Report
Data mining – Assignment 2

Group members

Pranav Kumar Asthana : 2015A7PS0961H
Uttara Ravi : 2015A7PS0032H
Sahil Sangwan : 2015A7PS0965H
Utkarsh Grover : 2017A7PS1428H

Dataset used

Amino acid sequences:
http://genome.crg.es/datasets/ggalhsapgenes2005/hg16.311.putative.aa.fa

Preprocessing done

The data was read in the FASTA format (.fa). Since the data is small and easily fits on memory, we can load the entire data at once.
Once loaded, we computed the distances between each pair of data points. This is a time-intensive step since the number of comparisons that need to be made is quadratic in the number of data points. Moreover, the computation of each distance is of the order of the product of the lengths of the two individual data points.
Total time complexity = $O(N^2.m^2)$ where N is the number of data points, and m is the average length of each sample.
Since this step is time-intensive and common to all the further steps, it can be performed as part of preprocessing of the data.

Formulae used

1. Distance between clusters:

MIN: $dist(cluster1, cluster2) = min(\{dist(p1, p2) \forall\ p1 \in cluster1, p2 \in cluster2\})$

MAX: $dist(cluster1, cluster2) = max(\{dist(p1, p2) \forall\ p1 \in cluster1, p2 \in cluster2\})$

AVG: $dist(cluster1, cluster2) = \frac{\sum(\{dist(p1, p2) \forall\ p1 \in cluster1, p2 \in cluster2\})}{|cluster1|.|cluster2|}$

2. Average dissimilarity of a point in a cluster:

$$dissimilarity(point, cluster) = \frac{\sum(\{dist(point, p) \forall\ p \in cluster\})}{|cluster|}$$

3. Diameter of a cluster

$$diameter(cluster) = max(\{dist(p1, p2) \forall\ p1, p2 \in cluster\})$$

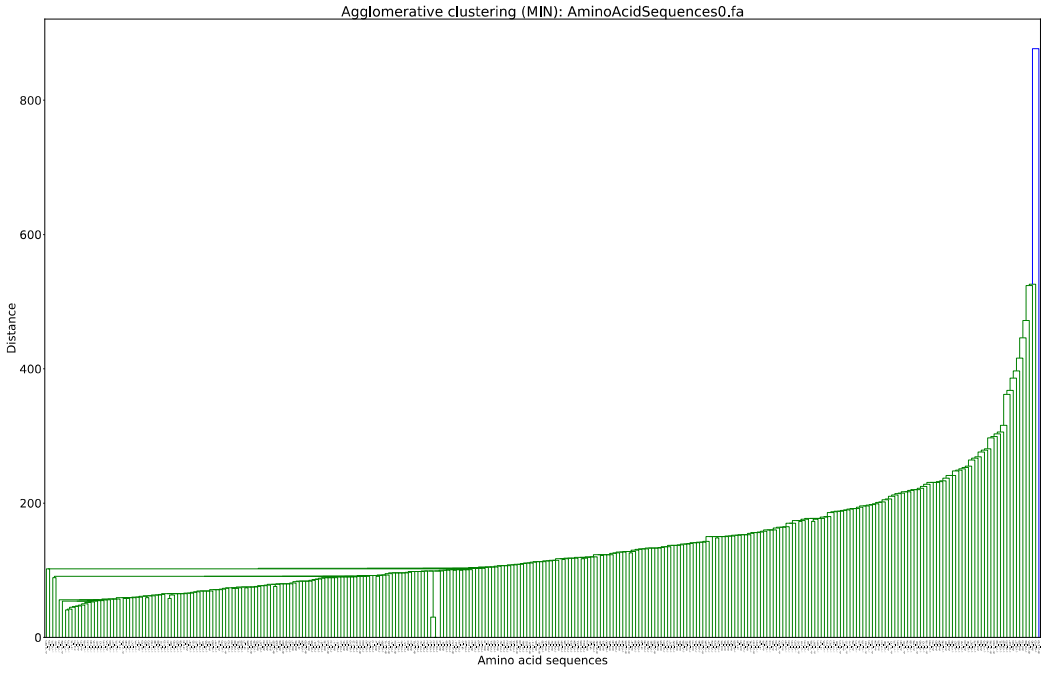
Linkage and distance metric used

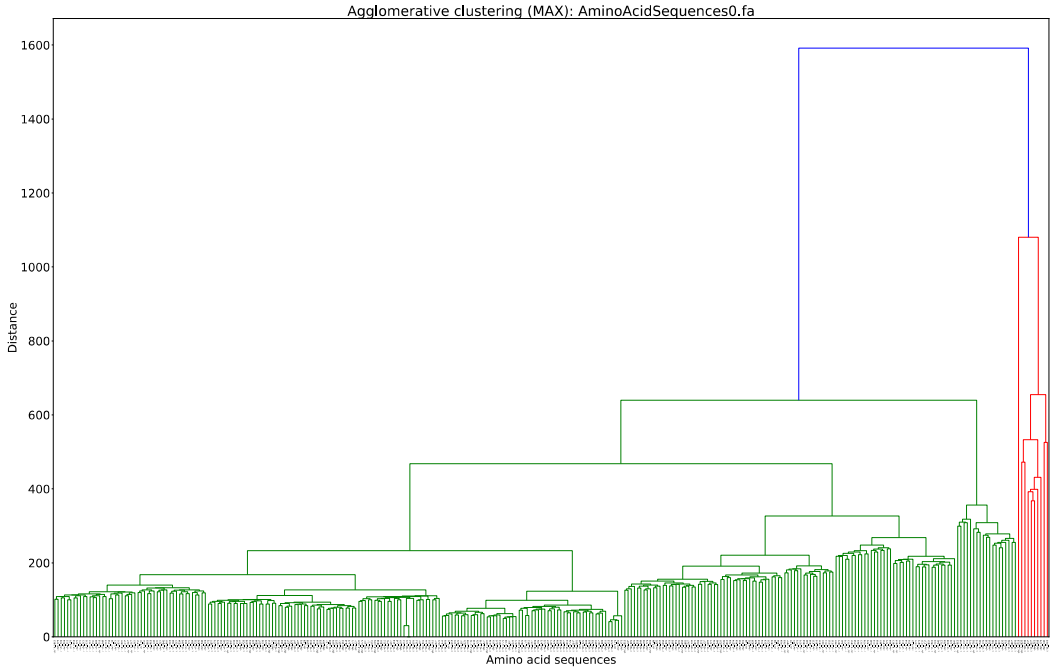
1. Linkage metric: We have used 3 linkage metrics for agglomerative clustering: Single (MIN), complete (MAX), average (AVG). For computing inter-cluster distance in divisive clustering, we have used the average metric.
2. Distance metric: For computing distance between two points(string), we have used a weighted edit distance with different weights for *match*, *substitution*, and *insert/delete.*

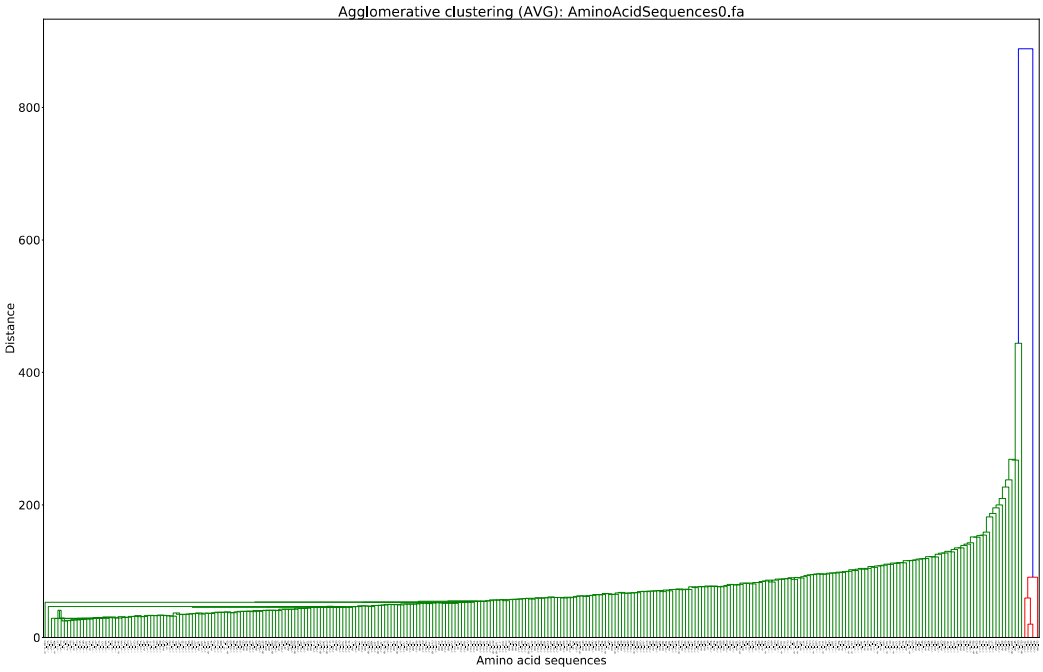
Type of data that can be clustered

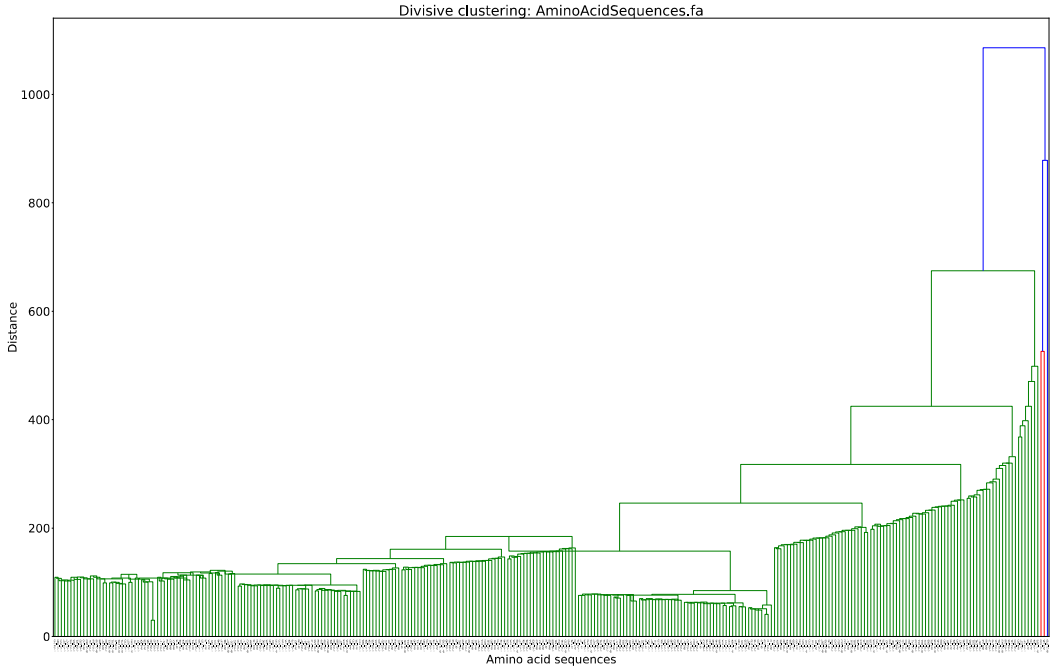
We can cluster any kind of short text data where the sequence of letters is of importance, such as gene sequences, amino acid sequences, words, names, etc.

Dendrograms






Agglomerative clustering (AVG): AminoAcidSequences0.fa
Distance
0
200
400
600
800
Amino acid sequences


Divisive clustering: AminoAcidSequences.fa
Distance
0
200
400
600
800
1000
Amino acid sequences

Comparison of k-means and hierarchichal clustering

Runtime for k-means:

```
$ time python kmeans.py 2
Reached end in 2 iterations
Final clusters
Cluster 0: 262 members
Cluster 1: 49 members
real    0m0.516s
user    0m0.445s
sys     0m0.543s

$ time python kmeans.py 5
Reached end in 3 iterations
Final clusters
Cluster 0: 74 members
Cluster 1: 1 members
Cluster 2: 55 members
Cluster 3: 130 members
Cluster 4: 51 members
```

Runtime for agglomerative:

```
$ time python agglomerative.py MIN
Merging 311 initial clusters
Clustering complete
[00:16<00:00, 19.35it/s]
Generating and saving dendrogram
real    0m16.285s
user    0m16.128s
sys     0m1.721s

$ time python agglomerative.py MAX
Merging 311 initial clusters
Clustering complete
Generating and saving dendrogram
[00:04<00:00, 70.30it/s]
real    0m4.634s
user    0m5.132s
sys     0m1.636s

$ time python agglomerative.py AVG
Merging 311 initial clusters
Clustering complete
Generating and saving dendrogram
[00:20<00:00, 15.47it/s]
real    0m20.322s
user    0m20.455s
sys     0m1.663s
```

Runtime for divisive:

```
$ time python divisive.py
Split initial cluster into 310 clusters
Clustering complete
Generating and saving dendrogram
310it [00:05, 54.84it/s]
real    0m6.053s
user    0m6.391s
sys     0m1.763s
```