UNIVERSIDADE FEDERAL DE SÃO PAULO
INSTITUTO DE CIÊNCIA E TECNOLOGIA
TÓPICOS ESPECIAIS II
Prof. Dr. Vinícius Veloso de Melo
2º semestre de 2015

Agrupamento de Sequências Biológicas

Marcos Castro de Souza

Instituto de Ciência e Tecnologia, Universidade Federal de São Paulo (ICT/UNIFESP)
E-mail para contato: mcastrosouza@live.com

RESUMO

Agrupamento é uma técnica de Data Mining que tem como objetivo agrupar dados de forma automática conforme um grau de semelhança. Nesse trabalho, os dados são sequências biológicas (nucleotídeos por exemplo) que foram separados por grupos de acordo com algum método de comparação de sequências. Foi construída uma ferramenta utilizando a linguagem de programação C++ para o agrupamento de sequências biológicas. Essa ferramenta irá funcionar como um módulo de um trabalho em andamento que visa identificar variações genéticas entre indivíduos. A ferramenta é baseada na técnica de agrupamento K-Means, é open-source e possui diversas opções que, ao combiná-las, poderão gerar bons resultados.

Palavras Chave: *K-Means; Algoritmo de Agrupamento; Agrupamento híbrido; Heurísticas; Bioinformática; Sequências Biológicas; Grafos; Programação Dinâmica;*

1. INTRODUÇÃO

Agrupamento (clustering) é o nome dado para um grupo de técnicas computacionais cujo propósito consiste em separar objetos em grupos (clusters), baseando-se nas características que estes objetos possuem (LINDEN, 2009). Clustering é uma técnica de Data Mining (mineração de dados) que tem como objetivo agrupar dados em clusters de forma que um agrupamento em um determinado cluster seja uma coleção de objetos que são semelhantes entre si e diferentes de objetos que estão em outros clusters. Um bom agrupamento assegura que a similaridade inter-cluster seja baixa e a similaridade intra-cluster seja alta.

As técnicas de agrupamento têm sido utilizadas nas mais variadas áreas como por exemplo no setor bancário (agrupar clientes para selecionar aqueles que tem o menor risco de dar calote num empréstimo), marketing (descobrir grupos para que possam ser desenvolvidos programas de marketing direcionados), bioinformática (encontrar padrões em uma grande quantidade de dados).

1.1. K-Means

K-Means é um processo para particionar uma população N-dimensional em K conjuntos de uma amostra (MacQueen, 1967). Trata-se de uma heurística de agrupamento não-hierárquico que busca minimizar a distância dos elementos a um conjunto de K centros.

O K-Means é um algoritmo baseado em distância que particiona os dados em um número pré-determinado de clusters (JAIN, 2012). Cada ponto (data point) é atribuído ao cluster mais próximo de acordo com uma função de distância (euclidiana por exemplo). A função de distância é utilizada para medir a similaridade entre os pontos. O recálculo dos centros é feito utilizando a média aritmética. A inicialização dos clusters pode ser feita de forma aleatória.

Figura 1. Algoritmo K-Means.

1. begin
2. initialize N, K, $C_1, C_2, \ldots, C_K$;
   where N is size of data set,
   K is number of clusters,
   $C_1, C_2, \ldots, C_K$ are cluster centers.
3. do assign the n data points to the closest C_i;
   recompute $C_1, C_2, \ldots, C_K$ using Simple Mean function;
   until no change in $C_1, C_2, \ldots, C_K$;
4. return $C_1, C_2, \ldots, C_K$;
5. End

Fonte: A Hybrid Clustering Algorithm for Data Mining.

1.2. Agrupamento híbrido

Ravindra Jain propôs um algoritmo de agrupamento híbrido para Data Mining (JAIN, 2012). A ideia da proposta é aplicar duas técnicas para encontrar a média, ou seja, hora recalcula os K centros utilizando a média aritmética e hora recalcula utilizando a média harmônica.

Figura 2. Algoritmo de agrupamento híbrido.

1. begin
2. initialize Dataset D_N, K, $C_1, C_2, \ldots, C_K$, CurrentPass=1;
   where D is dataset, N is size of data set,
   K is number of clusters to be formed,
   $C_1, C_2, \ldots, C_K$ are cluster centers.
   CurrentPass is the total no. of scans over the dataset.
3. do assign the n data points to the closest C_i;
   if CurrentPass%2==0
   recompute $C_1, C_2, \ldots, C_K$ using Harmonic Mean function;
   else
   recompute $C_1, C_2, \ldots, C_K$ using Arithmetic Mean function;
   increase CurrentPass by one.
   until no change in $C_1, C_2, \ldots, C_K$;
4. return $C_1, C_2, \ldots, C_K$;
5. End

Fonte: A Hybrid Clustering Algorithm for Data Mining.

De acordo com o artigo de Ravindra Jain, a acurácia é melhor do que utilizando algoritmos tradicionais como o K-Means e K-Harmonic Means. O principal problema de um algoritmo baseado

em média é que a média é afetada fortemente por outliers (valores extremos) (JAIN, 2012). A média harmônica converge bem mesmo quando a inicialização dos clusters for ruim (JAIN, 2012).

Tanto o algoritmo básico do K-Means quanto o algoritmo híbrido foram implementados e estão disponíveis na ferramenta.

1.3. K-Means++

A inicialização dos clusters no K-Means é feita normalmente de forma aleatória. O K-Means++ é uma forma de escolher os centros para o algoritmo K-Means (ARTHUR & VASSILVITSKII, 2007). O K-Means++ é uma tentativa de melhorar a acurácia do K-Means, trata-se de uma técnica baseada em sementes para inicialização dos clusters.

Considerando que D(x) denota a menor distância de um data point para o centro mais próximo que já foi escolhido, o algoritmo pode ser descrito da seguinte forma:

Figura 3. Algoritmo K-Means++.

1a. Take one center c_1, chosen uniformly at random from $\mathcal{X}$.
1b. Take a new center c_i, choosing $x \in \mathcal{X}$ with probability $\frac{D(x)^2}{\sum_{x \in \mathcal{X}} D(x)^2}$.
1c. Repeat Step 1b. until we have taken k centers altogether.
2-4. Proceed as with the standard k-means algorithm.

Fonte: k-means++: The Advantages of Careful Seeding.

O K-Means++ foi implementado como alternativa à inicialização aleatória do K-Means.

1.4. Método Elbow

Existem diferentes abordagens na literatura para escolher o parâmetro K (número de clusters) após várias execuções do K-Means (KODINARIYA & MAKWANA, 2013).

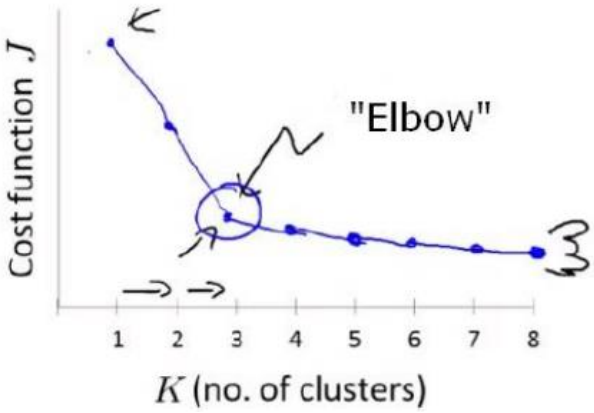
O método Elbow é um dos métodos tradicionais para determinar o número de clusters para um determinado conjunto de dados. Para implementá-lo, basta calcular a soma do erro quadrático (SSE em inglês) para alguns valores de K (por exemplo 2, 4, 6, 8, etc.). O SSE é definido como a soma da distância ao quadrado entre cada membro do cluster e seu centroide.

Figura 4. Cálculo do SSE.

$$SSE = \sum_{i=1}^{K} \sum_{x \in c_i} dist(x, c_i)^2$$

Após o cálculo do SSE para alguns valores de K, basta plotar cada K contra o seu SSE. Ao plotar, percebe-se que o erro decai com o aumento do K, isso acontece porque quando o número de clusters aumenta, a distorção é menor.

Figura 5. Elbow.



Fonte: Review on determining number of Cluster in K-Means Clustering.

A ideia do método Elbow é escolher um K no qual o SSE decai abruptamente. Isso produz o chamado "elbow effect" no gráfico. O método Elbow é uma heurística e, portanto, pode ou não pode funcionar bem em determinados casos. Às vezes, pode-se ter mais que um "elbow" ou não ter "elbow" algum.

Esse método foi implementado para que se possa ter uma dica do número de clusters. Para utilizá-lo, necessita da biblioteca externa koolplot que gera gráficos em C ou C++.

1.5. Detecção de outliers

Outliers são observações ruidosas, tais observações devem ser removidas para tornar o agrupamento mais confiável (GUHA, 1998). A eliminação de outliers permite utilizar qualquer método para o agrupamento, pois alguns métodos são muito afetados por esses outliers.

Para detecção de outliers, foi implementado o método ODIN (Outlier Detection using Indegree Number) (HAUTAMAKI, 2004). Esse método utiliza um grafo k-NN (k-nearest neighbour).

Figura 6. ODIN+K-Means.

$\{\text{ind}(\boldsymbol{x}_i)|i = 1,\ldots,N\} \leftarrow$ Calculate kNN graph
for $i \leftarrow 1,\ldots,N$ **do**
  $o_i \leftarrow 1/(\text{ind}(\boldsymbol{x}_i) + 1)$
  if $o_i > T$ **then**
    $X \leftarrow X \setminus \{\boldsymbol{x}_i\}$
  end if
end for
$(C, P) \leftarrow$ K-means(X)

Fonte: Improving K-Means by Outlier Removal.

O k-NNG (k-nearest neighbor graph) é um grafo direcionado no qual dois vértices *p* e *q* estão conectados por uma aresta se a distância entre *p* e *q* está entre as *k* menores distâncias de *p* para os outros vértices do grafo. Se o outlyingess for maior que o threshold T, então o vértice ***xi*** é considerado um outlier.

Caso o ODIN seja utilizado, a ferramenta gera um grupo somente de outliers removendo-os do restante do processo de agrupamento.

1.6. Métodos de comparação de sequências

O K-Means só trabalha com dados numéricos, portanto, precisa-se representar as sequências biológicas através de pontos em um espaço N-dimensional.

Métodos de comparação de sequências foram implementados para comparar duas sequências com o objetivo de definir uma pontuação (score) entre elas e, assim, representá-las como um ponto de forma que se possa utilizar a técnica de agrupamento K-Means.

Existem várias métricas de similaridade entre sequências de caracteres. Foram implementadas técnicas de alinhamento baseadas em programação dinâmica tais como Needleman-Wunsch (alinhamento ótimo global) e Smith-Waterman (alinhamento ótimo local). Outra técnica implementada que também utiliza programação dinâmica foi o LCS (longest common subsequence). Também foi implementada uma técnica baseada em ranking de similaridade chamada Dice's Coefficient e a distância Hamming que só pode ser utilizada para sequências de mesmo tamanho.

O custo computacional da escolha de um determinado método varia. Alinhamentos ótimos fornecem o melhor alinhamento possível, mas são custosos, por isso existem heurísticas que, se bem implementadas e com parâmetros devidamente ajustados, garantem bons resultados.

2. MOTIVAÇÃO

A determinação de clusters de sequências biologicamente relacionadas é de suma importância para análises em Bioinformática. Através de agrupamentos, é possível identificar similaridades em sequências biológicas e apresentá-las de uma forma compreensível aos biólogos para que se possa fazer análises e identificar padrões.

A ferramenta desenvolvida é open-source, pode ser utilizada facilmente por pessoas que não possuem conhecimento em linguagens de programação como também por pessoas que queiram utilizar como uma biblioteca incluindo em seus projetos. A princípio, os parâmetros obrigatórios são apenas o número de clusters e um arquivo no formato FASTA com as sequências biológicas.

Essa ferramenta poderá ser utilizada em um projeto que visa à identificação de variações genéticas utilizando grafos De Bruijn. No projeto em questão, um grafo de sobreposição é gerado tendo como entrada reads (sequências de nucleotídeos produzidas por máquinas sequenciadoras). Através de métodos de busca nesse grafo, é possível identificar topologias que podem ser detectadas como variações. Essas variações são importantes no estudo de genomas como é o caso da cana-de-açúcar. A cana-de-açúcar possui um genoma bastante complexo que ainda não foi totalmente sequenciado, portanto, essa espécie é alvo de vários estudos que tentam compreender melhor o seu genoma

objetivando o melhoramento genético visto que a cana-de-açúcar é uma importante matéria-prima da qual é produzido, por exemplo, o etanol.

3. TESTES

Como forma de validar as implementações, testes foram feitos com data sets retirados do UCI Machine Learning Repository (LICHMAN, 2013) que possui vários conjuntos de dados que servem para testar algoritmos de aprendizado de máquina.

3.1. Promoter data set

O primeiro data set testado foi o Promoter Gene Sequences. O objetivo é agrupar sequências de DNA em duas classes: se é promotor ou não. Promotores são sequências de DNA específicas importantes para o início da transcrição. Esse data set possui um total de 106 sequências de mesmo comprimento sendo que 53 são promoters (representadas por "+") e as outras 53 não são promoters (representadas por "-").

Foram feitos vários testes variando os parâmetros, os melhores resultados foram obtidos utilizando a distância Hamming.

Para o teste a seguir, foi utilizado K-Means++, distância Hamming e a proposta híbrida de agrupamento. Como são duas classes, o K (número de clusters) é igual a 2.

Figura 7. Cluster 1 – Promotor dataset – Teste 1.

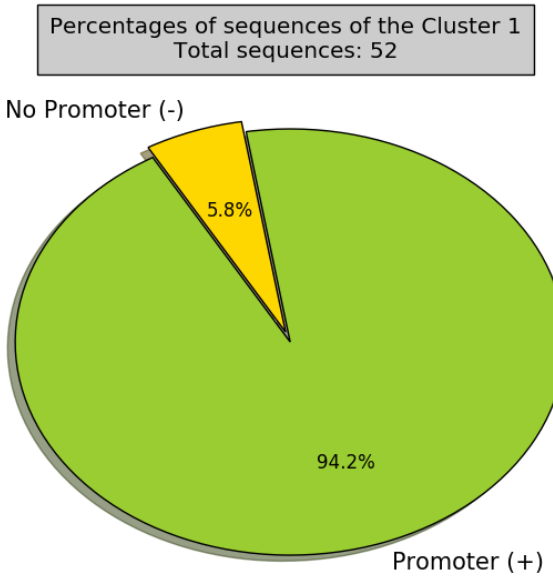
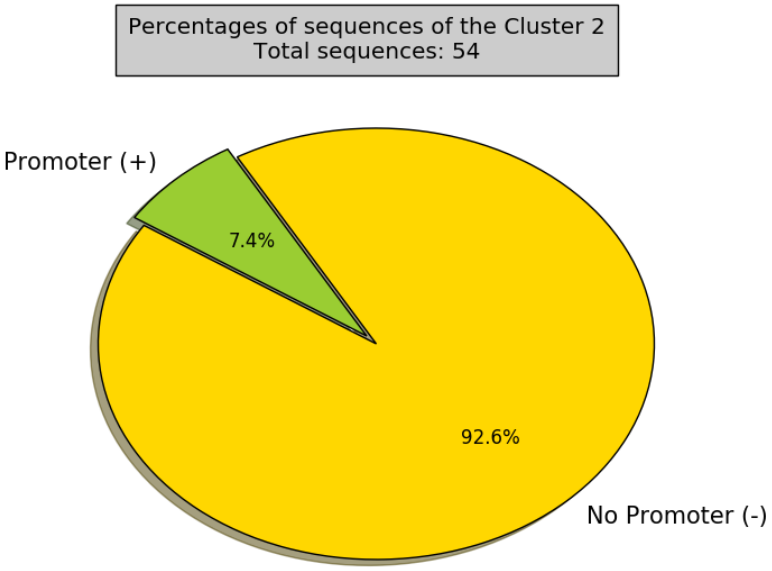


Figura 8. Cluster 2 – Promotor dataset – Teste 1.



O próximo teste ilustra o resultado de uma execução utilizando K-Means++, distância Hamming, mas sem utilizar a proposta híbrida de agrupamento.

Figura 9. Cluster 1 – Promotor dataset – Teste 2.

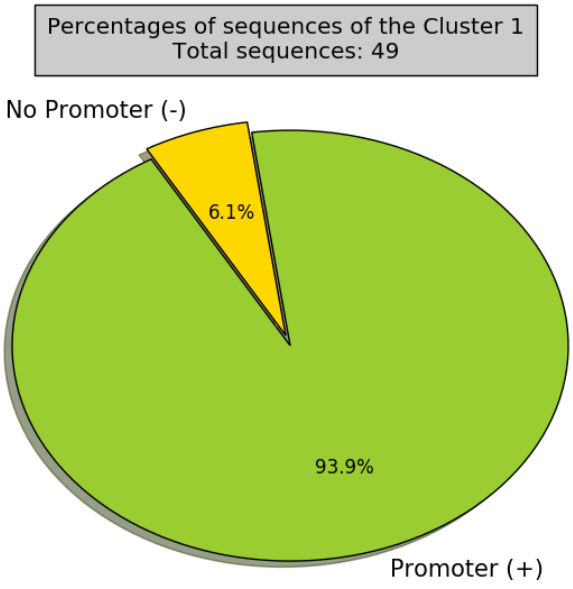
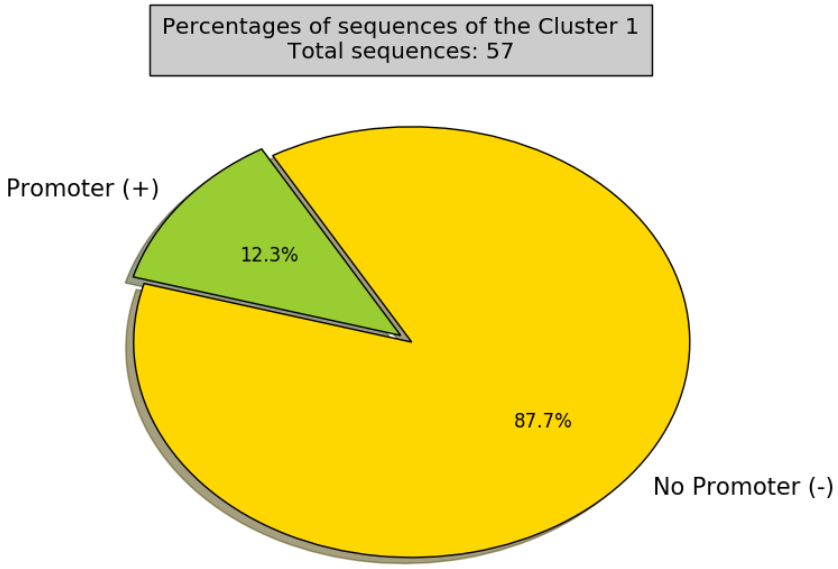


Figura 10. Cluster 2 – Promotor dataset – Teste 2.



Esses testes foram pontuais para demonstrar resultados para determinadas execuções, mas isso não significa que com uma mesma combinação de parâmetros o resultado será o mesmo. Por isso, são necessários estabelecer critérios com o intuito de tornar os testes mais robustos objetivando detectar a confiabilidade do agrupamento.

Tendo em vista que o Promoter data set possui ao todo 106 sequências, 2 classes e 53 sequências de cada classe, o primeiro critério estabelecido para determinar se um agrupamento foi aceito para esse data set é se o total de sequências de cada cluster for pelo menos 40 e, para cada cluster, uma das classes deve representar pelo menos 80% do total de sequências daquele cluster. A tabela a seguir ilustra o resultado para 100 testes para cada configuração.

Tabela 1. Testes Promoter Dataset – 106 sequências - 100 testes por configuração

	Porcentagem de agrupamentos aceitos	Porcentagem de agrupamentos não aceitos
K-Means++; Agrupamento Híbrido; distância Hamming	93%	7%
K-Means++; distância Hamming	61%	39%

K-Means; distância Hamming	58%	42%
K-Means; Agrupamento Híbrido, distância Hamming	94%	6%

A tabela 1 demonstra de forma clara que para esse data set, a combinação de K-Means (ou K-Means++) utilizando agrupamento híbrido e distância Hamming é a melhor configuração.

O último teste que feito com o Promoter data set teve um critério mais rigoroso. Será considerado um agrupamento aceito se o total de sequências de cada cluster for pelo menos 50 e, para cada cluster, uma das classes deve representar pelo menos 90% do total de sequências daquele cluster.

Tabela 2. Testes Promoter Dataset com critério mais rigoroso – 106 sequências - 100 testes por configuração

	Porcentagem de agrupamentos aceitos	Porcentagem de agrupamentos não aceitos
K-Means++; Agrupamento Híbrido; distância Hamming	88%	12%
K-Means++; distância Hamming	20%	80%
K-Means; distância Hamming	23%	77%
K-Means; Agrupamento Híbrido, distância Hamming	86%	14%

A tabela 2 mostra uma diminuição das porcentagens de agrupamentos aceitos. Isso era esperado pelo fato do critério ter sido mais rigoroso.

3.2. Splice data set

O segundo data set testado foi o Splice data set. Trata-se de um data set que possui 3 classes (EI, IE, N), portanto, o número de clusters será 3. O data set original contém um total de 3190 sequências (767 sequências da classe EI, 768 sequências da classe IE e 1655 sequências da classe N).

Para um primeiro teste, esse data set foi reduzido para 900 sequências com 300 sequências por classe. O critério utilizado para determinar se o agrupamento foi aceito é se o número de sequências de cada cluster for pelo menos 250 e, para cada cluster, uma das classes deve representar pelo menos 85% do total de sequências daquele cluster.

Tabela 3. Testes Splice Dataset – 900 sequências - 100 testes por configuração

	Porcentagem de agrupamentos aceitos	Porcentagem de agrupamentos não aceitos
K-Means++; distância Hamming	90%	10%
K-Means; distância Hamming	89%	11%

Nenhuma combinação testada com agrupamento híbrido produziu algum agrupamento aceito com o critério estabelecido. Para produzir agrupamentos aceitos com agrupamento híbrido, o critério teve que ser menos rigoroso, ou seja, um agrupamento agora é considerado aceito se o número de sequências de cada cluster for pelo menos 200 (não mais 250) e, para cada cluster, uma das classes deve representar pelo menos 80% (não mais 85%) do total de sequências daquele cluster.

Tabela 4. Testes Splice Dataset com critério menos rigoroso – 900 sequências - 100 testes por configuração

	Porcentagem de agrupamentos aceitos	Porcentagem de agrupamentos não aceitos
K-Means++; Agrupamento Híbrido; distância Hamming	92%	8%
K-Means++; distância Hamming	96%	4%
K-Means; distância Hamming	95%	5%
K-Means; Agrupamento Híbrido, distância Hamming	87%	13%

Pela a tabela 4 é possível visualizar que as configurações que não possuem agrupamento híbrido tiveram um aumento da porcentagem de agrupamentos aceitos em comparação com a tabela 3 e em comparação com as configurações que utilizam o agrupamento híbrido. Isso era esperado pelo resultado da tabela 3 e pelo fato do critério ter sido menos rigoroso.

Para o terceiro teste com o Splice data set, foram consideradas 2100 sequências com 700 sequências por classe. O critério utilizado para determinar se o agrupamento foi aceito é se o número de sequências de cada cluster for pelo menos 600 e, para cada cluster, uma das classes deve representar pelo menos 85% do total de sequências daquele cluster.

Tabela 5. Testes Splice Dataset – 2100 sequências - 50 testes por configuração

	Porcentagem de agrupamentos aceitos	Porcentagem de agrupamentos não aceitos
K-Means++; distância Hamming	100%	0%
K-Means; distância Hamming	100%	0%

Como bem mostra a tabela 5, todos os agrupamentos produzidos pelas configurações que não utilizaram a abordagem híbrida foram aceitos. Nenhuma combinação testada com agrupamento híbrido produziu algum agrupamento aceito com o critério estabelecido. Para produzir agrupamentos aceitos com o agrupamento híbrido, o critério teve que ser menos rigoroso, ou seja, um agrupamento agora é considerado aceito se o número de sequências de cada cluster for pelo menos 550 (não mais 600) e, para cada cluster, uma das classes deve representar pelo menos 80% (não mais 85%) do total de sequências daquele cluster.

Tabela 6. Testes Splice Dataset com critério menos rigoroso – 2100 sequências - 50 testes por configuração

	Porcentagem de agrupamentos aceitos	Porcentagem de agrupamentos não aceitos
K-Means++; Agrupamento Híbrido; distância Hamming	92%	8%
K-Means++; distância Hamming	100%	0%
K-Means; distância Hamming	100%	0%
K-Means; Agrupamento Híbrido, distância Hamming	94%	6%

O último teste com Splice data set considera todas as 3190 sequências (767 sequências da classe EI, 768 sequências da classe IE e 1655 sequências da classe N). O critério utilizado para determinar se o agrupamento foi aceito é se os clusters tiverem a seguinte distribuição de sequências: pelo menos 700 sequências em dois clusters e pelo menos 1300 sequências no terceiro cluster. Além disso, em cada cluster uma classe deve representar pelo menos 75% do total de sequências daquele cluster.

Tabela 7. Testes Splice Dataset– 3190 sequências – 50 testes por configuração

	Porcentagem de agrupamentos aceitos	Porcentagem de agrupamentos não aceitos
K-Means++; distância Hamming	80%	20%
K-Means; distância Hamming	90%	10%

Nenhum teste feito com agrupamento híbrido produziu agrupamentos aceitos com o critério estabelecido.

4. CONCLUSÃO

O agrupamento de dados é algo desafiador. Várias são as abordagens e técnicas que podem ser utilizadas para obter bons resultados, mas conhecer os dados é de suma importância para escolher a melhor abordagem e, assim, obter melhores resultados.

Esse trabalho teve como objetivo principal dá suporte a um projeto em andamento e fornecer uma ferramenta (ou biblioteca em C++) na área de Bioinformática que possa ajudar a compreender dados biológicos.

O projeto é open-source, o código está disponível no GitHub para livre acesso, exemplos de como executar assim como data sets, referências dentre outros arquivos estão disponíveis na página do projeto: https://github.com/marcoscastro/clustering_bio_sequences.

5. TRABALHOS FUTUROS

Vários melhoramentos podem ser feitos tais como um refinamento do agrupamento após a execução do K-Means utilizando heurísticas tais como Algoritmos Genéticos (AGs) ou ACO (Ant Colony Optimization).

Quanto aos métodos de comparação de sequências, poderia ser implementado um AG para realizar o alinhamento entre sequências como alternativa aos métodos tradicionais de alinhamentos tais como Needleman-Wunsch ou Smith-Waterman ou mesmo fazer um híbrido de alinhamento utilizando a saída do BLAST (Basic Local Alignment Search Tool) para a população do AG com o objetivo de possivelmente melhorar os resultados do AG.

6. REFERÊNCIAS

Linden, R. Técnicas de Agrupamento. **Revista de Sistemas de Informação da FSMA**. Rio de Janeiro, n. 4, pp. 18-36, 2009.

MacQueen, J. Some methods for classification and analysis of multivariate observations. 1967 Proc. **Fifth Berkeley Sympos. Math. Statist. and Probability** (Berkeley, Calif., 1965/66) Vol. I: Statistics, pp. 281–297 Univ. California Press, Berkeley, Calif.

Ravindra Jain, "A Hybrid Clustering Algorithm for Data Mining", **IEEE Transaction on Neural Networks**, June 2012.

Arthur, D. and Vassilvitskii, S. (2007). "k-means++: the advantages of careful seeding". **Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms. Society for Industrial and Applied Mathematics Philadelphia**, PA, USA. pp. 1027–1035.

Kodinariya, M. and Makwana, R. Review on determining number of Cluster in K-Means Clustering. **International Journal of Advance Research in Computer Science and Management Studies**. Vol. 01, 2013.

Guha, S., Rastogi, R., Shim, K. CURE na efficient clustering algorithm for large databases. **Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data**, Seattle, Washington, p. 73-84, 1998.

Hautamaki, V., Karkkainen, I., Franti, P. Outlier detection using k-nearest neighbour graph. **17th International Conference on Pattern Recognition**, Cambridge, United Kingdom, 2004, 430-433.

Lichman, M. (2013). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.