CptS 475
Zicheng Gu
Yi Chou
475 final report - protein clustering

Shout_sequence repeats(SSRs) occur in both prokaryotic and eukaryotic DNA, inter and intragenically, and may be exact or inexact copies. When heterogeneous SSRs are present in a given locus, we can take advantage of the pattern of different repeats to genotype strains based on the SSRs. Cataloging and tracking these repeats can be difficult as diverse groups of researchers are involved in the identification of the repeats. Additionally, the task is error-prone when done manually. So we need to use clustering methods to find what is the different pattern and if the clustering methods fit this data it will help a lot on the tracking and cataloging.
[1]
The goal we want to achieve is when heterogeneous SSRs are present in a given locus, we can take advantage of the pattern of different repeats to genotype strains based on the SSRs.The task of recognizing repetition is difficult to complete manually, so various clustering methods can effectively help us identify different repetitive patterns to avoid some manual mistakes. The importance of solving this problem is that if the genotyping can be completed quickly, this result will greatly speed up the development of various vaccines, and may also help us to understand strain variation and distribution more deeply. So our basic approach is to try different clustering methods to analyze the short sequence repeats of the species A. marginale. Considering each clustering method has to be compared with each other, so we decide to try three clustering methods which is hierarchical clustering, k-means clustering based on the euclidean distance and k-means clustering based on the term frequency. After getting the result from each clustering method then we will compare the result we get, and see what is the difference between the clustering methods. In addition to analyzing the clustering method itself we also have to figure out if there is any particular pattern in the data based on the result we get. If there is any particular pattern, we need to check will this pattern help us deal with this data and if the clustering methods we try are the best approach for this kind of data file. The present result will be the graph of hierarchical clustering, k-means clustering based on euclidean distance and k-means clustering based on the term frequency and get the conclusions after we compare each result we get. Then find out which clustering method is most suitable for processing this data type.
[1][2]
If we want to accurately define our problem definition then we have to clearly understand what SSRs are and need to understand how analyzing this data will help us in real life. Short sequence repeats(SSRs), also known as variable number of tandem repeats or microsatellites, are inherently unstable entities that frequently change the number of repeat units through slip-strand mismatches during DNA synthesis. In

humans, the single-number variation of SSRs is related to the occurrence of specific genetic diseases, while in microorganisms, SSRs is closely related to the regulation of gene expression. Understanding the functional limitations of SSRs reveals their potential use as molecular clocks to monitor the evolution of microbial genomes. Although microbial SSRs genotypes have been used more and more frequently to study the epidemiology and evolution of microbial strains and isolates, such methods should be used with caution. So the problem we are exploring right now is how can we use different repeated patterns to type genotypic strains based on SSR. Slip-strand mismatches(SSM) I mentioned above this complex phenomena can occur relatively easily. SSM is closed to the frequency of 10(-4) per bacterial cell division and allows high-frequency genetic transformation. Bacteria use this stochastic strategy to adjust their genetic pools in response to selective environmental pressures. SSR-mediated variation has important implications for bacterial pathogenesis and evolutionary fitness. Molecular analysis of SSRs changes allows epidemiological studies on the spread of pathogens. The occurrence, evolution and function of SSRs, as well as the molecular methods used to analyze them, are discussed in the context of responsiveness to environmental factors, bacterial pathogenicity, epidemiology, and the availability of whole genome sequences for more and more applications. The more microorganisms, especially those that are medically related. Short sequence repeats(SSRs) are very important in the medical field because it cover a wide range of aspects in the medical field. And its role in epidemiology is particularly important. So how to solve this problem faster is very important, and it is also very interesting to understand the mechanism of SSRs in depth.

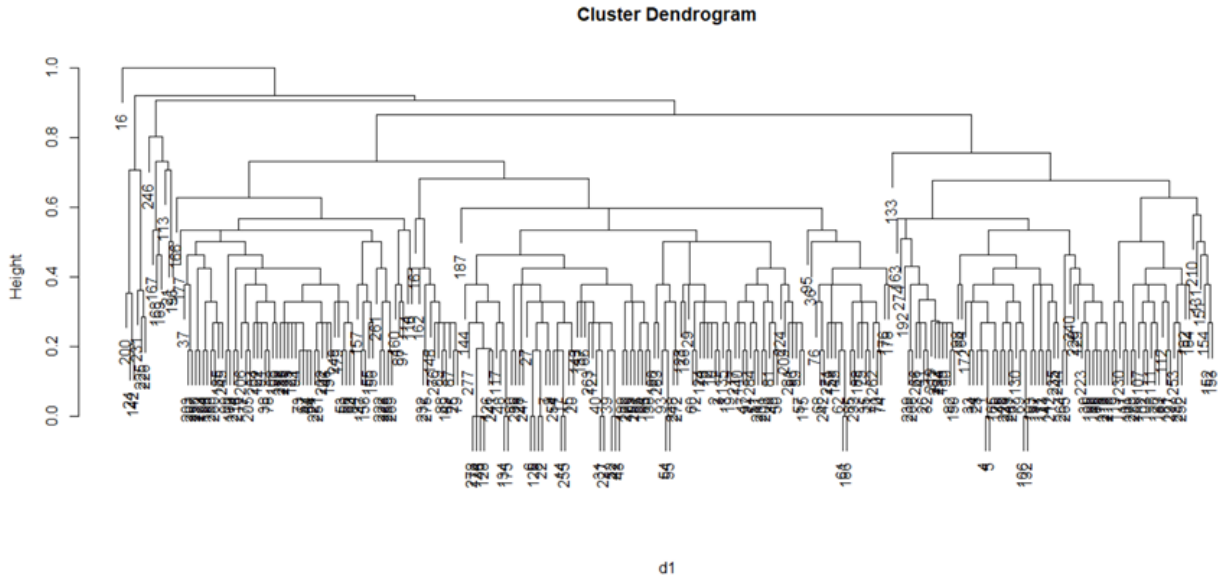
In this project, we use clustering models, including Kmeans clustering model, Hierarchical Clustering model, and clustering analysis implemented by k-means and hclust function of R language. Because we are analyzing protein short sequence data sets The protein data set has no category label data, so it is not easy for us to define the metric of the clustering, and it is impossible to know the accuracy of the clustering, so the total sum of squares of all samples is used here to measure the effect of the clustering. The total sum of squares can be obtained through the totss attributes of the kmeans function (code km$totss). The smaller the totss, the better the clustering effect. Because the original data is sequence data such as ‘ddsssasgqqqessvssqseastssqlg’, we need to convert it into a distance metric. In this article, we tried 3 kinds of distance metric, the first one is the alignment score with biological significance, It can express the difference in biological sequence with distance; Second is to perform one-hot encoding of the original data into 0-1 numerical data, and then calculate the euclidean distance; the third is we introduce the technology of text mining, which treats sequences such as ‘ddsssasgqqqessvssqseastssqlg’ as text data, and treats each word as one words, then

use the term frequency to count the number of letters that appear in each sequence, and then calculate the euclidean distance.

Hierarchical Clustering

We use ama from the seqinr package<read.alignment("AmargRepeats.fasta",format = "fasta") to read AmargRepeats.fasta data. Then use the dist.alignment function to obtain the alignment score, alignment score is Pairwise Distances from Aligned Protein or DNA/RNA Sequences: These functions compute a matrix of pairwise distances from aligned sequences using similarity (Fitch matrix, for protein sequences only) or identity matrix (for protein and DNA sequences). The resulting matrix contains the square root of the pairwise distances. For example, if the identity between 2 sequences is 80 the square root of (1.0-0.8) ie 0.4472136.

Then use the Hierarchical Clustering model for alignment score As a result, 284 sequences were clustered to obtain Cluster Dendrogram.



Generally, we can choose the appropriate number of clusters K from Cluster Dendrogram, but from the above Cluster Dendrogram we find that the cluster structure is very complicated, and the number of clusters K cannot be simply determined. So we choose K by comparing the number of samples in the clusters corresponding to different K, and try to make the samples in different clusters more even. By using the for loop, we tried clustering results of K from 2 to 10, and output different K When, the number of samples in each cluster.

When k=2, the number of samples in the cluster is: 283,1, which is very unbalanced.

When k=3, the number of samples in the cluster is: 277, 1, 6, which is also very unbalanced.

When k=4, the number of samples in the cluster is: 270,1,7,6,

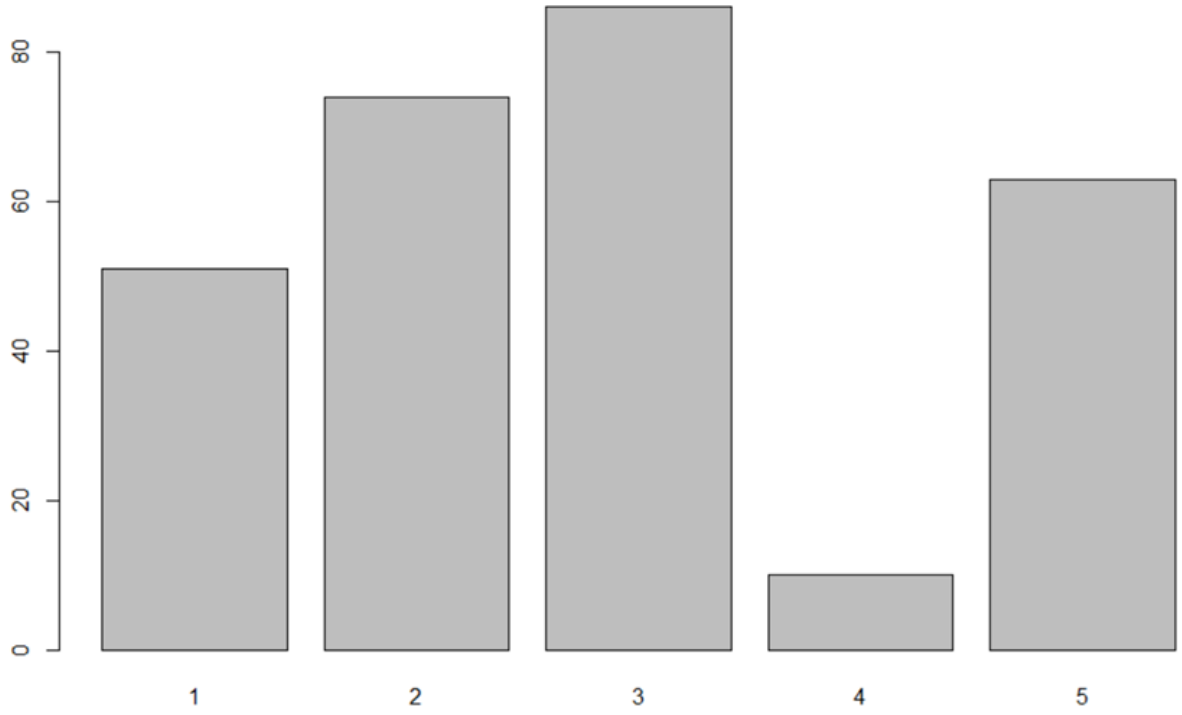
When k=5, the number of samples in the cluster is: 269,1,7,6,1

When k=6, the number of samples in the cluster is: 269 1 6 6 1 1
When k=7, the number of samples in the cluster is: 83 186 1 6 6 1 1
When k=8, the number of samples in the cluster is: 83 186 1 6 3 1 3 1;
When k=9, the number of samples in the cluster is: 83 186 1 3 3 1 3 3 1;
When k=10, the number of samples in the cluster is: 83 124 1 3 3 62 1 3 3 1;

From the different K statistics above, the number of samples in each cluster shows that the sample distribution is not very balanced. When K=5, the sample distribution is much more balanced than others, so we choose K=5.
Since this data set is protein sequence data with special biological meaning, we use the alignment score K of the basic biological meaning metric as the K of all subsequent clusters, in order to facilitate the comparison of the effects of different clusters.

One-hot encoding sequence, then take euclidean distance, cluster with k--means [3]

Because clustering needs to calculate the distance, we convert the sequence such as: ‘ddsssasgqqqessvssqseastssqlg’ into numerical data. Through statistical analysis, we find that the sequence length is between 28-31, and the sequence letter does not have the value o, so we want the sequence to be unified as the length 31 so we filled the sequence that length less than 31 with the letter “o”. In this way, the length of all sequences is 31, and then the sequences are divided, and each sequence is regarded as 31 features, and each feature takes a letter with a different value. Then for each of the 31 features, use one-hot encoding as 0-1, so that the sequence data ‘ddsssasgqqqessvssqseastssqlg’ is converted into numerical data with a value of 0-1, and then the euclidean distance can be calculated on the numerical matrix, so the distance of one-hot encoding takes into account the sequence information of the sequence. Then use the kmeans function in R to perform k-means clustering. The value of K is 5, and the category result of the clustering is output (code km$cluster). The distribution of the number of samples in the cluster is 51 74 86 10 63, which is more concentrated.



Text feature extraction based on tf term frequency

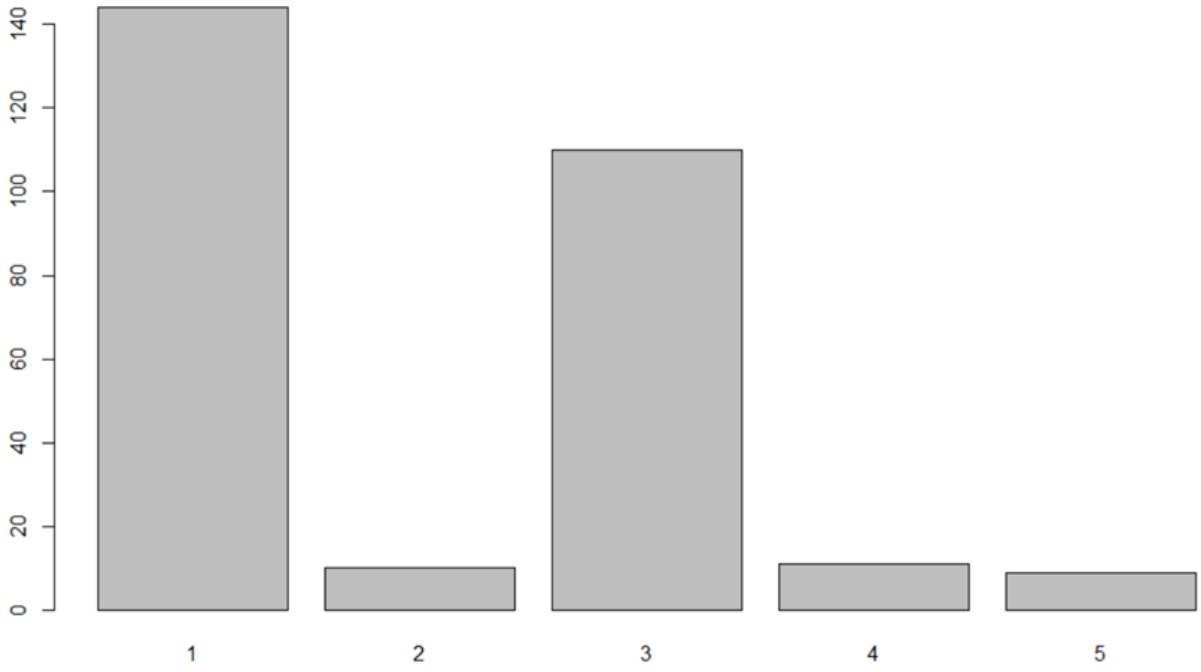
Based on the technology of statistical word frequency of text mining, we treat the data of each protein sequence as a text, and then each letter in the sequence is regarded as a word, so that the term frequency in text mining can be used to calculate the sequence. For example: ddsssasgqqqessvssqseastssqlg is converted into numerical data, where the data counts the number of times each letter appears in each sequence, but this statistical method ignores the sequence information of the sequence and only counts the frequency of the letters.

First divide the sequence into single characters, and then use the data processing code to calculate the number of different letters appearing in the specific sequence, and finally the sequence is processed into the following result, displaying the first 6 lines:

```
##   a c d e f g h i k l m n p q r  s t v w x y
## A 2 0 2 2 0 2 0 0 0 1 0 0 0 5 0 12 1 1 0 0 0
## B 3 0 2 1 0 3 0 0 0 1 0 0 0 6 0 11 1 1 0 0 0
## C 3 0 1 1 0 4 0 0 0 1 0 0 0 6 0 11 1 1 0 0 0
## D 3 0 1 2 0 3 0 0 0 1 0 0 0 5 0 12 1 1 0 0 0
## E 3 0 1 2 0 2 0 0 0 1 0 0 0 5 0 12 1 1 0 0 0
## F 2 0 1 1 0 3 0 0 0 1 0 0 0 6 0 12 2 1 0 0 0
```

Each row corresponds to a sequence, each column represents a letter, and the number represents the number of times the letter appears in the specified sequence. The matrix is called the document term matrix.

Then, in order for different letters to have different weights when calculating the distance, we standardize the data, and then use the kmeans function to perform the k-means distance. The K value is 5, and the clustering category results (code km$cluster) are output at the same time, the samples in the cluster The distribution of numbers is 144 10 110 11 9, and the distribution is quite different.



Comparison of clustering results

Compare the k-means clustering effect of two numerical methods

km$totss

[1] 2354.954

km2$totss

[1] 5943

Use The total sum of squares of all samples to measure the effect of clustering. The total sum of squares can be obtained through the totss attributes of the kmeans function (code km$totss). The smaller the totss, the better the clustering effect. From The Looking at the total sum of squares, we found that using one-hot to encode the sequence, and then take the euclidean distance, the effect of using k-means clustering is better than the clustering effect of text feature extraction based on tf term frequency, which may be due to the tf term frequency term The text feature extraction of frequence only considers the frequency of letters, while ignoring the sequence information of the protein. On the contrary, the one-hot encoding sequence considers each sequence information.

Compare the category distributions of the three clusters

table(AmargRepeats$y1,AmargRepeats$y2)

##

1 2 3 4 5

1 51 135 31 51 1

2 0 0 0 0 1

3 1 0 6 0 0

4 0 0 0 0 6

5 1 0 0 0 0

table(AmargRepeats$y1,AmargRepeats$y3)

##

1 2 3 4 5

1 9 6 110 137 7

```
##   2   0   0   0   1   0

##   3   1   2   1   3   0

##   4   0   0   0   6   0

##   5   0   1   0   0   0
```

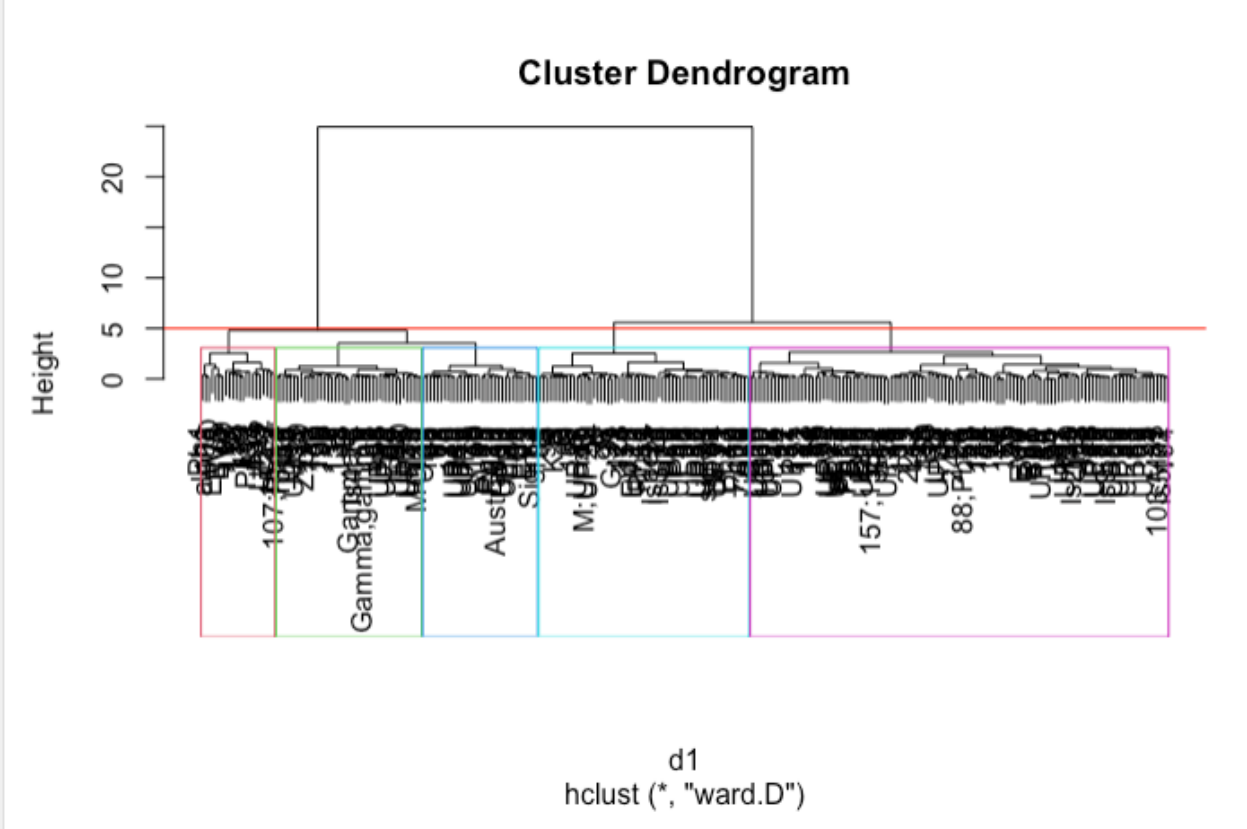
```
table(AmargRepeats$y2,AmargRepeats$y3)
```

```
##

##      1  2  3  4  5

##   1  4  1 17 28  3

##   2  6  3 65 59  2

##   3  0  3 14 19  1

##   4  0  2 15 33  1

##   5  0  0  0  8  0
```

Here we only compare the differences between different clustering methods, and the biological differences will be reflected in the result.

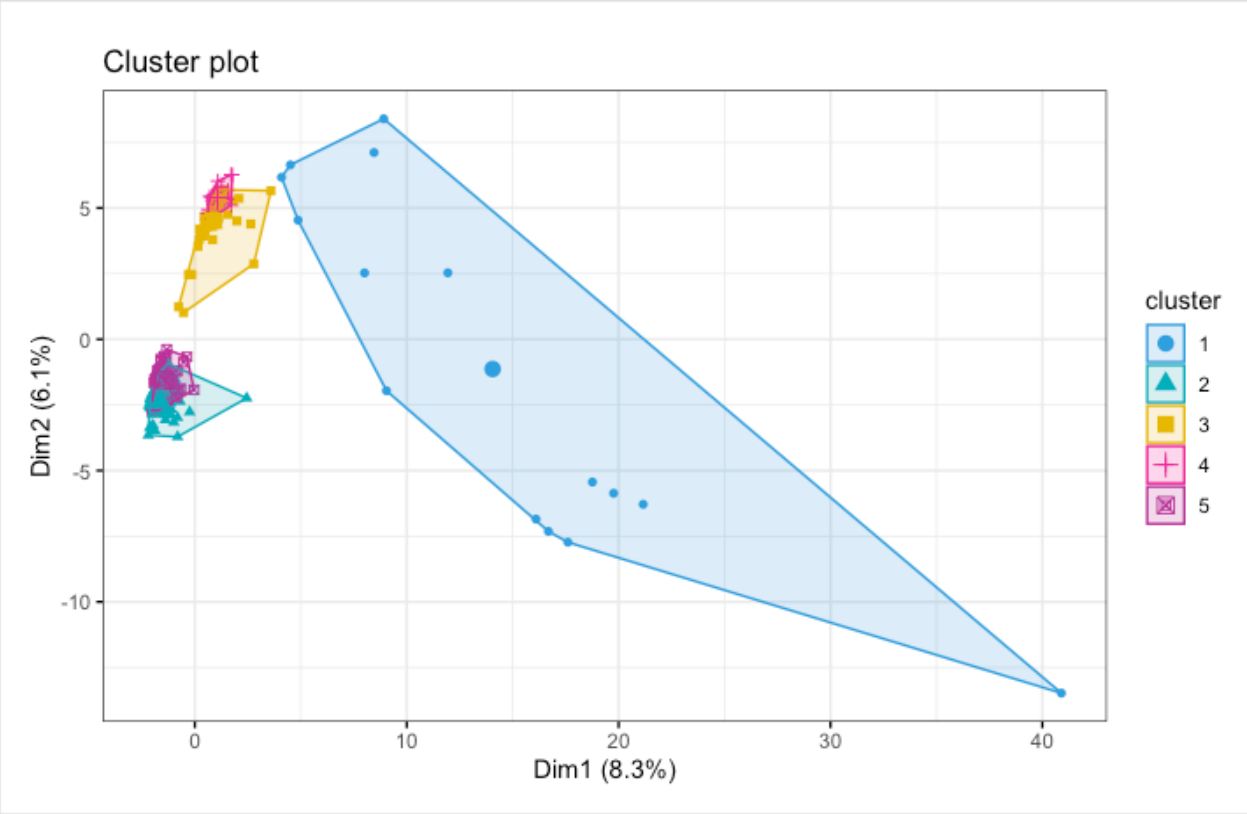
From the comparison results above, we find that different distance measures, even if we choose the same K, will produce very different clustering results. For example, we use the one-hot encoding sequence for y2, and then use K-means to gather the Euclidean distance. Class to get the result, y3 is that we treat the sequence as text content, and then perform K-means clustering. We find that the results of y2 and y3 are very different. It shows that the clustering model relies heavily on distance metrics. Different distance metrics produce different clustering results. At the same time, we have different interpretations for the results of different clusters. This interpretation needs to be combined with specific data meanings to illustrate the clustering results. It may be uncontrollable, and is usually used to explore the laws of data and find interesting patterns in the data.

1. Hierarchical clustering graph



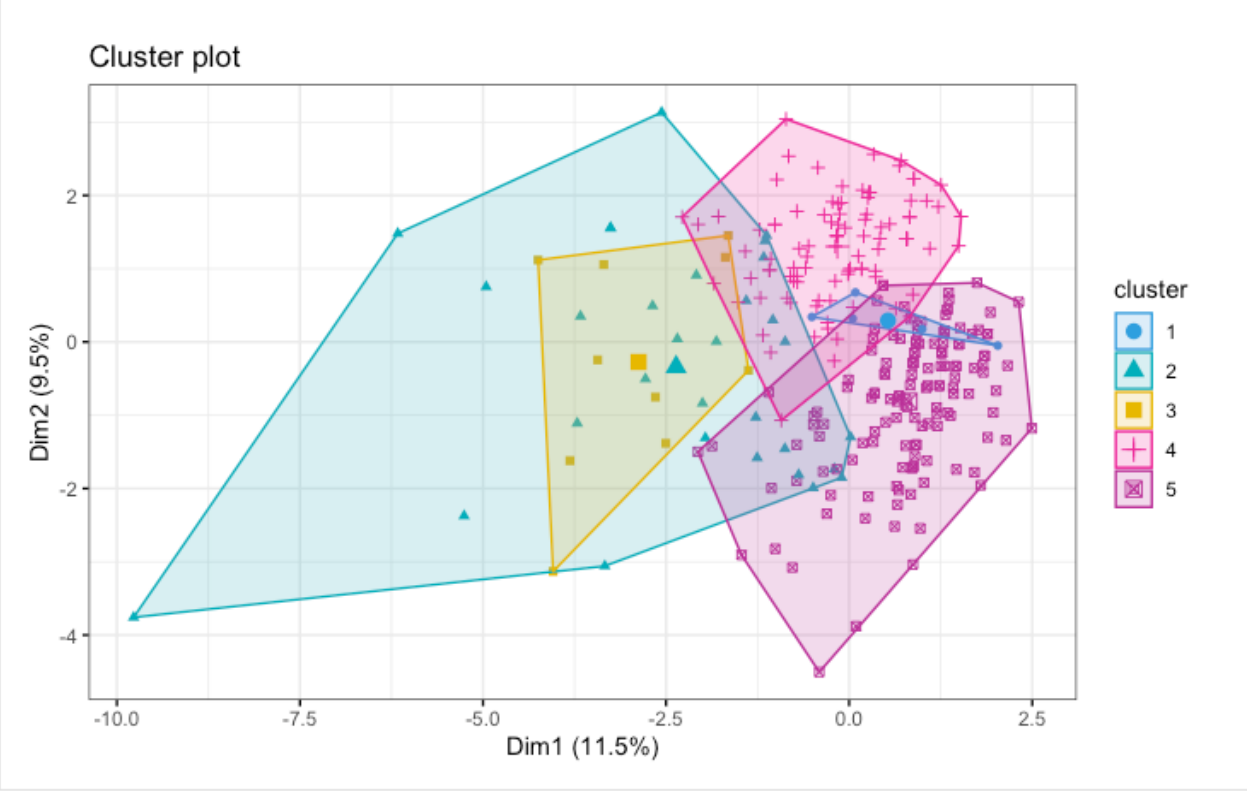
From this figure, we can see that the data files we use are divided into five major categories. The first clustering method we used is hierarchical clustering, which means that we don't need to give any limits to the algorithm and every calculation will be done by the algorithm itself. But if we leave all the calculation to the algorithm itself, which has both advantages and disadvantages. The advantage is the similarity between distance and rules is easy to define, with few restrictions, no need to specify the number of clusters in advance and can discover the hierarchical relationship of classes. The disadvantage is the computational complexity is too high, singular values can also have a great impact and the algorithm is likely to cluster into chains format.

2.k-means clustering based on the euclidean distance graph



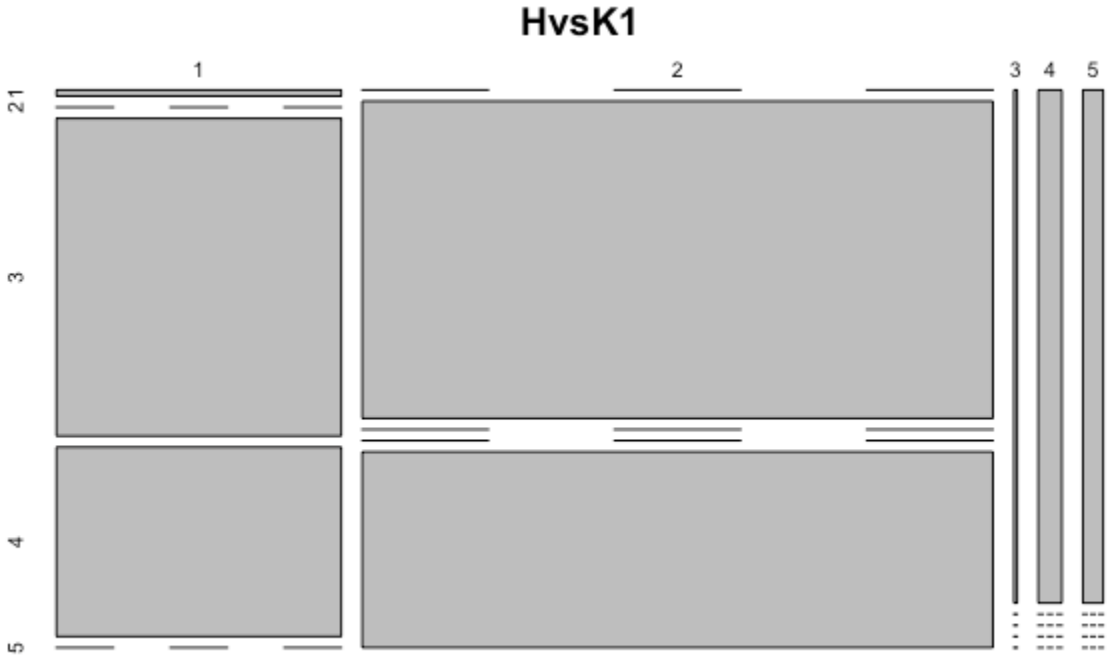
The second clustering method we try is k-means clustering based on the euclidean distance. From this figure, we can see that cluster 1 occupies most of the area. Clusters 3 and 4 are in the upper left corner and the areas overlap. Clusters 2 and 5 are located below clusters 3 and 4, and like clusters 3 and 4 do not occupy a lot of positions in the whole picture. And like clusters 3 and 4, some areas are overlapped. I think the reason for the overlap is that their euclidean distances are very close, which means their similarity is very high. We can see that the points in cluster 1 are very rare, but the area across is very large. So we can say that cluster 1 will not have a huge impact on the result we want to get. And the advantage of this method is simple, easy to understand and implement, and also low time complexity. The disadvantage is first, we need to manually enter the number of classes, which is very sensitive to the initial value setting. Second, k-means mainly finds round or spherical clusters, and cannot identify non-spherical clusters. Third, k-means clustering is very sensitive to noise and outliers.

3.k-means clustering based on the term frequency graph

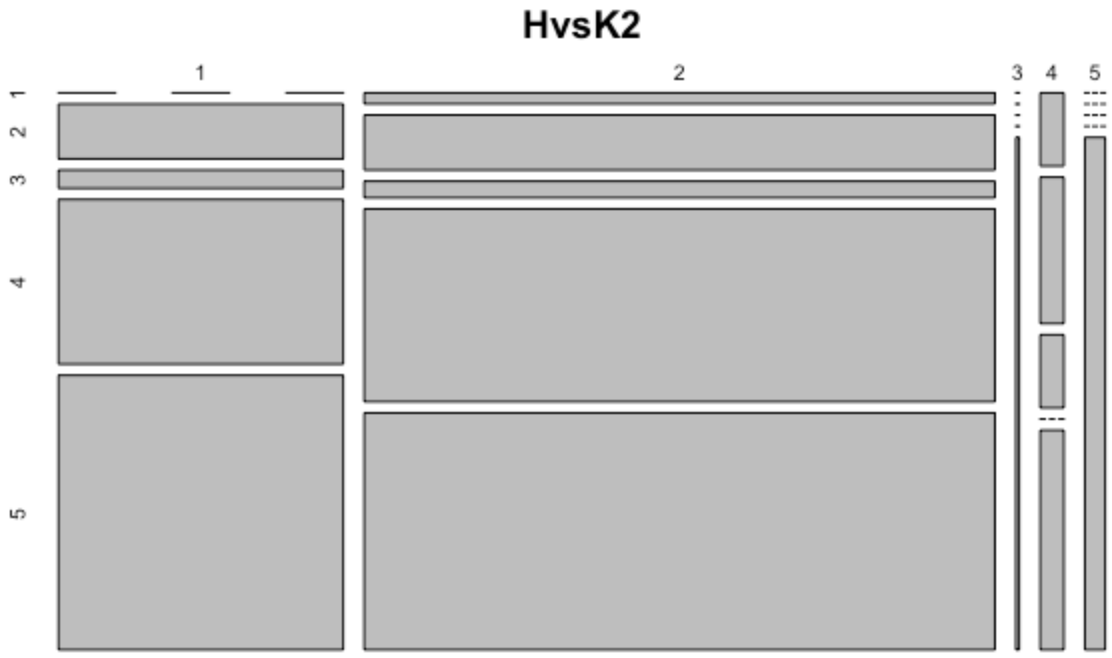


The third method we use is k-means clustering based on the term frequency. Same as the previous figure, we also set the k value equal to 5. But the figure this time is very different from the previous one. From the figure we can see this time cluster 2 occupies most of the area and completely contains cluster 3, and it also overlaps with clusters 4 and 5. Half of cluster 1 is in cluster 4, and half is in cluster 5, and because the areas of clusters 4 and 5 overlap it is completely contained in clusters 4 and 5. Except for cluster 3 and cluster 1, which do not overlap, the remaining clusters basically overlap with other clusters. We can see that the points in cluster 2 are very rare, but the spanning area is very large, which shows that k-means is indeed very sensitive to outliers. At the same time we can determine that the types of outlier sequence are very different from other clusters. Since we use the same clustering method the advantage and disadvantage are the same with the previous method.

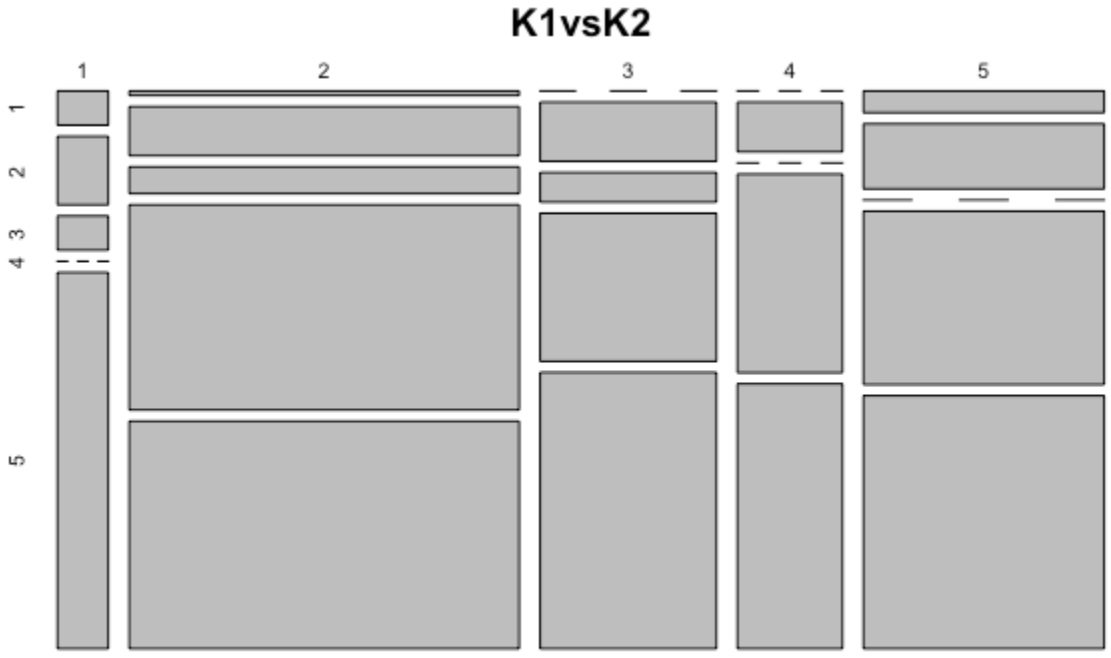
4. Hierarchical clustering vs k-means clustering based on the euclidean distance graph



5. Hierarchical clustering vs k-means clustering based on the term frequency graph



6. K-means clustering based on the euclidean distance vs k-means clustering based on the term frequency graph



Above three is all the comparison results we get. The 1,2,3,4, and 5 on the x and y axis represent the 5 clusters that each clustering method gets, and the number on the

columns and rows show us how the data is distributed in the cluster. From the first two graphs we compare hierarchical clustering and other two k-means clustering methods and we can see the graph is not that balanced.most of the data is concentrated on the (2,2),(1,3),(1,4),(2,4) these four points. This imbalance indicates that the results obtained by the two clustering methods have a relatively large deviation. On the second graph we also can see this kind of problem, which means different clustering methods we use will cause different clusters. Maybe we can get silmar cluster result but not for the hierarchical and k-means clustering. On the third graph, we can see it is much more balanced than the first one and second one. It is because we use the same clustering method for the cluster result we get and it means the coincidence rate of each of their corresponding clusters is very high. For this project I think hierarchical clustering and k-means clustering are all very suitable for this project but if this data file gets bigger and bigger the k-means will be better than the hierarchical clustering. Because hierarchical clustering computational complexity is too high, and if we can find a good k value the k-means clustering is probably more suitable than hierarchical clustering.
[1]
We found a project that is related to our work on the internet. The problem they are solving is pretty much the same problem we are exploring.And it is called RepeatAnalyzer, which is a software tool that can use marginal slurries as a model species to track, manage, analyze, and catalog SSR and genotypes. For the visualization function, RepeatAnalyzer can generate genotypes in the region of interest or high-quality maps of the geographical distribution of SSRs. RepeatAnalyzer’s repeat identification functionality was validated for all SSRs and genotypes reported in 21 publications, using 380 A. marginale isolates gathered from the five publications within that list that provided access to their isolates. When applied to A.marginale, the software will show genotype lengths for a given region follow a normal distribution, while SSR frequencies follow a power-law-like distribution. Our project does not have that many functions but I think the way we do the classification is similar, their way is more like the k-means clustering but it depends on 380 A. marginale isolates gathered from the five publications. And their method is much more complex than our method.

Conclusion

Short sequence repeats (SSRs) are inherently unstable entities. In the process of DNA synthesis, slip chain mismatches often change the number of repeat units. From the two k-means we have completed, we can see that this kind of mismatch has a specific pattern. From the k-means graph we get the synthesized DNA can be roughly divided into 3 categories, of which 2 categories occupy most of the data, and the remaining category is very small but has a wide range of distribution. Therefore, we need to pay special attention to these special cases. And after knowing the pattern, we can gain a deeper understanding of gene expression. Also because of sliding chain mismatch

(SSM), bacteria will use this stochastic strategy to adjust their genetic pool in response to selective environmental pressure. Therefore, SSR-mediated mutation is of great significance to the pathogenesis and evolutionary adaptability of bacteria. At the same time, we also found that the clustering model relies heavily on distance metrics. Different distance metrics produce different clustering results. At the same time, we have different interpretations for the results of different clusters. This interpretation needs to be combined with specific data meanings. We believe that In the future, we will continue to deepen the analysis of SSM and SSRs, kmeans is still very applicable, providing us with tools to explore the laws of data.

Bibliography

[1]Catanese, H. N., Brayton, K. A., & Gebremedhin, A. H. (2016, June 3). *RepeatAnalyzer: A tool for analysing and managing short-sequence repeat data - BMC Genomics*. BioMed Central. Retrieved December 12, 2021, from https://bmcgenomics.biomedcentral.com/articles/10.1186/s12864-016-2686-2.

[2]A;, van B. (n.d.). *The role of short sequence repeats in epidemiologic typing*. Current opinion in microbiology. Retrieved December 12, 2021, from https://pubmed.ncbi.nlm.nih.gov/10383858/.

[3] Data Tricks, (3 July 2019). *One-hot encoding in R: three simple methods,* from https://datatricks.co.uk/one-hot-encoding-in-r-three-simple-methods

The link for the github: https://github.com/luuis1234567/475-final-project.git