

Um Modelo Fuzzy-Genético para Problemas de Classificação

A.S. Koshiyama, T. Escovedo, D.M. Dias, M.M.B.R. Vellasco e R. Tanscheit

Departamento de Engenharia Elétrica
Pontifícia Universidade Católica do Rio de Janeiro

Rio de Janeiro, RJ, Brasil

{adriano, tatiana, douglasm, marley, ricardo}@ele.puc-rio.br

Resumo— Este trabalho apresenta um Sistema Fuzzy-Genético para Classificação (SFGC), chamado de *Genetic Programming Fuzzy for Classification* (GPF-CLASS). Este modelo difere da abordagem tradicional dos SFGC, que usa a meta-heurística como uma forma de aprender regras fuzzy do tipo "se-então". Essa abordagem clássica tem várias alterações e restrições sobre o uso de operadores genéticos, avaliação e seleção, que dependem principalmente da meta-heurística usada. A Programação Genética faz com que essa implementação se torne dispendiosa, além de explorar pouco suas próprias características e potencialidades. O modelo GPF-CLASS difere da abordagem convencional, visando a uma maior integração com a meta-heurística: Programação Genética Multigênica (PGMG). Foram realizados testes com 22 benchmarks para classificação, além de ter sido efetuadas comparações com demais SFGC propostos na literatura.

Palavras-chave— *Classificação; Sistemas Fuzzy-Genéticos; Programação Genética Multigênica.*

I. INTRODUÇÃO

Problemas de classificação são amplamente relatados na literatura [1]. Em geral, a associação de um determinado padrão a uma classe específica deve ser identificada, mas o problema pode tornar-se mais complexo, como em aprendizagem on-line ou em ambientes não estacionários, quando novas classes podem ser adicionadas, ou alteradas ao longo do tempo [2]. Redes Neurais [3], Máquina de Vetores de Suporte [4] e Programação Genética [5] tornam possível solucionar problemas de classificação com grande precisão. No entanto, uma alta precisão não é sempre acompanhada por uma interpretabilidade satisfatória, ou seja, pode ser difícil de identificar, em termos linguísticos, a relação entre os padrões de entrada e a classe de saída.

Sistemas de Inferência Fuzzy (SIF) constituem uma boa alternativa que alia relativa precisão e compreensão [6]: o sistema permite ao usuário identificar linguisticamente quais características fazem um padrão de entrada pertencer a uma classe específica. No entanto, em muitas situações um especialista, que pode não existir, é necessário para a modelagem e ajuste [7]. Geralmente, em problemas de classificação, uma base de dados é disponível, sendo possível efetuar uma aprendizagem supervisionada dos parâmetros do SIF. Uma maneira de realizar isto é através dos Sistemas Fuzzy Genéticos (SFG) [8].

Um SFG integra a busca global de um algoritmo evolutivo com a capacidade de inferência e compreensão

linguística de um SIF [9]. A meta-heurística é empregada apenas para encontrar os parâmetros do SIF e não como mecanismo que pode mudar sua estrutura de raciocínio ou funcionamento. Em geral, este é um aspecto positivo, mas em arquiteturas mais híbridas, em que a meta-heurística tem maior participação, a precisão do SFG pode se tornar mais acurada, além de possivelmente ocorrer variações na interpretabilidade interessantes ao usuário.

O objetivo deste trabalho é desenvolver um Sistema Fuzzy Genético para Classificação com uma arquitetura diferente daquela de um SIF tradicional. Este novo modelo é chamado *Genetic Programming Fuzzy for Classification* (GPF-CLASS), uma vez que faz uso da Programação Genética Multigênica [10]. O objetivo é obter uma estrutura mais híbrida para atingir alta precisão e interpretabilidade adequada.

Este trabalho está organizado da seguinte forma: seção 2 descreve alguns conceitos da meta-heurística usada no modelo GPF-CLASS: Programação Genética Multigênica [10]. A seção 3 apresenta o modelo e seus aspectos teóricos, enquanto que a seção 4 trata do estudo de caso. Seção 5 conclui o trabalho.

II. PROGRAMAÇÃO GENÉTICA MULTIGÊNICA

Programação Genética (PG) [5] é um método pertencente à Computação Evolutiva, biologicamente inspirado e que sintetiza programas de computador para desenvolver alguma tarefa. Tipicamente, a PG usa uma população de indivíduos, cada um deles denotado por uma estrutura em árvores onde é codificada uma equação matemática que descreve um vetor de saída Y a partir de um conjunto de J atributos de entradas X_j ($j=1, \dots, J$). Com base nessa estrutura realiza-se a PG tradicional. Já a Programação Genética Multigênica (PGMG) [10,11] denota um indivíduo como um complexo de estruturas em árvores (equações), que da mesma forma recebem X_j , buscando prever Y .

Cada resultado G_d ($d=1, \dots, D$) das D equações é uma solução parcial para o problema, sendo que a saída final \hat{Y} pode ser resultado de uma combinação linear dos G_d . A partir disto é possível avaliar a qualidade global do indivíduo. É fácil verificar que, quando $D=1$, a PGMG reduz-se à PG clássica.

Em relação aos operadores genéticos, a operação de mutação na PGMG é similar à efetuada na PG clássica. No caso da operação de cruzamento, é necessário efetuar uma

distinção no nível em que a operação é realizada: é possível aplicar o cruzamento no baixo e no alto nível. O baixo nível é o espaço onde é possível manipular as estruturas (terminais e funções) das equações presentes em um indivíduo. No caso, tanto a mutação quanto o cruzamento de baixo nível na PGMG são semelhantes ao que é efetuado na PG clássica.

Um exemplo de cruzamento de alto nível para um indivíduo com cinco equações é apresentado na Figura 1. O alto nível é o espaço em que se manipulam de forma macro as equações presentes no indivíduo. Logo, verifica-se que, a partir de dois pontos aleatórios, são permutadas equações de um indivíduo para o outro. Este ponto de corte aleatório pode ser simétrico - quando troca-se o mesmo número de equações entre indivíduos - ou assimétrico - caso contrário. A saída resultante é mais afetada pelo cruzamento de alto nível do que pela operação de cruzamento de baixo nível e pela mutação.

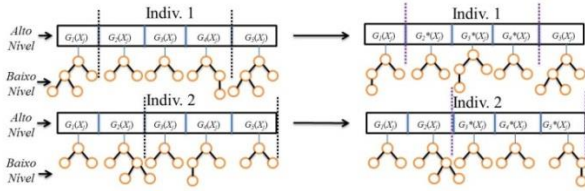


Fig. 1. Exemplo de cruzamento de alto-nível entre indivíduos.

A próxima seção aborda o modelo GPF-CLASS, que, a partir das bases apresentadas da PGMG, utiliza essa meta-heurística em consórcio com a Lógica Fuzzy para classificação, constituindo-se, assim, em um Sistema Fuzzy-Genético.

III. MODELO GPF-CLASS

O modelo GPF-CLASS é descrito por quatro etapas: fuzzificação, inferência, decisão e avaliação. O procedimento de estimação inicia-se com a fuzzificação dos atributos de entrada X_j . Depois disso, as funções de cada indivíduo são sintetizadas através da PGMG e é tomada uma decisão sobre qual classe pertence um padrão. Esta solução é avaliada e o processo é repetido até que seja alcançada uma solução que atende a um determinado critério. A Figura 2 ilustra o modelo GPF-CLASS.

A. Fuzzificação

A fuzzificação consiste em mapear valores precisos em graus de pertinência a conjuntos fuzzy [6]. Um conjunto fuzzy corresponde a um termo linguístico e é caracterizado por uma função de pertinência, que normalmente é definida por um especialista. Na falta deste, a opção comumente usada é dispor as funções de pertinência de forma normalizada e uniforme no universo de discurso de cada atributo [11,12,13] (ver Figura 3, por exemplo).

Seja A_{mj} o m-ésimo conjunto fuzzy para o j-ésimo atributo. Considere-se, para exemplificação, um caso em que $j=1,2,\dots,5$. É possível definir $A_{mj} = \{MP_j, P_j, M_j, G_j, MG_j\}$, tal que cada conjunto fuzzy manifeste linguisticamente, por exemplo, os conceitos muito pequeno (MP_j), pequeno (P_j), médio (M_j), grande (G_j) e muito grande (MG_j) para o j-ésimo atributo. Na presença de variáveis categóricas, sugere-se que

estas sejam inseridas na base de dados codificadas no formato 1 de N, para tornar o mais claro possível a aceção fornecida pelo modelo.

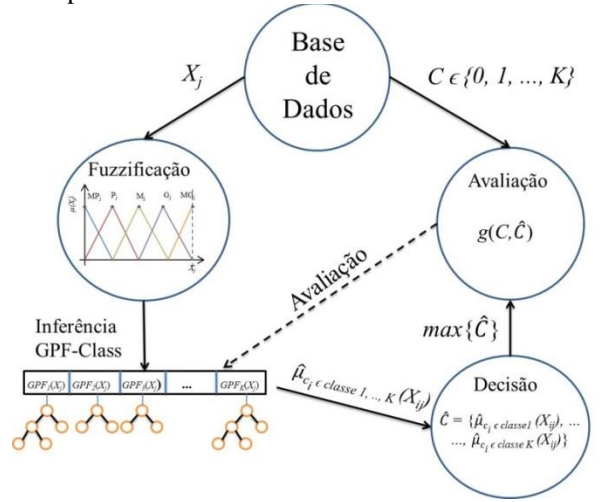


Fig. 2. Modelo GPF-CLASS.

B. Inferência GPF-CLASS e Decisão

O modelo GPF-CLASS busca estabelecer uma associação entre os conjuntos fuzzy dos atributos com a respectiva classe de saída. Em outras palavras, como e em que sentido os conceitos linguísticos de cada atributo está relacionado com uma determinada classe. Existem diferentes representações para problemas binários e de múltiplas classes. No caso de múltiplas classes, seja $C \in \{0,1,\dots,K\}$ em que K é o número de possíveis classes. GPF-CLASS pode ser descrito por:

$$C_{\epsilon 1} = f_1(\mu_{A_{mj}}(X_j)) + \epsilon_1 = \hat{\mu}_{C \in 1}(X_j) + \epsilon_1 \quad (2)$$

$$C_{\epsilon 2} = f_2(\mu_{A_{mj}}(X_j)) + \epsilon_2 = \hat{\mu}_{C \in 2}(X_j) + \epsilon_2 \quad (3)$$

...

$$C_{\epsilon K} = f_K(\mu_{A_{mj}}(X_{ij})) + \epsilon_K = \hat{\mu}_{C \in K}(X_{ij}) + \epsilon_K \quad (4)$$

onde $C_{\epsilon K}$ é um valor binário ($C_{\epsilon K} = 1$ quando C é classe K , e $C_{\epsilon K} = 0$ caso contrário). $\hat{\mu}_{C \in K}(X_j)$ é o grau de pertinência de X_j a classe K e ϵ_K é o erro de classificação do modelo. Os $f_k(\cdot)$ são funções a serem buscadas pela PGMG. Tem-se $\hat{\mu}_{C \in K}(X_j) \in [0,1]$, $\hat{C} = \{\hat{\mu}_{C \in 1}(X_{ij}), \hat{\mu}_{C \in 2}(X_{ij}), \dots, \hat{\mu}_{C \in K}(X_{ij})\}$. Um determinado padrão pertence à classe que $\max(\hat{C})$, ou seja, pertence a classe que tem maior compatibilidade. No caso de $\hat{\mu}_{C \in 1}(X_j) = \hat{\mu}_{C \in 2}(X_j) = \dots = \hat{\mu}_{C \in K}(X_j)$, a classe é indefinida ou alguma heurística pode ser usada para determinar a que classe este padrão qualquer pertence.

Em classificação binária, o GPF-CLASS é descrito por:

$$C_{\epsilon 1} = f_1(\mu_{A_{mj}}(X_j)) + \epsilon_1 = \hat{\mu}_{C \in 1}(X_j) + \epsilon_1 \quad (5)$$

onde $C_{\epsilon 1}$ é um valor binário, tal que $C_{\epsilon 1}$ é igual a 1 se o padrão pertence a classe 1, e 0 caso contrário; $\hat{\mu}_{C \in 1}(X_j)$ é o grau de pertinência do padrão X_j a classe 1 e ϵ_1 é o erro de

classificação do modelo. Como $\hat{\mu}_{C \in 1}(X_j) \in [0,1]$, então é possível definir $\hat{\mu}_{C \in 0}(X_j)$ como o grau de pertinência de um dado padrão a classe 0, como:

$$\hat{\mu}_{C \in 0}(X_j) = 1 - \hat{\mu}_{C \in 1}(X_j) \quad (6)$$

é a operação fuzzy complementar clássica para $\hat{\mu}_{C \in 1}(X_j)$. Considere, por exemplo, $\hat{C} = \{\hat{\mu}_{C \in 0}(X_{ij}), \hat{\mu}_{C \in 1}(X_{ij})\}$. De forma a definir a classe precisa para um dado padrão, $\max(\hat{C})$ deve ser encontrada. Se $\hat{\mu}_{C \in 1}(X_j) > \hat{\mu}_{C \in 0}(X_j)$, o resultado é classe 1 e vice-versa. Caso $\hat{\mu}_{C \in 1}(X_j) = \hat{\mu}_{C \in 0}(X_j)$, o procedimento mencionado para o caso de múltiplas classes pode ser aplicado.

A partir de \hat{C} estimado e a C observada, um indivíduo da população da PGMG é avaliado por:

$$ECMP = \frac{\sum |C - \hat{C}|}{n} * 100 \quad (7)$$

sendo ECMP o erro de classificação médio percentual e n o número total de padrões na base de dados. O indivíduo que melhor selecionar os $\mu_{A_{mj}}(X_j)$ será o que minimizar o ECMP.

Para estabelecer a função de ligação $f_k(\cdot)$ é necessário definir a forma funcional e escolher quais conjuntos fuzzy das variáveis de entrada devem figurar em cada mapeamento. Esta decisão é complexa, pois envolve escolhas adversas, entre as quais: Forma funcional: linear ou não linear; idêntico para todas as equações ou diferente em algumas; Seleção dos conjuntos fuzzy: quais, quanto e em quais mapeamentos $f_k(\cdot)$ eles devem ingressar.

Além destes dois fatores, deve-se ter em conta que uma função com muitos parâmetros e concebida de forma não linear torna a aceção do modelo mais obscura, apesar de possivelmente tornar a classificação mais acurada. Logo, deve-se buscar ao máximo a preservação da inteligibilidade linguística do modelo, acompanhada de um mínimo de perdas em precisão. Uma forma de possibilitar o alcance de ambos os objetivos é pelo uso da PGMG.

Uma vez que na PGMG um indivíduo representa um complexo de programas, é possível definir que cada indivíduo possui um número de programas necessário para produzir um número de saídas necessárias. No caso de cinco conjuntos fuzzy associados à variável de saída, é necessário manter cinco programas por indivíduo, com cruzamento de alto nível simétrico e com inicialização estrita de cinco programas por indivíduo.

Por exemplo, para a PGMG sintetizar a equação (5), considere-se o conjunto de terminais (funções de pertinência dos conjuntos fuzzy associados aos atributos de entrada e constantes) e funções mostrados na Tabela 1 e Figura 4.

TABELA 1. TERMINAIS E FUNÇÕES DO MODELO GPF-CLASS.

Terminais	Funções
$\mu_{MP_1}(X_j)$, $\mu_{P_1}(X_j)$, $\mu_{M_1}(X_j)$, $\mu_{G_1}(X_j)$ e $\mu_{MG_1}(X_j)$ e constantes no intervalo [a,b].	Soma, subtração, raiz quadrada, potência quadrática e produto tipo 1 e 2.

Do conjunto de funções, as operações de soma e subtração denotam o sentido em que variações expressas pelos termos linguísticos dos atributos de entrada implicam em variações positivas – afirmação, ou negativas – negação, no grau de compatibilidade a uma k-ésima classe. As operações de raiz quadrada e de potência quadrática funcionam como os intensificadores e redutores de Zadeh [6]. A função produto possui duas funcionalidades: Produto tipo 1: Associar uma constante a um $\mu_{A_{mj}}(X_j)$; Produto tipo 2: Combinar um $\mu_{A_{mj}}(X_j)$ com outro conjunto fuzzy de entrada.

Sem perda de generalidade, considere-se $\hat{\mu}_{C \in 1}(X_j) = 0,8 \mu_{MG_1}(X_j)$ como exemplo da primeira característica do produto. Então, se $\mu_{MG_1}(X_j) = 1$, segue-se que $\hat{\mu}_{C \in 1}(X_j) = 0,8$. Logo, a constante usada em conjunto com a função produto tipo 1 denota o nível da transmissão que a ativação específica de um conjunto fuzzy de entrada acarreta em graus de pertinência a uma determinada classe.

Para o segundo caso, seja a expressão abaixo:

$$\hat{\mu}_{C \in 1}(X_j) = \mu_{MP_1}(X_1)\mu_{P_1}(X_1) + \mu_{MP_1}(X_1)\mu_{G_1}(X_1) + \mu_{MG_1}(X_1)\mu_{M_2}(X_2) \quad (8)$$

A primeira parte de (8), $\mu_{MP_1}(X_1)\mu_{P_1}(X_1)$, considerando-se partições uniformes e normalizadas (veja exemplo na figura 4), pode ser interpretada linguisticamente como “aproximadamente entre muito pequeno e pequeno”. Na segunda parte da expressão, $\mu_{MP_1}(X_1)\mu_{G_1}(X_1) = 0$, pois os conjuntos não são adjacentes e, portanto o produto interno entre eles é nulo (ver Figura 3).

Para a terceira parte, $\mu_{MG_1}(X_1)\mu_{M_2}(X_2)$, apesar de possivelmente existir um X_1 e um X_2 tal que $\mu_{MG_1}(X_1)\mu_{M_2}(X_2) > 0$, a interpretação linguística é mais complexa. Pode-se entender como “quanto maior for a correspondência (ou proximidade) de X_1 a MG_1 e X_2 a M_2 , maior será o grau de pertinência desse padrão a classe 1”. De forma mais simplificada: “se X_1 é MG_1 e X_2 é M_2 então X é classe 1 com grau de pertinência estimado em $\mu_{MG_1}(X_1)\mu_{M_2}(X_2)$ ”. Em outras palavras, este segundo caso do produto modela a interação entre conceitos de diferentes variáveis, revelando inter-relações e peculiaridades do comportamento destas, que, dependendo da situação, podem ser significativas para o analista.

Por fim, pode ser verificado que em problemas de classificação binária a PGMG se torna a PG clássica, pois somente uma única equação precisa ser encontrada; a outra classe é definida complementarmente. No caso de múltiplas classes, é necessário que a PGMG use K funções decodificadas em um indivíduo. Também, a métrica ECMP (7) pode ser substituída por outra qualquer.

IV. ESTUDOS DE CASO

A. Base de dados e Procedimentos Experimentais

Para demonstrar o potencial do modelo GPF-CLASS foram efetuados alguns experimentos. Os resultados obtidos

foram comparados com os do modelo GP-COACH [13]. Este trabalho trata de um Sistema Fuzzy-Genético de Base de Regras para Classificação (SFGBRC) aplicada a problemas de alta dimensionalidade, empregando a PG com a base de regras fuzzy codificada através da formulação Genetic Competitive-Cooperative Learning (GCCL) [14] (uma regra por indivíduo). Além do GP-COACH, o GPF-CLASS foi comparado a outros modelos de classificação:

- PCA-RAVI [15]: obtém uma base de regras fuzzy para problemas de alta dimensionalidade, extraindo um subconjunto de atributos por análise de componentes principais;
- 2SLAVE [16]: usa um Algoritmo Genético para aprender uma base de regras fuzzy, codificada em um indivíduo pela abordagem Iterated Rule Learning (uma regra por indivíduo);
- GP-PITT-Tsakonas [17]: um SFGBRC que usa a Programação Genética Orientada a Gramática com o conceito de decodificação Pittsburgh (uma base de regras por indivíduo);
- GCCL-Ishibuchi [14]: um SFGBRC que usa um Algoritmo Genético, com tamanho fixo, que segue uma abordagem GCCL para codificar as regras;
- FRBCS-GP [12]: similar ao GP-COACH, mas sem alguns ferramentais com respeito a operadores genéticos e de dinâmica populacional.

No experimento foram usadas 22 bases de dados, disponível no repositório KEEL [19]. Os testes foram realizados conforme [13]: três execuções para cada base de dados, onde em cada execução foi feita uma validação cruzada separando em 10 pedaços cada base de dados. O acerto médio de classificação percentual foi calculado para as 30 execuções na fase de treinamento e teste de cada modelo. Uma análise estatística não paramétrica [20] foi realizada a partir da acurácia média na fase de teste para as 22 bases de dados. O nível de significância adotado foi de 5% ($\alpha=5\%$). O procedimento estatístico adotado: (1) teste de Friedman e de Iman-Davenport, de modo a avaliar a existência de alguma diferença significativa entre os modelos e, no caso de existir diferença, (2) teste de Holm para detectar qual modelo obteve uma maior acurácia na fase de teste em relação aos demais. Parâmetros e operadores genéticos usados nos demais modelos são descritos em [13]. O critério de parada estabelecido foi de 20000 avaliações.

As funções de pertinência foram dispostas de forma normalizada e uniforme conforme Figura 3; atributos categóricos foram codificados no formato 1 de N. A Pressão Lexicográfica Parcimoniosa [21] foi usada para reduzir o tamanho das expressões de cada indivíduo da PGMG.

Os parâmetros, terminais e funções usadas nos GPF-CLASS são apresentados na Tabela 2. A grande parte destes parâmetros seguiram os de [13], de modo a permitir uma maior comparação com o GP-COACH. GPF-CLASS foi implementado em MATLAB R2010a [22]. Para as análises estatísticas foi usado o software KEEL [19].

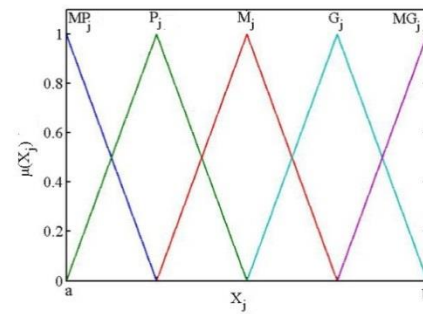


Fig. 3. Conjuntos fuzzy para o j-ésimo atributo.

TABELA 2. CONFIGURAÇÕES PRINCIPAIS DA PGMG.

Parâmetros	Valores
Tamanho da População	100
Máxima Profundidade do Indivíduo	7
Número de Gerações	200
Tamanho do Torneio	2
Taxa de Cruzamento de Alto Nível	65%
Taxa de Cruzamento de Baixo-Nível	65%
Taxa de Mutação	20%
Taxa de Reprodução Direta	15%
Taxa de Elitismo	1%
Pressão Lexicográfica [21]	Sim
Terminais	$\mu_{MP_j}(X_j), \mu_{P_j}(X_j), \mu_{M_j}(X_j), \mu_{G_j}(X_j)$ e $\mu_{MG_j}(X_j)$. Categóricos transformados no formato 1 de N. Constantes entre [0, 1]
Funções	Soma, subtração e produto tipo 1

B. Resultados e Discussões

Tabela 3 dispõe os resultados de acurácia de cada modelo, na fase de treinamento e teste, para as 22 bases de dados. Os valores em negrito indicam o melhor desempenho na fase de teste para um dado benchmark. GPF-CLASS demonstrou maior acurácia em 12 bases de dados (55%), enquanto o GP-COACH obteve em 6 bases de dados (27%). Em problemas de classificação binária, GPF-CLASS obteve acurácias superiores em 91% dos casos. Já em problemas de múltiplas classes, o modelo GP-COACH proporcionou melhores resultados em 6 bases de dados, enquanto o GPF-CLASS obteve melhores resultados em 2 casos.

Figura 4 exibe o rank (posto) médio para cada modelo, considerando os resultados na fase de teste para as 22 bases de dados. Quanto menor o rank, melhor o desempenho do modelo. É verificável que o GPF-CLASS foi o que obteve menor rank (2,00) quando comparado aos demais modelos. Os testes de Friedman e Iman-Davenport foram aplicados, resultando nas seguintes estatísticas de teste: Friedman = 61,63; Iman-Davenport = 18,39. Em ambos os testes, a hipótese de igualdade dos ranks entre os modelos foi rejeitada (p -valor < 0,001). GPF-CLASS foi selecionado como o modelo de controle para o teste de Holm (devido ao seu menor rank entre os demais). Os resultados do teste de Holm são apresentados na tabela 4.

Os resultados apresentados na Tabela 4 demonstram que o GPF-CLASS foi significativamente superior do que GP-PITT-Tsakonas, PCA-Ravi, 2SLAVE e GCCL-Ishibuchi em termos de acurácia. O mesmo não pode ser afirmado, quando

se comparam GPF-CLASS com FRBCS-GP e GP-COACH, apesar de numericamente os resultados do GPF-CLASS terem sido melhores (como nos problemas de classificação binária, por exemplo).

Figura 5 dispõe boxplots da acurácia relativa do c -ésimo modelo ($c = 1, \dots, 7$) para uma determinada base de dados, quando comparado ao modelo com maior acurácia. Por exemplo, na base de dados Bupa, GPF-CLASS obteve 70,72% de acurácia na fase de teste, enquanto que o resultado do GP-COACH foi de 63,63%. Como GPF-CLASS foi o modelo mais acurado nesta base de dados, portanto, a acurácia relativa do GPF-CLASS é $70,72\%/70,72\% = 1$ e GP-COACH é $63,63\%/70,72\% = 0,89$. Ao aplicar isto, é obtido um número adimensional para todos os resultados. Usando o teste de Wilcoxon [30] para comparar GP-COACH e GPF-CLASS a partir desta transformação, os resultados são: $Z=-1,96$; $p\text{-valor}=0,05$. Então, pode ser concluído que GPF-CLASS foi significativamente melhor do que GP-COACH em obter melhores resultados (ou próximo do melhor).

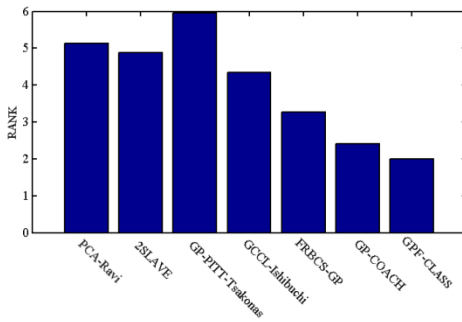


Fig. 4. Rank médio dos modelos.

Em geral, para bases de dados simples como Iris, o tempo computacional tomado pelo GPF-CLASS para 20000 avaliações foi de 1 minuto. Para bases de dados maiores, como Magic ou Penbased, o tempo computacional requerido foi em torno de 5 minutos. Foi usado o ambiente MATLAB R2010a executado em um Windows PC, processador Intel Quad-Core com 4 GB RAM.

Por fim, equações (14-16) descrevem uma típica expressão do GPF-CLASS para a base de dados Iris.

$$Setosa = \mu_M(SL) \quad (14)$$

$$Versicolor = \mu_{MP}(PL) \quad (15)$$

$$Virginica = 2.0\mu_{MG}(PL) + \mu_G(PW) - \mu_P(PL) - \mu_M(PL) - \mu_{MP}(PL) - 2.0\mu_M(SL) - 2.0\mu_G(SL) \quad (16)$$

É possível interpretar equações (14) e (15) como:

Se o Tamanho da Sépala é Médio

Então a planta pertence à espécie Setosa

Se o Tamanho da Pétala é Muito Pequena

Então a planta pertence à espécie Versicolor

Além disso, ao analisar a equação (16), é possível concluir que quando uma planta possui um Tamanho de Pétala Muito Grande esta está positivamente relacionada com a espécie Virginica; por outro lado, pelas relações

negativas em (16) é possível deduzir que quando uma planta possui um Tamanho da Pétala Médio, Pequeno ou Muito Pequeno, menor é seu grau de pertinência a espécie Virginica. Adicionalmente, se um padrão possui uma Largura de Pétala Grande, então tem maior compatibilidade a espécie Virginica; alternativamente, se uma planta possui um Tamanho de Sépala Médio ou Grande, então esta não tende a estar relacionada com a espécie Virginica.

V. CONCLUSÕES

Este trabalho apresentou a concepção do modelo Fuzzy-Genético para classificação GPF-CLASS. Estudos de casos demonstraram o potencial do modelo. Este obteve melhores resultados do que abordagens bem estabelecidas na literatura. Deve se ter em conta que os parâmetros dos modelos usados para comparação foram otimizados para melhores resultados nessas bases de dados [13], enquanto que o GPF-CLASS não sofreu nenhum ajuste fino para obter melhores resultados.

Trabalhos futuros devem incluir uma análise quantitativa das equações geradas pelo GPF-CLASS, possivelmente usando uma medida de complexidade (número de atributos, nível de suporte ou confiança gerada pelas equações, etc.). Mais execuções e avaliações devem ser efetuadas para investigações do baixo êxito do modelo em problemas de múltiplas classes. Por fim, um ajuste fino dos parâmetros, visando obter melhores resultados e compreender o comportamento do modelo para diferentes parametrizações.

REFERÊNCIAS

- [1] R.O. Duda, P.E. Hart and D.G. Stork, Pattern Classification. 2ed. New York: Wiley, 2000.
- [2] R. Elwell and R. Polikar, Incremental learning of concept drift in nonstationary environments. IEEE Transactions on Neural Networks, vol. 22, pp. 1517-1531, 2011.
- [3] S. Haykin, Neural Networks: A comprehensive foundation, ed.2. New Jersey: Prentice Hall, 2004.
- [4] S. Abe, Support vector machines for pattern classification. Berlin: Springer, 2010.
- [5] R. J. Koza, Genetic Programming.: On the Programming of Computers by Means of Natural Selection. Massachusetts: MIT Press, 1992.
- [6] W. Pedrycz and F. Gomide, Fuzzy systems engineering: Toward Human-Centered Computing. Hoboken: Wiley-Interscience, 2007.
- [7] J.S.R. Jang, C.T. Sun and E. Mizutani, Neuro-Fuzzy and Soft Computing. New Jersey: Prentice-Hall, 1997.
- [8] F. Herrera, Genetic fuzzy systems: taxonomy, current research trends and prospects. Evolutionary Intelligence, vol. 1, pp. 27-46, 2008.
- [9] H. Ishibuchi, T. Nakashima and M. Nii, Classification and Modeling with Linguistic Information Granules: Advanced Approaches to Linguistic Data Mining. Berlin: Springer, 2004.
- [10] M.P. Hinchliffe, M.J. Willis, H. Hiden, M.T. Tham, B. McKay and G.W. Barton, Modeling chemical process systems using a multi-gene genetic programming algorithm. In: GP: Proceedings of the First Annual Conference. Massachusetts: MIT Press, 1996, pp. 56-65.
- [11] D.P. Seanson, M.J. Willis and G.A. Montague, Co-evolution of non-linear PLS model components. Journal of Chemometrics, vol. 2, pp. 592-603, 2007.
- [12] F. Berlanga, M. Del Jesus and F. Herrera, Learning fuzzy rules using genetic programming: context-free grammar definition for high-dimensionality problems. In Proceedings of the I Workshop on Genetic Fuzzy Systems, Granada, 2005, pp. 136-141.

- [13] F. Berlanga, A.J. Riveira, M. Del Jesus and F. Herrera, GP-COACH: Genetic Programming-based learning of COmpact and ACcurate fuzzy rule-based classification systems for High-dimensional problems. *Information Sciences*, vol.180, pp.1183-1200, 2010.
- [14] H. Ishibuchi, T. Nakashima and T. Murata, Design of accurate classifiers with a compact fuzzy-rule base using an evolutionary scatter partition of feature space. *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol. 29, pp. 601–618, 1999.
- [15] H. Ishibuchi and T. Nakashima, Effect of rule weights in fuzzy rule-based classification systems. *IEEE Transactions on Fuzzy Systems*, vol. 9, pp. 506-515, 2001.
- [16] V. Ravi and H. Zimmermann, Fuzzy rule based classification with FeatureSelector and modified threshold accepting, *European Journal of Operational Research*, vol. 123, pp. 16–28, 2000.
- [17] A. González and R. Pérez, Selection of relevant features in a fuzzy genetic learning algorithm. *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol. 31, pp. 417-425, 2001.
- [18] A.A. Tsakonas, A comparison of classification accuracy of four genetic programming-evolved intelligent structures. *Information Sciences*, vol. 176, pp. 691-724, 2006.
- [19] J. Alcalá-Fdez, L. Sánchez, S. García, M.J. Del Jesus, S. Ventura, J.M. Garrell, J. Otero, C. Romero, J. Bacardit, V.M. Rivas, J.C. Fernández and F. Herrera, KEEL: A Software Tool to Assess Evolutionary Algorithms to Data Mining Problems. *Soft Computing*, vol. 13, pp. 307-318, 2009.
- [20] J. Demsar, Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.
- [21] S. Luke and L. Panait, Lexicographic parsimony pressure. In: *Proceedings of the Genetic and Evolutionary Computation Conference*, W. B. Langdon et al. (Ed.). New York: Morgan Kaufmann Publishers, 2002, pp. 829-836.
- [22] MATLAB. MATLAB 7.10.0 (R2010a). Massachusetts: The MathWorks Inc., 2010.

TABELA 3. ACURÁCIA DOS MODELOS EM DIFERENTES BASES DE DADOS. * -TR – TREINAMENTO. TS. - TESTE

Base de Dados com Duas Classes														
Dados	PCA-Ravi		2SLAVE		GP-PITT-Tsakonas		GCCL-Ishibuchi		FRBCS-GP		GP-COACH		GPF-CLASS	
	Tr. (%)	Ts. (%)	Tr. (%)	Ts. (%)	Tr. (%)	Ts. (%)	Tr. (%)	Ts. (%)	Tr. (%)	Ts. (%)	Tr. (%)	Ts. (%)	Tr. (%)	Ts. (%)
Bupa	67,08	54,46	64,75	58,58	59,80	56,45	60,36	58,27	64,49	62,20	69,04	63,63	80,52	70,72
HillValley1	52,35	51,48	52,52	51,76	50,86	49,97	20,26	20,02	50,43	49,78	53,96	52,89	65,82	64,13
HillValley2	52,00	50,85	52,53	51,21	50,57	49,20	28,62	28,00	51,28	50,69	55,68	53,99	65,45	64,88
Magic	77,47	77,59	74,23	74,29	64,89	64,79	76,02	76,02	74,59	74,51	79,78	79,82	84,30	84,11
Parkinsons	88,2	73,63	84,10	81,75	77,87	74,53	84,22	83,27	86,86	85,75	89,74	86,48	96,56	89,33
Pima	81,77	68,31	67,00	66,45	65,85	64,28	70,37	69,11	74,79	73,16	77,02	74,37	81,55	76,67
Ringnorm	38,4	30,12	80,12	79,64	50,87	50,51	91,81	91,70	94,1	93,84	91,24	91,13	93,30	92,53
Sonar	92,9	27,65	78,45	70,72	65,24	52,42	83,49	72,40	83,3	71,15	80,25	67,48	92,18	77,00
Spambase	81,7	64,93	69,87	70,14	82,30	81,89	69,77	69,87	75,03	74,55	83,17	82,80	90,76	90,37
Twonorm	24,81	20,04	84,67	84,35	49,02	48,80	90,7	90,12	92,4	91,97	85,42	84,83	96,96	96,47
Wdbc	94,73	86,77	92,42	91,80	65,66	63,09	92,69	91,09	95,6	95,02	95,09	93,90	98,87	96,43
Média	68,31	55,08	72,79	70,97	62,08	59,63	69,85	68,17	76,62	74,78	78,22	75,57	86,02	82,06

Base de Dados com Múltiplas Classes														
Cleveland	79,97	49,24	54,18	46,19	61,93	56,46	62,83	54,15	60,91	56,69	64,93	55,23	63,52	57,33
Ecoli	86,19	55,46	58,18	57,49	45,77	43,94	74,47	71,17	81,22	76,75	83,58	77,72	75,17	73,82
Glass	74,18	46,56	49,29	44,39	48,03	45,12	69,63	60,69	61,28	56,61	71,26	65,33	66,87	62,13
Iris	94,27	88,44	94,67	94,67	54,10	48,44	95,55	94,67	97,65	97,11	97,78	97,56	98,22	96,67
Libras	74,83	42,41	33,15	25,83	10,08	5,28	28,70	20,74	56,24	47,69	74,22	45,56	27,30	23,00
Page-blocks	91,15	90,70	91,4	91,42	93,03	92,92	90,41	90,34	91,24	91,09	91,3	91,23	94,17	94,22
Pen-based	81,36	81,81	81,32	81,16	44,86	44,67	82,53	82,18	75,74	75,53	82,29	82,20	49,88	49,84
Satimage	78,36	76,54	33,39	33,45	23,82	23,82	63,14	63,12	68,08	68,06	72,83	72,50	76,30	76,02
Segment	80,17	78,50	73,37	72,81	21,85	21,62	84,64	84,07	81,23	80,38	86,55	85,96	79,21	79,35
Wine	99,33	93,17	92,22	91,53	46,30	38,19	97,98	91,21	95,84	91,13	98,96	95,10	97,69	93,89
Yeast	58,94	39,45	15,22	14,51	32,36	31,76	50,69	49,01	52,79	52,16	49,74	48,56	50,07	48,93
Média	81,70	67,48	61,49	59,40	43,83	41,11	72,78	69,21	74,75	72,11	79,40	74,27	70,76	68,65

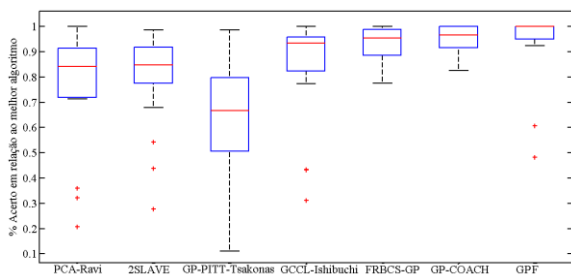


Fig 5. Boxplot da acurácia relativa dos modelos.

TABELA 4. RESULTADOS DO TESTE DE HOLM

i	Modelo	z	p-valor	α/i	Hipótese
6	GP-PITT-Tsakonas	6,0714	< 0,0001	0,0083	Rejeita
5	PCA-Ravi	4,8152	< 0,0001	0,0100	Rejeita
4	2SLAVE	4,4314	< 0,0001	0,0125	Rejeita
3	GCCL-Ishibuchi	3,5939	0,0003	0,0167	Rejeita
2	FRBCS-GP	1,9540	0,0507	0,0250	Não Rejeita
1	GP-COACH	0,6281	0,5299	0,0500	Não Rejeita