Trabalho Prático 2
Solução para problemas difíceis

Vitor Emanuel F. Vital[1]

[1]Departamento de Ciência da Computação – Universidade Federal de Minas Gerais (UFMG)
31.270-901 – Belo Horizonte – MG – Brasil

Abstract. *This meta-paper addresses practical aspects related to the implementation of approximate algorithms. Specifically, we discuss the k-center problem, which is useful in machine learning clustering tasks. We also examine issues regarding empirical comparison of algorithms/programs. In this regard, we compare the implementation of the 2-approximation algorithm for the k-center problem with the classical algorithm for clustering (K-Means). The comparisons aim to evaluate both computational demands and solution quality aspects.*

Resumo. *Este meta-artigo aborda os aspectos práticos da implementação de algoritmos aproximativos, com foco no problema dos k-centros, que desempenha um papel importante na tarefa de agrupamento em aprendizado de máquina. Além disso, são discutidas questões relacionadas à comparação empírica de algoritmos/programas. Especificamente, realizamos uma comparação entre a implementação do algoritmo 2-aproximado para o problema do k-centro e o algoritmo clássico de agrupamento (K-Means). Essas comparações abrangem aspectos de demanda computacional e qualidade da solução, visando fornecer uma avaliação abrangente desses algoritmos.*

1. Introdução

Agrupamento de dados desempenha um papel crucial em uma variedade de aplicações de aprendizado de máquina. No contexto específico, o problema dos k-centros emerge como um desafio comum, que envolve a identificação de k centros representativos em um conjunto de dados. Neste estudo, dedicamo-nos a explorar os aspectos práticos relacionados à implementação de algoritmos aproximativos destinados a solucionar o problema dos k-centros.

Além disso, uma abordagem amplamente adotada é a comparação empírica de algoritmos/programas, já que desempenha um papel fundamental na seleção de abordagens eficientes. Com isso em mente, comparamos a implementação do algoritmo 2-aproximado para o problema dos k-centros com o algoritmo clássico de agrupamento conhecido como K-Means.

Nosso objetivo é fornecer uma avaliação abrangente desses algoritmos, abordando tanto a demanda computacional quanto a qualidade das soluções obtidas. Essa análise visa contribuir para o conhecimento sobre os métodos de agrupamento, permitindo a seleção de abordagens mais eficientes e eficazes para lidar com problemas de agrupamento em diversos domínios de aplicação.

2. O problema dos k-centros (clustering)

Existem várias formulações para o problema de agrupamento (clustering). Neste trabalho, consideramos a implementação que envolve um conjunto de n pontos e um parâmetro k. Nosso objetivo é encontrar k centros de tal forma que os pontos estejam o mais próximo possível desses, com uma distância máxima de r de qualquer um dos centros.

Formalmente, adotamos a seguinte formulação:

- **ENTRADA:** $S = s_1, s_2, ..., s_n$ um conjunto de pontos; $dist : S \times S \rightarrow R^+$ uma função de distância; e um inteiro k.
- **SAÍDA:** um conjunto $C = c_1, c_2, ..., c_k$ de pontos (centros) que particiona o conjunto de pontos em k grupos (s_i pertence ao centro mais próximo).
- **OBJETIVO:** minimizar o raio máximo dos clusters, $r(C) = maxdist(s_i, C)$.

Essa formulação serve como base para o desenvolvimento do algoritmo de clustering abordado neste artigo. Ao aplicar esse algoritmo, buscamos encontrar uma solução que minimize o raio máximo dos clusters, proporcionando uma forma eficiente de agrupamento dos pontos em conjuntos coerentes.

3. Modelagem

A implementação do algoritmo k-centers tem como objetivo encontrar k centros representativos em um conjunto de dados, de forma que os pontos estejam o mais próximo possível desses centros.

Uma abordagem comumente utilizada para a implementação do algoritmo k-centers é baseada em heurísticas aproximativas. Essas heurísticas buscam encontrar uma solução próxima da ideal, mas sem garantir a otimalidade global. Isso é especialmente relevante, considerando que o problema de agrupamento é NP-difícil e, portanto, encontrar a solução ótima pode ser computacionalmente inviável para conjuntos de dados grandes.

O algoritmo k-centers é projetado para minimizar o raio máximo dos clusters, que representa a maior distância entre um ponto e seu centro correspondente. Durante a implementação desse algoritmo, várias técnicas podem ser aplicadas para encontrar uma solução aproximada eficiente. Uma abordagem comum envolve a seleção inicial dos centros de forma aleatória ou com base em critérios heurísticos. No caso deste algoritmo, a seleção inicial é feita aleatoriamente. No entanto, para mitigar a possível má qualidade do resultado devido à escolha inicial, são realizados 30 testes ou execuções do algoritmo.

Em cada iteração do algoritmo, os pontos restantes são alocados aos centros de acordo com a distância euclidiana ou outra métrica adequada, com base nisso, foi implementada a função de *Distância de Minkowski*.

No contexto do algoritmo k-center, é importante entender conceitos-chave: o *Raio da Solução* e as medidas clássicas usadas para avaliação de agrupamentos, como a *Silhueta* e o *Ìndice de Rand Ajustado*.

Essas medidas são importantes para avaliar a qualidade e a consistência dos clusters gerados pelo algoritmo k-center. Ao utilizar essas métricas, é possível ter uma avaliação mais precisa do desempenho do algoritmo e tomar decisões embasadas na seleção dos parâmetros ou na comparação com outros métodos de agrupamento.

A título de comparação da qualidade da solução, os experimentos serão implementados também com a utilização do *K-Means*. Tal prática é comum no campo de agrupamento de dados. Ambos os algoritmos têm o objetivo de particionar um conjunto de dados em k clusters, porém, eles diferem em suas abordagens e resultados esperados.

3.1. Distância de Minkowski

A distância de Minkowski é uma medida de distância utilizada em várias áreas, como aprendizado de máquina e processamento de sinais. Essa métrica geraliza as medidas de distância mais conhecidas, como a distância Euclidiana e a distância de Manhattan, distâncias utilizadas no algoritmo k-center em questão.

A distância Euclidiana é usada em espaços euclidianos, calculando a raiz quadrada da soma dos quadrados das diferenças entre as coordenadas dos pontos.

Por outro lado, a distância de Manhattan, também conhecida como distância retangular ou métrica L1, é uma medida de distância que considera apenas as diferenças absolutas entre as coordenadas dos pontos.

A distância de Minkowski permite ajustar o parâmetro 'p' para adaptar-se a diferentes necessidades, sendo reduzida à distância Euclidiana quando $p = 2$ e equivalente à distância de Manhattan quando $p = 1$.

3.2. Raio da Solução

O raio da solução refere-se à distância máxima entre um ponto e seu centro correspondente em um determinado conjunto de clusters. Em outras palavras, é a medida que representa o tamanho máximo de um cluster. O objetivo do algoritmo k-center é minimizar esse raio, buscando encontrar uma distribuição de centros que minimize a dispersão dos pontos em relação aos seus respectivos centros.

Além disso, para avaliar a qualidade dos clusters obtidos pelo algoritmo k-center, são comumente utilizadas medidas clássicas de avaliação. Duas dessas medidas são a silhueta e o índice de Rand ajustado.

A implementação utilizada baseia-se no ponto, não central, com maior distância ao centro mais próximo. A distância utilizada baseia-se na função da *Distância de Minkowski* com ambas métricas abordadas (euclidiana e de Manhattan).

3.3. Silhueta

A medida de silhueta é uma técnica que avalia o quão bem um ponto se encaixa em seu próprio cluster em comparação com outros clusters. Ela considera a distância média entre o ponto e todos os outros pontos dentro do mesmo cluster, bem como a distância média entre o ponto e os pontos dos clusters vizinhos mais próximos. Quanto mais próximo de 1 for o valor da silhueta, melhor é a atribuição do ponto ao seu cluster.

A implementação utilizada provém da biblioteca Scikit Learn.

3.4. Ìndice de Rand Ajustado

O índice de Rand ajustado é outra medida amplamente utilizada para avaliar a concordância entre dois agrupamentos. Ele compara a similaridade dos pares de pontos

dentro dos agrupamentos com a similaridade dos pares de pontos entre os agrupamentos. O índice de Rand ajustado varia de -1 a 1, em que um valor próximo de 1 indica alta concordância entre os agrupamentos.

A implementação utilizada provém da biblioteca Scikit Learn.

3.5. K-Means

O algoritmo k-means é amplamente utilizado e conhecido. Ele busca minimizar a soma dos quadrados das distâncias entre cada ponto e o centro do seu cluster atribuído. A principal métrica utilizada é a distância Euclidiana. O k-means requer uma inicialização adequada dos centros iniciais e geralmente é executado várias vezes para evitar mínimos locais.

Em relação à eficiência computacional, o k-means é geralmente mais rápido, pois envolve apenas a atualização iterativa dos centros e a realocação de pontos. O k-center, por outro lado, pode ser mais exigente computacionalmente, pois requer o cálculo do raio máximo em cada iteração.

4. Implementação

O programa foi implementado utilizando a linguagem Python 3 e executado no sistema operacional Linux Ubuntu 20.04.6 LTS. Para auxiliar na implementação, foram utilizadas as seguintes bibliotecas:

Scikit-Learn: biblioteca foi empregada para calcular as métricas de avaliação dos agrupamentos, como a Silhueta, o Índice de Rand Ajustado e os resultados do algoritmo K-Means.

Pandas: utilizada para obter os dados dos datasets. Essa biblioteca oferece uma ampla gama de funcionalidades para manipulação e análise de dados tabulares.

NumPy: biblioteca foi empregada para manipular estruturas de dados auxiliares durante os cálculos do algoritmo K-Center. O NumPy fornece suporte eficiente para operações matemáticas e manipulação de arrays multidimensionais.

5. Análise dos Resultados

A análise comparativa entre a implementação pessoal do algoritmo K-Centers e a implementação da biblioteca do K-Means é fundamental para avaliar a performance e qualidade dos resultados obtidos por essas abordagens de clusterização. O objetivo dessa análise é identificar suas vantagens e desvantagens em relação a métricas específicas de avaliação.

Neste estudo, foram coletados dados relacionados aos resultados obtidos a partir da aplicação dos algoritmos K-Centers e K-Means em conjuntos de dados coletados de dez datasets disponibilizados na UCI Machine Learning Repository. Foram consideradas métricas como o raio máximo (manhattan e euclidiano), a métrica silhouette, a métrica Rand e o tempo de execução. Para o algoritmo K-Centers, como a escolha dos centros é arbitrária e pode influenciar na qualidade da solução fazemos trinta iterações a fim de minimizar tais consequências.

A análise dos resultados visa responder a perguntas importantes, como: qual algoritmo apresenta uma melhor capacidade de separação dos clusters? Qual algoritmo é

mais eficiente computacionalmente? Quais são as características e diferenças dos clusters gerados por cada algoritmo?

Ao comparar a implementação pessoal do K-Centers com a implementação da biblioteca do K-Means, é possível identificar as peculiaridades de cada método e avaliar sua adequação para diferentes conjuntos de dados e problemas. Essa análise permitirá compreender o desempenho e a qualidade dos resultados obtidos por essas abordagens de clusterização e fornecer insights valiosos para a seleção do algoritmo mais adequado em diferentes cenários.

A seguir, serão apresentados os resultados obtidos em termos de métricas específicas, como o raio máximo, a métrica silhouette, a métrica Rand e o tempo de execução. Essa análise detalhada fornecerá uma visão mais completa e embasada sobre as diferenças entre o K-Centers e o K-Means, permitindo uma compreensão mais profunda de seus pontos fortes e limitações.

5.1. Resultados Obtidos

A seguir, apresentamos os resultados das iterações realizadas nos 10 conjuntos de dados obtidos, todos eles provenientes de datasets disponibilizados no UCI Machine Learning Repository. Esses resultados fornecem insights valiosos sobre o desempenho e a eficácia dos algoritmos em diferentes contextos de dados.

O conjunto de dados utilizados estão ordenados conforme os experimentos realizados:

Balance Scale: *Balance scale weight distance database*

Banknote Authentication: *Data were extracted from images that were taken for the evaluation of an authentication procedure for banknotes.*

Blood Transfusion Service Center: *Data taken from the Blood Transfusion Service Center in Hsin-Chu City in Taiwan – this is a classification problem.*

Breast Cancer Wisconsin (Original): *Original Wisconsin Breast Cancer Database*

Facebook Live Sellers in Thailand: *Facebook pages of 10 Thai fashion and cosmetics retail sellers. Posts of a different nature (video, photos, statuses, and links). Engagement metrics consist of comments, shares, and reactions.*

Iranian Churn Dataset: *This dataset is randomly collected from an Iranian telecom companyâ€™s database over a period of 12 months.*

Seoul Bike Sharing Demand: *The dataset contains count of public bicycles rented per hour in the Seoul Bike Sharing System, with corresponding weather data and holiday information*

Shill Bidding Dataset: *We scraped a large number of eBay auctions of a popular product. After preprocessing the auction data, we build the SB dataset. The goal is to share the labelled SB dataset with the researchers.*

Spambase: *Classifying Email as Spam or Non-Spam*

Wine Quality: *Two datasets are included, related to red and white vinho verde*

wine samples, from the north of Portugal. The goal is to model wine quality based on physicochemical tests (see [Cortez et al., 2009]

Observação: O último dataset 'Wine Quality' possui dois conjuntos de dados, um para vinho branco e o segundo para vinho vermelho, ambos são unidos para posterior análise.

Nº de instâncias	K-Centers									
	Manhattan		Eucliadiano		Silhueta	Desvio Padrão	Rand	Desvio Padrão	Tempo de Execução(s)	Desvio Padrão
	Raio	Desvio Padrão	Raio	Desvio Padrão						
625	16.0	0.0	8.0	0.0	0.13	0.01	0.56	0.05	0.07	0.0
1372	57.48	1.96	34.50	0.0	0.45	0.04	0.54	0.01	0.12	0.01
748	12395.0	0.0	12250.47	0.0	0.85	0.01	0.64	0.0	0.05	0.01
699	70.03	1.99	25.11	0.0	0.54	0.02	0.72	0.07	0.06	0.01
7050	29567.0	0.0	21763.62	0.0	0.80	0.02	0.48	0.0	1.50	0.03
3150	18391.13	33.99	17113.19	0.0	0.68	0.02	0.58	0.04	0.30	0.01
8760	5235.33	270.72	3910.56	0.0	0.38	0.05	0.52	0.06	2.11	0.62
6321	15.74	0.37	9.33	0.0	0.50	0.08	0.57	0.11	1.00	0.05
4601	19984.16	505.8	15840.01	0.0	0.96	2.22	0.52	0.04	0.64	0.03
6500	730.48	0.0	519.80	0.0	0.76	0.05	0.62	0.0	1.26	0.21

Figura 1. Resultados das trinta iterações do algoritmo K-Centers para o conjunto de dados selecionados

Nº de instâncias	K-Means				
	Manhattan	Euclidiano	Silhueta	Rand	Tempo de Execução(s)
625	10.41	5.30	0.17	0.59	0.06
1372	43.76	27.11	0.43	0.52	0.02
748	11720.88	11597.76	0.70	0.59	0.01
699	65.42	23.27	0.57	0.91	0.01
7050	23081.62	20898.42	0.81	0.52	0.78
3150	15058.03	14256.02	0.67	0.55	0.11
8760	4624.01	3457.30	0.48	0.62	1.11
6321	12.63	8.13	0.58	0.50	0.64
4601	20100.29	15662.14	0.84	0.53	0.34
6500	652.20	463.54	0.51	0.66	1.11

Figura 2. Resultado do algoritmo K-Means para o mesmo conjunto de dados

Com base nas tabelas apresentadas, é possível realizar uma análise comparativa entre o desempenho do algoritmo K-Centers e o algoritmo K-Means. Os resultados obtidos indicam que o algoritmo K-Centers demonstrou ser menos eficiente do que o algoritmo K-Means na maioria dos casos avaliados. Embora os valores sejam próximos, observa-se que o algoritmo K-Means obteve uma melhor qualidade de aproximação entre os elementos e seus respectivos centros.

Além disso, destaca-se que o tempo de execução do algoritmo K-Means também foi inferior em todos os experimentos realizados. Esses resultados sugerem que o algoritmo K-Means apresenta uma maior eficiência computacional e melhor capacidade de agrupamento em relação ao algoritmo K-Centers.

Uma análise mais aprofundada entre a implementação do K-Centers e o algoritmo K-Means, ao analisar os valores do raio máximo, observamos que tanto na métrica de distância de Manhattan quanto na métrica de distância euclidiana, o algoritmo K-Centers apresentou valores mais altos em relação ao algoritmo K-Means. Isso indica que o K-Centers teve uma maior dispersão dos pontos em relação ao centro de cada cluster, enquanto o K-Means conseguiu agrupar os pontos de forma mais compacta.

No que diz respeito à métrica de silhueta, verificamos que o algoritmo K-Centers obteve um valor médio inferior ao do K-Means, isso indica que o K-Means conseguiu estabelecer clusters com uma maior coesão interna e uma maior separação entre os clusters, resultando em uma silhueta média superior.

Ao analisar o índice de Rand ajustado, observamos que o algoritmo K-Centers obteve um valor médio também inferior ao do K-Means, isso indica que o K-Means teve uma maior concordância com as classificações reais dos dados, comparado ao K-Centers.

Por fim, em relação ao tempo de execução, o algoritmo K-Centers apresentou um tempo de execução menor que o K-Means. Isso demonstra que o K-Means foi significativamente mais rápido em sua execução, fornecendo uma vantagem em termos de eficiência computacional.

6. Conclusão

Em resumo, com base nos resultados obtidos, podemos concluir que o algoritmo K-Means apresentou um desempenho geralmente superior ao algoritmo K-Centers. O K-Means mostrou-se mais eficiente na formação de clusters compactos, com uma melhor qualidade de aproximação entre os pontos e seus centros. Além disso, o K-Means obteve uma maior concordância com as classificações reais dos dados e um tempo de execução mais rápido.

No entanto, é importante destacar que essas conclusões são específicas para o conjunto de dados e parâmetros utilizados neste experimento. Outros conjuntos de dados podem apresentar resultados diferentes, e é recomendado realizar análises adicionais e considerar diferentes métricas e cenários para uma avaliação mais completa e precisa dos algoritmos.

Os algoritmos de clusterização, como o K-Centers e o K-Means, desempenham um papel fundamental na análise de dados e no campo da aprendizagem de máquina. Embora o K-Means tenha mostrado um desempenho geralmente superior ao K-Centers com base nos resultados deste experimento, é importante ressaltar que ambos os algoritmos têm suas próprias vantagens e são úteis em diferentes contextos.

Os algoritmos de clusterização, como o K-Centers e o K-Means, desempenham um papel fundamental na análise de dados e no campo da aprendizagem de máquina.

Além disso, a realização desse trabalho permitiu explorar a importância da avaliação e comparação de algoritmos, fornecendo uma base sólida para a tomada de decisões informadas em projetos futuros. O estudo realizado no contexto da implementação pessoal do K-Centers versus a biblioteca do K-Means é relevante tanto no aspecto acadêmico quanto na compreensão das capacidades e características desses algoritmos.

7. References

Slides da disciplina, [Kleinberg and Tardos 2009]

Referências

Kleinberg, J. and Tardos, E. (August 6, 2009). *Algorithm Design*.