Relatório Técnico
TP2 - Algoritmos II

Henrique Soares Assumpção e Silva[1]

[1]Departamento de Ciência da Computação – Universidade Federal de Minas Gerais (UFMG)

henriquesoares@ddc.ufmg.br

Abstract. *This report describes the technical evaluation of a 2-approximate algorithm for solving the k-clusters problem in many public datasets available for free. This study focuses on empirically evaluating the developed method for solving the aforementioned problem, as well as comparing such method with other state-of-the-art algorithms for clustering.*

Resumo. *Este relatório fornece uma análise detalhada da performance de um algoritmo 2-aproximado para o problema dos k-centros em diversos conjuntos de dados públicos disponíveis gratuitamente. Este estudo tem como objetivo principal avaliar de maneira empírica o método desenvolvido para solucionar o problem em questão, além de comparar tal método com outros algoritmos do estado da arte para a tarefa de clusterização.*

1. Introdução

O problema dos k-centros, também conhecido como problema de clusterização, é uma tarefa clássica em mineiração de dados e aprendizado de máquina. Podemos formalizar a tarefa da seguinte forma: dado um conjunto de pontos $S = \{s_1, s_2, ..., s_m\}, s_i \in \mathbb{R}^n$, uma métrica $\mathcal{D} : S \times S \mapsto \mathbb{R}^+$ e um inteiro k, desejamos encontrar um conjunto $C = \{c_1, c_2, ..., c_k\} \subseteq S$ de pontos, denominados centros, que particiona S em k grupos, i.e., cada ponto em S pertence ao grupo cujo centro está mais próximo. Tais grupos devem ser escolhidos de forma a minimizar o raio máximo dos clusters. Primeiramente definimos a distância de um ponto s_i qualquer a um conjunto de pontos C da seguinte maneira:

$$\mathcal{D}_r(s_i, C) = \min_j \mathcal{D}(s_i, c_j) \tag{1}$$

Então definimos o raio máximo de um conjunto de pontos C:

$$r(C) = \max_i \mathcal{D}_r(s_i, C) \tag{2}$$

Para o problema em questão, utilizamos a distância de Monkowski como métrica. Sejam $X \in \mathbb{R}^n, Y \in\in \mathbb{R}^n$ vetores contínuos. Define-se a distância entre X e Y, parametrizado por um valor natural p, como:

$$D(X, Y) = (\sum_{i=1}^{n} |x_i - y_i|^p)^{\frac{1}{p}} \tag{3}$$

O problema dos k-centros é comprovadamente NP-completo, e neste artigo iremos analisar a performance de um algoritmo guloso 2-aproximado para o problema, além de comparar tal agoritmo com o método KMeans [MacQueen 1967], que é um dos modelos mais utilizados para clusterização atualmente.

2. Modelagem

Nesta seção iremos discutir o algoritmo guloso que será análisado, bem como as métricas de avaliação escolhidas. O algoritmo em si é extremamente simples e seu funcionamento é descrito pelo algoritmo 1. A ideia é selecionar um ponto inicial arbitrário e adiciona-lo aos centros, e então progressivamente selecionar pontos que estejam mais distantes dos centros até que o conjunto possua o tamanho desejado. Tal algoritmo, apesar de simples, é comprovadamente 2-aproximado para o problema, i.e., no pior dos casos sua solução é duas vezes pior que o ótimo w.r.t. à métrica definida na equação 2. A implementação do

Algorithm 1 Solução Gulosa
Entrada: Conjunto S de pontos, inteiro k

if $k > |S|$ **then return** S
else
  Selecione $s_i \in S$ arbitrário
  $C \leftarrow \{s_i\}$
  while $|C| < k$ **do**
    Escolha $s_i \in S \setminus C$ de maior $\mathcal{D}_r(s_i, C)$
    $C \leftarrow C \cup \{s_i\}$
  end while
end ifreturn C

algoritmo realizada para o projeto possui complexidade de tempo de aproximadamente O(nk^2), pois para cada uma das k iterações do algoritmo devemos computar a distância de n pontos para os $k-1$ pontos dos clusters.

Também foram definidas duas métricas de avaliação para a qualidade dos clusters computados pelo algoritmo. Tais métricas são descritas a seguir:

- *Rand Score* (RS): Esta métrica computa a similaridade entre duas clusterizações distintas de maneira simples. Ela considera todos os pares de pontos e computa a razão entre os pares que pertencem ao mesmo cluster pelo total de pares. Tal métrica pode ser vista como uma espécia de acurácia, que mede o quão similar dois clusters distintos são. A métrica varia de 0 até 1, sendo 0 o pior resultado possível e 1 o melhor;
- *Silhouette Score* (SS): Está métrica mede a qualidade da clusterização feita por meio da comparação entre a distância média interna entre os pontos de cada cluster e seu respectivo centro e a distância média entre os pontos de cada cluster e o cluster vizinho mais próximo. Tal métrica essencialmente avalia se a classificação dos pontos foi adequada. A métrica varia de -1 até 1, sendo -1 o pior resultado possível e 1 o melhor.

3. Experimentos

Nesta seção iremos discutir os experimentos realizados para avaliar empiricamente o algoritmo em questão. O algoritmo de clusterização foi testado em dez conjuntos de dados públicos disponíveis gratuitamente. Todos os links para os conjuntos de dados estão disponíveis nos arquivos de implementação do projeto. Segue uma descrição de cada conjunto utilizado:

- *Mammograpic Mass* (MM): Conjunto de dados relativos à presença de tumores malignos ou benignos em pacientes que realizaram exames de mama. A tarefa em questão é, dado as informações relativas ao exame da cada paciente, prever o tipo do tumor. Esse dataset possui 830 instâncias, cada uma com 6 atributos, incluindo o atributo alvo;
- *South German Credit* (SGC): Conjunto de dados relativos à indivíduos no sul da alemanha que aplicaram para receber créditos financeiros. Existem diversas variáveis de interesse no conjunto, e para esse estudo desejamos prever o histórico de crédito do indivíduo com base nas demais informações cadastrais disponíveis. Esse dataset possui 1000 instâncias, cada uma com 21 atributos, incluindo o atributo alvo;
- *Blood Transfusion Service Center* (BTSC): Conjunto de dados relativos à indivíduos que doaram (ou não) sangue em março de 2007. A tarefa em questão é tentar prever se o indivíduo doou sangue com base em informações cadastrais do indivíduo. Esse dataset possui 748 instâncias, cada uma com 5 atributos, incluindo o atributo alvo;
- *Audit Data* (AD): Conjunto de dados relativos à firmas indianas que podem ser consideradas como suspeitas pelo governo. A tarefa é prever o nível de risco que uma firma apresenta com base nos demais dados disponíveis sobre a firma. Esse dataset possui 775 instâncias, cada uma com 26 atributos, incluindo o atributo alvo;
- *Drug Consumption* (DC): Conjunto de dados relativos ao uso de drogas por um conjunto de indivíduos. A tarefa é classificar o indivíduo com respeito a quais classes de drogas ele já utilizou, i.e., dado as 18 classes possíveis, atribuir um valor de 0 a 17 para cada indivíduo. Esse dataset possui 1885 instâncias, cada uma com 13 atributos, incluindo o atributo alvo;
- *Myocardial infarction complications* (MIC): Conjunto de dados relativos à informações de exames médicos sobre indivíduos que sofreram infarto do miocardio. A tarefa em questão é tentar prever qual foi a consequência do infarto com base nos dados disponíveis. Esse dataset possui 1700 instâncias, cada uma com 111 atributos, incluindo o atributo alvo;
- *Cervical cancer (Risk Factors)* (CC): Conjunto de dados relativos à informações de exames médicos e outras informações pessoais sobre diversos indivíduos, com o objetivo de prever indicadores de cancêr cervical. Esse dataset possui 858 instâncias, cada uma com 33 atributos, incluindo o atributo alvo;
- *Room Occupancy Estimation* (ROE): Conjunto de dados relativos à informações sobre quartos em diversos apartamentos. A tarefa em questão é classificar os quartos com respeito à quantidade de moradores com base em informações não intrusivas, e.g., temperatura média. Esse dataset possui 10129 instâncias, cada uma com 17 atributos, incluindo o atributo alvo. Devido a quantidade significativa de instâncias, realizamos uma amostra aleatória de 3000 instâncias devido à restrições computacionais, i.e., como é necessário computar uma matriz das distâncias entre todos os pontos, tal matriz requer um espaço significativo se utilizarmos todas as instâncias disponíveis;
- *Wine Quality (Red)* (WQ): Conjunto de dados relativos à informações químicas de diversos tipos de vinhos tintos. A tarefa em questão é classificar o vinho com uma nota de 0 a 10, com base em suas características. Esse dataset possui 1599

instâncias, cada uma com 12 atributos, incluindo o atributo alvo;

- *Spambase* (SB): Conjunto de dados com informações diversas sobre emails, como frequência de palavras e tamanho médio de frases. A tarefa em questão é determinar se um dado email é um spam ou não. Esse dataset possui 4601 instâncias, cada uma com 58 atributos, incluindo o atributo alvo, e por motivos similares aos descritos previamente, realizamos uma amostragem aleatória de 3000 instâncias.

Para testar o algoritmo proposto, testamos dois valores distintos para o parâmetro p da distância de Minkowski, nominalmente $p = 1$ (Distância Manhattan) e $p = 2$ (Distância Euclidiana). Para cada valor de p, realizamos 30 execuções com sementes aleatórias distintas do algoritmo para cada conjunto de dados citado. Computamos os valores médios das métricas discutidas na seção 2, bem como seus desvios padrão. Além disso, computamos o tempo médio (e desvio padrão) da execução de ambos os algoritmos testados, além do valor médio (e desvio padrão) do raio máximo para os clusters computados.

4. Resultados

Nesta seção iremos discutir os resultados dos experimentos descritos na seção anterior. Para cada tabela a seguir, os valores apresentados apresentam precisão de quatro casas decimais e eles representam as estatísticas para os resultados de interesse ao longo dos 30 experimentos realizados por conjunto de dados e por valor p para distância.

Dataset	p	Model: RS Média	Model: RS Dp	KMeans: RS Média	KMeans: RS Dp
MM	1	0.5016	0.0093	0.5683	0.0000
MM	2	0.5089	0.0200	0.5683	0.0000
SGC	1	0.5154	0.0188	0.5564	0.0024
SGC	2	0.5154	0.0188	0.5564	0.0024
BT	1	0.6433	0.0007	0.5992	0.0000
BT	2	0.6433	0.0007	0.5992	0.0000
AD	1	0.5226	0.0000	0.5249	0.0000
AD	2	0.5226	0.0000	0.5249	0.0000
DC	1	0.8176	0.0486	0.8852	0.0004
DC	2	0.8059	0.0570	0.8852	0.0004
MI	1	0.3939	0.0172	0.3824	0.0001
MI	2	0.3942	0.0152	0.3824	0.0001
CC	1	0.8726	0.0040	0.5272	0.0000
CC	2	0.8710	0.0034	0.5272	0.0000
ROE	1	0.8232	0.0396	0.7817	0.0024
ROE	2	0.8052	0.0359	0.7817	0.0024
WQ	1	0.5491	0.0214	0.5987	0.0014
WQ	2	0.5549	0.0201	0.5987	0.0014
SB	1	0.5240	0.0000	0.5398	0.0000
SB	2	0.5242	0.0002	0.5398	0.0000

Tabela 1. Tabela com os resultados dos experimentos para o *Rand Score* para o algoritmo aproximado (Model) e para o KMeans.

A tabela 1 nos mostra os resultados w.r.t. *Rand Score*. É possível observar que, de forma geral, o modelo proposto se manteve competitivo quando comparado com o KMeans, sendo que em alguns conjuntos de dados como o CC ele superou os resultados do KMeans por uma margem considerável. Por outro lado, é possível observar que os desvios padrão para os resultados do modelo é significativamente maior que os desvios padrão do KMeans, ou seja, os resultados do modelo são mais sensíveis às condições iniciais, o que era de se esperar tendo em vista ao passo inicial aleatório que o modelo faz.

A tabela 2 nos mostra os resultados w.r.t. *Silhouette Score*. Também é possível observar que, de forma geral, o modelo proposte se manteve competitivo quando comparo ao KMeans. Em alguns datasets, como CC e BT, o resultado do modelo superou o resultado do KMeans por uma margem considerável. Além disso, é possível observar o impacto que diferentes valores de p fornecerem para a métrica em questão. Assim como observado previamente, o desvio padrão do algoritmo proposto é significativamente maior em geral que o desvio padrão do KMeans.

Dataset	p	Model: SS Média	Model: SS Dp	KMeans: SS Média	KMeans: SS Dp
MM	1	0.6665	0.1133	0.4741	0.0000
MM	2	0.6158	0.1219	0.4882	0.0141
SGC	1	0.5960	0.0462	0.5608	0.0016
SGC	2	0.5961	0.0461	0.5610	0.0016
BT	1	0.8654	0.0034	0.6859	0.0000
BT	2	0.8661	0.0034	0.6873	0.0013
AD	1	0.9720	0.0000	0.9557	0.0000
AD	2	0.9715	0.0006	0.9581	0.0024
DC	1	-0.0193	0.0145	0.0426	0.0035
DC	2	-0.0133	0.0143	0.0472	0.0056
MI	1	0.4078	0.0900	0.5821	0.0009
MI	2	0.3844	0.1092	0.6103	0.0283
CC	1	0.7771	0.0132	0.3843	0.0000
CC	2	0.7863	0.0165	0.4284	0.0441
ROE	1	0.6642	0.0819	0.6732	0.0061
ROE	2	0.6828	0.0614	0.7028	0.0303
WQ	1	0.3251	0.0421	0.3610	0.0056
WQ	2	0.3189	0.0367	0.3826	0.0220
SB	1	0.9704	0.0026	0.8676	0.0000
SB	2	0.9690	0.0023	0.8667	0.0008

Tabela 2. Tabela com os resultados dos experimentos para o *Silhouette Score* para o algoritmo aproximado (Model) e para o KMeans.

A tabela 3 nos mostra os resultados w.r.t. raio máximo dos clusters. Os raios do algoritmo, de forma geral, são menores que os raios encontrados pelo KMeans, mostrando que os clusters encontrados pelo algoritmo possuem raio máximo menor, o que, de acordo com a métrica definida na equação 2, significa que os clusters do algoritmo são em geral melhores que os clusters do KMeans.

A tabela 4 nos mostra os resultados w.r.t. tempo de execução dos algoritmos. De modo geral, o algoritmo proposto é significativamente mais lento quando comparado ao KMeans. Existem diversas otimizações de tempo que podem ser feitas para acelerar o algoritmo proposto, e.g. utilizar árvores de busca para estruturar os pontos, mas a implementação em questão possui uma performance de tempo inferior à performance do KMeans.

Dataset	p	Model: $r(C)$ Média	Model: $r(C)$ Dp	KMeans: $r(C)$ Média	KMeans: $r(C)$ Dp
MM	1	56.6667	9.4493	57.8246	0.0000
MM	2	52.9246	8.5683	54.3994	3.4252
SGC	1	2459.4667	402.4334	5281.6146	119.3210
SGC	2	2438.9870	402.0158	5266.1825	120.3989
BT	1	5594.0333	610.6268	8627.5299	0.0000
BT	2	5567.9162	608.7854	8590.5455	36.9844
AD	1	2548.8741	119.7671	4437.5244	0.0000
AD	2	1886.9432	667.6358	3117.4336	1320.0908
DC	1	11.5797	0.1924	11.4479	0.3699
DC	2	8.0375	3.5454	7.8168	3.6413
MI	1	321.7610	12.4256	268.3506	2.6834
MI	2	239.5328	82.9547	200.6152	67.7626
CC	1	98.0378	4.1337	98.3013	0.0000
CC	2	74.5448	23.6741	76.0951	22.2062
ROE	1	545.5541	35.6741	562.7580	15.8998
ROE	2	436.8233	112.0918	437.8522	125.4442
WQ	1	42.4331	2.5608	73.9825	0.6826
WQ	2	33.6869	9.0199	60.0069	13.9885
SB	1	15704.7276	258.8580	17519.6652	0.0000
SB	2	13614.6166	2098.1105	15534.5112	1985.1540

Tabela 3. Tabela com os resultados dos experimentos para o raio máximo do cluster para o algoritmo aproximado (Model) e para o KMeans.

5. Conclusão

De forma geral, observa-se que o algoritmo 2-aproximado proposto é tão bom quanto, e em diversas situações significativamente melhor, que o KMeans, quando analizado no contexto dos conjuntos de dados em questão. Tal vantagem no entanto vem ao custo de um tempo de execução significativamente pior que o tempo de execução do KMeans, tendo em vista a implementação simples proposta neste artigo.

Os experimentos propostos no projeto nos permitiram avaliar empiricamente a performance do algoritmo aproximativo proposto para o problema dos k-centros, e, dentro do escopo de dados dos experimentos, é possível concluir que o algoritmo é competitivo quando comparado à métodos como o KMeans, que estão entre os métodos mais utilizados para a tarefa de clusterização na atualidade.

Dataset	p	Model: RT Média	Model: RT Std	KMeans: RT Média	KMeans: RT Std
MM	1	0.0338	0.0011	0.0196	0.0016
MM	2	0.0346	0.0016	0.0206	0.0042
SGC	1	0.1725	0.0062	0.0460	0.0060
SGC	2	0.1704	0.0057	0.0450	0.0054
BT	1	0.0313	0.0029	0.0163	0.0020
BT	2	0.0310	0.0023	0.0165	0.0035
AD	1	0.0349	0.0035	0.0165	0.0080
AD	2	0.0350	0.0028	0.0154	0.0059
DC	1	3.1459	0.0587	0.2132	0.0151
DC	2	3.1971	0.0701	0.2139	0.0166
MI	1	0.6950	0.0127	0.1071	0.0140
MI	2	0.7258	0.0341	0.1083	0.0170
CC	1	0.0365	0.0015	0.0285	0.0120
CC	2	0.0370	0.0015	0.0277	0.0087
ROE	1	0.3432	0.0063	0.0564	0.0085
ROE	2	0.3510	0.0101	0.0555	0.0073
WQ	1	1.0387	0.0171	0.1124	0.0101
WQ	2	1.0734	0.0405	0.1122	0.0095
SB	1	0.1241	0.0088	0.0344	0.0101
SB	2	0.1286	0.0090	0.0340	0.0085

Tabela 4. Tabela com os resultados dos experimentos para o tempo de execução em segundos (RT) para o algoritmo aproximado (Model) e para o KMeans.

Referências

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations.