Jacobs University Bremen

Sparse Subspace Clustering for Dimension Reduction of Mutation Matrix

Semester Project III

By: Dawit Nigatu
Supervisor: Prof. Dr. Werner Henkel

Transmission Systems Group (TrSyS)
School of Engineering and Science

January 2014

Introduction

In the previous project, we used classical multidimensional scaling (CMD) to scale down the 64×64 ECM mutation matrix and 20×20 amino acid chemical distance matrix to 2 dimensions (2-D). From the 2-D representations, we were able to see similar clusterings, which gives a meaning that highly probable mutations are between codons of similar chemical properties, at least in terms of polarity, chemical composition, and molecular volume. However, we have also observed some inconsistencies and become suspicious that it may be from the dimension reduction method we employed. Hence, in this project, we liked to implement sparse subspace clustering (SCC) proposed by Elhamifar and Vidal [1]. For a detailed and comprehensive explanation of subspace clustering, we refer the reader to [2].

First, how the SCC algorithm works is presented, followed by the description of spectral clustering, which is used in the SCC algorithm itself. Then, we will adopt the algorithm to our objectives and present the result.

Sparse subspace clustering

The algorithm is based on the sparse representation of the data. Usually, a high-dimensional data resides in multiple low-dimensional subspaces. Even if the representation might not be unique, the core idea behind the algorithm is to represent every data point as a linear combination of other points from its own subspace. The algorithm consists of two steps. First, a sparse optimization program is implemented which finds the membership of data points in the underling subspaces. Then, the spectral clustering technique is used for clustering of the data.

Let the data points that lie in the union of the n subspaces be represented by $\{\mathbf{y_i}\}_{i=1}^{N}$ and denote the matrix containing all the data points as $\mathbf{Y} = [\mathbf{y_1}|\dots|\mathbf{y_N}]$. Each data point in the union of subspaces can be written as

$$\mathbf{y_i} = \mathbf{Y}\mathbf{c_i},\ c_{ii} = 0\ , \tag{1}$$

where $\mathbf{c_i} \triangleq [c_{i1}\ c_{i2}\ \dots c_{iN}]^T$ and the constraint $c_{ii} = 0$ eliminates the trivial solution of writing a point as a linear combination of itself.

Among the solutions, there exists a sparse solution whose non-zero entries correspond to data points from the same subspace as $\mathbf{y_i}$ and it is called a subspace-sparse representation. Finding the subspace-sparse representation can be formulated as an optimization

problem in matrix form as

$$\min ||\mathbf{C}||_1 \quad \text{s.t.} \quad \mathbf{Y} = \mathbf{YC},\ \text{diag}(\mathbf{C}) = 0\ , \tag{2}$$

where $\mathbf{C} \triangleq [\mathbf{c_1}\ \ \mathbf{c_2}\ \ \ldots \mathbf{c_N}]$ is a matrix whose i^{th} column corresponds to the sparse representation of $\mathbf{y_i}$. The solution of Eq. (2) ideally represents a sparse representation whose non-zero elements correspond to points from the same subspace.

After obtaining the sparse representation, a symmetric similarity matrix $\mathbf{W} = |\mathbf{C}| + |\mathbf{C}|^T$ is formed, representing the weights of the edges of a graph. Ideally, the graph will have nodes connected only if they are from the same subspace. Using this similarity matrix clustering is obtained by applying spectral clustering. We will briefly describe how spectral clustering works in the next section.

Spectral Clustering

We will only give a brief introduction of spectral clustering. A more detailed tutorial can be obtained from [3].

Spectral clustering takes a similarity (adjacency) matrix $\mathbf{W} = (w_{ij})\ i, j = 1, ..., N$ as an input, uses graph Laplacian matrices, performs dimension reduction, and clusters the data points. There are different types of graph Laplacian matrices.

The unnormalized graph Laplacian is defined as

$$\mathbf{L} = \mathbf{D} - \mathbf{W}\ , \tag{3}$$

where $\mathbf{D}$ is a degree matrix which is a diagonal matrix with the degrees $d_i = \sum_{j=1}^{N} w_{ij}$.

The normalized graph Laplacians are defined as

$$\mathbf{L_{sym}} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{W}\mathbf{D}^{-\frac{1}{2}}\ , \tag{4}$$

$$\mathbf{L_{rw}} = \mathbf{I} - \mathbf{D}^{-1}\mathbf{W}\ . \tag{5}$$

The clustering or dimension reduction to k groups or dimensions is done by taking the first k eigenvectors (assuming the eigenvalues are sorted in increasing order) of the graph Laplacian matrix. For the clustering problem a k-means algorithm is applied on the rows of the matrix containing the eigenvectors as its columns. In case of the dimension reduction, the first k eigenvectors will represent the data in a reduced dimension.

Dimension reduction of the mutation matrix using SSC

The SSC algorithm needs the data to be first contained in the $\mathbf{Y}$ matrix with columns containing N data points. However, the ECM mutation matrix is a transition probability matrix with entries in a column representing the transition probabilities from one codon to every other codon. To adjust the matrix in accordance with the requirement of the algorithm, we have used an intermediate result from the classical multidimensional scaling (CMD) that we applied in the previous project. We took the whole set of eigenvectors instead of the two as a representation in 61 dimensions.

The SSC algorithm was basically addressing the data segmentation (clustering) problem, meaning that the output tells us which of the data points belongs to which group. Therefore, we need to deviate at the final stages of the algorithm, where it uses the eigenvectors of the graph Laplacian to apply the k-means algorithm. Instead, we use the first two eigenvectors as the 2-D representation.

The result is presented in Fig. 1. As can be seen, the result does not seem to show a clear difference and clustering. Many codons exist at overlapping positions, which makes it even worst than the result we obtained with CMD.

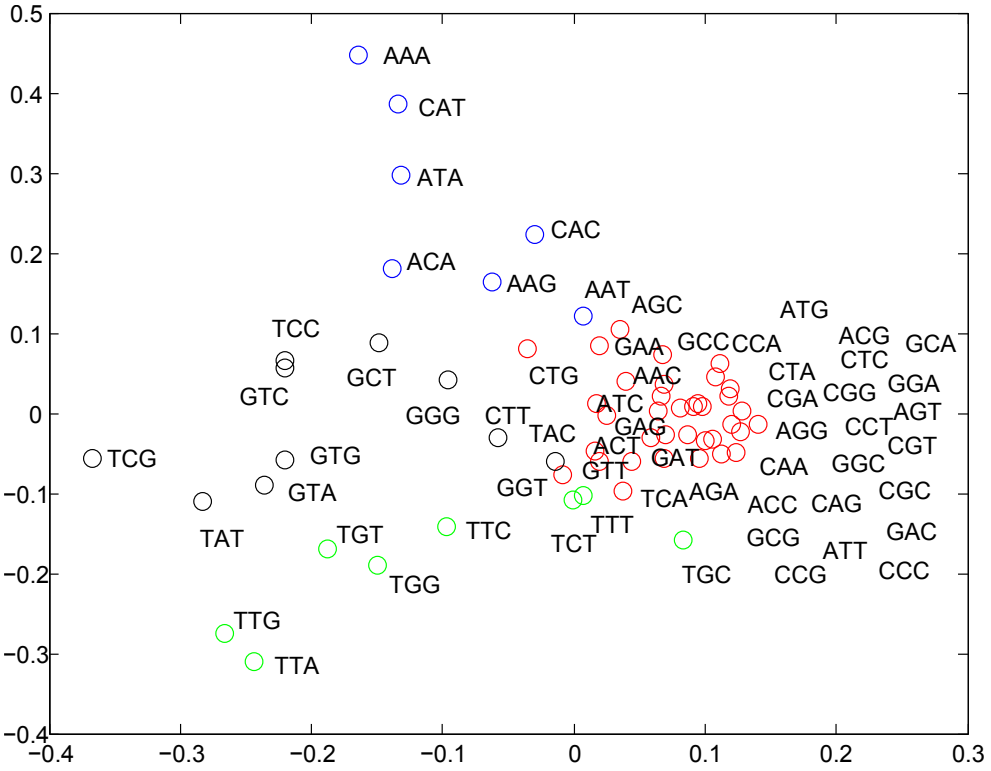

FIGURE 1: 2-D representation using sparse subspace clustering algorithm.

Conclusion

The obtained result is not what we expected. Almost half of the codons are located extremely close to each other, making it difficult for observing any clear relation. The CMD result was much better in terms of showing the similarity and differences of codon distances.

Additional Work

We have also corrected and updated a paper submitted to BioMed Central (BMC) according to the reviewers comments and suggestions. The title of the paper is "**The Empirical Codon Mutation Matrix as a Communication Channel**". The manuscript can be found in a separate file.

Bibliography

[1] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, 2009, pp. 2790–2797.

[2] R. Vidal, "Subspace clustering," *Signal Processing Magazine, IEEE*, vol. 28, no. 2, pp. 52–68, 2011.

[3] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.