UM ALGORITMO DE OTIMIZAÇÃO LIVRE DE DERIVADAS PARA A PROTEÇÃO DE DADOS INDIVIDUAIS

Arthur Nóbrega Baptista de Araújo
Universidade Federal do Rio Grande do Norte
Campus Universitário s/n, Natal-RN, Brasil, 59072-970
araujo.arthur@ymail.com

Daniel Aloise
Universidade Federal do Rio Grande do Norte
Campus Universitário s/n, Natal-RN, Brasil, 59072-970
aloise@dca.ufrn.br

RESUMO

O maior desafio ao divulgar dados privados é compartilhar informações contidas em bancos de dados e ao mesmo tempo proteger as pessoas de serem identificadas individualmente. Microagregação é uma família de métodos para o controle da divulgação desses dados. O princípio da microagregação é que regras de confidencialidade permitam a publicação de registros individuais divididos em *clusters* de *g* ou mais dados, onde nenhum registro é mais representativo do que os outros do mesmo *cluster*. A aplicação de tais regras leva à substituição de valores individuais por aqueles calculados a partir dos *clusters*. O processo de *clusterização* deve ser desenvolvido para reduzir a perda de informação causada por essa substituição. Este trabalho propõe um algoritmo livre de derivadas para microagregação de dados numéricos baseado em avaliações de uma função substituta (*surrogate)*. Experimentos computacionais mostram que o algoritmo leva a perdas mínimas de informação para algumas instâncias *benchmark* da literatura.

PALAVARAS-CHAVE. Microagregação, Proteção de dados, Otimização livre de derivadas.

ABSTRACT

The biggest challenge when disclosing private data is to share information contained in databaseswhile protecting people from being individually identified. Microaggregation is a family of methods for statistical disclosure control. The principle of microaggregation is that confidentiality rules permit the publication of individual records if they are partitioned into groups of g or more data, where none is more representative than the others in the same group. The application of such rules leads to replacing individual values by those computed from small groups, denoted microaggregates, before data publication. The microaggregation procedure should be developed to reduce as much as possible the information loss caused by this replacement process. This work proposes a free-derivative optimization algorithm based on the evaluation of a simplified surrogate. Computational experiments show that the proposed method finds minimum information losses for a number of benchmark instances in the literature.

KEYWORDS. Microaggregation, Data privacy, Free-derivative optimization.

1. Introdução

O aumento da quantidade de dados gerados a partir do preenchimento de questionários e formulários *online* em nossa sociedade exige uma reflexão sobre o acesso de organizações públicas e privadas a todas essas informações. A mineração de dados é uma forma automatizada de explorar essa valiosa fonte de conhecimento e, devido à sua grande aplicabilidade, muitas vezes levanta preocupações sobre a privacidade de dados individuais (KANTARCIOGLU *et al.*, 2004; TAVANI, 1999).

O objetivo do Controle Estatístico de Divulgação (CED) é permitir que os dados sejam explorados ao passo que a segurança de informações privadas seja garantida (WILLENBORG e DeWAAL, 2001). De fato, a melhor forma de proteção de dados é através da criptografia, entretanto, dados criptografados não podem ser utilizados por técnicas de mineração de dados. O CED atua no *tradeoff* entre a utilização de dados e a proteção fornecida pela criptografia.

As técnicas de CED muitas vezes resultam na modificação dos dados antes de se tornarem públicos. Esta modificação pode ser produzida a partir do mascaramento de dados ou da geração de dados sintéticos que preservam algumas propriedades estatísticas dos dados originais. Os métodos de mascaramento do CED podem se classificar em duas classes: perturbativos, para os quais os dados são realmente modificados antes da sua publicação; ou não perturbativos, onde os dados são parcialmente publicados (supressão de alguns atributos) ou há apenas a publicação de uma amostra dos mesmos. Microagregação é uma classe de métodos perturbativos de CED que tem sido extensivamente estudada (o artigo seminal de Domingo-Ferrer e Mateo-Sanz (2002) já foi citado mais de 350 vezes segundo o Google Scholar). O princípio da microagregação é que os registros individuais podem ser substituídos por valores calculados em pequenos grupos homogêneos (*clusters*), antes de sua divulgação. Uma vez que o conjunto protegido de dados contém apenas os dados mascarados, é menos provável que sua divulgação viole a privacidade dos indivíduos. A microagregação pode ser feita de diversas formas, seja para dados numéricos ou categóricos, e suas variações diferenciam-se pela forma como o método específico lida com as três seguintes questões a seguir: (i) como os registros individuais são mascarados? (ii) como a perda de informação é minimizada? e (iii) como a privacidade dos dados é garantida?

Domingo-Ferrer e Mateo-Sanz (2002) definem um modelo de programação matemática para microagregação de dados numéricos. Nesse modelo, os dados são agrupados através do critério da soma mínima dos quadrados em *clusters* de tamanho maior ou igual a um parâmetro g usando tantos *clusters* k quanto necessário. O modelo pode ser expresso da seguinte forma:

$$SEQ = \min_{x,y,k} \sum_{i=1}^{n} \sum_{j=1}^{k} x_{ij} \left\| p_i - y_j \right\|^2 \quad (1)$$

$$\text{s.à.: } \sum_{j=1}^{k} x_{ij} = 1, \quad \forall i = 1, \dots, n \quad (2)$$

$$\sum_{i=1}^{n} x_{ij} \geq g, \quad \forall j = 1, \dots, k \quad (3)$$

$$x_{ij} \in \{0,1\}, \forall i = 1, \dots, n; \ \forall j = 1, \dots, k \ ; y_j \in \mathbb{R}^s, \forall j = 1, \dots, k; \ k \in \mathbb{N}^* \quad (4)$$

Os dados numéricos com n registros individuais são representados por pontos $p_i = (p_i^r, r = 1, \dots, s)$ em $\mathbb{R}^s$para $i = 1, \dots, n$; k centros de *clusters* devem estar localizados em pontos desconhecidos $y_j \in \mathbb{R}^s$, para $j = 1, \dots, k$; a norma $\| \cdot \|$ indica a distância euclidiana entre os dois pontos em seu argumento considerando um espaço Euclidiano com s dimensões. Os variáveis de decisão binárias x_{ij} expressam a atribuição do ponto p_i ao *cluster* j. O conjunto de restrições (2) garante que cada ponto p_i, $i = 1, \dots, n$, seja atribuído a exatamente um *cluster*. As restrições em (3) definem que o tamanho de cada *cluster* seja maior ou igual a g.

O modelo definido em (1)-(4) lida com as questões (i)-(iii) da seguinte maneira: em

primeiro lugar, para um *x* fixo, condições de primeira ordem aplicadas sobre o gradiente da função objetivo exigem que, em uma solução ótima, as variáveis *y* estejam sempre nos centróides dos *clusters*. Assim, cada dado pode ser mascarado para as coordenadas de seu centróide antes da divulgação, o que responde à questão (i). No que diz respeito à questão (ii), a soma dos erros quadrados (SEQ) é um critério que permite expressar tanto a homogeneidade quanto a separação dos *clusters* formados (SPATH, 1980). Homogeneidade significa que pontos no mesmo *cluster* devem ser similares e separação que pontos em *clusters* distintos devem ser diferentes entre si. Desta forma, a microagregação com o critério SEQ permite maximizar a preservação das características dos dados originais. Finalmente, o modelo definido em (1)-(4) aborda a questão (iii) por meio do conjunto de restrições (3). Sem ele, o modelo é trivialmente resolvido localizando um centro de *cluster* na posição de cada ponto. No entanto, isto não é interessante sob o ponto de vista da proteção de dados, pois se os pontos e centróides são os mesmos, os dados não são mascarados e, portanto, não são protegidos. As restrições em (3) são definidas para evitar a divulgação de informações individuais, porque agrupam os dados em *clusters* de tamanho igual ou superior a *g*. Assim, cada dado está "escondido em seu *cluster*". Quanto maior o valor de *g*, maior a proteção desejada.

O problema é NP-difícil para uma dimensão $s > 1$ qualquer (OGANIAN e DOMINGO-FERRER, 2001). Por se tratar de um problema NP-difícil, muitas heurísticas tem sido propostas na literatura para o modelo em questão (veja, por exemplo, DOMINGO-FERRER *et al*, 2006; PANAGIOTAKIS e TZIRITAS, 2012 para um survey). Neste trabalho, é proposto um algoritmo de otimização livre de derivadas para o modelo (1)-(4). A estrutura do artigo está organizada da seguinte forma: a Seção 2 faz uma breve explicação de duas heurísticas “means” conhecidas e usadas na literatura para o problema de *clusterização* por SEQ: *k*-means e *h*-means (HANSEN e MLADENOVIC, 2001). Apresenta, ainda, a nossa adaptação delas para abordar o modelo de microagregação (1)-(4). As duas heurísticas adaptadas são usadas dentro de um algoritmo de otimização livre de derivadas apresentado na Seção 3, que procura o melhor valor de *k* através de avaliações de uma função SEQ substituta (*surrogate*) simplificada. Experimentos computacionais em conjuntos de dados de referência são relatados na Seção 4. Finalmente, as conclusões são apresentadas na Seção 5.

2. Heurísticas *k*-means e *h*-means para microagregação numérica

As heurísticas *k*-means e *h*-means são amplamente utilizadas na literatura devido à sua simplicidade e convergência rápida para um ótimo local observada na prática. A partir de um conjunto inicial de *k* pontos vistos como centróides iniciais (ou uma partição inicial), o *k*-means procede (após calcular esses centróides iniciais, se necessário) reatribuindo os pontos a seus centróides mais próximos e atualizando suas posições até a estabilidade ser atingida, o que ocorre quando nenhuma reatribuição que melhore a solução seja possível. Por sua vez, a heurística *h*-means funciona da seguinte maneira: uma partição inicial é escolhida aleatoriamente, então se consideram realocamentos de um ponto de cada vez. Tais alterações são calculadas para todos os deslocamentos possíveis, ou seja, testam-se as mudanças de cada ponto para cada *cluster*. Se existe uma ou mais mudanças que causem melhorias, a melhor é feita, e os valores dos centróides envolvidos são atualizados. Caso contrário, a heurística para.

As heurísticas (*k*, *h*)-means padrões não podem ser diretamente utilizadas para o problema de microagregação (1)-(4) devido às seguintes limitações: (a) trabalham com um número pré-definido de *clusters*, e (b) não levam em consideração as restrições de cardinalidade dos *clusters*.

Apresentamos nesta seção duas estratégias diferentes para contornar (b), enquanto (a) é abordado na próxima seção. A primeira estratégia restringe deslocamentos de pontos de dados apenas para aqueles que produzem uma solução viável para o problema (1)-(4), ou seja, se uma mudança deixa um *cluster* com menos de *g* elementos, o movimento é proibido. Assim, a heurística para se todos os remanejamentos permitidos não melhorarem o custo da solução.

Na segunda estratégia, a restrição de cardinalidade é satisfeita através da aplicação de uma rotina de viabilidade. Neste caso, primeiramente, as heurísticas *k*-means e *h*-means padrão são executadas. Em sequência, para cada *cluster j* da solução com menos de *g* elementos, atribui-se um novo elemento – proveniente de *clusters* com mais de *g* elementos – cuja realocação cause a menor deterioração no custo da solução atual. Observe que é sempre possível encontrar tal elemento quando $n / k \geq g$. As heurísticas modificadas com esta estratégia são interrompidas quando, após a aplicação da rotina de viabilidade, o algoritmo não é capaz de melhorar a solução *viável* da iteração anterior.

Quando o k-means e o h-means são modificados com a primeira estratégia, uma segunda condição de parada é usada em conjunto com a condição de se chegar à estabilidade. Desta forma, as heurísticas modificadas com essa estratégia são interrompidas também quando os deslocamentos que melhoram a solução deixam a partição resultante inviável. Logo, esta condição de parada combinada faz com que os algoritmos sejam mais rápidos do que suas versões originais. Já por meio da segunda estratégia, o emprego de uma rotina de viabilidade no final do k-means e h-means significa mais tempo para a conclusão do algoritmo. No entanto, a rotina de viabilidade é executada em tempo O(kn), não resultando em aumento da complexidade geral dos algoritmos modificados.

3. Um algoritmo de otimização livre de derivadas

O valor exato de *k* para ser usado pela heurística (*k*, *h*)-means modificada (Algoritmos 1 e 2) permanece desconhecido. Uma abordagem possível consiste em aplicar os algoritmos para cada valor possível de *k* e armazenar a melhor solução obtida.

A partir de Domingo-Ferrer e Mateo-Sanz (2002), sabe-se que os tamanhos dos *clusters* na solução ótima estão entre *g* e *2g - 1*. De posse desta informação, é possível definir um intervalo onde o valor ótimo de *k* se encontra e, então, aplicar as heurísticas apenas para os valores dentro do intervalo.

Proposição 1 *Uma solução ótima para o problema de microagregação definido em (1)-(4) existe de forma que o número de clusters k* é maior ou igual a* $\left\lfloor \frac{n}{2g-1} \right\rfloor$ *e menor ou igual a* $\left\lfloor \frac{n}{g} \right\rfloor$.

Prova: Começaremos provando que o limite superior de *k** é $\left\lfloor \frac{n}{g} \right\rfloor$. Suponha que $k^* > \left\lfloor \frac{n}{g} \right\rfloor$. Isso implica que pelo menos um *cluster* é composto por (*n* mod *g*) pontos. Consequentemente, esse *cluster* tem cardinalidade menor que *g*, o que é uma contradição. Para o limite inferior, suponha que $k^* < \left\lfloor \frac{n}{2g-1} \right\rfloor$, ou seja, $k^* \leq \left\lfloor \frac{n}{2g-1} \right\rfloor - 1$. Dado que o tamanho máximo de um *cluster* numa partição ótima é *2g – 1*, temos:

$$n \leq \left(\left\lfloor \frac{n}{2g-1} \right\rfloor - 1 \right)(2g-1) \leq \left(\frac{n}{2g-1} - 1 \right)(2g-1) \leq n - 2g + 1 < n.$$

A demonstração acima prova a contradição, ou seja, $\left\lfloor \frac{n}{2g-1} \right\rfloor \leq k^* \leq \left\lfloor \frac{n}{g} \right\rfloor$ □

Com a Proposição 1, pode-se argumentar que o problema (1)-(4) poderia ser resolvido através de uma sequência de subproblemas, obtidos fixando-se *k* para cada um dos valores inteiros entre $\left\lfloor \frac{n}{2g-1} \right\rfloor$ e $\left\lfloor \frac{n}{g} \right\rfloor$. Entretanto, tais subproblemas são tão difíceis de resolver quanto o problema (1)-(4), conforme demonstra a Proposição 2.

Proposição 2 *O subproblema de microagregação definido pela fixação do valor de k no modelo (1)-(4) é NP-difícil para dimensões gerais maiores do que 1.*

Prova: A dedução vem do problema de microagregação (1)-(4) em si, e baseia-se no seguinte raciocínio: o problema de microagregação (1)-(4) pode ser minimizado através da resolução

de uma sequência de subproblemas, sendo os subproblemas obtidos fixando-se k inteiro com $\left\lfloor \frac{n}{2g-1} \right\rfloor \leq k \leq \left\lfloor \frac{n}{g} \right\rfloor$, e salvando a solução mínima para cada um deles. Como este número de subproblemas é polinomialmente delimitado em n, provamos que o problema (1)-(4) é reduzido em tempo polinomial para seus subproblemas com k fixo □

Seja F: $k \rightarrow SEQ_k$, uma função que, para cada $k \in \mathbb{N}^*$, devolve o valor de SEQ_k em (8). A análise das funções F: $k \rightarrow SEQ_k$ não revela muito sobre o valor ótimo de k para (1)-(4). A função F aparenta ser multimodal, em geral. Além disso, sabemos a partir da Proposição 2 que resolver os subproblemas (8) para todos os valores de k qualificados é impraticável para grandes instâncias. Com isso em mente, propomos aqui a otimização de um substituto simplificado (*surrogate*) (AUDET *et al*, 2010) de F, denominado F', que se aproxima da verdadeira função F. Nossa ideia é usar as heurísticas (k, h)-means modificadas, descritas na seção anterior, para avaliar heuristicamente F. Para cada valor qualificado de k, k-means, h-means ou qualquer combinação dos dois é utilizada para obter a imagem aproximada de k em F. Assim, F' é avaliada em tempo computacional muito inferior à avaliação verdadeira função F.

O modelo heurístico proposto para minimizar (1)-(4) é constituído por um algoritmo livre de derivadas para otimização univariada (NOCEDAL e WRIGHT, 2000). Ele procura um mínimo local em F', como mostrado no pseudocódigo do Algoritmo 2. Primeiro, na linha 1, o algoritmo inicializa a variável γ com um comprimento do passo inicial γ_0 e a variável k com um chute inicial k_0. O valor de k_0 pode ser recuperado a partir de uma solução aleatória ou heurística. Ainda na linha 1, o valor da variável *last_succ*, que serve para guiar a busca para o lado esquerdo ou direito de k, é inicializado com um valor de sentinela. No laço das linhas 2-26, o algoritmo itera até $\gamma < 1$ (ressaltando que $k \in \mathbb{N}^*$). A variável *dir* é inicializada com valor zero na linha 3. O *loop* nas linhas 4-25 controla a avaliação de F' com um passo de uma unidade de comprimento ao longo de cada uma das direções γ e $-\gamma$. Os blocos *se/senão* nas linhas 5-9 reforçam a busca para um ou outro sentido (γ ou $-\gamma$), dependendo do estado da última avaliação realizada.

Se a última avaliação não for bem sucedida, a variável *dir* armazena γ ou $-\gamma$ e, portanto, a nova direção *dir* é apenas um reflexo do último valor utilizado. No entanto, se o algoritmo chega os blocos *se/senão* nas linhas 5-9 após uma tentativa bem sucedida, a busca é reforçada na mesma direção de *dir*. Na linha 10, o ponto k_t é calculado, e será a nova tentativa. Se for um valor qualificado de acordo com a Proposição 1 na linha 11, a função F' é avaliada em k_t. Se F' é melhor em k_t, em comparação ao atual valor k, este último tem seu valor atualizado para k_t na linha 13, e a variável *last_succ* é atualizada de acordo com o senso de direção usado (*dir*). Na linha 17, o tamanho do *frame* é aumentado e o ciclo é interrompido na linha 18. Caso contrário, o tamanho do *frame* é diminuído na linha 23, se os dois pontos experimentais, um em cada direção, são piores do que o ponto de k, com respeito a F'. A variável *last_succ* também é reiniciada, na linha 23. Finalmente, a solução completa é obtida na linha 27 através da utilização de k-means, h-means, ou qualquer combinação dos dois. O algoritmo é convergente, uma vez que: (i) k pertence a um intervalo finito, (ii) k é atualizado somente se houver melhoria e (iii) γ é reduzido a metade quando as buscas em ambas as direções não são bem sucedidas.

Algoritmo 2: Algoritmo de otimização livre de derivadas para microagregação

```
1)  γ ← γ0; k ← k0; last_succ ← NULO;
2)  Enquanto γ ≥ 1, faça:
3)          dir ← 0;
4)          Para cada i = 1, 2, faça:
5)                  Se  dir = −γ,  então: dir ← γ;
6)                  Senão: Se dir = γ, então: dir ← −γ;
7)                  Senão: Se last_succ = ESQUERDA, então: dir ← −γ;
8)                  Senão: dir ← γ;
9)                  Fim;
```

```
10)                    k_t ← k + dir;
11)                    Se ⌊n/2g − 1⌋ ≤ k ≤ ⌊n/g⌋, então:
12)                        Se F'(k_t) < F'(k), então:
13)                            k ← k_t;
14)                            Se dir = γ, então: last_succ ← DIREITA;
15)                            Senão: last_succ ← ESQUERDA;
16)                            Fim;
17)                            γ ← γ * 2;
18)                            break;
19)                        Fim;
20)                    Fim;
21)                    Se i = 2, então:
22)                        last_succ ← NULO;
23)                        γ ← γ/2;
24)                    Fim;
25)            Fim;
26)    Fim;
27)    (x, y) ← HEURISTICA(k);
28)    Retorna(x, y, k).
```

4. Experimentos computacionais

Os experimentos computacionais foram realizados em um Pentium Dual Core com plataforma de 64 bits, relógio de 1,2 GHz e 4 GB de memória RAM. Os algoritmos foram implementados em C++ e compilados pelo gcc 4.4.

Nosso primeiro conjunto de experimentos teve como objetivo avaliar a performance das heurísticas (k, h)-means modificadas para ambas as estratégias definidas na Seção 2. Os algoritmos foram testados em 216 instâncias sintéticas propostas por Panagiotakis e Tziritas (2012), para g = 3, 5 e 10, totalizando 216 × 3 = 648 conjuntos de dados. Para cada conjunto, os quatro algoritmos , i.e., *k-means* modificado com estratégia I, *k-means* modificado com estratégia II, *h-means* modificado com estratégia I e *h-means* modificado com estratégia II, foram executados para cada valor possível de k e com soluções iniciais aleatórias, de acordo com a Proposição 1. Através dos experimentos, identificou-se que a estratégia II encontra melhores resultados em quase 100% das vezes, tanto para o *k*-means quanto para o *h*-means.

Uma vez que trabalham com vizinhanças diferentes, a aplicação de uma heurística após a outra pode levar a melhorias na solução final obtida. A fim de saber qual a melhor ordem de aplicação para as heurísticas, foi realizada uma série de experimentos com o mesmo conjunto de instâncias acima. Como resultado, em quase 60% das vezes a aplicação do *h*-means seguido do *k*-means foi a melhor opção. Essa combinação foi denominada HM+KM. Para avaliar o algoritmo proposto aqui, foi usado o HM+KM com a segunda estratégia na avaliação da função de otimização.

Nosso segundo conjunto de experimentos computacionais avaliaram nosso algoritmo em um conjunto de instâncias reais propostas por Domingo-Ferrer e Mateo-Sanz (2002). As tabelas 1, 2 e 3 mostram os resultados obtidos a partir da aplicação do nosso algoritmo, com g = 3, 5 e 10, nas instâncias reais usadas: (i) Tarragona, com n = 834 e s = 13; (ii) Census, com n = 1080 e s = 13 e por fim (iii) EIA, com n = 4092 e s = 11. A primeira coluna mostra o valor de γ_0 que foi utilizado. O algoritmo de otimização livre de derivadas é testado a partir de duas soluções iniciais diferentes: a primeira advinda da heurística mais popularmente utilizada para o problema (MDAV) (DOMINGO-FERRER e MATEO-SANZ, 2002), e a segunda fornecida pela heurística estado-da-arte (GSMS-T2) (PANAGIOTAKIS e TZIRITAS, 2012). A ideia deste conjunto de experimentos é mensurar o quanto nosso algoritmo pode refinar soluções obtidas por outras heurísticas propostas na literatura.

Para cada conjunto de dados das tabelas, a primeira coluna (*IL*) refere-se à medida de perda de informação: $IL = SEQ / SST$, onde a *SST* é a soma das distâncias ao quadrado de cada ponto ao centróide da instância toda. Perda de informação (*IL*) é uma medida comum de precisão utilizada na

literatura de microagregação. Cada heurística é qualificada por uma medida de *IL*, e o menor *IL* indica a heurística mais precisa. A coluna *imp* apresenta as melhorias (em porcentagem) sobre as soluções fornecidas pelas heurísticas MDAV e GSMS-T2. Finalmente, a coluna *Time (s)* relata os tempos médios de computação em segundos.

		Tarragona			Census			EIA		
Heurística	**γ0**	**IL**	**imp**	**Time (s)**	**IL**	**imp**	**Time (s)**	**IL**	**Imp**	**Time (s)**
MDAV	**10**	16.11	4.84	5.77	5.69	0.00	6.71	0.48	0.00	278.13
	20	16.11	4.84	5.99	5.69	0.00	8.57	0.48	0.00	348.39
	30	15.44	8.79	5.94	5.69	0.00	8.46	0.48	0.00	355.15
GSMS-T2	**10**	16.36	0.00	8.69	5.54	0.00	12.08	0.41	0.00	690,10
	20	16.36	0.00	8.85	5.54	0.00	13.08	0.41	0.00	701,35
	30	16.36	0.00	7.55	5.54	0.00	13.33	0.41	0.00	714,67

Tabela 1 Resultados do algoritmo de otimização livre de derivadas com $g = 3$ obtidos a partir de soluções iniciais fornecidas por MDAV e GSMS-T2.

		Tarragona			Census			EIA		
Heuristica	**γ0**	**IL**	**imp**	**Time (s)**	**IL**	**imp**	**Time (s)**	**IL**	**Imp**	**Time (s)**
MDAV	**10**	21.19	5.64	0.71	9.08	0.00	1.80	1.66	0.00	206.45
	20	20.93	6.80	3.54	8.56	5.80	3.42	1.24	25.33	257.60
	30	21.07	6.15	2.35	8.78	3.33	2.76	1.51	8.83	330.68
GSMS-T2	**10**	22.70	18.02	7.30	8.56	0.00	9.67	0.86	24.49	654.02
	20	21.19	23.46	7.22	8.56	0.00	10.81	0.89	21.79	662,77
	30	21.66	21.76	5.40	8.56	0.00	10.80	**0.79**	29.82	690.35

Tabela 2 Resultados do algoritmo de otimização livre de derivadas com $g = 5$ obtidos a partir de soluções iniciais fornecidas por MDAV e GSMS-T2.

		Tarragona			Census			EIA		
Heuristica	**γ0**	**IL**	**imp**	**Time (s)**	**IL**	**imp**	**Time (s)**	**IL**	**Imp**	**Time (s)**
MDAV	**10**	33.19	0.00	1.08	14.15	0.00	1.60	2.17	43.55	104.96
	20	33.19	0.00	1.52	13.27	6.19	3.07	2.52	34.33	113.67
	30	33.19	0.00	2.83	14.15	0.00	2.55	2.11	45.00	132.63
GSMS-T2	**10**	39.03	0.00	3.86	13.36	0.00	8.03	3.02	8.02	481.34
	20	39.03	0.00	4.41	13.19	1.30	8.79	2.05	37.37	502.44
	30	39.03	0.00	4.58	**13.16**	1.53	9.24	**2.05**	37.43	515.37

Tabela 3 Resultados do algoritmo de otimização livre de derivadas com $g = 10$ obtidos a partir de soluções iniciais fornecidas por MDAV e GSMS-T2.

Os resultados nas Tabelas 1, 2 e 3 revelam que:

i. Não há nenhuma evidência de que o comprimento do passo inicial (γ_0) possua um papel importante na eficácia do algoritmo. Assim, embora γ_0 seja o único parâmetro do algoritmo, o método não parece ser muito influenciado pelo seu valor.

ii. Os tempos de computação do algoritmo normalmente diminuem com *g*. Isto não é surpreendente, uma vez que a parte mais custosa (em termos de tempo) do algoritmo é a avaliação da função *F'*, que é realizada por meio das heurísticas *k*-means e *h*-means, cuja complexidade é dada como uma função do número de *clusters*. De fato, o número de *clusters* na solução ótima de (1)-(4) tende a ser menor para maiores valores de *g*. Além disso, é importante mencionar que a diferença nos tempos de execução quando o algoritmo parte das soluções fornecidas por MDAV e GSMS-T2 se dá em virtude da diferença de complexidade das duas heurísticas, não sendo ocasionada pela utilização do algoritmo livre de derivadas. Naturalmente, quando em uma execução, a solução é melhorada com maior frequência, o algoritmo livre de derivadas gasta mais tempo computacional, pois há um maior número de contrações e expansões da janela de busca de *k*.

iii. Em 25 dos 54 casos (cerca de 46%), o algoritmo é capaz de melhorar a solução fornecida por MDAV e GSMS-T2. As melhorias decorrentes da aplicação do algoritmo variam de 1% a 45%.

iv. Três novas soluções de referência foram encontradas pelo refinamento de soluções advindas das heurísticas MDAV e do GSMS-T2. Elas são mostradas em negrito nas tabelas.

5. Conclusões

Este trabalho apresenta um algoritmo de otimização livre de derivadas para o problema de microagregação numérica. As heurísticas *k*-means e *h*-means são adaptadas neste trabalho considerando as restrições de cardinalidade por meio de uma rotina de viabilidade, que mostrou-se a melhor estratégia de acordo com os nossos experimentos. Os algoritmos são então utilizados dentro do algoritmo livre de derivadas no intuito de fornecer um *surrogate* para a função otimizada.

O algoritmo é eficaz em reduzir o custo de soluções advindas de duas heurísticas conhecidas (i.e., MDAV GSMS-T2) em até 45%. Em particular, o método proposto obteve as melhores soluções conhecidas para três instâncias da literatura. O algoritmo exige mais esforço computacional do que os métodos de aglomeração anteriormente propostos por outros autores. No entanto, seu esforço adicional não é crítico para a maioria das aplicações da microagregação, já que o tempo não é normalmente uma questão restritiva para a divulgação dos dados.

Agradecimentos Este trabalho foi desenvolvido com apoio financeiro do CNPq.

Referências

Audet, C., Dennis, J. E. e Le Digabel, S. (2010), Globalization strategies for mesh adaptive direct search, *Computational Optimization and Applications*, 46, 193-215.

Domingo-Ferrer, J., Martínez-Ballesté, A., Mateo-Sanz, J. M. e Sebé, F. (2006), Efficient multivariate data-oriented microaggregation, *The VLDB Journal*, 15, 355-369.

Domingo-Ferrer, J. e Mateo-Sanz, J. M. (2002), Practical data-oriented microaggregation for statistical disclosure control, *IEEE Transactions on Knowledge and Data Engineering*, 14, 189-201.

Hansen, P. e Mladenovic, N. (2001), J-means: a new local search heuristic for minimum sum of squares clustering, *Pattern Recognition*, 34, 405-413.

Kantarcioglu, M., Jin, J. e Clifton, C. (2004), When do data mining results violate privacy?, *Atas da X ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 599-604.

Nocedal, J. e Wright, S. J. *Numerical Optimization*, Springer, 2000.

Oganian, A. e Domingo-Ferrer, J. (2001) On the complexity of optimal microaggregation for statistical disclosure control, *Stat. J. United Nat. Econ. Com. Eur.*, 18, 345-354.

Panagiotakis, C. e Tziritas, G. (2012), Sucessive group selection for microaggregation, Aceito para publicação em *IEEE Transactions on Knowledge and Data Engineering*.

Spath, H. *Cluster analysis algorithm for data reduction and classification of objects*, Wiley, 1980.

Tavani, H. T. (1999), Informational privacy, data mining, and the internet, *Ethics and Information Technology*, 1, 137-145.
Willenborg, L. e DeWaal, T. *Elements of Statistical Disclosure Control*, Springer, New York, 2001.