação (BISWAS; NAAZ; ALAM, 2011):

$$z^* = \frac{\int_z \mu_A(z) z dz}{\int_z \mu_A(z) dz} \quad (2.1)$$

Onde z^* é a saída *crisp*, $\mu_A(z)$ é a função agregada e z é a variável de saída. Dessa forma, obtém-se a saída do sistema, um valor real para ser utilizado de forma prática na aplicação desejada.

2.1.4 Classificação com Sistemas *Fuzzy*

Para que o sistema *fuzzy* possa ser utilizado como classificador, realiza-se um pós-processamento da saída. A saída de um sistema *fuzzy* tradicional consiste em conjuntos *fuzzy* (*Mamdani*) ou uma função matemática (*Takagi-Sugeno*). Por isto, faz-se necessário utilizar um método para relacionar a saída do sistema com classes discretas ω . Três métodos gerais de pós-processamento para classificação *fuzzy* podem ser citados (KAYMAK; BERG; BERGH, 2002):

- Método de regressão
- Método discriminante
- Método de compatibilidade máxima

O método de regressão é realizado dividindo o espaço de saída do sistema *fuzzy* em regiões não sobrepostas, onde cada região corresponde a uma classe. Isso consiste em dividir a saída Y em C intervalos

disjuntos I_C e então atribuir a cada intervalo uma classe ω_{I_c} . O problema com este método ocorre quando as classes não são facilmente divididas e ordenadas.

O método discriminante utiliza uma função $d_c(x)$ que é associada a cada classe ω_c . Estas funções podem ser até novos sistemas *fuzzy* com as regras *fuzzy* necessárias para definir as classes. Neste caso, as regras são construídas da seguinte maneira:

$$\text{Se } x_1 \text{ é } A_1 \text{ e } \dots x_M \text{ é } A_{jM}, \text{ então } d_c(x) = f_j^c(x) \quad (2.2)$$

onde f_j^c é a função consequente que está associada a classe ω_c .

O método de máxima compatibilidade é uma variação do método discriminante, onde o consequente de cada regra *fuzzy* consiste em uma classe ω_j e o que realiza a distinção entre estas é o grau de pertinência das FP definidas nos antecedentes. As regras apresentam o seguinte formato:

$$\text{Se } x_1 \text{ é } A_1 \text{ e } \dots x_M \text{ é } A_{jM}, \text{ então } \omega_j, j = 1, 2, \dots, N \quad (2.3)$$

Onde o sistema *fuzzy* deverá possuir N regras para um número C de classes. Note que um sistema tradicional como Mamdani ou Takagi-Sugeno não pode ser utilizado para este método, devido ao fato de os consequentes consistirem em classes *crisp*. Neste método, a interpretabilidade das regras pode ser menos clara e um algoritmo de aprendizagem deve ser aplicado para extrair as regras de um conjunto de dados.

2.1.5 Clusterização *Fuzzy C-Means*

Como mencionado anteriormente, a definição do sistema *fuzzy* consiste no envolvimento de um especialista na área de interesse, na utilização de algoritmos de aprendizagem, ou ambos (SAGGAF; NEBRIJA, 2003). O método *fuzzy c-means* (FCM) é uma das técnicas mais populares e poderosas para aprendizado de máquina *fuzzy*, utilizado para encontrar *clusters fuzzy* em um conjunto de dados, proposto originalmente por Dunn (1973). *Clusters* são grupos de pontos amostrais orientados ao redor de um centróide. Um *cluster* pode ter seus limites bem definidos, conhecido como *clusters crisp*. Por outro lado, quando os limites do *cluster* não são bem definidos, ou seja, ocorre sobrepo-

sição entre *clusters*, estes se tornam *fuzzy* (ZHANG et al., 2014), como ilustrado na Figura 4.

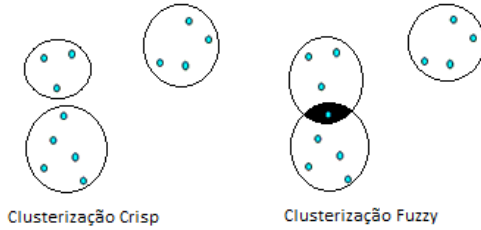


Figura 4 – Exemplos de *clusters crisp* e *clusters fuzzy*

O processo de *clusterização* consiste em um aprendizado não-supervisionado dos dados, de forma a separá-los em grupos, maximizando a similaridade dos pontos amostrais. O FCM determina um grau de pertinência para cada ponto amostral em relação aos centróides de cada *cluster* e os atualiza iterativamente durante o treinamento até atingir um critério de parada ϵ . Desta forma, um ponto amostral pode pertencer a todos os *clusters* com seus graus de pertinência correspondentes. O algoritmo *fuzzy c-means* consiste em 8 passos (CHATTOPADHYAY; PRATIHAR; SARKAR, 2011):

- **1^a passo:** suponha N pontos amostrais M dimensionais, representados por $x_i (i = 1, 2, \dots, N)$ a serem *clusterizados*
- **2^a passo:** defina o número C de *clusters* a serem encontrados no espaço amostral, onde $2 \leq C \leq N$ e um critério ϵ de parada
- **3^a passo:** defina um valor de *cluster fuzziness* $f > 1$
- **4^a passo:** inicializa-se uma matriz de graus de pertinência U com valores aleatórios de tamanho $N \times C \times M$, de forma que $U_{ijm} \in [0, 1]$ e $\sum_{j=1}^C U_{ijm} = 1$ para cada i e valores fixos de m
- **5^a passo:** determina-se os centróides CC_{jm} para o *cluster* j e a dimensão m usando a equação:

$$CC_{jm} = \frac{\sum_{i=1}^N U_{ijm}^f x_{im}}{\sum_{i=1}^N U_{ijm}^f} \quad (2.4)$$

- **6ª passo:** calcula-se a distância Euclidiana entre as amostras i e os centróides CC de cada *cluster* j no eixo m de acordo com a equação:

$$D_{ijm} = \|(x_{im} - CC_{jm})\| \quad (2.5)$$

- **7ª passo:** atualiza-se então a matriz de graus de pertinência U de acordo com D_{ijm} . Se $D_{ijm} > 0$, então calcula-se por

$$U_{ijm} = \frac{1}{\sum_{c=1}^C \left(\frac{D_{ijm}}{D_{icm}}\right)^{\frac{2}{f-1}}} \quad (2.6)$$

- **8ª passo:** repita dos passos 5 a 7 até que as mudanças efetuadas em $U \leq \epsilon$

O quão *fuzzy* os *clusters* serão dependerá do f definido no 3ª passo. Observando a equação 2.6, se $f = 1$, os graus de pertinência U_{ijm} convergem a 0 ou 1, implicando em uma *clusterização crisp*. Quanto maior o valor de f , menor serão os graus U_{ijm} , tornando a *clusterização* mais *fuzzy*. No geral, utiliza-se $f = 2$ quando não se há conhecimento prévio do domínio do problema (CHATTOPADHYAY; PRATHAR; SARKAR, 2011).

2.2 SISTEMAS FUZZY PROBABILÍSTICOS

2.2.1 Introdução

Apesar de sistemas *fuzzy* modelarem a imprecisão linguística de maneira eficiente, eles carecem na capacidade de modelar a incerteza estocástica. Vários estudos analisam a semelhança entre a modelagem *fuzzy* e a teoria de probabilidade, e estas mostram que o conceito de pertinência e probabilidade são diferentes. Devido a isso, a pesquisa atrás de sistemas *fuzzy* com propriedades probabilísticas, capazes de modelar os dois tipos de incerteza no mesmo sistema tem crescido (WALTMAN, 2005; KATARZYNA; WALASZEK-BABISZEWSKA, 2011; KAYMAK; BERG, 2013).

Existem várias implementações de sistemas *fuzzy* probabilísticos, mas, no geral, o ponto de partida são os conceitos de probabilidade *fuzzy* definidos por Zadeh (1968). Porém, Zhang e Li (2012) também citam sistemas *fuzzy* probabilísticos como um caso específico dos sistemas

Tipo de Incerteza	Incerteza subjetiva resultante da descrição humana linguística	Incerteza objetiva resultante das características do problema	Incerteza resultante de aleatoriedade
Métodos para modelar a incerteza	Lógica Fuzzy		Teoria de probabilidade

Figura 5 – Tipos de incerteza e seus métodos, adaptado de Katarzyna e Walaszek-Babiszewska (2011)

fuzzy tipo-2, em que cada variável de entrada possui pares de funções de pertinência, aumentando a precisão da modelagem da imprecisão linguística, gerando conjuntos *fuzzy* de três dimensões. Neste caso, uma destas funções de pertinência será substituída por uma função de densidade, como ilustrado na Figura 6 (MENDEL; JOHN, 2002; ZHANG; LI, 2012; KAYMAK; BERG, 2013).

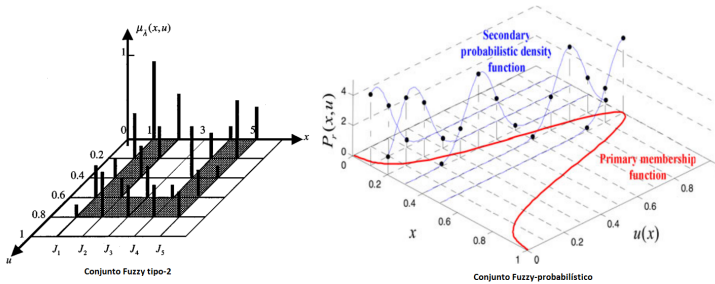


Figura 6 – Conjuntos *fuzzy* tipo-2 e probabilístico, retirados de Mendel e John (2002), Zhang e Li (2012)

As características e operações de sistemas *fuzzy* probabilístico (fuzzificação, formato das regras, defuzzificação, etc.) também variam entre as implementações. Sua saída pode ser intervalos ou até funções de densidade que, neste caso, representarão as probabilidades condicionais dos eventos. Kaymak e Berg (2013) mostram, também, como é possível obter estimativas das probabilidades condicionais utilizando a distribuição a priori dos dados e a função de pertinência de um dado evento *fuzzy*, como explicado na seção 2.2.2, estabelecendo uma relação quantificável entre os conceitos de probabilidade e pertinência.

A saída de um sistema *fuzzy* probabilístico pode ser apresentada como uma função de densidade, como mostra a Figura 7, onde y é a saída do sistema, y_0 a média e σ_y a variabilidade da função de saída (ZHANG; LI, 2012).

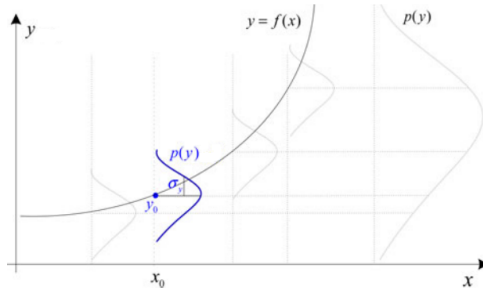


Figura 7 – Exemplo de relação entrada-saída de um sistema *fuzzy* probabilístico, adaptado de Zhang e Li (2012)

Portanto, considere classes *crisp* $y \in C_1, \dots, C_2$, classificadas por uma variável $x = (x_1, \dots, x_d)$. Desta forma, sistemas *fuzzy* probabilísticos para classificação podem assumir o seguinte formato de regras, de forma a oferecer uma medida de confiabilidade na tarefa de classificação (WALTMAN, 2005):

Se x_i é A_i , então $y = C_1$ com probabilidade $p_{j,1}$

...

$y = C_i$ com probabilidade $p_{j,i}$

Zhang e Li (2012) realizam uma comparação de complexidade entre algumas implementações *fuzzy*, incluindo a modelagem clássica, tipo-2 e probabilística. Este estudo mostra que a implementação probabilística é mais eficiente que a tipo-2, mas menos eficiente que a clássica. Porém, realizando uma comparação entre a acurácia dos sistemas em um problema onde existe incerteza estocástica, encontra-se melhores resultados nos sistemas *fuzzy* probabilísticos.

Nas próximas seções, serão discutidos os conceitos utilizados na implementação de sistemas *fuzzy* probabilísticos desenvolvidos por Kaymak e Berg (2013), que servirão de base para o desenvolvimento da rede PFNB.

2.2.2 Probabilidade de Eventos *Fuzzy*

Zadeh (1968) define o conceito de probabilidade de um evento *fuzzy*. Assumindo uma variável aleatória x em um espaço amostral X . Um subconjunto A de X define um evento, e sua probabilidade $P(A)$ é definida por:

$$P(A) = \int_{x \in A} f(x) dx = \int_{-\infty}^{+\infty} \chi_A(x) f(x) dx \quad (2.7)$$

onde $\chi_A(x)$ é a função característica bivalente (*crisp*) do evento A , de forma que $\chi_A(x) = 1 \leftrightarrow x \in A$ e $\chi_A(x) = 0 \leftrightarrow x \notin A$. Ou seja, a probabilidade de um evento é dada pela expectativa da sua função característica.

Substituindo a função característica na equação 2.7 por uma função de pertinência $\mu(x) : X \rightarrow [0, 1]$, a medida de probabilidade para eventos *crisp* é estendida para a medida de probabilidade de eventos *fuzzy*. Neste caso, a probabilidade de um evento *fuzzy* A é dada pela expectativa da função de pertinência, de acordo com a seguinte equação (KAYMAK; BERG, 2013):

$$P(A) = \int_{-\infty}^{+\infty} \mu_A(x) f(x) dx = E(\mu_A(x)) \quad (2.8)$$

A equação 2.8 é ilustrada na Figura 8, onde, assumindo uma variável aleatória x , sua função de densidade $f(x)$ e uma função de pertinência $\mu(x)$ que define um conjunto *fuzzy*, o produto entre estas funções é uma estimativa *fuzzy* da função de densidade do evento A . Além disso, sua área estima a média do evento A . Desta forma, leva-se em consideração tanto a imprecisão linguística (definida pelo conjunto *fuzzy*), quanto a incerteza estocástica (definida pela função de densidade) (WALTMAN, 2005; KAYMAK; BERG, 2013).

2.2.3 Espaço Amostral *Fuzzy* Bem Definido

Assumindo vários eventos *fuzzy* A_c ($c = 1, \dots, C$) definidos no espaço amostral X , cada ponto amostral x corresponde a um conjunto de graus de pertinência $\mu_{A_1}, \mu_{A_2}, \dots, \mu_{A_c}$. Desta forma, interpreta-se cada ponto amostral x como uma amostra *fuzzy*. Uma amostra *crisp* apenas pertence ou não pertence a um evento, enquanto uma amostra *fuzzy* pode pertencer a vários eventos A_c com diferentes graus

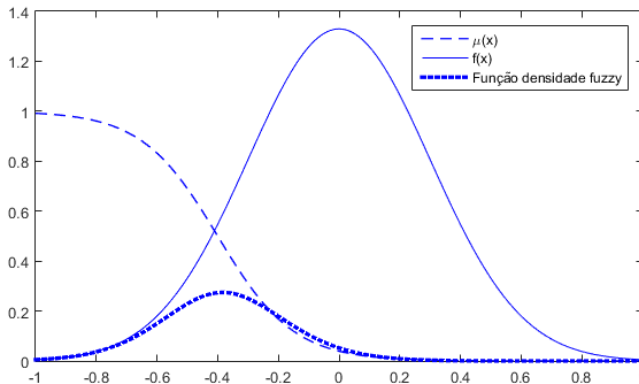


Figura 8 – Exemplo de estimação de uma função de densidade, através do produto $f(x)\mu(x)$

de pertinência.

Para que o conjunto de eventos *fuzzy* A_c seja considerado bem definido, é necessário que seja observada a condição (KAYMAK; BERG; BERGH, 2002):

$$\forall x : \sum_{A_c} \mu_{A_c}(x) = 1 \quad (2.9)$$

ou seja, a soma de todos os graus de pertinência μ_{A_c} de uma amostra *fuzzy* é sempre igual a 1. A Figura 9 ilustra um espaço amostral *fuzzy* bem definido:

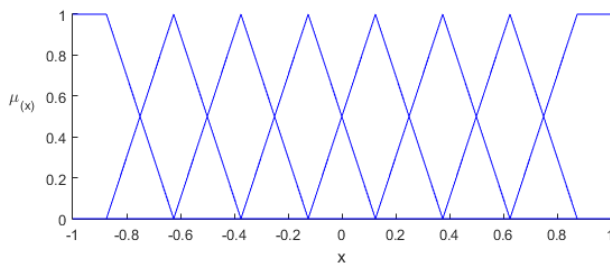


Figura 9 – Espaço amostral *fuzzy* bem definido

No entanto, Waltman (2005) e Ryhajlo (2013) afirmam que não é necessário que o espaço amostral *fuzzy* seja bem definido, desde que realizada a normalização dos graus de pertinência por:

$$\bar{\mu}_{A_c}(x) = \frac{\mu_{A_c}(x)}{\sum \mu_{A_c}(x)} \quad (2.10)$$

onde $\bar{\mu}_{A_c}(x)$ é o grau de pertinência normalizado de x , satisfazendo a condição 2.9.

2.2.4 Estatística de Eventos *Fuzzy*

Possuindo um número S de pontos amostrais $x_s, s = 1, \dots, S$, de um espaço amostral X com uma função de densidade $f(x)$, é possível utilizar a estatística para estimar probabilidades, tanto no caso *crisp* quanto no caso *fuzzy*.

No caso *crisp*, o espaço amostral X é dividido em eventos A_c . Desta forma, a probabilidade incondicional de um evento é dada por:

$$P(A_c) \approx \tilde{f}_{A_c} = \frac{f_{A_c}}{S} = \frac{1}{S} \sum_{x_s} \chi_{A_c}(x_s) \quad (2.11)$$

onde χ_{A_c} representa a função característica do evento, \tilde{f}_{A_c} a frequência relativa e f_{A_c} a frequência absoluta de pontos amostrais $x_s \in A_c$. Assim, o vetor $(\tilde{f}_{A_1}, \tilde{f}_{A_2}, \dots, \tilde{f}_{A_c})$ é a estimativa da distribuição de probabilidade de todas os eventos A_c .

Generaliza-se este conceito *crisp* para um espaço amostral *fuzzy*, substituindo as funções características por funções de pertinência. Neste caso, X é dividido em A_c eventos *fuzzy*, e suas probabilidades incondicionais podem ser estimadas por (BERG; BERGH; KAYMAK, 2001):

$$P(A_c) \approx \tilde{f}_{A_c} = \frac{f_{A_c}}{S} = \frac{1}{S} \sum_{x_s} \mu_{A_c}(x_s) \quad (2.12)$$

onde μ_{A_c} representa a função de pertinência da classe, \tilde{f}_{A_c} a frequência relativa e f_{A_c} a frequência absoluta de pontos amostrais $x_p \in A_c$. Novamente, o vetor $(\tilde{f}_{A_1}, \tilde{f}_{A_2}, \dots, \tilde{f}_{A_c})$ é a estimativa da distribuição de probabilidade de todas os eventos *fuzzy* A_c .

Para estimar probabilidades condicionais $P(A_c|A_b)$ no caso *crisp*, utiliza-se a equação:

$$P(A_c|A_b) = \frac{P(A_c \cap A_b)}{P(A_b)} \approx \frac{\tilde{f}_{A_c \cap A_b}}{\tilde{f}_{A_b}} = \frac{\sum_{x_s} \chi_{A_c}(x_s) \chi_{A_b}(x_s)}{\sum_{x_s} \chi_{A_b}(x_s)} \quad (2.13)$$

utilizando a propriedade $\chi_{A_c \cap A_b}(x_s) = \chi_{A_c}(x_s) \chi_{A_b}(x_s)$. Estendendo mais uma vez ao caso *fuzzy*, modifica-se a fórmula e obtém-se (BERG; BERGH; KAYMAK, 2001):

$$P(A_c|A_b) = \frac{P(A_c \cap A_b)}{P(A_b)} \approx \frac{\tilde{f}_{A_c \cap A_b}}{\tilde{f}_{A_b}} = \frac{\sum_{x_s} \mu_{A_c}(x_s) \mu_{A_b}(x_s)}{\sum_{x_s} \mu_{A_b}(x_s)} \quad (2.14)$$

2.2.5 Histogramas *Fuzzy*

Na teoria de probabilidade clássica, é possível aproximar uma função de densidade através de um histograma. Para construir um histograma divide-se o espaço amostral em classes uniformes Γ_c (*bins*) e então calcula-se a frequência de valores que pertencem a cada classe. São desenhados, então, retângulos onde a altura é igual a frequência de valores de cada classe e sua base é o tamanho do intervalo de cada classe, como exemplificado na Figura 10

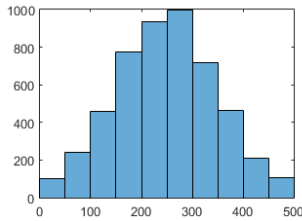


Figura 10 – Exemplo de histograma com 10 *bins* de tamanho 50

Dessa forma, a função de densidade $f(x)$ é então aproximada por $\hat{f}(x)$:

$$\hat{f}(x) = \sum_c \frac{P(\Gamma_c) \chi_{\Gamma_c}(x)}{\int_{-\infty}^{+\infty} \chi_{\Gamma_c}(x) dx} \quad (2.15)$$

onde $\int_{-\infty}^{+\infty} \chi_{\Gamma_c}(x) dx$ representa o tamanho de cada classe, isto é,

em um espaço bidimensional, a largura de cada retângulo.

Se os retângulos forem substituídos por funções de pertinência de áreas equivalentes, como ilustrado na Figura 11, dividimos o espaço amostral em classes *fuzzy* A_c .

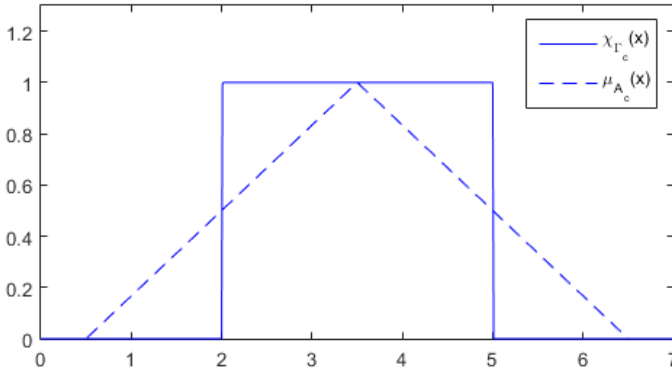


Figura 11 – Exemplo de função de pertinência equivalente a um *bin crisp*, ambos com área 3

Assim, modifica-se a equação 2.15, substituindo a função característica χ_{A_c} por uma função de pertinência μ_{A_c} . Logo, é possível estimar uma função de densidade $\hat{f}(x)$ por (ALMEIDA et al., 2014):

$$\hat{f}(x) = \sum_c \frac{P(A_c)\mu_{A_c}(x)}{\int_{-\infty}^{+\infty} \mu_{A_c}(x)dx} \quad (2.16)$$

Devido ao fato de os *bins fuzzy* se sobreporem, a estimativa $\hat{f}(x)$ torna-se mais “suave” do que em um histograma clássico, como ilustrado na Figura 12. Observa-se que $\hat{f}(x)$ obtida através do histograma *fuzzy* é mais próxima da distribuição real dos dados (WALTMAN, 2005).

2.3 REDES BAYESIANAS

2.3.1 Introdução

As redes bayesianas (RBs), originalmente definidas por Pearl (1985), são modelos gráficos que representam um conjunto de variáveis e os relacionamentos probabilísticos existentes entre elas. RBs são grafos

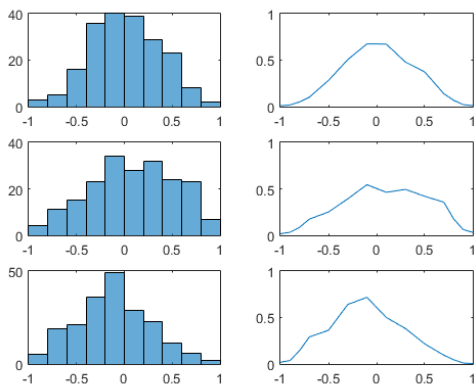


Figura 12 – Comparação entre histogramas tradicionais e funções de densidade *fuzzy* de dados normalmente distribuídos

direcionados acíclicos, onde cada nodo representa um uma variável. Cada variável contém um conjunto de estados possíveis. Por exemplo, na Figura 13, as variáveis A e B influenciam diretamente na variável C , que por sua vez influencia a variável E , e assim por diante. As variáveis que influenciam outras são chamadas de pais e as variáveis que sofrem influência são chamadas de filhos.

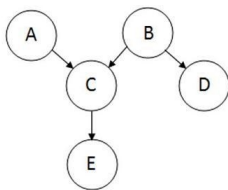


Figura 13 – Exemplo de rede bayesiana

Além disso, cada nodo possui um conjunto de estados e está associado a uma tabela de probabilidade correspondente. Suponha os nodos da Figura 13. A é uma variável discreta com dois estados $A = Verdadeiro$ e $A = Falso$. A variável B , por outro lado, é contínua no intervalo $[1, 10]$. Para que se possa modelar a variável B em uma rede bayesiana tradicional, é necessário discretizar a variável. Ainda neste

exemplo, tanto A quanto B são variáveis independentes. Por isso, suas tabelas de probabilidades conterão suas probabilidades incondicionais $P(A)$ e $P(B)$. C não é independente, pois possui A e B como pais. Neste caso, sua tabela de probabilidade será referente a distribuição de probabilidade condicional $P(C|A, B)$, como na Figura 14

A	P(A)
V	30%
F	70%

B	P(B)
1-5	75%
6-10	25%

A	B	P(C A,B)
V	1-5	20%
V	6-10	30%
F	1-5	25%
F	6-10	25%

Figura 14 – Exemplo de tabelas de probabilidade incondicional $P(A)$ e $P(B)$ e condicional $P(C|A, B)$. A variável B foi discretizada em duas classes.

A distribuição de probabilidade conjunta representada por uma rede bayesiana com as variáveis V_1, V_2, \dots, V_n é dada pela regra da cadeia (LIMA, 2014):

$$P(V_1, V_2, \dots, V_n) = \prod^n P(V|pais(V)) \quad (2.17)$$

A distribuição de probabilidade conjunta obtida pela equação 2.17 pode não ser computacionalmente eficiente. A quantidade de parâmetros em uma tabela de probabilidade conjunta cresce exponencialmente, e é calculada por e^n onde e é o número de estados possíveis das variáveis em V .

As RBs são amplamente utilizadas para modelar e explicar o domínio de um problema onde a incerteza estocástica está presente, de maneira simples e facilmente visualizável, auxiliando na tomada de decisão. Além disso, a inserção de novas informações na rede pode ser feita apenas atualizando os parâmetros probabilísticos de cada nodo (JENSEN; NIELSEN, 2007).

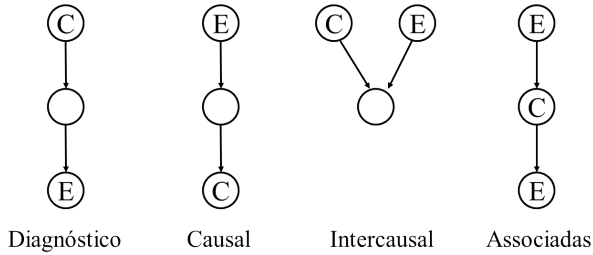


Figura 15 – Tipos de inferências bayesianas. (KORB; NICHOLSON, 2011)

2.3.2 Inferência

A inferência de uma rede bayesiana consiste em computar uma distribuição de probabilidade condicional $P(C|E)$ para um conjunto de variáveis de consulta C , dado um conjunto de variáveis de evidência E . Vários algoritmos foram desenvolvidos e podem ser utilizados para este fim, e estes podem ser mais adequados para os diferentes requisitos de performance e estruturas de rede. O método tradicional para realizar a inferência probabilística consiste no Teorema de Bayes (KORB; NICHOLSON, 2011):

$$P(C|E) = \frac{P(E|C)P(C)}{P(E)} \quad (2.18)$$

A inferência bayesiana pode ocorrer de quatro maneiras:

- **Inferência por Diagnóstico:** quando ocorre de efeitos para causas. Por exemplo: $P(C|E)$
- **Inferência Causal:** quando ocorre de causas para efeitos. Por exemplo $P(E|C)$
- **Inferência Intercausal:** quando ocorre entre causas de um efeito comum. Por exemplo: $P(A|C, E)$
- **Inferências Associadas:** quando dois ou mais tipos de inferência são combinados

2.3.3 Classificador Naïve Bayes

A *Naïve Bayes* (rede bayesiana ingênua) é uma estrutura amplamente conhecida das redes bayesianas que consiste em assumir independência entre as variáveis de evidência E , todas possuindo apenas um nodo pai C , como na Figura 16:

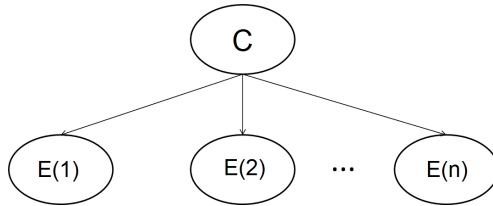


Figura 16 – Exemplo de rede bayesiana do tipo *naïve bayes*

A simplicidade presente na rede *naïve bayes* torna o processo mais computacionalmente eficiente. Ao assumir independência entre as variáveis de evidência, as propriedades probabilísticas de cada nodo em relação a variável de consulta podem ser aprendidas uma de cada vez e o tamanho das tabelas de probabilidade conjunta calculadas pela regra da cadeia, equação 2.17, torna-se linear de acordo com o número de variáveis de evidência E (MCCALLUM; NIGAM, 1998).

Como diz seu próprio nome, assumir esta forte independência ingênua entre as variáveis modeladas não é a melhor forma de se modelar um grande número de problemas do mundo real (BUSTAMANTE; GARRIDO; SOTO, 2007). Porém, as redes *naïve bayes* mostram performances boas em vários problemas complexos, além de não necessitar de um grande número de amostras para estimar seus parâmetros (RENNIE et al., 2003).

2.3.4 Redes *Fuzzy* Bayesianas

As redes bayesianas possuem a capacidade de modelar apenas uma incerteza, a incerteza estocástica, resultante de aleatoriedade. De forma a inserir o conhecimento complementar presente na modelagem da imprecisão linguística, técnicas híbridas denominadas de redes *fuzzy* bayesianas foram desenvolvidas (RYHAJLO, 2013; BRIGNOLI et al., 2015; TANG; LIU, 2007; PAN, 1999; BUSTAMANTE; GARRIDO; SOTO, 2007;

PENZ et al., 2012; TIBIRICA, 2005; PENG-CHENG et al., 2012).

Bustamante, Garrido e Soto (2007) citam também como vantagem dos modelos híbridos o fato de que a discretização *fuzzy* reduz a perda de informação disponível nos dados, ao contrário da discretização tradicional *crisp*. Ao discretizar variáveis da maneira tradicional, claramente há perda de informações ao comparar-se a conjuntos *fuzzy*. Um dos trabalhos mais antigos, Pan (1999), propôs a abordagem *fuzzy* na discretização de variáveis contínuas.

Ryhajlo (2013) afirma que é importante que um sistema de prevenção a falhas mecânicas ou elétricas apresente e leve em consideração *graus* de degradação dos componentes, exemplificando a vantagem de expressar as variáveis em termos de graus de pertinência. Ele também menciona que é uma prática comum entre as implementações de redes *fuzzy* bayesianas que seja feita a normalização dos graus de pertinência, como visto na seção 2.2.3.

Assim como os sistemas *fuzzy* probabilísticos, existem vários métodos diferentes de integrar sistemas *fuzzy* e redes bayesianas. Em geral, estas técnicas são diferentes entre si por terem sido desenvolvidas para representar diferentes tipos de problemas. Isso pode tornar mais difícil a tarefa de observar as diferenças e semelhanças entre estas abordagens (RYHAJLO, 2013).

Uma formalização da inferência da abordagem *fuzzy* bayesiana, referenciada por alguns autores (PENZ et al., 2012; BRIGNOLI et al., 2015), origina de Tibirica (2005) e Brignoli et al. (2015):

$$P(H_i|\tilde{e}_j) = \frac{P(\tilde{e}_j|H_i)P(H_i)}{P(\tilde{e}_j)} \quad (2.19)$$

Onde H_i é a hipótese a ser testada e \tilde{e}_j as evidências *fuzzy* associadas a ela. Para cada evidência \tilde{e}_j , funções de pertinência são definidas de forma a descrever a vagueza ou imprecisão presente na variável.

Desta forma, as abordagens *fuzzy* bayesianas tendem a inserir o conhecimento *fuzzy* direto no mecanismo de inferência, enquanto a rede proposta neste trabalho aplica o raciocínio *fuzzy* desde a estimação dos parâmetros, no treinamento. Tang e Liu (2007) concluem que a junção dos dois conceitos possibilita a modelagem de vários problemas práticos que apresentam os dois tipos de incerteza.

Bustamante, Garrido e Soto (2007) lista como trabalho futuro a possibilidade de usar um algoritmo de clusterização *fuzzy* para gerar as funções de pertinência, assim como na rede PFNB. Neste capítulo foram apresentados conceitos acerca de sistemas *fuzzy*, sistemas *fuzzy* probabilísticos, redes bayesianas e redes *fuzzy* bayesianas que servem

como base para o próximo capítulo. A seguir, será demonstrado como estes conceitos se relacionam no desenvolvimento da rede PFNB.

3 REDE PROBABILÍSTICA *FUZZY NAÏVE-BAYES*

3.1 INTRODUÇÃO

Utilizando como base o modelo de sistema *fuzzy* probabilístico proposto por Kaymak e Berg (2013) e os conceitos de inferência bayesiana, foi implementada a rede probabilística *fuzzy* *naïve-bayes* (PFNB). A rede PFNB é capaz de modelar simultaneamente a imprecisão linguística e a incerteza estocástica e, através da sua interpretabilidade, pode produzir conhecimentos acerca de um determinado conjunto de dados após o seu treinamento.

Seu treinamento é supervisionado, recebendo S vetores de entrada (amostras) $x_k (k = 1, 2, \dots, K)$ de um conjunto de dados, onde K é a quantidade de atributos (dimensões), e suas respectivas saídas y . Além disso, a rede necessita que valores C e b sejam selecionados pelo usuário antes do treinamento, que serão utilizados posteriormente no *fuzzy c-means* (FCM) e na estimação de funções de densidade, respectivamente. Os valores C e b precisam ser estudados individualmente para cada problema.

A estrutura da rede PFNB possui 3 componentes principais: o fuzzificador (funções de pertinência), seu mecanismo de inferência bayesiana e a camada de saída. Estes componentes são apoiados pelos parâmetros de treinamento da rede (funções de densidade e as probabilidades a priori e condicionais) estimados do conjunto de dados, além de pesos ω que mapeiam a relação entre a saída da rede e as classes do problema. Esta estrutura geral da rede proposta é ilustrada na Figura 17.

3.2 FUNÇÕES DE PERTINÊNCIA

O sistema utiliza funções de pertinência gaussianas, que realizam a fuzzificação dos vetores de entrada x_k . Funções de pertinência gaussianas foram escolhidas devido a facilidade para calculá-las e sua semelhança com a distribuição normal. Antes do treinamento, um valor C é especificado, utilizado no FCM, para definir a quantidade de *clusters fuzzy* que serão estimados.

A partir de uma estimativa por máxima verossimilhança dos dados de cada *cluster*, obtém-se a média e desvio padrão, que serão utilizados para gerar as FPs gaussianas. Desta forma, dadas amostras

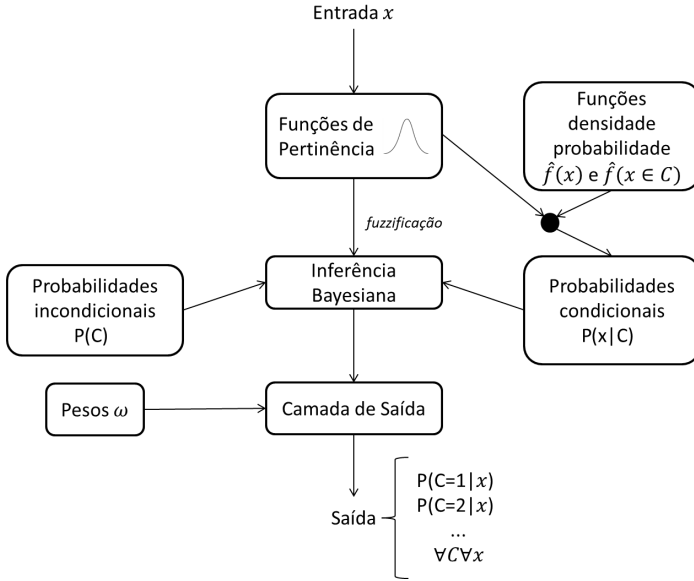


Figura 17 – Estrutura geral do modelo proposto

$x_k (k = 1, 2, \dots, K)$, a quantidade total de FPs será dada pela multiplicação $C.K$, ou seja, a quantidade de *clusters* definidos anteriormente e a quantidade de atributos de cada amostra. O grau de pertinência de uma amostra x é dado pela multiplicação de todos os graus em cada dimensão:

$$\mu_C(x) = \prod_1^K \mu_C(x_k) \quad (3.1)$$

Por se tratar de funções de pertinência geradas automaticamente a partir de um algoritmo não-supervisionado, o espaço amostral *fuzzy* resultante não será bem definido. Por isto, normaliza-se todos os graus de pertinência calculados utilizando a equação 2.10:

$$\bar{\mu}_C(x) = \frac{\mu_C(x)}{\sum \mu_C(x)} \quad (3.2)$$

Como nos sistemas *fuzzy*, na rede PFNB as funções de pertinência também representam a modelagem linguística do problema.

3.3 FUNÇÕES DE DENSIDADE

A rede armazena as estimativas $\hat{f}(x)$ das funções de densidade $f(x)$ de cada dimensão do conjunto de dados. Estas são estimadas através dos histogramas fuzzy, e os dados de cada eixo são divididos em b *bins fuzzy* (funções de pertinência triangulares, como na Figura 11). Primeiro, a probabilidade a priori de cada *bin fuzzy* é calculada pela equação 2.12:

$$P(b) = \frac{1}{S} \sum_{x_s} \mu_b(x_s) \quad (3.3)$$

Então, $\hat{f}(x)$ é aproximada pela equação 2.16:

$$\hat{f}(x) = \sum_B \frac{P(b)\mu_b(x)}{\int_{-\infty}^{+\infty} \mu_b(x)dx} \quad (3.4)$$

A rede também armazena estimativas $\hat{f}(x \in C)$ referentes aos dados pertencentes a cada *cluster*. $\hat{f}(x \in C)$ também é aproximada pelas equações 2.12 e 2.16, utilizando apenas amostras onde $\mu_C > 0$.

Como nas redes bayesianas, na rede PFNB as funções de densidade também são responsáveis pela modelagem da incerteza estocástica.

3.4 PROBABILIDADES A PRIORI E CONDICIONAIS

A rede armazena uma tabela discreta de probabilidades a priori $P(C)$ para cada *cluster*, calculadas pela equação 2.12:

$$P(C) = \frac{1}{S} \sum_{x_s} \mu_C(x_s) \quad (3.5)$$

Além disso, utilizando as estimativas $\hat{f}(x \in C)$ calculadas anteriormente, a rede também armazena as probabilidades condicionais $P(x|C)$, obtidas através da equação 2.8:

$$P(x|C) = \int_{-\infty}^{+\infty} \mu_C(x)\hat{f}(x \in C)dx \quad (3.6)$$

As probabilidades condicionais representam a junção das duas incertezas, imprecisão e estocástica, pois levam em consideração tanto

as estimativas probabilísticas, quanto os graus de pertinência dos *clusters fuzzy*.

3.5 MECANISMO DE INFERÊNCIA BAYESIANA

Após a estimação dos parâmetros, o modelo é capaz de realizar o processo de inferência baseado no Teorema de Bayes (equação 2.18), assim como nas redes bayesianas. A inferência é feita em 3 passos:

- **pré-processamento dos dados:** normalização dos dados no intervalo $[-1, 1]$
- **1^a passo:** calcula-se a probabilidade condicional $P(C|x)$ utilizando as $P(x|C)$ estimadas no treinamento, de acordo com o Teorema de Bayes, equação 2.18
- **2^a passo:** multiplica-se a saída do sistema pela matriz de pesos da camada de saída
- **3^a passo:** normaliza-se o output com $\frac{P(C|x)}{\sum P(C|x)}$ para satisfazer $\sum P(C|x) = 1$

Desta forma, obtém-se as $P(C|x)$ do conjunto de entrada, estimando a probabilidade de cada ponto amostral pertencer a cada *cluster C*

3.6 CAMADA DE SAÍDA

A camada de saída é onde será feita a relação entre os *clusters fuzzy* obtidos com o FCM e as classes do problema através de um pós-processamento. A saída do sistema consiste nas probabilidades condicionais $P(C|x)$, ou seja, a probabilidade de uma classe C dado uma entrada x . A camada de saída utiliza o resultado da inferência bayesiana e a multiplica por pesos ω , estimados através de regressão linear durante o treinamento, dados pela seguinte equação:

$$\omega = (X^T X)^{-1} X^T y \quad (3.7)$$

Onde X é a saída do sistema e y é a classe de saída real do conjunto de dados. Isto é necessário pois é possível definir um número C de *clusters* diferentes da quantidade real de classes de saída dos dados ou a ordem pode estar inconsistente, devido a natureza não-supervisionada

do FCM (ABONYI; SZEIFERT, 2013). Além disso, classes de problemas reais podem ser melhor modeladas por dois ou mais *clusters* gaussianos.

3.7 TREINAMENTO

Como citado anteriormente, para o treinamento da PFNB é necessário dispor de um conjunto de dados com S amostras $x_k (k = 1, 2, \dots, K)$ e suas respectivas saídas y . Além disso, deve-se especificar a quantidade de *clusters* C a serem encontrados pelo FCM e a quantidade de *bins fuzzy* b na estimativa das funções de densidade dos dados. O treinamento segue 8 passos:

- **pré-processamento dos dados:** normalização dos dados no intervalo $[-1, 1]$
- **1ª passo:** clusterização utilizando FCM em C *clusters*, definidos anteriormente
- **2ª passo:** obtenção de funções de pertinência gaussianas com o desvio padrão e média dos dados de cada cluster, utilizando máxima verossimilhança.
- **3ª passo:** calcula-se o $\mu_C(x)$ de cada ponto amostral x a cada *cluster* (equação 3.1).
- **4ª passo:** normalização dos $\mu_C(x)$ de acordo com a equação 3.2
- **5ª passo:** cálculo das probabilidades a priori $P(C)$ de cada *cluster* dadas pela equação 3.5
- **6ª passo:** estima-se as funções de densidade $\hat{f}(x_{k\dots K})$ e $\hat{f}(x_{k\dots K} \in C)$, em todo o intervalo $[-1, 1]$, dadas pelas equações 3.5 e 3.4, particionando o espaço amostral em b *bins fuzzy*, definidos anteriormente
- **7ª passo:** calcula-se as probabilidades condicionais $P(x|C)$ utilizando as estimativas $\hat{f}(x \in C)$ calculadas anteriormente e as funções de pertinência obtidas no 2ª passo, de acordo com a equação 3.6
- **8ª passo:** calcula-se a matriz de pesos da camada de saída através de regressão linear, equação 3.7

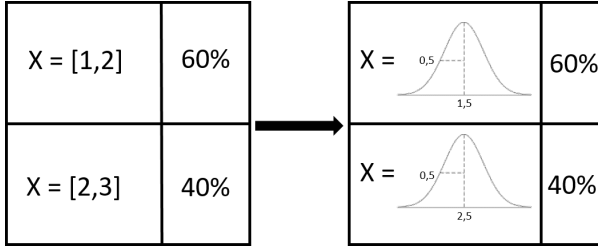


Figura 18 – Comparação entre a *naïve bayes* tradicional e a rede PFNB proposta

3.8 INTERPRETABILIDADE

A rede PFNB, ao fundir características de sistemas *fuzzy* probabilísticos e redes bayesianas *naïve*, mantém um alto nível de interpretabilidade após o treinamento. É possível interpretá-la da mesma maneira que pode-se interpretar os dois modelos. Por exemplo, a PFNB gera regras *fuzzy* no seguinte formato:

Se x_1 é A_1 , então y é C_1 com $P(C_1|A_1)$

Se x_1 é A_1 , então y é C_2 com $P(C_2|A_1)$

...

Se x_1 é A_1 , então y é C_j com $P(C_j|A_1)$

Este formato de regras é semelhante ao mecanismo de inferência *Takagi-Sugeno*, onde a saída é uma função matemática. Neste caso, a função é referente a inferência bayesiana (equação 2.18) e depende dos graus de pertinência $\mu_{(x_k)}$ obtidos na fuzzificação.

É possível, também, interpretar a rede como uma *naïve bayes* tradicional, ao notar-se que os intervalos discretos utilizados para representar as classes, no modelo clássico, tornam-se FPs gaussianas, como na Figura 18. Neste exemplo, a variável X está discretizada nos intervalos $[1, 2]$ e $[2, 3]$ e observa-se a conversão em FPs de variância 0,5 e médias 1,5 e 2,5. Desta forma, a modelagem de cada variável se torna de natureza contínua e permite caracterizar uma amostra em graus de pertinência.

Unindo esta interpretabilidade com a capacidade de aprendizado da rede proposta, esta é capaz de gerar conhecimento a respeito de um

conjunto de dados, analisando a rede após seu treinamento.

3.9 CONJUNTO DE DADOS *SIMPLE CLASS*

De forma a demonstrar a capacidade da rede e sua vantagem em relação aos modelos tradicionais, foram realizados experimentos no conjunto de dados *Simple Class*. O *Simple Class* está disponível no software *MATLAB*[®] como exemplo didático para testar modelos de classificação. É um conjunto de dados bidimensional (x, y) , contendo 1000 pontos amostrais e 4 *clusters*. Os *clusters* possuem 233, 277, 243 e 247 pontos amostrais e seus centros possuem maior densidade de amostras, como mostra a Figura 19. Para treinar a rede, primeiro normaliza-se os dados entre $[-1, 1]$. O *Simple Class* é um conjunto de dados simples e com *clusters* muito bem definidos, por isso, todos os valores de C e b testados atingiram 100% de acurácia. Neste exemplo, apenas por motivos didáticos, C e b foram especificados como 4 e 12 respectivamente.

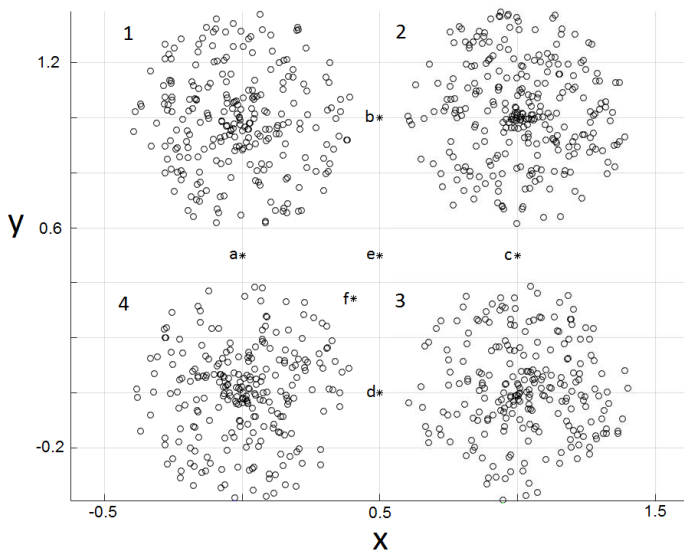


Figura 19 – Conjunto de dados *Simple Class*

3.9.1 Funções de pertinência

Foram definidos 4 *clusters* como parâmetro para o *fuzzy c-means*. Então, obtém-se as funções de pertinência μ , como mostra a Figura 20. Por ser um problema bidimensional, ou seja, $K = 2$, o produto $C.K$ resulta em 8 funções de pertinência.

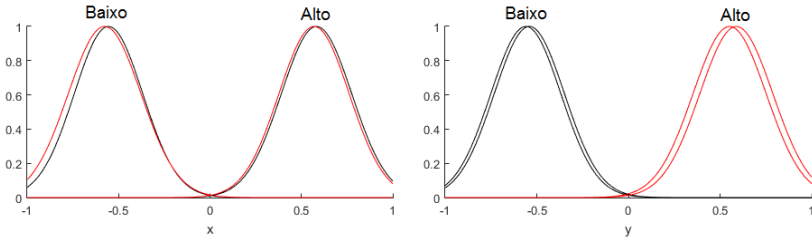


Figura 20 – Funções de pertinência de x e y obtidas no treinamento da rede

3.9.2 Funções de densidade

Em seguida, são estimadas as funções de densidade do eixo x $\hat{f}(x)$ e do eixo y $\hat{f}(y)$, mostradas na Figura 21, além das $\hat{f}(x \in C)$ e $\hat{f}(y \in C)$, particionando o espaço de entrada em 12 *bins fuzzy*. Por tratarem-se de variáveis muito semelhantes, x e y possuem distribuições em cada C quase idênticas. Para simplificar, serão demonstradas apenas as estimativas em x , mostradas na Figura 22, onde $FD(x)$ é a função de densidade *fuzzy* estimada pelos histogramas *fuzzy*, $FP(x|C)$ é a função de pertinência aprendida pela rede e $FD \text{ Fuzzy}$ é o produto entre as duas, estimando a função de densidade de cada *cluster*.

3.9.3 Probabilidades condicionais e a priori

Então, estima-se as probabilidades a priori de cada *cluster* $P(C = 1, \dots, 4) = 0.2330, 0.2769, 0.2429$ e 0.2472 , respectivamente, através da equação 2.12.

Além disso, de posse das estimativas $\hat{f}(x \in C)$ e $\hat{f}(y \in C)$, podemos, como demonstra Kaymak e Berg (2013), obter estimativas

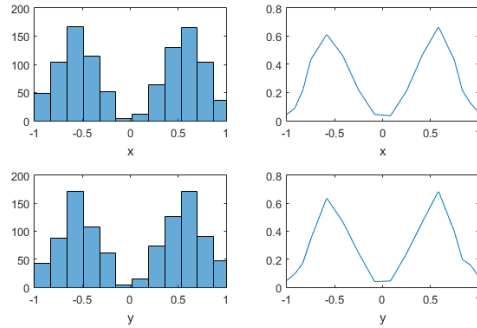


Figura 21 – Histogramas tradicionais e funções de densidade estimadas do eixo x e y através de histogramas *fuzzy* com 12 *bins fuzzy*

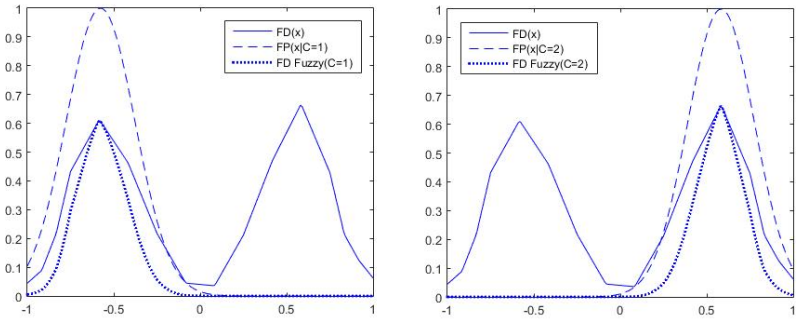


Figura 22 – Funções de densidade *fuzzy* calculadas com a equação 3.1 para dois *clusters* do eixo x

das probabilidades condicionais $P(x|C)$ multiplicando-as pelas funções de pertinência obtidas anteriormente, como na Figura 23:

3.9.4 Camada de saída

Por último, é simulada a saída do sistema com as entradas do conjunto de dados, a fim de calcular os pesos ω com a equação 3.7. A Tabela 1 demonstra a matriz ω responsável por mapear a saída da rede

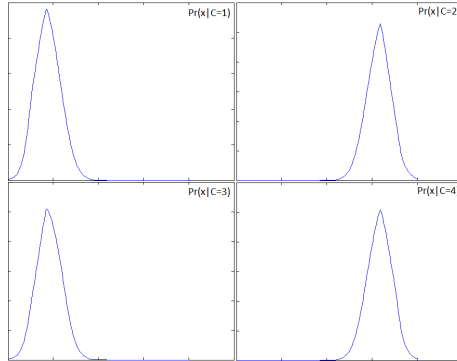


Figura 23 – $P(x|C)$ estimadas através da multiplicação das $\hat{f}(x \in C)$ pelas $\mu(x)$

para as classes do problema, devido ao algoritmo de *clusterização* não-supervisionado. A partir deste momento, o treinamento é concluído e o sistema está pronto para receber novas entradas.

Tabela 1 – Matriz de pesos da camada de saída

	ω			
1	0	0	0	0
0	1	0	0	0
0	0	0	0	1
0	0	0	1	0

3.9.5 Discussão

Observe a rede bayesiana *naïve* demonstrada na Figura 24. Esta rede foi obtida com o software *NETICA*[®], utilizando o método *expectation maximization* para aprender os parâmetros da rede através do conjunto de dados, discretizando as variáveis de entrada em 4 estados.

Os resultados das simulações de 6 pares (x, y) de entrada na rede PFNB e na rede tradicional, referentes aos pontos $[a, b, c, d, e, f]$ da Figura 19, são demonstrados nas Tabelas 2 e 3, respectivamente.

Comparando das duas tabelas, é possível fazer algumas observações em relação a diferença entre a rede proposta e a rede bayesiana

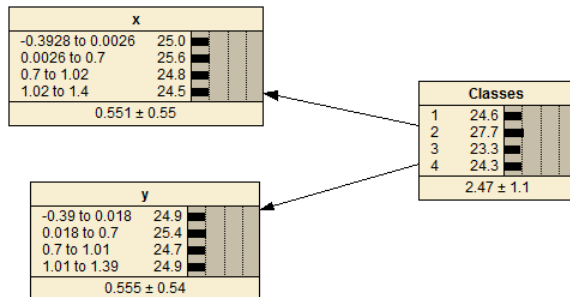


Figura 24 – Rede *naïve bayes* aprendida no software *NETICA*[®] para o *Simple Class*

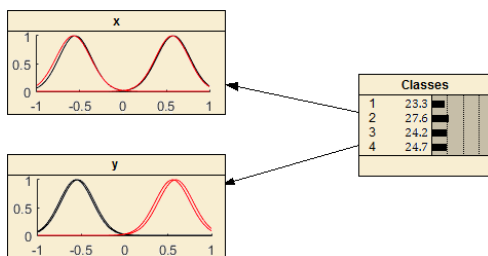


Figura 25 – Representação *naïve bayes* da rede PFNB treinada para o *Simple Class*

naïve tradicional.

Os pontos $[a, b, c, d]$ estão localizados aproximadamente na metade da distância entre o centro de cada *cluster*. A rede PFNB foi capaz de identificar este comportamento e classificou-os com porcentagens próximas de 50% em relação a seus respectivos *clusters* mais próximos. A rede *naïve bayes* tradicional não foi capaz de modelar este comportamento com acurácia, apresentando, em todos os casos, probabilidades acima de 84% em um só *cluster*.

O ponto *e* está localizado aproximadamente no centro do conjunto de dados. A rede PFNB distribuiu as probabilidades de maneira uniforme, levando em consideração que existem *clusters* com uma densidade maior de pontos próximo do centro. A rede tradicional classificou o ponto *e* com maior probabilidade de pertencer ao *cluster* 4.

Por último, analisando a Figura 19, é possível observar que o

Tabela 2 – Saída da rede PFNB para os pontos da Figura 19

	a	b	c	d	e	f
1	0,582	0,514	0	0	0,325	0
2	0	0,486	0,422	0	0,205	0
3	0	0	0,578	0,467	0,267	0,024
4	0,418	0	0	0,533	0,203	0,976

Tabela 3 – Saída de uma rede *naïve bayes* para os pontos da Figura 19

	a	b	c	d	e	f
1	0,146	0,926	0	0	0,103	0,103
2	0	0,074	0,090	0	0,004	0,004
3	0,001	0	0,910	0,059	0,051	0,051
4	0,854	0,001	0,001	0,941	0,841	0,841

ponto f apresenta uma probabilidade maior de pertencer ao *cluster* 4 do que o ponto e , e a saída da rede PFNB sugere exatamente este comportamento. Quando simulamos o ponto f na rede bayesiana *naïve* tradicional, observa-se que os pontos e e f são tratados como se tivessem as mesmas probabilidades, devido a suas coordenadas serem classificadas no mesmo intervalo discreto da rede.

Em redes *naïve bayes* tradicionais, a probabilidade de uma amostra (x, y) de pertencer a uma dada classe é igual a probabilidade de outra amostra $(x + \Delta_x, y + \Delta_y)$, enquanto ambas estiverem dentro do mesmo intervalo discreto de x e y obtidos durante a discretização das variáveis. Ao contrário das redes tradicionais, a rede PFNB também leva em consideração a imprecisão presente na caracterização dos pontos em relação a seus graus de pertinência aos *clusters*. Neste caso, caracterizada pela distância de cada amostra ao centro de cada *cluster*. Isto torna a saída do sistema mais precisa para Δ_x e Δ_y menores, de forma que as fronteiras entre os *clusters* são mais suaves.

É possível perceber a relação entre os conceitos de probabilidade e pertinência na diferença de resultados entre o ponto e e f . Ao combinar as duas informações, a maior precisão da caracterização dos pontos na rede PFNB em relação a seus *clusters* demonstra o caráter complementar dos dois tipos de incerteza (MEGHADADI; AKBARZADEH-T, 2001). Além disso, a Figura 21 sugere que a rede faz uma estimativa bem precisa da função de densidade dos eixos x e y através de histogramas *fuzzy*, quando comparada aos histogramas tradicionais.

Como demonstrado na seção 3.9.1, a rede PFNB gera 8 funções

de pertinência para modelar os 4 *clusters* do *Simple Class*. Isto resulta em uma redundância, gerando FPs com médias e variâncias muito próximas, demonstradas em pares como “Baixa” e “Alta” na Figura 20. Utilizando estes nomes para os pares de FPs, é possível interpretar a rede com regras *fuzzy* como exemplificadas abaixo:

Se x = Baixo e y = Alto, então C = 1

Se x = Alto e y = Alto, então C = 2

Se x = Alto e y = Baixo, então C = 3

Se x = Baixo e y = Baixo, então C = 4

Isto auxilia na geração de conhecimento acerca do conjunto de dados e na interpretação da rede, após seu treinamento.

Neste capítulo foi detalhada a rede probabilística *fuzzy naïve-bayes*, demonstrando seus componentes principais, o mecanismo de inferência e o processo de treinamento, assim como um exemplo didático com o objetivo de esclarecer as diferenças para com uma rede *naïve bayes* tradicional. No próximo capítulo serão descritos os experimentos realizados com a finalidade de comparar a performance da rede com sistemas semelhantes.

4 EXPERIMENTOS E RESULTADOS

Para validar o modelo proposto, experimentos foram realizados em cinco conjuntos de dados de *benchmark*: *Chemical*, *Iris*, *Glass*, *Breast Cancer* e *Wine*, conforme a Tabela 4. Estes conjuntos de dados são utilizados para medir a performance de diferentes classificadores e algoritmos e, exceto pelo *Chemical*, estão disponíveis no repositório de aprendizado de máquina UCI (LICHMAN, 2013). O *Chemical* é distribuído junto com o software *MATLAB*[®]. Os conjuntos de dados possuem dimensionalidades e classes de saída variadas, além de diferentes quantidades de amostras, de forma a testar o aprendizado da rede em diferentes condições (JIANG; WANG, 2010). Além disso, em alguns casos, as amostras dos conjuntos foram embaralhadas, para evitar que a rede seja exposta às amostras de forma sequencial, favorecendo uma classe específica e prejudicando as estimativas.

Tabela 4 – Visão geral dos conjuntos de dados aplicados na validação

Dados	Qtd. de amostras	Atributos	Classes
Chemical	498	8	3
Iris	150	4	3
Glass	214	9	6
Breast Cancer	699	9	2
Wine	178	13	3

A performance das redes PFNB foi comparada com classificadores baseados em redes neurais (MLP, *Multilayer Perceptrons*), sistemas fuzzy neurais (ANFIS, *Artificial Neural Fuzzy Inference System*) e redes neurais de base radial (RBF, *Radial Basis Function*), utilizando o método de validação *k-fold*.

Este método consiste em dividir o conjunto de dados em k subconjuntos mutuamente exclusivos, utilizando $k - 1$ subconjuntos para o treinamento e o restante para a validação. Isto é repetido até que todos os conjuntos foram utilizados uma vez para a validação, e assim obtém-se a média e desvio padrão de acertos. O método *k-fold* avalia a capacidade de generalização do classificador, ou seja, a sua acurácia para amostras não vistas no treinamento (ABONYI; SZEIFERT, 2013). Cada modelo realiza esta análise cinco vezes, com um $k = 10$, possibilitando o cálculo do desvio padrão da acurácia, para verificar a constância dos resultados. Os valores b e C utilizados na rede PFNB

(seção 3.7) foram encontrados através de tentativa e erro.

Os experimentos foram realizados dentro do ambiente MATLAB[®]. Os classificadores foram escolhidos de forma a representar uma variedade distinta de métodos bem conhecidos e documentados (WANG; FU, 2006). As seguintes seções apresentarão os detalhes de cada experimento, bem como um pequeno resumo sobre o conjunto de dados utilizado e uma comparação com modelos similares, propostos por outros autores.

4.1 CHEMICAL

O conjunto de dados *Chemical* consiste em dados retirados de 8 sensores químicos, e seu objetivo é prever o valor de um nono sensor. O experimento realizado com o *Chemical* é um pouco diferente dos demais experimentos, pois este trata-se de um problema de regressão. A saída deste conjunto são valores que variam de 485 a 527. A variável de saída foi discretizada nos intervalos [485, 500], [501, 510] e [511, 527], representando, a grosso modo, três classes: baixo, médio e alto. São dispostas 498 amostras. Os valores utilizados para encontrar a melhor rede foram $b = 12$ e $C = 9$, obtendo uma acurácia de 77,26% e desvio padrão de 4,79%.

Devido as alterações feitas no conjunto de dados, transformando-o em um problema de classificação, não foi possível comparar a performance da rede com trabalhos de outros autores. Um estudo comparativo realizado apenas com os modelos ANFIS e MLP pode ser visto na Tabela 5.

Tabela 5 – Estudo comparativo do *Chemical*

Modelo	Acurácia	σ
Rede PFNB	77,26%	0,047
ANFIS	66,00%	0,076
MLP	84,73%	0,006

Observe na Tabela 6 a relação entre a saída da rede PFNB e os valores alvo do *Chemical*:

Através da discretização da saída, é possível dar uma interpretação linguística para os valores possíveis do nono sensor. Desta forma, a rede possibilita que o usuário interprete o resultado em termos de probabilidade ao invés de valores absolutos. A rede PFNB classifica as amostras em relação ao quão próxima a mesma está das características

Tabela 6 – Valores de saída reais e probabilidades estimadas pela rede PFNB

Valor de Saída Real (%)								
Classe	489	494	498	503	507	511	517	523
<i>Baixo</i>	59,2	50	50,6	44,6	46,4	1,2	0,8	0
<i>Médio</i>	40,7	49,9	49,4	53,2	53,5	44,8	43,7	23,5
<i>Alto</i>	0	0	0	2,1	0	53,9	55,3	76,4

de cada classe. Por exemplo, a rede apresenta valores de probabilidade muito próximos para os valores 498 e 503, pertencentes ao final do intervalo da classe *Baixa* e início da *Média*. A medida que a amostra se torna menos característica de determinada classe, como, por exemplo, os valores 517 e 523, a rede apresentará mais certeza na classificação, com valores maiores de probabilidade.

Como demonstrado na Tabela 5, as redes MLP obtiveram o melhor resultado no experimento, mas nota-se a vantagem qualitativa da rede PFNB em relação a como este resultado é exibido.

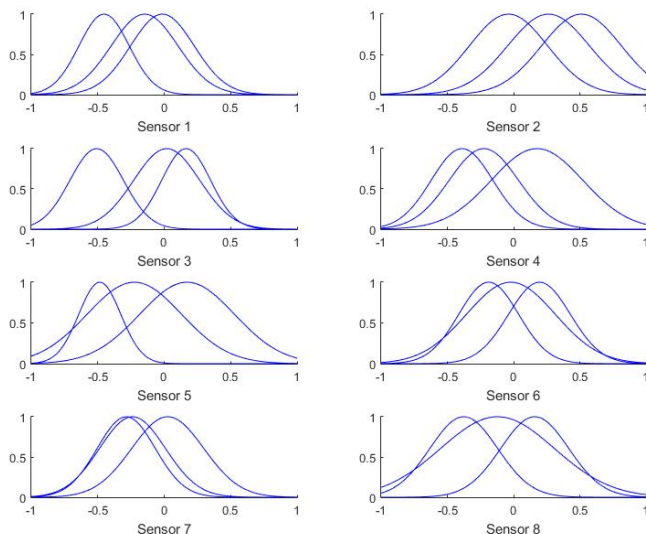


Figura 26 – Funções de pertinência extraídas com o FCM do *Chemical*

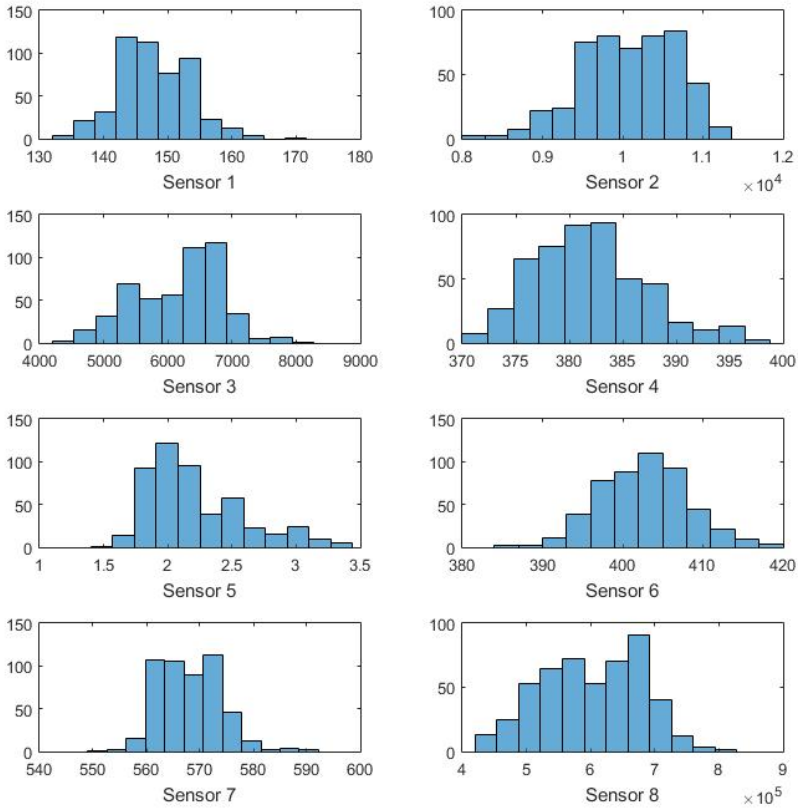


Figura 27 – Histogramas das variáveis do conjunto *Chemical*

4.2 IRIS

O conjunto de dados *Iris* foi criado por Fisher (1936) e é um dos conjuntos de classificação mais populares, sendo utilizado até hoje para medir a performance de classificadores (LICHMAN, 2013). O objetivo é classificar plantas de iris a partir da observação de suas flores, em três espécies diferentes: *Iris setosa*, *Iris versicolor* e *Iris virginica*. As 150 amostras são divididas igualmente entre as espécies, contendo 4 atributos sobre as flores: largura e comprimento de pétala e sépala. Os valores utilizados que resultaram na melhor acurácia foram $b = 12$ e $C = 6$, obtendo uma acurácia de 95,33% e desvio padrão de 5,49%

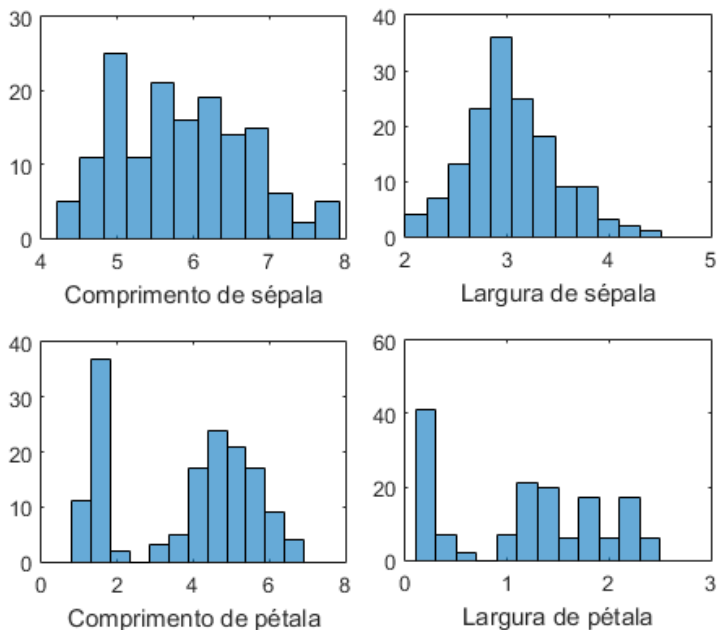
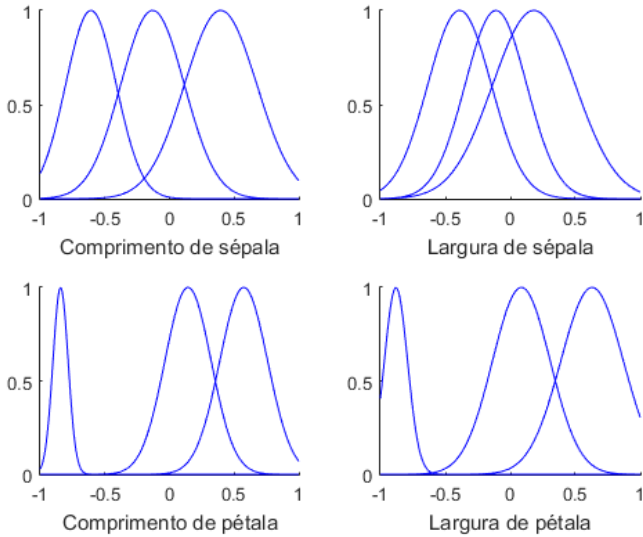


Figura 28 – Histogramas das variáveis do conjunto *Iris*

Tabela 7 – Estudo comparativo do *Iris*

Modelo	Acurácia	σ
Rede PFNB	95,33%	0,054
ANFIS	97,33%	0,046
RBF	96,7%	0,047
MLP	96,0%	0,034
ProFNN (HENGJIE et al., 2011)	97,3%	-
HFRBC-GA (DANDREA; LAZZERINI, 2013)	100%	-
Fuzzy DT (ABONYI et al., 2003)	96,11%	-
UFSBOC (JIANG; WANG, 2010)	95,29%	-
<i>Swarm</i> RCE+ (YUWONO et al., 2012)	91,6%	-

Figura 29 – Funções de pertinência extraídas com o FCM do *Iris*

4.3 GLASS

O conjunto de dados *Glass* consiste em 214 amostras com 10 atributos cada. Seu objetivo é classificar vidros, indicando se estes são planos (*float*) ou não, de veículos ou edifícios, entre outros. Ao total, os dados são classificados em 6 possíveis classes. Vidros *float* passam por um processo diferente de produção, onde este é fundido e derramado continuamente em alguma espécie de metal derretido, de forma que se espalhe uniformemente pela solução (PILKINGTON, 1969). Desta forma, é possível diferenciá-los através de atributos como seu índice de refração, quantidade de sódio, magnésio, alumínio e outros elementos. Essa informação é utilizada em investigações criminológicas, pois o vidro encontrado na cena do crime pode oferecer pistas caso identificado corretamente (LICHMAN, 2013). A rede que obteve melhor performance recebeu como parâmetro $C = 2$ e $b = 8$, com 94% de acurácia e um desvio padrão de 5,73%.

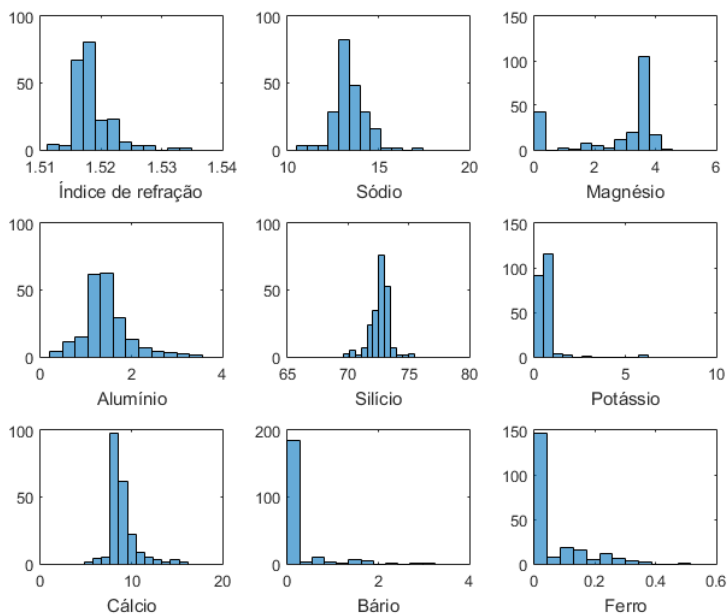
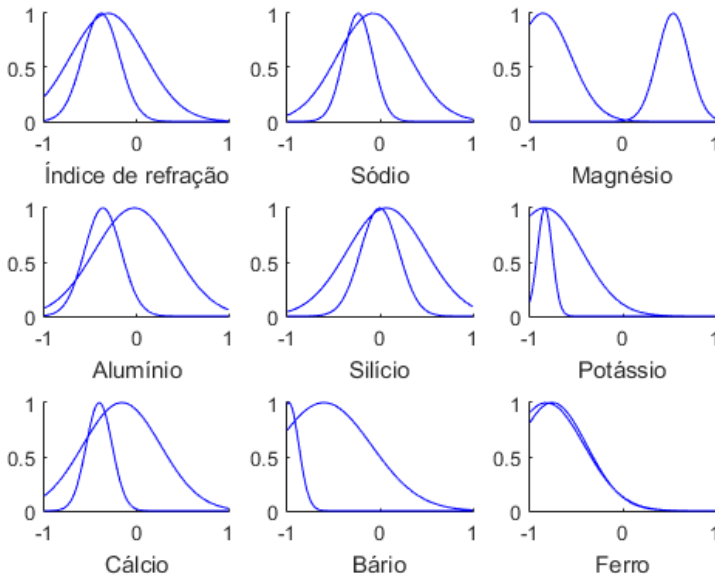


Figura 30 – Histogramas das variáveis do conjunto *Glass*

Tabela 8 – Estudo comparativo do *Glass*

Modelo	Acurácia	σ
Rede PFNB	94,00%	0,057
ANFIS	93,60%	0,054
RBF	92,40%	0,047
MLP	94,40%	0,054
Ramp-MCSVM (PHOUNGPOL et al., 2012)	97,9%	-
UFSBOC (JIANG; WANG, 2010)	94,12%	-
COB-GA (SITE; MISHRA, 2013)	96,35%	-
MNGA (TARIQUE; ZAMEE; KHAN, 2014)	69,82%	-
Swarm RCE+ (YUWONO et al., 2012)	90,3%	-

Figura 31 – Funções de pertinência extraídas com o FCM do *Glass*

4.4 BREAST CANCER

O objetivo do *Breast Cancer* é classificar amostras de câncer de mama, separando-os em maligno e benigno, de acordo com dados adquiridos através de biópsias realizadas ao longo de três anos (LICHMAN, 2013). O conjunto de dados contém 9 atributos, como a espessura da massa celular, uniformidade do tamanho da célula, entre outros, com 699 amostras. O melhor resultado foi obtido quando $b = 8$ e C igual ao número de classes do problema, com uma acurácia de 97,44% e desvio padrão 1,48%.

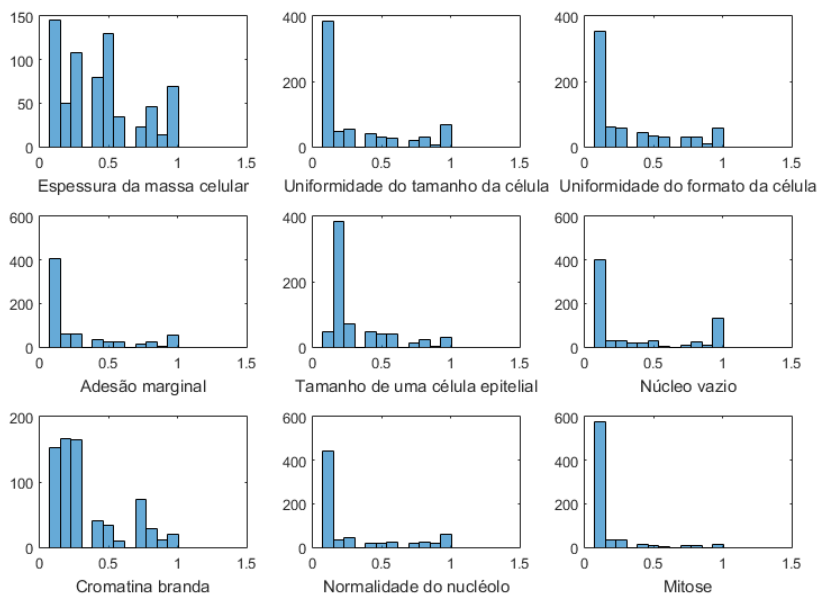
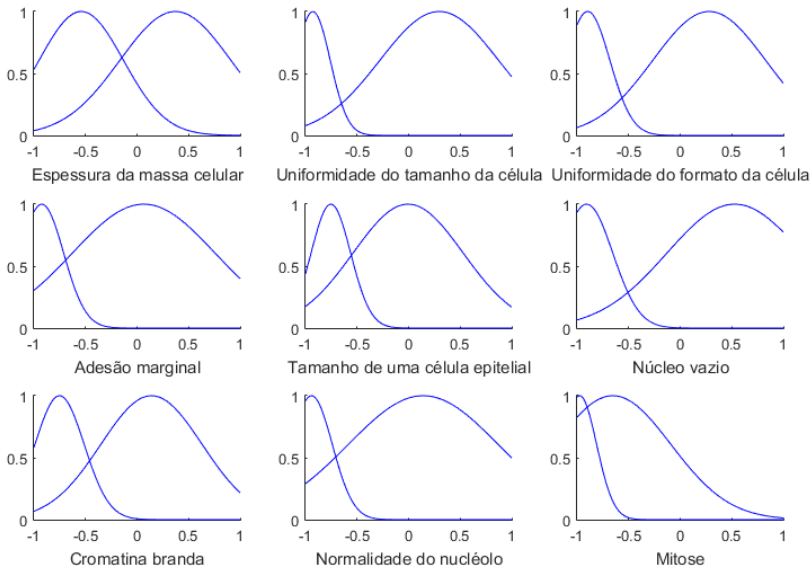


Figura 32 – Histogramas das variáveis do conjunto *Breast Cancer*

Tabela 9 – Estudo comparativo do *Breast Cancer*

Modelo	Acurácia	σ
Rede PFNB	97,44%	0,014
ANFIS	96,28%	0,024
RBF	97,05%	0,012
MLP	95,26%	0,031
<i>SFC</i> (ABONYI; SZEIFERT, 2013)	95,70%	-
Sistema Fuzzy Probabilístico (WALTMAN, 2005)	96,30%	-
HFRBC-GA (DANDREA; LAZZERINI, 2013)	98,32%	-
Fuzzy DT (ABONYI et al., 2003)	96,82%	-
UFSBOC (JIANG; WANG, 2010)	95,34%	-
MNGA (TARIQUE; ZAMEE; KHAN, 2014)	97,58%	-
<i>Swarm</i> RCE+ (YUWONO et al., 2012)	95,89%	-

Figura 33 – Funções de pertinência extraídas com o FCM do *Breast Cancer*

4.5 WINE

O conjunto de dados *Wine* consiste no resultado de uma análise química de vinhos italianos de três diferentes cultivos. Seu objetivo é utilizar esta análise, que consiste de 178 amostras com 13 atributos, e classificar o vinho de acordo com sua vinícola correspondente. A rede treinada com $C = 6$ e $b = 4$ teve a maior precisão, com 96,8% de acurácia e um desvio padrão de 3,10%.

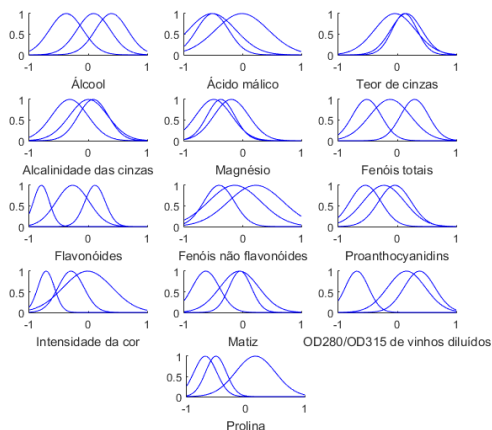


Figura 34 – Funções de pertinência extraídas com o FCM do *Wine*

Tabela 10 – Estudo comparativo do *Wine*

Modelo	Acurácia	σ
Rede PFNB	96,8%	0,031
ANFIS	93,6%	0,051
RBF	94,7%	0,075
MLP	97,2%	0,046
<i>SFC</i> (ABONYI; SZEIFERT, 2013)	97,7%	-
Fuzzy Probabilístico (WALTMAN, 2005)	97,7%	-
<i>Fuzzy Learning</i> (ROUBOS et al., 2001)	99,4%	-
HFRBC-GA (DANDREA; LAZZERINI, 2013)	99,41%	-
Fuzzy DT (ABONYI et al., 2003)	99,41%	-
UFSBOC (JIANG; WANG, 2010)	92,3%	-
<i>Swarm RCE+</i> (YUWONO et al., 2012)	68%	-

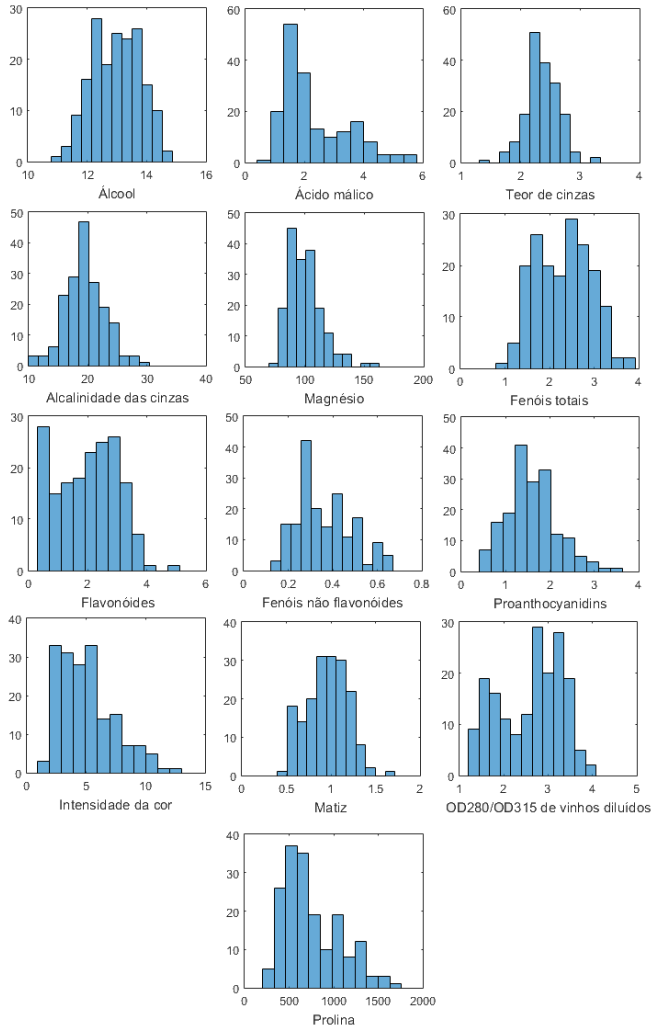


Figura 35 – Histogramas das variáveis do conjunto *Wine*

4.6 DISCUSSÃO

A partir dos resultados dos experimentos, algumas observações podem ser feitas acerca da rede PFNB. A Tabela 11 mostra os valores dos parâmetros C e b onde foram obtidas as melhores acurácias e a Figura 36 demonstra a relação entre a acurácia e os parâmetros.

Tabela 11 – Média da acurácia em por cento da rede PFNB para cada conjunto de dados e variações dos valores b e C

Dados	C	b = 4		b = 8		b = 12		b = 20	
		média	σ	média	σ	média	σ	média	σ
Chemical	9	76,3	6,1	75,9	5,2	77,2	4,7	73,5	5,9
Iris	3	88,6	7,7	92	6,1	90,6	5,6	87,3	4,9
Glass	2	93,6	5	94	5,7	91,2	5,5	91,6	3,5
B. Cancer	2	97,3	2,2	97,4	1,4	97,1	1,6	97,1	1,7
Wine	3	96,8	4,1	95,6	2,2	94,8	3,2	89,2	5,6
Chemical	12	75,7	5,1	75,7	5,8	72,9	6,2	71,8	6,7
Iris	6	86	5,8	94,6	5,2	95,3	5,4	94	8,5
Glass	4	93,2	5,6	92,8	4,1	92,4	7,6	90,4	5,4
B. Cancer	4	97,3	1,4	97,4	2,1	96,9	1,5	96	2,2
Wine	6	96,8	3,1	95,6	3,9	90,8	8	84	5,9

O *Breast Cancer* foi o conjunto de dados que apresentou o melhor resultado no geral, incluindo seus desvios padrões, possivelmente devido a maior quantidade de amostras disponíveis para o treinamento e o menor número de classes (apenas duas). Este conjunto de dados também obteve o melhor resultado no estudo comparativo, apresentando resultados muito próximos em relação aos outros modelos semelhantes encontrados na literatura. Os conjuntos de dados *Iris* e *Glass* apresentaram os maiores desvios padrões, indicando que estes conjuntos são os que apresentam maior variabilidade na acurácia, ou seja, ocorrem resultados menos consistentes no treinamento da rede. No caso do conjunto *Glass*, isto deve ser em razão da maior quantidade de classes a serem identificadas.

Na maioria dos casos, a rede sofreu sobreajuste (*overfitting*) com valores mais altos de b . Por exemplo, na variável *Espessura da massa celular* do conjunto de dados *Breast Cancer*, como ilustra a Figura 37, o treinamento com $b = 20$ produz uma estimativa de função de den-

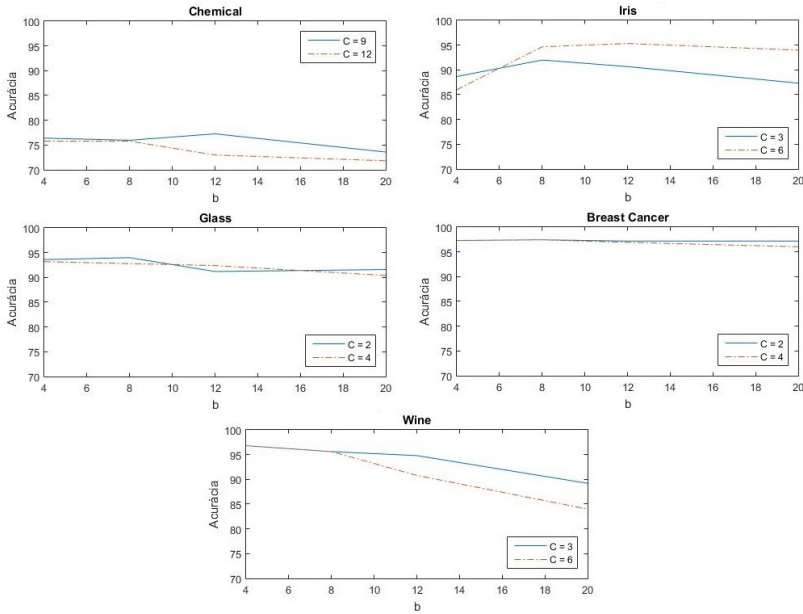


Figura 36 – Relação entre acurácia e os parâmetros de treinamento da rede para cada conjunto de dados

tidade *fuzzy* com variações muito acentuadas, enquanto $b = 4$ torna a estimativa mais suave. O conjunto *Wine* sofreu a maior perda em acurácia ao aumentar os valores de b e C , e isto pode indicar que o *overfitting* ocorre de maneira mais expressiva em conjuntos de dados com muitas dimensões. Por outro lado, o conjunto de dados *Iris* não teve um bom desempenho com valores baixos de b e C , demonstrando que o caso contrário também pode acontecer. O *Iris* foi o conjunto de mais difícil treinamento para a rede, devido ao menor número de amostras e atributos. Isso implica que aumentar a precisão das estimativas com valores mais altos de b e C pode melhorar a performance para conjuntos de dados com menos informações disponíveis. Como evidenciado na Figura 36, o *Breast Cancer* sofreu pouca ou nenhuma alteração em performance em todos os valores de b e C e apresenta uma acurácia consistente em torno de 97%.

Em alguns conjuntos de dados, como no *Wine* e *Iris*, utilizar valores de C maiores do que a quantidade de classes de saída aumenta

a performance da rede, indicando que estas classes são modeladas mais precisamente com mais do que um *cluster* gaussiano. Ambos apresentaram seu maior desempenho quando C foi especificado como o dobro da quantidade de classes dos dados. Por outro lado, o conjunto *Glass* foi treinado com $C = 2$ e $C = 4$, apesar de apresentar 6 classes de saída. Valores mais altos de C geravam *clusters* com poucos dados, dificultando a estimação das funções de densidade. Foram realizados, também, experimentos com $C = 3$, apresentando 91,12% de acurácia e 5,84% de desvio padrão em seu melhor resultado, com $b = 8$.

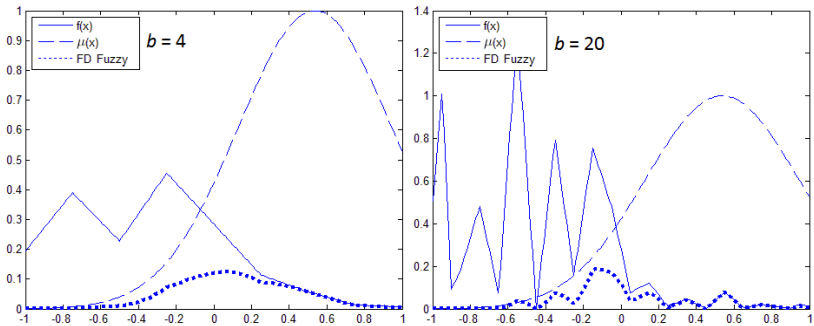


Figura 37 – Comparação entre as funções de densidade aproximadas por histogramas *fuzzy* com $b = 4$ e $b = 20$

Outra observação pode ser feita em casos como a variável *Ferro* do conjunto *Glass*, a variável *Teor de cinzas* do conjunto *Wine* ou no *Sensor 7* do conjunto *Chemical*, nas Figuras 31, 34 e 26: assim como no *Simple Class*, na seção 3.9.1, nota-se uma redundância, pois algumas FPs possuem médias e variâncias muito próximas. Estas poderiam ser combinadas em somente uma FP, aumentando ainda mais a interpretabilidade e diminuindo o custo computacional da rede. Isto mostra que relacionar os valores de C com a quantidade de classes do conjunto de dados pode não ser a maneira mais eficiente de definir o melhor valor.

No geral, a rede apresenta resultados similares a outros sistemas encontrados na literatura, mas é importante observar que apenas os métodos tradicionais (ANFIS, redes RBF e redes MLP) foram validados utilizando exatamente o mesmo método de validação (*k-fold*). Resultados como o do modelo MNGA (TARIQUE; ZAMEE; KHAN, 2014), que obteve 97,58% e 69,82% de acurácia no treinamento dos conjuntos *Breast Cancer* e *Glass*, respectivamente, mostram que a rede PFNB apresenta consistência nos resultados diante de conjuntos de dados com

características variadas. Isto também acontece com o modelo *Swarm RCE+* (YUWONO et al., 2012), que atinge resultados consistentes acima de 90% em todos os conjuntos de dados, exceto no *Wine*, onde apresenta uma acurácia de apenas 68%.

5 CONCLUSÃO

Neste trabalho, a rede probabilística *fuzzy naïve-bayes* foi proposta, um novo classificador com o objetivo de unir as melhores características de sistemas *fuzzy* probabilísticos e redes bayesianas *naïve*. Desta forma, foi possível modelar a imprecisão linguística e a incerteza estocástica simultaneamente, mantendo o alto nível de interpretabilidade encontrado em sistemas *fuzzy* e redes *naïve bayes* tradicionais. A rede PFNB é capaz de aprender e representar conhecimento acerca de conjuntos de dados e ser interpretada tanto como um sistema *fuzzy* quanto como uma rede bayesiana *naïve*. Isto pode ser considerado uma vantagem em relação aos modelos comparados, pois a rede apresentou resultados menos precisos em alguns casos, por uma pequena margem. Esta interpretabilidade, quando aliada a capacidade de aprendizagem da rede, possibilita a geração de conhecimento acerca da informação contida nos dados de treinamento. Além disso, como pode ser visto através da utilização do método *k-fold* de validação, a rede apresenta boa capacidade de generalização.

A quantidade de bins *fuzzy* b (para os histogramas *fuzzy*) e a quantidade de *clusters* C (para o FCM), parâmetros utilizados no treinamento da rede, precisaram ser especificados manualmente para cada conjunto de dados. Neste trabalho, os melhores valores foram encontrados através de tentativa e erro, e um método de otimização para defini-los automaticamente deveria ser investigado. Ainda, outros métodos de aprendizagem das funções de pertinência, assim como outros tipos, devem ser experimentados. Por exemplo, um algoritmo de clusterização *fuzzy* supervisionado pode aumentar a performance quando comparado ao FCM (ABONYI; SZEIFERT, 2013). Outra observação que pode ser feita a partir dos resultados do capítulo 3 e 4, diz respeito a redundância encontrada em algumas das funções de pertinência geradas. Um método para eliminar esta redundância teria de ser realizado em cada dimensão de entrada, pois esta nem sempre está presente em todas as dimensões. Isto tenderia a elevar a performance e a interpretabilidade da rede PFNB.

Os conjuntos de dados foram escolhidos com vista em facilitar a pesquisa de resultados de sistemas de outros autores. Porém, ainda houveram dificuldades em encontrar trabalhos que executassem os experimentos de maneira em que pudesse ser realizada uma comparação mais direta. Entender os conceitos acerca de sistemas *fuzzy* probabilísticos também se provou um desafio, devido a diferentes definições

dadas por vários autores. Por isto, os trabalhos de Kaymak e Berg (2013) foram tomados como base para que houvesse padronização dos conceitos utilizados para a implementação da rede PFNB.

A rede PFNB foi implementada apenas como prova de conceito, e não há otimização do código. Não houve estudo comparativo da performance computacional da rede em relação aos outros métodos, mas notou-se empiricamente que esta possui performance inferior aos modelos tradicionais, mais consolidados e otimizados. A reescrita do código utilizando técnicas de otimização ou heurísticas será necessária para o aprendizado de conjuntos de dados muito grandes. Também não houve discussão acerca da memória utilizada na execução e no armazenamento da rede treinada.

Como trabalho futuro, a rede poderia sofrer adaptações de forma a implementar qualquer estrutura de redes bayesianas, e não apenas a estrutura *naïve*. A implementação de uma interface gráfica facilitaria a visualização da rede, dando ênfase a esta vantagem qualitativa em relação a outros modelos. Além disso, a rede foi aplicada apenas a problemas de classificação. O modelo pode ser estendido para problemas de regressão, apresentando na saída do sistema a distribuição de probabilidade como um intervalo de confiança da predição.

REFERÊNCIAS

- ABONYI, J. et al. Data-driven generation of compact, accurate, and linguistically sound fuzzy classifiers based on a decision-tree initialization. *International Journal of Approximate Reasoning*, 2003.
- ABONYI, J.; SZEIFERT, F. Supervised fuzzy clustering for the identification of fuzzy classifiers. *Pattern Recognition Letters* 24, 2013.
- ALMEIDA, R. J. et al. Probabilistic fuzzy systems as additive fuzzy systems. *Information Processing and Management of Uncertainty in Knowledge-Based Systems*, 2014.
- BARRICO, C. M. C. S. *Optimização Evolucionária Multi-Objectivo em Ambientes Incertos Pesquisa de Soluções Robustas*. Tese (Doutorado) — Departamento de Engenharia Electrotécnica de Computadores, Universidade de Coimbra, 2007.
- BERG, J. V. den; BERGH, W. V. den; KAYMAK, U. Probabilistic and statistical fuzzy set foundations of competitive exception learning. *ERIM Report Series Reference No. ERS-2001-40-LIS*, 2001.
- BISWAS, R.; NAAZ, S.; ALAM, A. Effect of different defuzzification methods in a fuzzy based load balancing application. *IJCSI International Journal of Computer Science Issues*, 2011.
- BRIGNOLI, J. T. et al. A fuzzy-bayesian model based on the superposition of states applied to the clinical reasoning support. *SAI Intelligent Systems Conference*, 2015.
- BUSTAMANTE, C.; GARRIDO, L.; SOTO, R. Fuzzy naive bayesian classification in robosoccer 3d: A hybrid approach to decision making. *RoboCup 2006: Robot Soccer World Cup X*, 2007.
- CHATTOPADHYAY, S.; PRATIHAR, D. K.; SARKAR, S. C. de. A comparative study of fuzzy c-means algorithm and entropy-based fuzzy clustering algorithms. *Computing and Informatics*, 2011.
- DANDREA, E.; LAZZERINI, B. A hierarchical approach to multi-class fuzzy classifiers. *Expert Systems with Applications*, 2013.
- DAS, B. Generating conditional probabilities for bayesian networks: Easing the knowledge acquisition problem. *Computing Research Repository (CoRR)*, 2004. <<http://arxiv.org/abs/cs.AI/0411034>>.

- DUNN, J. C. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. *J. Cybernet*, 1973.
- FISHER, R. A. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, Blackwell Publishing Ltd, v. 7, n. 2, p. 179–188, 1936.
- HENGJIE, S. et al. A probabilistic fuzzy approach to modeling nonlinear systems. *Neurocomputing* 74, 2011.
- JENSEN, F. V.; NIELSEN, T. D. *Bayesian Networks and Decision Graphs*. [S.l.]: Springer, 2007.
- JIANG, S.; WANG, L. Unsupervised feature selection based on clustering. *2010 IEEE Fifth International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA)*, IEEE, 2010.
- KATARZYNA, R.; WALASZEK-BABISZEWSKA, A. Probabilistic-fuzzy knowledge-based system for managerial applications. *Management and Production Engineering Review*, 2011.
- KAUR, A.; KAUR, A. Comparison of mamdani-type and sugeno-type fuzzy inference systems for air conditioning system. *International Journal of Soft Computing and Engineering (IJSCE)*, 2012.
- KAYMAK, U.; BERG, J. van den; BERGH, W.-M. van den. Fuzzy classification using probability-based rule weighting. *IEEE*, 2002.
- KAYMAK, U.; BERG, J. van der. Conditional density estimation using probabilistic fuzzy systems. *IEEE Transactions on Fuzzy Systems*, 2013.
- KORB, K. B.; NICHOLSON, A. E. *Bayesian Artificial Intelligence*. [S.l.]: CRC Press, 2011.
- LI, H.-X.; LIU, Z. Probabilistic fuzzy logic system: a tool to process stochastic and imprecise information. *FUZZ-IEEE*, 2009.
- LICHMAN, M. UCI machine learning repository [<http://archive.ics.uci.edu/ml>]. *University of California, Irvine, School of Information and Computer Sciences*, 2013. <<http://archive.ics.uci.edu/ml>>.
- LIMA, M. D. C. *Método de Discretização de Variáveis para Redes Bayesianas utilizando Algoritmos Genéticos*. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, 2014.

MCCALLUM, A.; NIGAM, K. A comparison of event models for naive bayes text classification. *AAAI-98 workshop on learning for text categorization*, 1998.

MEGHDAZI, A. H.; AKBARZADEH-T, M.-R. Probabilistic fuzzy logic and probabilistic fuzzy systems. *IEEE International Fuzzy Systems Conference*, 2001.

MENDEL, J. M.; JOHN, R. I. B. Type-2 fuzzy sets made simple. *IEEE Transactions on Fuzzy Systems*, 2002.

MOURA, G.; ROISENBERG, M. Probabilistic fuzzy bayesian network. *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, p. 476–482, 2015.

MOURA, G.; ROISENBERG, M. Probabilistic fuzzy naive bayes. *2015 Brazilian Conference on Intelligent Systems (BRACIS)*, p. 246–251, 2015.

PAN, H. Fuzzy bayesian networks - a general formalism for representation, inference and learning with hybrid bayesian networks. *International Journal of Pattern Recognition and Artificial Intelligence*, 1999.

PEARL, J. Bayesian networks: A model of self-activated memory for evidential reasoning. *UCLA Proceedings of the 7th Conference of the Cognitive Science Society*, 1985.

PEDRYCZ, W.; GOMIDE, F. *Fuzzy Systems Engineering: Toward Human-Centric Computing*. [S.l.]: Wiley-IEEE Press, 2007. ISBN 0471788570, 9780471788577.

PENG-CHENG, L. et al. A fuzzy bayesian network approach to improve the quantification of organizational influences in hra frameworks. *Safety Science* 50, 2012.

PENZ, C. A. et al. Fuzzy-bayesian network for refrigeration compressor performance prediction and test time reduction. *Expert Systems with Applications*, 2012.

PHOUNGPOL, P. et al. Multiclass svm with ramp loss for imbalanced data classification. *2012 IEEE International Conference on Granular Computing*, IEEE, 2012.

PILKINGTON, A. Review lecture. the float glass process. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, The Royal Society, p. 1–25, 1969.

RENNIE, J. et al. Tackling the poor assumptions of naive bayes classifiers. *ICML*, 2003.

REZAEI, M. R. et al. Petrophysical data prediction from seismic attributes using committee fuzzy inference system. *Computers and Geosciences* 35, 2009.

ROUBOS, H. et al. *Developments in Soft Computing*. Norwell, MA, USA: Physica-Verlag HD, 2001. 108-115 p. ISBN 978-3-7908-1361-6.

RYHAJLO, N. F. *Fuzzy Bayesian Networks for Prognostics and Health Management*. Dissertação (Mestrado) — Montana State University, 2013.

SAGGAF, M. M.; NEBRIJA, E. L. A fuzzy logic approach for the estimation of facies from wire-line logs. *AAPG Bulletin*, 2003.

SITE, S.; MISHRA, S. K. Model for measuring accuracies of majority voting of ensemble classifier with cob and genetic algorithm. *2013 International Conference on Information Communication and Embedded Systems (ICICES)*, 2013.

TANG, H.; LIU, S. Basic theory of fuzzy bayesian networks and its application in machinery fault diagnosis. *Fourth International Conference on Fuzzy Systems and Knowledge Discovery*, 2007.

TARIQUE, T. A.; ZAMEE, M. A.; KHAN, M. I. A new approach for pattern recognition with neuro-genetic system using microbial genetic algorithm. *2013 International Conference on Electrical Engineering and Information Communication Technology (ICEEICT)*, 2014.

TIBIRICA, C. A. G. *Uma abordagem híbrida Fuzzy-bayesiana para modelagem de incertezas*. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, 2005.

WALTMAN, L. *A Theoretical Analysis of Probabilistic Fuzzy Systems*. Tese (Doutorado) — Erasmus University Rotterdam, 2005.

WANG, L.; FU, X. *Data Mining with Computational Intelligence*. Springer, 2006. (Advanced Information and Knowledge Processing). ISBN 9783540288039.
<<https://books.google.com.br/books?id=eeDYyp098s8C>>.

- YUWONO, M. et al. Optimization strategies for rapid centroid estimation. *34th Annual International Conference of the IEEE EMBS*, IEEE, 2012.
- ZADEH, L. A. Fuzzy sets: Information and control. *8*. 338-358, 1965.
- ZADEH, L. A. Probability measure of fuzzy events. *Journal of Mathematical Analysis and Applications*, 1968.
- ZHANG, G.; LI, H.-X. An efficient configuration for probabilistic fuzzy logic system. *IEEE Transactions on Fuzzy Systems*, 2012.
- ZHANG, G. et al. A hybrid approach to integrate fuzzy c-means based imputation method with genetic algorithm for missing traffic volume data estimation. *Transportation Research Part C*, 2014.