Mineração de regras de associação temporais quantitativas por meio de algoritmo genético

Alternative title: Quantitative temporal association rule mining by genetic algorithm

Sérgio F. da Silva [*] , Marcos A. Batista [†]
Instituto de Biotecnologia – IBiotec/UFG
Av. Dr. Lamartine P. Avelar, 1120
CEP: 75704-020, Catalão, GO
sergio@ufg.br, marcos.batista@cnpq.br

Agma J. M. Traina [‡]
Instituto de Ciências Matemáticas e de Computação – ICMC/USP
Av. Trabalhador São-carlense, 400
CEP: 13566-590, São Carlos, SP
agma@icmc.usp.br

RESUMO

Mineração de regras de associação tem mostrado grande potencial para extrair conhecimento de conjunto de dados multidimensionais. Contudo, os métodos existentes na literatura não são efetivamente aplicáveis a dados temporais quantitativos. Este artigo estende os conceitos de mineração de regras de associação da literatura. Com base nestes conceitos é apresentado um método para mineração de regras de conjuntos de dados multidimensionais temporais quantitativos por meio de algoritmo genético, denominado GTARGA em referência à *Quantitative Temporal Association Rule Mining by Genetic Algorithm*. Experimentos com QTARGA em várias bases de dados reais mostram que este permite minerar várias regras de alta confiança em uma única execução do método.

Palavras-Chave

Algoritmos genéticos, mineração de regras de associação, dados temporais quantitativos.

ABSTRACT

Association rule mining has shown great potential to extract knowledge from multidimensional data sets. However, existing methods in the literature are not effectively applicable to quantitative temporal data. This article extends the concepts of association rule mining from the literature. Based on the extended concepts is presented a method to mine rules from multidimensional temporal quantitative data sets using genetic algorithm, called GTARGA, in reference to Quantitative Temporal Association Rule Mining by Genetic Algorithm. Experiments with QTARGA in four real data sets show that it allows to mine several high-confidence rules in a single execution of the method.

Categorias e Descritores do Assunto

Intelligent Information Systems [**Data Mining**]: Genetic algorithms, Association rules mining, Temporal quantitative data

Termos Gerais

Concepts and methods

Keywords

Genetic algorithms, association rules mining, temporal quantitative data.

1. INTRODUÇÃO

Muitos fenômenos do mundo real, incluindo atividades e processos, apresentam variáveis correlacionadas. Desta forma, fenômenos reais podem ser melhor compreendidos por descobrir implicações dos valores de variáveis ao longo do tempo, ou seja, implicações de certos episódios em episódios subsequentes. Por exemplo, quebra de safras não são completamente compreendidas sem a análise de variáveis meteorológicas. Chuva ácida não é compreendida sem a análise de poluentes na atmosfera. Também, as implicações entre episódios são úteis para a geração de previsões (predição). Por exemplo, se tem-se que determinados poluentes na atmosfera acarretam chuva ácida e quantidade de tais poluentes tem aumento significativamente no decorrer dos últimos anos, então pode-se prever maior incidência de chuva ácida em um futuro próximo, com alta confiança.

Fontes de dados temporais quantitativos oriundas de atividades do mundo real (ou áreas do conhecimento) são oni-

[*]Sérgio F. da Silva é o autor principal da pesquisa.
[†]Marcos A. Batista contribuiu no desenvolvimento dos algoritmos desenvolvidos.
[‡]Agma J. M. Traina é a supervisora da pesquisa.
[*]Sérgio F. da Silva é o autor principal da pesquisa.
[†]Marcos A. Batista contribuiu no desenvolvimento dos algoritmos desenvolvidos.
[‡]Agma J. M. Traina é a supervisora da pesquisa.

SBSI 2015, May 26th-29th, 2015, Goiânia, Goiás, Brazil

presentes. Dentre algumas destas fontes pode-se citar: economia, comunicações, astronomia, energia, agronomia, meteorologia and agrometeorologia. Contudo, os dados produzidos por estas fontes têm sido praticamente inúteis devido a ausência de técnicas para extrair conhecimentos concretos e não-triviais destes. Atualmente, muitos destes dados são analisados por métodos estatísticos e/ou gráficos que têm capacidade limitada para a análise de múltiplas variáveis simultaneamente.

Nos últimos anos, vários métodos de mineração de dados temporais têm sido propostos [1, 2, 3, 4, 5, 7, 10, 11, 15, 17, 18, 19], assim como de mineração de dados quantitativos [8, 6, 12, 13, 14, 20]. Contudo, os métodos existentes na literatura não são efetivamente aplicáveis para a mineração de regras (implicações) de dados temporais quantitativos, devido a deficiências em satisfazer os seguintes critérios:

- Minerar regras (implicações temporais entre variáveis) e não somente identificar padrões em uma série temporal.
- Considerar o tempo de ocorrência de episódios e não somente a ordem em que eles ocorrem;
- Operar sem a necessidade do usuário informar parâmetros críticos tais como, limiares de suporte e de confiança das regras pretendidas;
- Minerar várias regras com diversidade e de alta confiança em uma única execução do método;
- Apresentar escalabilidade computacional, de modo a possibilitar sua aplicação para grandes conjuntos de dados.

Métodos da literatura atual não têm a capacidade de minerar regras como:

$$\langle\langle \text{Prec. Acumulada (mm)} = [150, 220], \text{mês} = [\text{Nov, Dez}]\rangle$$
$$\text{AND}$$
$$\langle \text{Temperatura Média}(^o\text{C}) = [25, 32], \text{mês} = [\text{Nov, Dez}]\rangle$$
$$\Rightarrow$$
$$\langle \text{Crescimento de Planta (cm)} = [40, 50], \text{mês} = [\text{Dez, Jan}]\rangle\rangle,$$

a qual provê conhecimento útil sobre a produção da cana-de-açúcar. Tais tipos de regras de associação são mineradas pelo método proposto neste artigo.

O método proposto, de agora em diante denominado QTARGA, em referência à *Quantitative Temporal Association Rule Mining by Genetic Algorithm*, se baseia em um algoritmo genético de código real. Algoritmos genéticos de código real têm sido bastante aplicados para mineração de regras de associação de dados quantitativos. Uma das principais dificuldades do uso de algoritmos genéticos para mineração de regras de associação, a qual tem sido ignorada em várias pesquisas, é a natureza unimodal destes algoritmos [16]. Algoritmos genéticos tradicionais são projetados para encontrar uma solução ótima ou sub-ótima. Desta forma, a população (conjunto de soluções candidatas) tende a concentrar em torno da solução ótima no decorrer das gerações (iterações) do algoritmo. Para solucionar este problema, foi usado um mecanismo de preservação de diversidade que permite ao AG explorar simultaneamente várias regiões do espaço de busca e, consequentemente, encontrar várias soluções ótimas e/ou sub-ótimas em uma única execução do método. Vale destacar que o método de preservação de diversidade, a saber, *clearing* [16], é amplamente conhecido pela comunidade de algoritmos genéticos. Contudo, para viabilizar a utilização de *clearing* foi proposta uma medida de distância entre regras, sendo a distância entre duas regras proporcional à diversidade. Além disso, ao contrário de muitos métodos da literatura, QTARGA não usa de discretização prévia dos dados em intervalos, por ter mecanismos para descobrir tais intervalos no decorrer do processo de mineração.

O método QTARGA foi validado através de experimentos em quatro conjuntos de dados reais. Os resultados obtidos mostram que QTARGA consegue minerar várias regras com alta diversidade e confiança em uma única execução do método. Além disso, QTARGA apresenta boa escalabilidade computacional, o que permite sua aplicação para bases de dados grandes.

O restante deste artigo é organizado da seguinte forma. A Seção 2 define o formato das regras de interesse e a medida de qualidade destas. A Seção 3 apresenta o método QTARGA, desenvolvido para a mineração de regras temporais quantitativas, definidas conforme especificado na Seção 2. A Seção 4 reporta e discute os resultados obtidos pelo método QTARGA em quatro conjuntos de dados. Por fim, a Seção 5 apresenta as conclusões.

2. CONSIDERAÇÕES INICIAIS E DEFINIÇÕES

Este trabalho amplia a noção de regras de associação temporais apresentada em [9] para lidar efetivamente com dados temporais quantitativos. Os conceitos apresentados em [9], os quais formam uma das bases principais para mineração de regras de associação temporais, lidam somente com variáveis binárias (ou discretas) , ex., `choveu/não_choveu` em um instante de tempo específico. Este tipo de representação é obviamente não representativo o suficiente para uma análise temporal quantitativa, pois, a intensidade de cada evento é desconsiderada. Buscando contornar essa limitação, foi proposto o conceito de ***episodeset*** como uma extensão do conceito de ***itemset*** para o domínio de dados temporais quantitativos. A seguir é definido o conceito de ***episodeset***, além de conceitos adicionais que possibilitam a mineração de regras de associação temporais quantitativas.

Definição 1. ***Episodeset***. Seja $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m\}$ o conjunto de variáveis de observação. O intervalo entre as observações é denominado granularidade de tempo (τ), a qual pode ser diária, semanal, mensal, entre outras. Seja $E = \{e_1, e_2, \ldots, e_m\}$ um conjunto de episódios registrados no instante de tempo t, onde e_i é um episódio associado à variável v_i. Um *episodeset* $E^{(t)}, 1 \leq t \leq n$ (onde n é o número de períodos de tempo do banco de dados temporal), registrado na base de dados $\mathcal{D}$ é chamado de *super-episodeset* ou *m-episodeset* pois, supõe-se que todas as variáveis de observação têm seus valores registrados[1]. Assim, considera-se

[1]Na coleta de dados do mundo real, por falhas de equipamentos ou humanas, pode acontecer de variáveis não terem determinados valores registrados, resultando em valores ausentes, ou porventura, os valores podem ser sido registrado de forma distorcida. Neste caso, normalmente utiliza-se técnicas de tratamento de valores ausentes (ou distorcidos), descarta-se aquele período ou, até mesmo, a variável.

que uma base de dados temporal quantitativa $\mathcal{D}$ é um conjunto de n super-episódios E, correspondentes a n períodos de tempo.

Definição 2. **Ciclo base.** Um ciclo base Φ_i é uma sequência contígua de *super-episodesets* da base de dados. O comprimento (ou duração) l de um ciclo base é dado pelo número de *super-episodesets* por ciclo base, sendo este pré-estabelecido. Matematicamente, cada ciclo base corresponde ao intervalo de tempo $[t_{i.l}, t_{(i+1).l})$. O número de ciclos base n_{CB} é dado por $\lfloor |\mathcal{D}|/l \rfloor$, onde $|\mathcal{D}|$ é o número de *super-episodesets* da base de dados.

Definição 3. **Regra de associação temporal quantitativa.** Uma regra de associação temporal quantitativa é uma implicação da forma $\mathbf{X} \Rightarrow \mathbf{Y}$ (se $\mathbf{X}$ então $\mathbf{Y}$), onde $\mathbf{X}$ e $\mathbf{Y}$ são conjunções de episódios associados às variáveis de observação. $\mathbf{V_X}$ denota o subconjunto de variáveis associadas às condições episódicas de $\mathbf{X}$ e $\mathbf{V_Y}$ denota o subconjunto de variáveis associadas com as condições episódicas $\mathbf{Y}$. A interseção de $\mathbf{V_X}$ e $\mathbf{V_Y}$ deve ser vazia, ie., $\mathbf{V_X} \cap \mathbf{V_Y} = \emptyset$.

Definição 4. **Condição episódica**. Uma condição episódica é uma condição intervalar de uma variável em um dado intervalo de tempo. Condições episódicas são representadas da forma : $\langle v_i(\texttt{unidade de } v_i) = [v_{0i}, v_{1i}]$ no período $(\texttt{unidade de tempo}) = [t_0, t_1]\rangle$.

Definição 5. **Suporte de uma conjunção de condições episódicas.** Uma conjunção de condições episódicas é dada por condições episódicas conectadas por pelo operador lógico AND. O suporte de uma conjunção de condições episódicas $\mathbf{X}$ representa a frequência de ocorrência de $\mathbf{X}$ na base de dados. Considerando que $\mathbf{X}$ é analisado por ciclo base, o suporte de $\mathbf{X}$ corresponde a frequência em que $\mathbf{X}$ nos ciclos base. Os *super-episodesets* de um ciclo base Φ_i, denotado por $\mathbf{E}[i]$, suportam uma conjunção de condições episódicas $\mathbf{X}$ se e somente se todas as condições episódicas de $\mathbf{X}$ ocorre em $\mathbf{E}[i]$. Matematicamente, o suporte é de $\mathbf{X}$ é definido como:

$$supp(\mathbf{X}) = \frac{\sum_{i=1}^{n_{BC}} happen(\mathbf{X}, \mathbf{E}[i])}{n_{BC}}, \qquad (1)$$

onde $happen(\mathbf{X}, \mathbf{E}[i])$ returna 1 se $\mathbf{X}$ ocorre em $\mathbf{E}[i]$ e 0, caso contrário.

3. O MÉTODO QTARGA

O método de mineração de regras de associação temporais quantitativas proposto se baseia na proposição de um algoritmo genético específico para o problema. Assim, o método é descrito em termos dos passos e operadores do algoritmo genético, o qual é dado no Algoritmo 1.

Algoritmo 1: Método QTARGA.

Entrada: Codificação de cromossomo, função de aptidão, restrições (atributos antecedentes / consequentes), comprimento de intervalo de atributo máximo ($piv.(\max \mathbf{v}_i - \min \mathbf{v}_i)$), janela de tempo minima ($janTime$)
Saída: Regras de associação temporais quantitativas correspodentes a ótimos locais e globais.
1: Gere uma população de cromossomos ($\mathcal{C}$) aleatoriamente, de acordo com a codificação de cromossomo;
2: Avalie cada cromossomo $\mathcal{C}$ da população, conforme a função de aptidão;
3: Aplique o método de *niching*;
4: Selecione os cromossomos pelo método da roleta até completar o conjunto de pais (*matting pool*);
5: Aplique *crossover* uniforme tomando pares de indivíduos do *matting pool*;
6: Aplique mutação uniforme aos cromossomos recém gerados;
7: Selecione os melhores cromossomos entre e pais e filhos para a próxima geração;
8: Enquanto o número máximo de gerações não for atingido, retorne ao passo 2.
9: Retorne o conjunto de regras de associação codificadas pela população de cromossomos.

a) Codificação de cromossomo

Na codificação de cromossomo, cada variável da base de dados é associada a um gene. Considerando m variáveis de observação, tem-se m genes: G_1, G_2, G_3, ..., G_m. Cada gene G_i de um cromossomo representa um episódio relacionado a variável $\mathbf{v}_i, i = 1 \ldots m$ e é codificado conforme ilustrado na Figura 1. Na Figura 1, w é um peso que é comparado a um limiar para indicar se a condição episódica representada pelo gene fará ou não parte da regra. AC é um *flag* para indicar se a condição episódica representada pelo gene fará parte do antecedente ($flag = 0$) ou do consequente ($flag = 1$) da regra. v_0 e v_1 são os limites inferior e superior do intervalo da variável, respectivamente. t_0 e t_1 são os limites inferior e superior do intervalo de tempo, respectivamente. Os valores v_0, v_1, t_0 e t_1 são ajustados pelo algoritmo genético, respeitando as restrições das pelos parâmetros *piv* e *janTime*, que definem os tamanhos máximos de intervalos permitidos nas regras, para valores de variáveis e de tempo, respectivamente.

w	AC	v_0	v_1	t_0	t_1

Figura 1: Representação de gene.

b) Medida de aptidão

$$\text{RelativeConfidence}(\mathbf{X} \Rightarrow \mathbf{Y}) = \frac{supp(\mathbf{X} \cup \mathbf{Y}) - supp(\mathbf{X})\ supp(\mathbf{Y})}{supp(\mathbf{X})(1 - supp(\mathbf{Y}))} \qquad (2)$$

sendo $supp(\mathbf{Z})$ calculado conforme a Equação 1.

A *Relative Confidence* (confiança relativa) mede o grau de relação entre $\mathbf{X}$ e $\mathbf{Y}$. Ela retorna valores máximos quando $\mathbf{X}$ e $\mathbf{Y}$ ocorrem simultaneamente. Esta medida de qualidade de regras foi proposta recentemente na literatura [18] e vem sendo aplicada com sucesso para a mineração de regras interessantes. Contudo, o Método QTARGA funciona para qualquer medida quantitativa de qualidade de regras de associação.

c) Operadores genéticos

A seleção para reprodução é feita por meio do método da roleta. Pares de indivíduos selecionados para reprodução são cruzados por meio de *crossover* uniforme: sorteia-se uma máscara do tamanho do cromossomo, que indica qual cromossomo pai fornecerá cada gene ao primeiro filho; o segundo filho é gerado pelo complemento da máscara. Cada

cromossomo selecionado para mutação terá um de seus genes mutados. A mutação pode ocorrer no peso w, em AC (quando não for aplicadas restrições sobre as variáveis que compõem o antecedente e o consequente), ou nos limites inferiores (v_0 e t_0) e superiores (v_1 e t_1) dos intervalos de variáveis e de tempo.

d) *Niching*

Buscando realizar uma otimização multimodal, onde busca-se por vários ótimos locais e globais simultaneamente, foi usado um método de *niching*, denominado *clearing*, descrito em [16]. Para usar esse método, foi proposta uma medida de distância em regras de associação temporais quantitativas. A seguir é descrito o método de *clearing niching* e a medida de distância proposta.

O método de *clearing* corresponde ao conceito de *niching* enunciado por J. H. Holland em 1975: o compartilhamento de recursos por uma população de indivíduos caracterizados por alguma similaridade. Porém, ao invés de compartilhar os recursos disponíveis, o método de *clearing* provê os recursos de um nicho somente ao melhor indivíduo de cada subpopulação. Isto permite ao algoritmo genético realizar uma otimização multimodal. Além disso, o método de *clearing* permite ao AG reduzir o problema de deriva genética (*genetic drift*) [16], quando usado em conjunto com um operador de seleção apropriado.

Clearing é aplicado entre a avaliação da aptidão dos cromossomos e a aplicação do operador de seleção para cruzamento. O método faz uso de uma medida de distância (dissimilaridade) entre cromossomos (cujo fenótipo corresponde a regras de associação) para determinar se eles/(elas) pertencem a uma mesma subpopulação ou não. Cada subpopulação terá um cromossomo dominante: o que tem o maior valor de aptidão na subpopulação. Se um cromossomo pertence a uma subpopulação, então sua dissimilaridade com relação ao cromossomo dominante é menor que um dado limiar σ, denominado raio de *clearing*. O método de *clearing* preserva a aptidão do cromossomo dominante enquanto que diminui para zero a aptidão dos demais cromossomos da população. Assim, o método de *clearing* atribui todos os recursos de um nicho para um único cromossomo: o vencedor (*winner*). Tal médodo corresponde a remover imaginariamene da população todos os indivíduos dominados dentro de seus nichos.

Também, o método de *clearing* é generalizável para aceitar vários vencedores, escolhidos entre os melhores indivíduos do nicho [16]. A capacidade de um nicho é definida como o número máximo de cromossomos que um nicho pode comportar. Se a capacidade de nicho for igual ao tamanho da população o efeito de *clearing* desaparece e o método de busca torna-se um GA padrão. A escolha de capacidade de nicho entre 1 e o tamanho da população oferece situações intermediárias entre o efeito de *clearing* máximo e uma busca GA padrão.

O algoritmo 2 descreve o método de *clearing* [16]. Considere $\mathcal{C}$ (população de cromossomos) e $n_{\mathcal{C}}$ (número de cromossomos da população) como sendo variáveis globais. σ é o raio de *clearing* e κ é a capacidade de cada nicho. A variável *nbWinner* armazena o número de vencedores do nicho corrente. A população de cromossomos $\mathcal{C}$ é considerada como sendo um vetor de $n_{\mathcal{C}}$ cromossomos.

O algoritmo de *clearing* (Algoritmo 2) usa três funções:

- *OrdenaFitness*($\mathcal{C}$): ordena a população de cromossomos em ordem decrescente de acordo com a aptidão.
- *Fitness*($\mathcal{C}[i]$): retorna a aptidão do i-ésimo cromossomo da população $\mathcal{C}$.
- *Distância*($f(\mathcal{C}[i])$, $f(\mathcal{C}[j])$): retorna a distância entre o fenótipo de dois cromossomos da população;
- $f(\mathcal{C}[i])$: retorna o fenótipo do cromossomo $\mathcal{C}[i]$, o qual é uma regra de associação temporal quantitativa no presente trabalho.

Algoritmo 2: *Clearing niching.*

Entrada: σ (raio de *clearing*), κ (capacidade de cada nicho).
Saída: *Clearing* – atribuição dos recursos de um nicho ao indivíduo mais apto.

```
1: OrdenarFitness(C);
2: para i = 0 até n_C-1
3:     se Fitness(C[i]) > 0
4:         nbWinners = 1;
5:         para j = i + 1 até n_C − 1
6:             se Fitness(C[j]) > 0 AND Distância(f(C[i]),
   f(C[j])) < σ
7:                 se nbWinners < κ
8:                     nbWinners = nbWinners + 1;
9:                 senão Fitness(C[j]) = 0;
10:                    Fitness(C[j]) = 0;
```

e) Calculando a distância entre regras de associação

Sejam duas regras de associação:
$R = (CA_R(v_{j_1})$ AND ... AND $CA_R(v_{j_n})) \Rightarrow (CC^1_R(v_{j_1})$ AND ... AND $CC_R(v_{j_m}))$ e
$S = (CA_S(v_{j_1})$ AND ... AND $CA_S)(v_{j_o}) \Rightarrow (CC^1_S(v_{j_1})$ AND ... AND $CS_S(v_{j_p}))$, onde $CA_R(v_{j_i})$, $CC_R(v_{j_i})$, $CA_S(v_{j_i})$, $CC_S(v_{j_i})$ são condições episódicas relacionadas às variáveis v_{j_i} da base de dados. Cada condição episódica tem a forma:

$$v_i \in [v0_i, v1_i] \text{ no intervalo de tempo } [t0_i, t1_i]$$

onde v_i é uma variável qualquer da base de dados. Para uma dada regra R, a interseção das variáveis associadas às condições episódicas do antecedente (VA_R) com as variáveis associadas às condições episódicas do consequente (VC_R) deve ser vazia, isto é, $\mathrm{VA}_R \cap \mathrm{VC}_R = \emptyset$.

A distância entre R e S, denotada por *Distance*(R, S) é dada pelo algoritmo 3. Para cada condição episódica no antecedente da regra R, CA_R, é verificado se existe alguma condição episódica no antecedente da regra S, CA_S, que seja comparável com CA_R. Duas condições episódicas são comparáveis se elas se referem à mesma variável. Se existe duas condições episódicas comparáveis, calcula-se a distância entre elas. Senão incrementa-se 1 (um) no contador de distância. Em seguida o contador de distância é divido pelo número de condições episódicas n_{CA_R}. O mesmo cálculo é feito para as condições episódicas do consequente. Finalmente a distância é entre as regras é dada por $(dist_A + dist_C)/2$.

A distância entre condições episódicas, *DistanciaEp(C1, C2)*, é dada pelo Algoritmo 4.

4. RESULTADOS

Nesta seção são reportados quatro casos de estudo realizados para a validação da técnica desenvolvida. Todos os

Algoritmo 3: *Distância*(R, S) – Distância entre duas regras de associação temporais quantitativas.

Entrada: R, S (duas regras de associação temporais quantitativas).
Saída: distância entre R e S (dist).

$n_{CA_R} = numCond(CA_R)$;
$dist_A = 0$;
para i $= 0$ **até** $n_{CA_R} - 1$
  se $\exists CA_S(v_{j_k})$ tal que $v_{j_k} == v_{j_i}, v_{j_i} \in VA_R$
    $dist_A = dist_A + DistanciaEp(CA_R(v_{j_i}), CA_S(v_{j_k}))$;
  senão
    $dist_A = dist_A + 1$;
$dist_A = dist_A / n_{CA_R}$;

$n_{CC_R} = numCond(CC_R)$;
$dist_C = 0$;
para i $= 0$ **até** $n_{CC_R} - 1$
  se $\exists CC_S(v_{j_k})$ tal que $v_{j_k} == v_{j_i}, v_{j_i} \in VC_R$
    $dist_C = dist_C + DistanciaEp(CC_R(v_{j_i}), CC_S(v_{j_k}))$;
  senão
    $dist_C = dist_C + 1$;
$dist_C = dist_C / n_{CC_R}$
$dist = (dist_A + dist_C)/2$;

Algoritmo 4: *DistanciaEp*(C_1, C_2) – Distância entre dois episódios temporais associados à mesma variável.

Entrada: C_1, C_2, v (duas condições episódicas associadas a uma mesma variável v).
Saída: distância entre $C1$ e $C2$, representada por (*distEp*).

calcular:
  $\min v$ /*valor mínimo que v assume na base de dados*/
  $\max v$ /*valor máximo que v assume na base de dados*/
  $\min t$ /*valor mínimo que t assume na base de dados*/
  $\max t$ /*valor máximo que t assume na base de dados*/
$dv = \min\{(v_1^{(C_1)} - v_0^{(C_1)}), v_1^{(C_2)} - v_0^{(C_2)})\} - (\min\{v_1^{(C_1)}, v_1^{(C_2)}\} - \max\{v_0^{(C_1)}, v_0^{(C_2)}\})$;
$distV = dv/(\max v - \min v)$;
$dt = \min\{(t_1^{(C_1)} - t_0^{(C_1)}), t_1^{(C_2)} - t_0^{(C_2)})\} - (\min\{t_1^{(C_1)}, t_1^{(C_2)}\} - \max\{t_0^{(C_1)}, t_0^{(C_1)}\})$
$distT = dt/(\max t - \min t)$;
$distEp = (distV + distT)/2$;

experimentos foram processados em um MacBook Pro, Processador 2.9 GHz Intel Core i7 com 8GB de memória DDR3 1600 MHz usando o sistema operacional OS X 10.8.3. Os métodos foram implementados em linguagem C. Nos casos de estudo foram analisados o número de regras mineradas (*numRules*) e o tempo de execução (*exec. time*) dado em segundos, em função de configurações da técnica. Na configuração da técnica foi considerado variações no tamanho de população de cromossomos ($n_{\mathcal{C}}$), o número de gerações (*nGen*), o raio de *clearing* (σ) e o tamanho máximo de intervalo de variável admissível. O tamanho máximo de intervalo de variável admissível é definido por $piv.(\max v - \min v)$, onde *piv* é uma porcentagem. Nos resultados é reportado somente o valor de *piv* utilizado. Os demais parâmetros do AG, mantidos fixos nos casos de estudo, são: taxa de cruzamento de 80% e taxa de mutação de 3% por cromossomo. Os resultados reportados corresponde a uma média de três execuções para cada configuração da técnica.

4.1 Caso de estudo 1: *Stock Prices*

Neste caso de estudo foi utilizada a base de dados *Stock Prices* (http://www.stat.ucla.edu/cases/), que corresponde aos preços de ações diários de janeiro de 1988 à outubro de 1991, de dez companhias aéreas dos Estados Unidos (dez variáveis, pois o preço de ações de cada companhia é uma variável). Neste caso, o ciclo base é semanal, ou seja, $l = 7$.

Neste caso de estudo foi imposta a restrição que a companhia10 (company10) fará parte do consequente da regra. A tabela 1 mostra os resultados obtidos em termos da configuração da técnica (especificada no ínicio da seção de resultados), do número de regras mineradas e do tempo de execução. As conclusões gerais dos experimentos são dadas na subseção 4.5.

A Figura 2 mostra algumas regras mineradas para a base de dados *Stock Prices*. Pode-se verificar uma correlação entre os preços de ações das companhias 3 e 10: quando o preço de ações da companhia 3 (*company3*) é relativamente alto, o preço de ações da companhia 10 (*company10*) é relativamente baixo; e quando o preço de ações da companhia 3 (*company3*) é relativamente baixo, o preço de ações da companhia 10 (*company10*) é relativamente alto.

$n_{\mathcal{C}}$	*nGen*	σ	*piv*	*numRules*	*exec. time (sec.)*
100	350	0.3	0.1	44	4.18
100	350	0.3	0.2	36	3.76
100	350	0.5	0.1	63	4.05
100	350	0.5	0.2	54	4.20
50	250	0.3	0.1	28	0.96
50	250	0.3	0.2	23	0.90
50	250	0.5	0.1	23	0.83
50	250	0.5	0.2	31	0.98

Tabela 1: Configuração do AG *versus* número de regras mineradas (*numRules*) *versus* tempo de execução (*exec. time*) em segundos para a base de dados *Stock Prices*.

4.2 Caso de estudo 2: Araraquara

Neste caso de estudo foi utilizada a base de dados denominada Araraquara, coletada pelo Sistema de Monitoramento Agrometeorológico – Agritempo (http://www.agritempo.gov.br/). Ela contém dados agrometeorológicos mensais de Araraquara correspondentes aos valores de média da temperatura mínima (*Tmin*), média datemperatura máxima (*Tmax*), precipitação acumulada (*Prec*), *Normalized Difference Vegetation Index* (*NDVI*) médio e *Water Requirement Satisfaction Index* (*WRSI*) médio. Os dados correspondem ao período de abril de 2001 a janeiro de 2008. Neste caso, o ciclo base é anual, ou seja, $l = 12$.

Neste caso de estudo foi imposta a restrição que o *NDVI* fará parte do consequente da regra. A tabela 2 sumariza os resultados obtidos.

4.3 Caso de estudo 3: Piracicaba

Neste caso de estudo foi utilizada a base de dados Piracicaba, coletada pela Embrapa (Empresa Brasileira de Pesquisa Agropecuária). A base de dados consiste de três va-

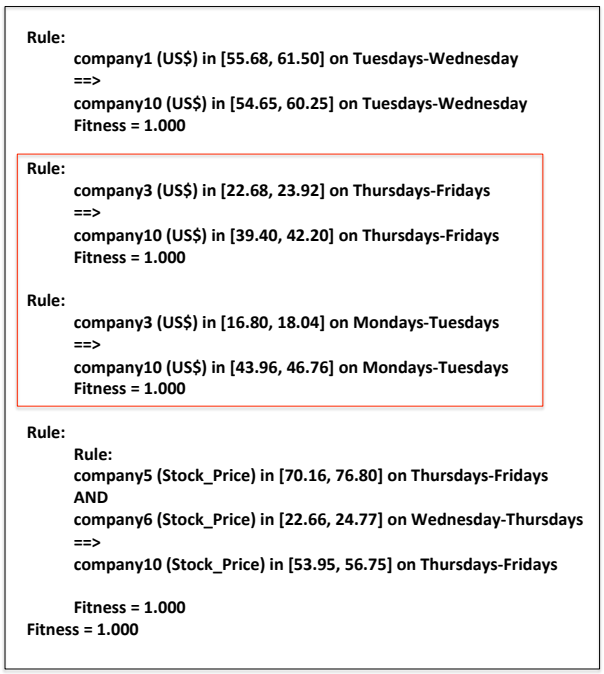

Figura 2: Exemplo de regras mineradas para a base de dados *Stock Prices*.

n_C	$nGen$	σ	piv	$numRules$	$exec.\ time\ (sec.)$
100	350	0.3	0.1	11	0.49
100	350	0.3	0.2	17	0.63
100	350	0.5	0.1	4	0.26
100	350	0.5	0.2	6	0.29
50	250	0.3	0.1	8	0.17
50	250	0.3	0.2	10	0.16
50	250	0.5	0.1	4	0.11
50	250	0.5	0.2	4	0.12

Tabela 2: Configuração AG *versus* número de regras mineradas (*numRules*) *versus* tempo de execução (*exec. time*) em segundos para a base de dados Piracicaba.

riáveis tomadas mensalmente: o valor médio de temperatura máxima ($Tmax$), o valor médio de temperatura mínima ($Tmin$), e a precipitação acumulada ($Prec$). Os dados correspondem a um período de 47 anos (de 1961 a 2008). Neste caso, o ciclo base é anual, ou seja, $l = 12$.

Neste caso de estudo foi imposta a restrição que a precipitação ($Prec$) fará parte do consequente da regra. A tabela 3 sumariza os resultados obtidos.

4.4 Caso de estudo 4: Produtividade da cana-de-açúcar em Piracicaba

Neste caso de estudo foi utilizada uma base de dados de produtividade da cana-de-açúcar em Piracicaba, fornecida pelo Cepagri (Centro de Pesquisas Agrometeorológicas e Climáticas Aplicadas a Agricultura)-UNICAMP. A base de dados consiste de quatro variáveis tomadas mensalmente no município de Piracicaba: média da temperatura mínima ($Tmin$), média da temperatura máxima ($Tmax$), precipitação média ($Prec$) e produtividade da cana-de-açúcar ($Prod$) em toneladas por hectare (ton/hec). A base de dados corresponde ao período de 2003 a 2009. Também neste caso, o

n_C	$nGen$	σ	piv	$numRules$	$exec.\ time\ (sec.)$
100	350	0.3	0.1	11	0.49
100	350	0.3	0.2	17	0.63
100	350	0.5	0.1	4	0.26
100	350	0.5	0.2	6	0.29
50	250	0.3	0.1	8	0.17
50	250	0.3	0.2	10	0.16
50	250	0.5	0.1	4	0.11
50	250	0.5	0.2	4	0.12

Tabela 3: Configuração do AG *versus* número de regras mineradas (*numRules*) *versus* tempo de execução (*exec. time*) em segundos para a base de dados Piracicaba.

ciclo base é anual, ou seja, $l = 12$.

No presente caso de estudo foi imposta a restrição que a produtividade ($Prod$) fará parte do consequente da regra. A tabela 3 sumariza os resultados obtidos.

n_C	$nGen$	σ	piv	$numRules$	$exec.\ time\ (sec.)$
100	350	0.3	0.1	10	0.29
100	350	0.3	0.2	14	0.34
100	350	0.5	0.1	4	0.12
100	350	0.5	0.2	8	0.20
50	250	0.3	0.1	9	0.12
50	250	0.3	0.2	8	0.12
50	250	0.5	0.1	4	0.06
50	250	0.5	0.2	5	0.08

Tabela 4: Configuração do AG *versus* número de regras mineradas (*numRules*) *versus* tempo de execução (*exec. time*) em segundos para a base de dados Produtividade da cana-de-açúcar em Piracicaba.

4.5 Análise dos resultados

No geral pode-se perceber que a técnica consegue minerar várias regras de associação, sendo elas diversas e de alta qualidade, de acordo com a medida de aptidão. Em média a aptidão das regras mineradas é próxima de 0.99, sendo que o valor máximo da função de aptidão (equação 2) é 1 (um). A configuração do tamanho de população (n_C) e número de gerações ($nGen$) se comportou conforme o esperado: quando maior os valores destes parâmetros, maior o número de regras mineradas. Contudo, aumentando-se os valores de n_C e $nGen$, aumenta-se o tempo de execução do método, conforme é esperado.

A intuição quanto raio de *clearing* é que, quanto maior este, maior é capacidade do ambiente para comportar nichos, e consequentemente, maior a quantidade de regras mineradas. A configuração deste parâmetro se comportou em média conforme o esperado, contudo o caso de estudo 1 foi uma exceção. Esta exceção aconteceu devido ao grande número de padrões (regras) existentes na base *Stoke Prices*. Este fato foi detectado aumentando o tamanho de população n_C. Com populações a partir de 300 cromossomos, o ajuste do raio de *clearing* (σ) passa a comportar conforme o esperado: quanto menor o raio de clearing, maior o número de regras mineradas. O parâmetro piv também se comportou conforme o esperado; em média, quanto maior o tamanho de intervalo admissível, maior o número de regras mineradas. Contudo, se piv for muito grade (próximo de 1), serão mineradas regras que não expressam nenhum conhecimento concreto (específico). Por exemplo, `temperatura (`^oC`) [0-42]`, sempre ocorre em Araraquara.

5. CONCLUSÕES

Partindo de conceitos existentes na literatura sobre mineração de regras de associações temporais, foi proposta uma extensão para lidar com dados temporais quantitativos, de forma natural e sem perda de informação. Para demostrar a validade dos conceitos propostos, foi elaborado um algoritmo genético que funciona de acordo com tais conceitos de forma a identificar implicações temporais entre episódios. Vale destacar que o algoritmo genético proposto não necessita de discretização prévia dos dados em intervalos, assim como, da especificação de parâmetros críticos específicos de bases de dados como os limiares de suporte e confiança, necessários em algoritmos de mineração de regras de associação clássicos. Como pode ser notado, o algoritmo genético proposto tem vários parâmetros ajustáveis, contudo, o ajuste destes parâmetros é intuitivo, conforme descrito na subseção 4.5. A limitação clássica de algoritmos genéticos tradicionais, de realizar busca unimodal, foi tratada através da utilização do mecanismo clássico de *clearing* para preservação de diversidade. Os resultados obtidos mostram que é possível minerar várias regras de alta qualidade em uma única execução do método.

Também vale destacar que, conforme é de conhecimentos dos autores, o tipo de regras mineradas pela abordagem proposta, além de altamente informativa, não é identificada por nenhum método de mineração regras de associação da literatura. Assim, o método proposto não foi comparado com métodos da literatura. Espera-se que os conceitos propostos sejam uma base para definição de novos métodos de mineração de regras de associação temporais quantitativos. Além disso, o método proposto pode ser aplicado em bases de dados temporais quantitativos advindos de variadas áreas do conhecimento, de forma a descobrir conhecimentos valiosos e não-triviais.

6. AGRADECIMENTOS

Agradecemos a FAPEG (Proc. 201210267900) e ao CNPq (Proc. 447984/2014-4 e Proc. 479792/2012-7) pelo suporte financeiro.

7. REFERÊNCIAS

[1] S. Amo, N. A. Silva, R. P. Silva, and F. S. Pereira. Tree pattern mining with tree automata constraints. *Information Systems*, 35:570–591, 2010.

[2] B. Catania and A. Maddalena. A unified framework for heterogeneous patterns. *Information Systems*, 37:460–483, 2012.

[3] D.-A. Chiang, C.-T. Wang, S.-P. Chen, and C.-C. Chen. The cyclic model analysis on sequential patterns. *IEEE Transactions on Knowledge and Data Engineering*, 21(11):1617–1628, 2009.

[4] J. K. Febrer-Hernández and J. Hernández-Palancar. Sequential pattern mining algorithms review. *Intelligent Data Analysis*, 16:451–466, 2012.

[5] T. Fu. A review on time series data mining. *Engineering Applications of Artificial Intelligence*, 24(1):164–181, 2011.

[6] E. Georgii, L. Richter, U. Rückert, and S. Kramer. Analyzing microarray data using quantitative association rules. *Bioinformatics*, 21(suppl 2):ii123–ii129, 2005.

[7] Y. Hai and X. Li. A general temporal association rule frequent itemsets mining algorithm. *International Journal of Advancements in Computing Technology*, 3(11):63–71, 2011.

[8] T.-P. Hong, C.-S. Kuo, and S.-C. Chi. Mining association rules from quantitative data. *Intelligent Data Analysis*, 3(5):363–376, 1999.

[9] C.-H. Lee, M.-S. Chen, and C.-R. Lin. Progressive partition miner: an efficient algorithm for mining general temporal association rules. *IEEE Transaction on Knowledge and Data Engineering*, 15(4):1004–1017, 2003.

[10] Y. J. Lee, J. W. Lee, D. J. Chai, B. H. Hwang, and K. H. Ryu. Mining temporal interval relational rules from temporal data. *Journal of Systems and Software*, 82(1):155–167, 2009.

[11] N. R. Mabroukeh and C. I. Ezeife. A taxonomy of sequential pattern mining algorithms. *ACM Computing Surveys*, 43(1):1–41, 2010.

[12] D. Martin, A. Rosete, J. Alcala-Fdez, and F. Herrera. A new multiobjective evolutionary algorithm for mining a reduced set of interesting positive and negative quantitative association rules. *Evolutionary Computation, IEEE Transactions on*, 18(1):54–69, 2014.

[13] M. Martínez-Ballesteros, A. Troncoso, F. Martínez-Álvarez, and J. C. Riquelme. Mining quantitative association rules based on evolutionary computation and its application to atmospheric pollution. *Integrated Computer-Aided Engineering*, 17(3):227–242, 2010.

[14] B. Pei, S. Zhao, H. Chen, X. Zhou, and D. Chen. Farp: Mining fuzzy association rules from a probabilistic quantitative database. *Information Sciences*, 237:242 – 260, 2013.

[15] M. Plantevit, A. Laurent, D. Laurent, M. Teisseire, and Y. W. Choong. Mining multidimensional and multilevel sequential patterns. *ACM Transactions on Knowledge Discovery from Data*, 4(1):1–37, 2010.

[16] A. Pétrowski. A clearing procedure as a niching method for genetic algorithms. In *Proceedings of 3rd IEEE International Conference on Evolutionary Computation*, pages 798–803, New York, USA, 1996.

[17] N. Tatti and B. Cule. Mining closed strict episodes. *Data Mining and Knowledge Discovery*, 25(1):34–66, 2011.

[18] X. Yan, C. Zhang, and S. Zhang. Genetic algorithm-based strategy for identifying association rules without specifying actual minimum support. *Expert Systems with Applications*, 36:3066–3076, 2009.

[19] M. Zhang and C. He. Survey on association rules mining algorithms. *Advancing Computing, Communication, Control and Management*, 56:111–118, 2010.

[20] H. Zheng, J. He, G. Huang, and Y. Zhang. Optimized fuzzy association rule mining for quantitative data. In *2014 IEEE International Conference on Fuzzy Systems*, pages 396–403, Beijing, China, 2014.