META-HEURÍSTICA GRASP APLICADA À EXTRAÇÃO DE REGRAS DE CLASSIFICAÇÃO

Genival Pavanelli
Universidade Federal do Paraná – UFPR
Programa de Pós-Graduação em Métodos Numéricos em Engenharia
CP19081 – Curitiba, PR; CEP 81531-990
genivalpavanelli@gmail.com

Maria Teresinha Arns Steiner
Universidade Federal do Paraná – UFPR
Programa de Pós-Graduação em Métodos Numéricos em Engenharia
CP19081 – Curitiba, PR; CEP 81531-990
tere@ufpr.br

Anderson Roges Teixeira Góes
Universidade Federal do Paraná – UFPR
Departamento de Expressão Gráfica
CP19081 – Curitiba, PR; CEP 81531-990
artgoes@ufpr.br

Alessandra Memari Pavanelli
Universidade Federal do Paraná – UFPR
Programa de Pós-Graduação em Métodos Numéricos em Engenharia
CP19081 – Curitiba, PR; CEP 81531-990
alessandracmc@bol.com.br

RESUMO

O processo de gestão do conhecimento nas mais diversas áreas da sociedade exige constante atenção a multiplicidade de decisões a serem tomadas acerca das atividades presentes nas organizações que as constituem. Para tomar estas decisões deve-se ter cautela ao basear-se somente no conhecimento pessoal adquirido com a experiência profissional, uma vez que a partir desse método o processo seria lento, caro e altamente subjetivo. Para auxiliar nesse gerenciamento, faz-se necessário o uso de ferramentas matemáticas que cumpram a finalidade de extração de conhecimento de base de dados. Este artigo propõe a aplicação de Procedimentos de Busca Gulosos, Aleatórios e Adaptativos (*Greedy Randomized Adaptive Search Procedure* – GRASP) (FEO; RESENDE, 1995), (PITSOULIS; RESENDE, 2002), (RESENDE; RIBEIRO, 2002) e (RESENDE; SILVA, 2013) como ferramenta de *Data Mining* (DM), dentro do processo denominado *Knowledge Discovery in Databases* (KDD) para a tarefa de extração de regras de classificação em bases de dados.

PALAVRAS CHAVE: Procedimentos de Busca Gulosos, Aleatórios e Adaptativos. *Data Mining*. Extração de Regras.

Área principal: MH – Metaheurísticas, OA – Outras Aplicações em PO e PM – Programação Matemática.

ABSTRACT

The process of knowledge management in the several areas of society requires constant attention to the multiplicity of decisions to be made about the activities in organizations that constitute them. To make these decisions one should be cautious in relying only on personal knowledge acquired through professional experience, since the whole process based on this

method would be slow, expensive and highly subjective. To assist in this management, it is necessary to use mathematical tools that fulfill the purpose of extracting knowledge from database. This article proposes the application of Greedy Randomized Adaptive Search Procedure (GRASP) (FEO; RESENDE, 1995), (PITSOULIS; RESENDE, 2002), (RESENDE; RIBEIRO, 2002) e (RESENDE; SILVA, 2013) as Data Mining (DM) tool within the process called Knowledge Discovery in Databases (KDD) for the task of extracting classification rules in databases.

KEYWORDS: Greedy Randomized Adaptive Search Procedure. Data Mining. Rules Extraction.

Main area: MH - Metaheuristics, OA - Other applications in OR and PM - Mathematical Programming,

1. Introdução

Atualmente a maioria das operações e atividades de diversas organizações é efetivada computacionalmente o que gera uma imensa quantidade de dados. À medida que aumentam os bancos de dados oriundos destas transações, aumenta também o interesse em extrair destes vastos bancos de dados o conhecimento.

Neste contexto, este trabalho aborda o processo de Descoberta de Conhecimento em Bases de Dados (*Knowledge Discovery in Databases*; KDD) que se trata de um processo de descoberta de padrões e tendências por análise de conjuntos de dados, tendo como principal etapa o processo DM, que produz uma relação entre padrões a partir dos dados (FAYYAD *et al.*, 1996).

A fim de executar o processo de *Data Mining* (DM, ou - do inglês - Mineração de Dados) este artigo propõe a aplicação de uma meta-heurística baseada no procedimento GRASP para extração de regras de classificação em bases de dados. Optou-se pelo GRASP por se tratar de uma meta-heurística multi-partida (RESENDE; SILVA, 2013), de fácil implementação e largamente utilizada em problemas de otimização combinatória.

O presente artigo está organizado da seguinte forma: na seção 2 são apresentados trabalhos correlacionados bem como os conceitos do processo KDD e da meta-heurística GRASP, os quais norteiam este trabalho. Na seção 3 está a proposta de adaptação à meta-heurística GRASP para a extração de regras de classificação em bases de dados. Em seguida, na seção 4, são apresentados os resultados obtidos com a referida proposta aplicada a três bases de dados. Finalmente, na seção 5 são apresentadas as conclusões.

2. Revisão de Literatura

Nesta seção primeiramente destacamos três trabalhos relacionados ao aqui apresentado seguido por uma breve revisão acerca de KDD e GRASP.

PLASTINO *et al.* (2011) apresentam uma versão híbrida da meta-heurística GRASP, que incorpora técnicas de *Data Mining* para a solução do problema de *p*-medianas, a qual denominaram DM-GRASP (Data Mining GRASP). A proposta desta hibridização baseia-se na hipótese de que os padrões extraídos de um conjunto de soluções sub-ótimas podem guiar a busca por soluções melhores. O algoritmo DM-GRASP é composto por duas fases. Na primeira, chamada fase de geração do conjunto de elite, o GRASP é executado *n* vezes a fim de obter um conjunto de soluções diferentes. As melhores soluções deste conjunto compõem o conjunto de elite. Neste momento é aplicado o processo de mineração de dados, a fim de se extrairem padrões (conjuntos de elementos) que aparecem com frequência em soluções do conjunto de elite. Na segunda fase, são executadas outras n iterações adaptadas, nas quais os padrões são utilizados para guiar a construção das soluções. Buscando melhorar ainda mais as soluções, propõem outra versão, na qual a mineração de dados é realizada não apenas uma vez como no DM-GRASP, mas sempre que o conjunto de elite se torna estável, ou seja, quando não ocorre nenhuma mudança no conjunto de elite ao longo de um número pré-definido de iterações. Essa versão foi denominada MDM-GRASP (*Multi Data Mining GRASP*). .

SEMAAN E OCHI (2011) apresentam um novo algoritmo heurístico baseado na meta-heurística GRASP para a extração de regras de associação. Com o objetivo identificar quais regras são realmente relevantes e úteis, são calculadas medidas de interesse para as regras de associação. Essas medidas empregam índices estatísticos para avaliar a força de cada regra. São elas: o fator de suporte, que representa o percentual de transações da base de dados que contêm os itens do conjunto; a confiança da regra $A \Rightarrow B$ (A implica em B), que é um valor que indica dentre as transações que contêm os itens de *A*, o percentual de transações que também contêm os itens de *B*; e o *lift*, que avalia as dependências entre o conjunto de itens antecedentes e consequentes das regras. Na primeira fase da heurística proposta foi considerada a formação de *itemsets* de tamanho *k*, submetido como parâmetro, e não a construção de soluções (regras de associação). Já a segunda fase atua na construção da solução e refinamento dos *itemsets* construídos na fase anterior, sendo que nesta fase ocorre efetivamente a extração das regras de associação. Com o objetivo de diversificar a formação de *itemsets* foram considerados quatro critérios relacionados ao suporte dos itens. Os resultados obtidos mostraram que a utilização do algoritmo proposto é uma alternativa interessante para a obtenção de regras de associação de qualidade, ainda que seus *itemsets* possuam baixo(s) suporte e/ou confiança.

BARBALHO *et al*. (2011) apresentam a hibridização da meta-heuristica GRASP que incorpora o processo de religação de caminhos (*Path Relinking*) e um módulo de mineração de dados. A aplicação proposta neste trabalho é para o problema 2PNPD (*2-path network design problem*). A contribuição deste trabalho é mostrar que não só a meta-heurística GRASP tradicional, mas também a hibridização do GRASP com a heurística *path-relinking* podem se beneficiar da incorporação de um processo de mineração de dados para extrair padrões de soluções sub-ótimas, a fim de orientar de forma mais eficiente à busca de melhores soluções. Foram aplicadas três versões híbridas do GRASP. Na primeira delas, denominada GRASP-PR, a religação de caminhos é usada após cada iteração GRASP, ligando a solução obtida da busca local com uma solução do conjunto de elite. A segunda versão, denominada DM-GRASP-PR é composta de duas fases: (1) execução de *n* iterações GRASP para obtenção do conjunto de elite; (2) nesta fase, denominada híbrida, são executas novamente *n* iterações GRASP. A terceira versão apresentada, denominada MDM-GRASP-PR, difere da anterior pelo fato de o processo de mineração de dados não ser executado apenas uma vez, mas toda a vez que o conjunto de elite se torna estável. Os resultados experimentais mostraram que a primeira versão da estratégia híbrida proposta, chamada DM-GRASP-PR, foi capaz de obter as melhores soluções em menos tempo computacional do que o GRASP original com religação de caminho. O MDM-GRASP-PR obteve resultados ainda melhores do que o DM-GRASP-PR.

A descoberta de conhecimento em bases de dados, é um processo não trivial de descoberta de padrões válidos, novos, úteis e acessíveis (FAYYAD *et al*., 1996; HAN, 2011). Em outras palavras, trata-se de um processo de extração de informação a partir de dados de uma base de dados, que contenha um conhecimento implícito, inicialmente desconhecido, compreensível e potencialmente útil.

Segundo FAYYAD *et al*. (1996), o processo KDD é composto de cinco etapas a saber: seleção dos dados; pré-processamento e limpeza dos dados; formatação ou transformação dos dados; Mineração de Dados; interpretação e avaliação dos resultados. A sequência destas etapas pode ser observada na Figura 1.

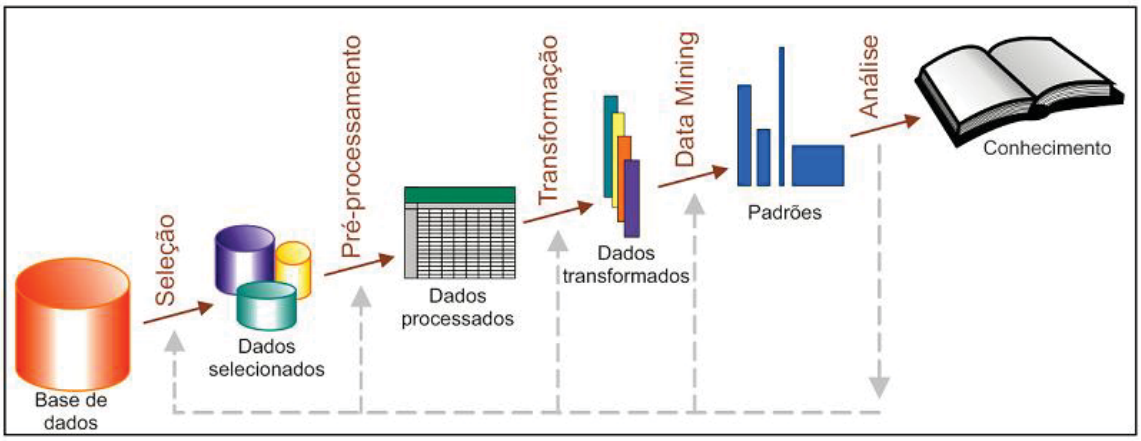

Figura 1 - Etapas do processo KDD, adaptada de FAYYAD et al. (1996).
Fonte: GÓES E STEINER 2012.

A quarta etapa deste processo é a Mineração de Dados (DM) que é a mais importante do processo (STEINER *et al.*, 2006; GÓES, 2012), onde são aplicadas as ferramentas para executar suas tarefas. A seguir são descritas as seguintes tarefas de DM: classificação (foco deste trabalho) e regras de associação.

A tarefa de classificação consiste em alocar um padrão (ou instância) a uma determinada classe dentre outras previamente estabelecidas. Ao se estabelecerem classes dentro de um conjunto de dados, busca-se corresponder cada uma delas a um conjunto de valores dos atributos previsores, valores esses considerados descritores da classe. Dessa forma, usando os descritores de cada uma das classes que constituem o conjunto de dados é possível construir um classificador que descreve cada instância como pertencente à determinada classe.

A tarefa de busca de regras de associação tem por objetivo estabelecer relacionamentos entre itens de uma base de dados (KAZIENKO, 2009). Regras de associação são expressões do tipo $A \Rightarrow B$, que significam: SE (A), ENTÃO (B), em que A e B são conjuntos de itens pertencentes a uma base de dados D, tais que: $A \subset D$, $B \subset D$, $A \neq \emptyset$, $B \neq \emptyset$ e $A \cap B = \emptyset$. Assim, o significado de uma regra de associação ($A \Rightarrow B$) é de que os conjuntos de itens A e B ocorrem juntos em uma mesma transação.

Neste trabalho a tarefa de extrair padrões da base de dados será executada pela meta-heurística baseada no procedimento iterativo GRASP. Cada iteração é composta de duas fases: uma de construção e a outra de busca local (FEO; RESENDE, 1995), (PITSOULIS; RESENDE, 2002) e (RESENDE; RIBEIRO, 2002).

Na fase de construção GRASP, a solução é obtida de forma iterativa, ou seja, a cada iteração desta fase, um elemento é acrescido à solução parcial até obter-se a solução completa. Os candidatos a comporem a solução são obtidos a partir do conjunto de elementos que não comprometem a viabilidade da solução. A avaliação do próximo elemento a compor a solução parcial é feita a partir de uma função de avaliação gulosa.

Os elementos melhor avaliados por esta função gulosa (aspecto guloso do algoritmo) compõem a Lista Restrita de Candidatos (LRC), cujo tamanho é definido pelo parâmetro α. A partir da LRC seleciona-se aleatoriamente (aspecto aleatório do algoritmo) o próximo elemento a compor a solução parcial do problema. Na próxima iteração desta fase os elementos que restaram são novamente avaliados e a LRC é atualizada (aspecto adaptativo do algoritmo) e a solução parcial é acrescida de mais um elemento, até que a solução esteja completa.

3. Meta-heurística Baseada no Procedimento GRASP para Extração de Regras de Classificação.

O algoritmo proposto neste trabalho se baseia na meta-heurística GRASP, assim é composto de duas fases: a primeira trata da construção da regra e a segunda realiza uma busca

local. Seu objetivo é extrair um conjunto de regras de classificação que apresentem alto grau de precisão preditiva.

- **Fase de Construção da meta-heurística proposta:**

Ao iniciar a fase de construção, o primeiro passo é definir o parâmetro *k* que indica quantos elementos irão compor o conjunto de antecedentes da regra de classificação. A regra inicialmente trata-se de um conjunto vazio. A cada uma das *k* iterações da fase de construção, um elemento é acrescido a regra parcial até obter-se a regra completa, ou seja, *k* elementos como antecedentes da regra que classificam corretamente o atributo consequente.

Os candidatos a comporem a regra em construção são obtidos a partir da "Lista Restrita de Candidatos" (LRC), a qual é estabelecida conforme se segue.

Considere o problema de extração de regras de classificação de uma base de dados. Seja $A= \{a_1, a_2, \ldots, a_n\}$ um conjunto de elementos a serem acrescidos a uma regra. Define-se $s(a_i)$ o valor do suporte da regra após a inclusão do elemento a_i. Sejam s^{max} e s^{min}, o maior e o menor suporte das regras, respectivamente. A LRC é composta por elementos a_i pertencente a *A* com os maiores suportes, de maneira que a sua inserção não destrua a viabilidade da regra. A lista fica associada ao parâmetro α cujo valor encontra-se no intervalo *[0,1]*. Os elementos pertencentes à LRC devem gerar, quando inseridos a regra, um suporte maior ou igual a um valor Δ pré-definido com base no parâmetro α. A equação (1) a seguir define Δ para a heurística proposta.

$$\Delta = s^{min} + \alpha\,(s^{max} - s^{min}) \qquad (1)$$

Como se pode observar na equação (1), o parâmetro α determina quão guloso ou aleatório será a inserção de um novo elemento a regra durante a sua construção. Neste caso para α igual a "0" o algoritmo é puramente aleatório, enquanto que para α igual a "1" o algoritmo é guloso.

A partir da LRC seleciona-se aleatoriamente o próximo elemento a compor a solução parcial do problema. Na próxima iteração desta fase os elementos que restaram são novamente avaliados, LRC é atualizada e a solução parcial é acrescida de mais um elemento, até que a solução esteja completa (*k* antecedentes), o que caracteriza o final da fase de construção.

A Figura 2 apresenta o pseudocódigo da fase de construção.

```
Construção Regra GRASP
    Regra = { }
    K = NrdeElementosdoAntecessor
    α = DeterminaTamanhoLRC
    Para 1 até k faça:
       Construir a LRC(α)={ai; s(ai)>= Δ};
         r = SeleçãoAleatória(LRCα);
       Regra = Regra ∪ r;
       AtualizaFunçãoGulosa(r)
    Fim Para
    Retornar Regra
Fim Construção Regra GRASP
```

Figura 2 - Pseudocódigo da fase de construção da regra aleatória e gulosa.
Fonte: o autor.

No final da fase de construção aleatória e gulosa do GRASP, a regra apresentada possui *k* elementos. Na próxima fase do procedimento GRASP para extração de regras de classificação o objetivo é realizar uma busca local nas vizinhanças da regra apresentada na fase anterior a fim de buscar outras regras de boa qualidade.

- **Fase de Busca Local da meta-heurística proposta:**

O processo de busca local da meta-heurística proposta parte de uma regra inicial a qual foi obtida da fase anterior (Fase de Construção) e que conta com *k* elementos no seu antecessor. A cada iteração desta fase gera-se o conjunto de todas as regras obtidas a partir da combinação dos *k* antecessores. Na primeira iteração geram-se todas as regras com "*k-1*" elementos no antecessor da regra; na segunda com "*k-2*"; e assim sucessivamente até estabelecer todas as combinações finalizando com "1" elemento no antecessor. Cada regra gerada é avaliada segundo critérios pré-estabelecidos, nesta proposta, um suporte e uma confiança mínimos, de maneira que todas as regras geradas que atenderem estes critérios (as quais a partir de agora serão chamadas de regras de boa qualidade) serão arquivadas. O final deste processo iterativo retorna um conjunto de regras de boa qualidade.

A Figura 3 apresenta o pseudocódigo de busca local partindo da regra (Regra) construída na primeira fase da meta-heurística proposta.

```
Busca Local GRASP
    Regra
    k = NrdeElementosdoAntecessor
    Enquanto  k >=1 faça:
      Estabelecer todas as Regras possíveis com (k-1)elementos de Regra;
      Avaliar todas as Regras estabelecidas;
      Armazenar as Regras de boa qualidade;
        k = k-1;
    Fim Enquanto
    Retornar Regras Armazenadas
Fim Busca Local GRASP
```

Figura 3 - Pseudocódigo da fase de busca local da proposta.
Fonte: o autor.

Ao final da fase de busca local encerra-se uma iteração da meta-heurística baseada em GRASP, proposta no presente trabalho. Uma nova iteração se inicia até que o critério de parada (número de iterações) seja atingido. Cabe ressaltar que cada iteração inicia-se sem levar em consideração os resultados anteriores. Ao final deste processo iterativo estarão arquivadas todas as "regras de boa qualidade" – aquelas que apresentam fator de suporte e confiança acima do pré-estabelecido, conforme definido anteriormente.

A partir das regras que foram pré-selecionadas durante a fase de busca local, em cada uma das iterações do processo, constrói-se um classificador.

O classificador é composto por um conjunto finito de regras sequencialmente dispostas, que tem por objetivo classificar corretamente o maior número possível de padrões. Os padrões que compõem a base de dados serão apresentados um a um a este classificador. Quando o padrão apresentado se enquadra aos antecedentes da regra ele é classificado e retirado do conjunto de dados. Se a sua classe for igual ao sucessor da regra a qual ele se enquadrou significa que ele foi classificado corretamente, caso contrário foi classificado de maneira incorreta.

Diante das inúmeras "regras de boa qualidade" que foram arquivadas durante a execução da meta-heurística proposta, devem-se escolher quais regras serão usadas sequencialmente a fim de o classificador obter maior precisão preditiva. Neste trabalho propõem-se o sequenciamento das regras segundo a sua confiança, ordenando-as neste critério da maior para a menor, de maneira a obter assim maior precisão preditiva.

4. Metodologia do Trabalho

Calcado no objetivo de apresentar uma meta-heurística baseada em GRASP para extração de regras de classificação, bem como a utilização destas regras na construção de um

classificador e, ainda, objetivando verificar sua versatilidade, esta meta-heurística foi aplicada a três bases de dados distintas:

- Base de dados da justiça do trabalho, composta de 100 instâncias, 10 atributos e três classes. Extraída de PAVANELLI (2007).
- Base de dados *wine*, composta de 178 padrões, 13 atributos e três classes. Extraída do *Machine Learning Repositttory* (*http://archive.ics.uci.edu/ml/datasets/wine*).
- Base de dados zoo, composta de 101 amostras, 16 atributos e sete classes. Extraída do *Machine Learning Repositttory* (*http://archive.ics.uci.edu/ml/datasets/zoo*).

A metodologia proposta neste trabalho foi detalhada na próxima seção quando aplicada à base de dados da justiça do trabalho. Também na próxima seção encontram-se os resultados das demais bases (*wine* e *zoo*). Cabe ressaltar que a metodologia empregada em todas as bases de dados possui três grandes blocos segundo o processo KDD: pré-processamento dos dados; aplicação da meta-heurística propriamente dita para extração de regras; construção do classificador.

4.1 Base de dados da Justiça do Trabalho

Esta base de dados foi extraída da 1ª Vara da Justiça do Trabalho de São José dos Pinhais, Paraná, no período de setembro a novembro de 2006, disponível em PAVANELLI (2007) e apresenta dados acerca de processos que já tiveram suas sentenças emitidas entre os anos de 1998 e 2005. É composta de 100 processos (instâncias ou padrões), cada qual com 10 atributos previsores, cujo objetivo é classificar cada instância quanto ao tempo de duração do processo em três classes distintas: tempo longo, tempo médio e tempo curto.

Os atributos classificadores foram definidos juntamente com um especialista da área (juiz do trabalho) e listam-se a seguir: Objeto do Processo, Salário do Reclamante, Rito, Perícia, Tempo de Serviço, Acordo, Profissão, Recurso Ordinário, Recurso de Revista e Número de Audiências.

A fim de que a heurística apresentasse consistência no seu desempenho, com relação à extração de regras que classificassem corretamente cada processo de acordo com o seu tempo de duração, cada um dos atributos acima citados foi "tratado" de maneira a corresponder a uma ou mais coordenadas binárias (PAVANELLI, 2007), (LU *et al*., 1996), (BAESENS *et al*., 2003), no vetor de entrada da heurística.

Para realizar a implementação da heurística proposta foi desenvolvido um programa no *Software Visual Studio* 2012. Os parâmetros envolvidos no procedimento são: o critério de parada, o qual ficou estabelecido de acordo com o número de iterações igual a 100; a cardinalidade da LRC (α), que ficou definida como 0,5 (α = 0,5), ou seja, um "meio termo" entre a total aleatoriedade e o procedimento guloso; a confiança mínima, que ficou estipulada como 0,5; o suporte mínimo, pré-estabelecido como 0,05.

Foram realizados testes, determinando-se, nesta base de dados, o valor máximo para o número de antecedentes igual a quatro (n_A = 4). É oportuno relembrar que a regra gerada na fase de construção apresentará o antecedente com número de atributos igual a *k*; na fase de busca local, porém, serão estabelecidas regras com n_A -1, n_A -2, ... , 1, até se chegar a "1" único atributo preditivo no antecedente da regra e desde que atendam aos critérios de suporte e de confiança mínimos.

Durante as 100 iterações do procedimento de construção e busca local, o algoritmo armazenou todas as regras obtidas que satisfaziam as condições pré-estabelecidas de suporte e de confiança mínimos. Na construção do classificador, devem-se apresentar as regras segundo uma determinada sequência.

Buscando aumentar a precisão preditiva este trabalho propõe a sequência de apresentação das regras segundo ordem decrescente da confiança de cada regra.

A avaliação da meta-heurística proposta foi realizada por meio do procedimento de validação cruzada *k-fold*, com *k* = 10. Desta forma, foram construídos 10 classificadores distintos. A Tabela 1 apresenta parte de um dos classificadores estabelecidos durante o processo de validação e aplicado ao seu conjunto de treinamento.

Tabela 1 – Classificador tempo de processo aplicado ao grupo de treinamento

	REGRA	Class correta	Class Errada	Restam
1	SE (Número de audiências = 1) E (Acordo = sim) ENTÃO (Tempo Curto)	15	0	75
2	SE (Tempo de serviço >28 meses) E (Acordo = sim) E (FGTS = sim) ENTÃO (Tempo Curto).	4	0	71
3	SE (Profissão = indústria) E (Rito = RT) E (Recurso ordinário = sim) E (Horas extras = sim) ENTÃO (Tempo Longo).	9	0	62
4	SE (Profissão = indústria) E (Rito = RT) E (Horas extras = sim) E (Tempo de serviço <=6 meses) ENTÃO (Tempo Longo).	2	0	60
...	...	...	...	...
23	SE (Rito = RT) ENTÃO (Tempo Longo).	6	3	0

A partir dos classificadores obtidos no teste, pode-se construir a matriz de confusão.

Tabela 2 – Matriz de confusão da base justiça do trabalho aplicada ao grupo de treinamento

Classe	Tempo Curto	Tempo Médio	Tempo Longo	Precisão	
				Classe	Classificador
Tempo Curto	22	2	0	22/24	
Tempo Médio	1	25	6	25/32	77/90
Tempo Longo	1	3	30	30/34	

A partir da matriz de confusão, observa-se que o classificador apresenta uma precisão preditiva de 85,6% para o grupo de treinamento.

O conjunto de teste (padrões que não foram utilizados no treinamento) deste "*fold*" foi submetido ao classificador apresentado na Tabela 1. Os resultados relevantes encontram-se na matriz de confusão, conforme a Tabela 3.

Tabela 3 – Matriz de confusão da base justiça do trabalho aplicada ao grupo de teste

Classe	Tempo Curto	Tempo Médio	Tempo Longo	Precisão	
				Classe	Classificador
Tempo Curto	3	0	0	3/3	
Tempo Médio	0	4	0	4/4	8/10
Tempo Longo	1	1	1	1/3	

A partir da matriz de confusão exibida na Tabela 3, observa-se que o classificador apresenta uma precisão preditiva de 80% para este grupo de teste.

Ao final do processo de validação cruzada, calcula-se a taxa de erro global, que é a média das taxas de erro calculadas em cada etapa. A Tabela 4 a seguir apresenta, além desta taxa, a acurácia global (1 – erro global) bem como seu desvio padrão e sua mediana, todos baseados nos 10 testes do procedimento de validação cruzada para os grupos de treinamento e teste.

Tabela 4 – Precisão preditiva da base justiça do trabalho

Conjunto	Acurácia Global	Desvio Padrão	Mediana
Treinamento	84%	0,009	0,83
Teste	78%	0,063	0,80

4.1.1 Comparação dos resultados obtidos pela meta-heurística proposta com a técnica de árvores de decisão para a base de dados tempo de processo

A aplicação aqui abordada visa comparar o desempenho da técnica de árvores de decisão com a meta-heurística proposta neste trabalho. Buscando ampliar o horizonte de comparações, foram estabelecidos três algoritmos envolvendo a técnica de árvores de decisão a partir do *software WEKA* (*Waikato Environment for Knowledge Analysis*): BFTree, REPTree e J4.8. Cabe ressaltar que em todos os testes – tanto com a meta-heurística proposta quanto com a dos algoritmos a partir do *software WEKA* – foram utilizados os mesmos conjuntos de dados.

A Tabela 5 a seguir apresenta o número de instâncias classificadas correta e incorretamente para todos os algoritmos.

Tabela 5 – Classificação das instâncias segundo os algoritmos aplicados à base justiça do trabalho

	BFTree	REPTree	J4.8	Proposta
Instâncias classificadas corretamente	70%	66%	68%	78%
Instâncias classificadas incorretamente	30%	34%	32%	22%

Como se pode observar na Tabela 5, a meta-heurística proposta apresentou melhores resultados quando comparados aos dos algoritmos de árvore de decisão.

Visando a uma comparação entre as precisões das classificações em cada uma das classes e também do classificador de cada uma das aplicações estabelecidas, foi construída a Tabela 6, na qual se tem um melhor detalhamento dos resultados.

Tabela 6 – Comparativo da precisão dos algoritmos aplicados à base justiça do trabalho

Algoritmos	Precisão			
	Tempo Curto	Tempo Médio	Tempo Longo	Classificador
BFTree	78%	58%	76%	70%
RepTree	78%	50%	73%	66%
J4.8	77%	61%	68%	68%
Proposta	100%	65%	75%	78%

A partir desta tabela, foi elaborado o gráfico de comparação entre as precisões apresentadas dentro de cada classe.

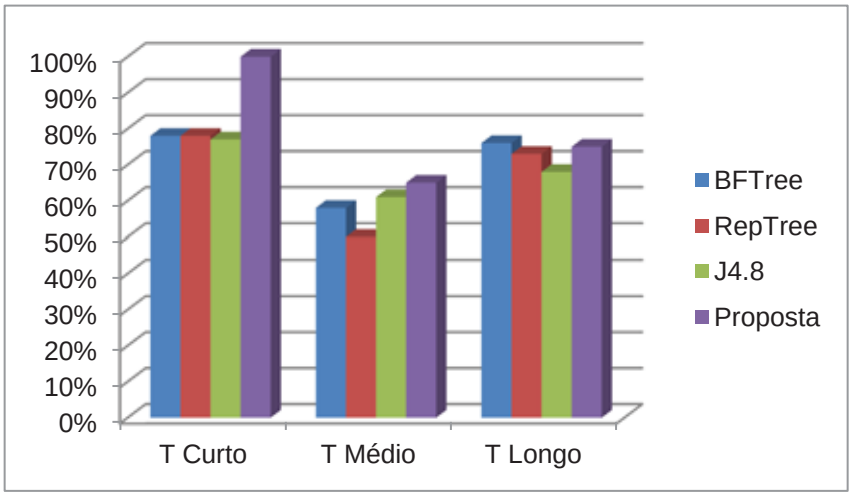

Figura 4 – Comparativo entre as precisões apresentadas dentro de cada classe para a base justiça do trabalho

4.1.2 Análise da aplicação da meta-heurística proposta

Conforme aludido no início desta seção, a meta-heurística proposta foi aplicada a três bases de dados. No item 4.1.1 foi apresentado detalhadamente o procedimento adotado neste trabalho para a base de dados justiça do trabalho. Nesta subseção, a partir da Tabela 7, serão apresentados de maneira sucinta os dados relevantes referentes às três bases analisadas.

Pela Tabela 7, verifica-se que a meta-heurística proposta apresentou, para duas bases de dados precisão preditiva superior à dos algoritmos de árvore de decisão. Em apenas uma das bases (*wine*) a acurácia obtida pela proposta deste trabalho foi igual à dos resultados obtidos pelos algoritmos comparativos. Estes resultados demonstram a superioridade (para as bases de dados analisadas) quanto à precisão preditiva da meta-heurística proposta em relação à dos demais algoritmos utilizados.

Tabela 7 – Resultados dos algoritmos aplicados às três bases de dados

Bases	Nr de Padrões	Nr de Atributos	Nr de Classes	Nr de Regras				Acurácia			
				BF Tree	Rep Tree	J4.8	Proposta	BF Tree	Rep Tree	J4.8	Proposta
Justiça	100	10	3	14	3	15	23	70%	66%	68%	78%
Wine	178	13	3	4	9	5	19	90%	94%	93%	94%
Zoo	101	16	7	1	1	9	9	41%	41%	92%	98%

5. Conclusões

Como comentado na introdução deste artigo, este trabalho aborda o processo de Descoberta de Conhecimento em Bases de Dados, mais especificamente a etapa de Mineração de Dados, a fim de executar a tarefa de extração de regras de classificação em bases de dados, utilizando como método uma meta-heurística baseada no procedimento GRASP.

Com o objetivo de construir um classificador com a máxima precisão preditiva, a metodologia adotada, que foi aplicada a três bases de dados distintas, fez uso das diversas etapas do processo *KDD*. Na etapa de pré-processamento de dados, tanto as variáveis qualitativas quanto as quantitativas foram codificadas de maneira a corresponder a uma ou mais coordenadas binárias para o vetor de entrada. Este pré-processamento fez-se necessário, pois a meta-heurística proposta foi desenvolvida para utilizar somente coordenadas binárias na entrada.

A utilização do procedimento GRASP como fundamento para elaboração da meta-heurística proposta atendeu as expectativas quanto à geração de regras de classificação. pois ao se variarem os parâmetros (número de antecedentes, suporte mínimo e confiança mínimo), pode-se variar o número de regras geradas, caracterizando assim a grande capacidade de adaptação desta meta-heurística às mais variadas bases de dados.

A construção do classificador é estabelecida a partir da ordenação das regras obtidas segundo o valor da sua confiança. Este critério atendeu ao objetivo proposto, uma vez que os classificadores apresentam elevada precisão preditiva.

A comparação estabelecida entre a meta-heurística proposta com a técnica de árvores de decisão proporcionou uma visão mais especifica das precisões preditivas que cada uma das bases analisadas apresenta para cada um dos algoritmos aplicados, quer seja o da meta-heurística proposta, foco deste trabalho, quer sejam os algoritmos envolvendo a técnica de árvores de decisão a partir do *software WEKA*: BFTree, REPTree e J4.8.

Esta comparação está apresentada de forma sucinta na Tabela 7. Pode-se observar que, para a base de dados *wine*, a meta-heurística proposta apresenta a mesma acurácia (0,94) da dos algoritmos J4.8 e RepTree, porém este resultado é superior ao do algoritmo BFTree (0,90). Nas outras duas bases de dados, a meta-heurística proposta apresenta acurácia superior à das demais. Conclui-se que a meta-heurística proposta é superior, quanto à precisão preditiva para as bases de dados analisadas, quando comparada aos algoritmos de árvore de decisão aqui apresentados.

Diante dos resultados obtidos a partir dos testes executados, nota-se que a meta-heurística proposta apresenta os requisitos básicos para executar a tarefa de *Data Mining*, mais especificamente a extração de regras de classificação. Os classificadores obtidos a partir destas regras apresentam elevadas precisões preditivas cumprindo, desta forma, o objetivo deste trabalho.

Sugere-se para trabalhos futuros a adoção de religamento de caminhos ou de outro processo de inserção de memória ao procedimento GRASP, a fim de que sejam alcançadas melhorias nos resultados. Outra sugestão é a realização de testes com o parâmetro α "trabalhado" de forma dinâmica, ou seja, ao iniciar a construção da regra, podem ser utilizados valores próximos a "0", tornando o algoritmo mais aleatório e à medida que cada antecedente vai sendo acrescido à regra em construção, o valor de α poderia ir se aproximando de "1", tornando a escolha dos próximos antecedentes mais gulosa.

REFERÊNCIAS

Baesens, B.; Setiono, R.; Mues, C. e Vanthienen, J. (2003). *Using Neural Network Rule Extraction and Decision Tables for Credit-Risk Evalution*. Management Science Informs, vol. 49, n° 3, p. 312-329.

Barbalho, H.; Rosseti, I. C. M.; Martins, S. L. e Plastino, A. *A Hybrid Data Mining GRASP with Path-Relinking*. XLIII SBPO, Ubatuba, SP, Ago. 2011.

Fayyad, U. M.; Piatetsky-Shapiro, G.; Smyth, P. e Uthrusamy, R. *Advances in knowledge Discovery& Data Mining.*California: AAAI/MIT, 1996.

Feo, T. e Resende, M. *Greedy Randomized Adaptive Search Procedures*. Journal of Global Optimization, v. 6, n. 2, p. 109133, 1995.

Góes, A. R. T. Uma metodologia para a criação de Etiqueta de Qualidade no contexto de Descoberta de Conhecimento em Bases de Dados: aplicação nas áreas elétrica e educacional. Curitiba, PR, 2012. Tese de Doutorado, Universidade Federal do Paraná.

Góes, A. R. T. e Steiner, M. T. A. O Processo *KDD* aplicado na extração de regras: um estudo de caso da área médica. XLIV SBPO. Rio de Janeiro – RJ. Setembro 2012.

Han, J. e Kamber, M. *Data Mining: Concepts and Techniques*, 3ª ed. Morgan Kaufmann Publishers, 2011.

Kazienko, P. *Mining Indirect Associations Rules for Web Recommendation*. International Journal of Applied Mathematics and Computer Science, v. 19, n. 1, p. 165-186, 2009.

Lu, H.; Setiono, R. e Liu, H. (1996). *Effective Data Mining Using Neural Networks*. IEE Transactions on Knowledge an Data Engineering, vol. 8, n° 6, p.957-961.

Pavanelli, G. Análise do tempo de duração de processos trabalhistas utilizando redes neurais artificiais como apoio a tomada de decisões. Dissertação de Mestrado, Universidade Federal do Paraná, Curitiba, PR, 2007.

Pitsoulis, L. e Resende, M. *Greedy Randomized Adaptive Search Procedures*. In: P.M.PARDALOS; M.G.C.RESENDE (Ed.). Handbook of Applied Optimization. [S.l.]: Oxford University Press, 2002. p. 168181.

Plastino, A.; Fuchshuber, R.; Martins, S. L.; Freitas, A. A. e Salhi, S. *A hybrid data mining metaheuristic for the p-median problem*. *Statistical Analysis and Data Mining*, v. 4, p. 313-335, 2011.

Resende, M. e Ribeiro, C. *Greedy Randomized Adaptive Search Procedures*. In: GLOVER, F.; KOCHENBERGER, G. (Ed.). Handbook of Metaheuristics. [S.l.]: Kluwer Academic Publishers, 2002. p. 219249.

Resende, M. G. C. e Silva, R. M. A., Meta-Heurísticas em Pesquisa Operacional. (Ed) Omnipax, 2013.

Semaan, G. S. e Ochi, L. S. Uma heurística baseada em GRASP para a extração de associações em bases de dados, XIV SPOLM, set. 2011.

Steiner, M. T. A.; Soma, N.Y.; Shimizu, T.; Nievola, J.C. e Steiner Neto, P. J.; Abordagem de um problema médico por meio do processo de KDD com ênfase à análise exploratória dos dados. Gest Prod. 2006; 13(2):325-37

.