Motif detection in biological network

Group 6 - Ankit Sharma, Akshita Sawhney and Sana Akhter

17 November, 2017

1 Introduction[3]

Scrutinizing biological systems allow us to study the system in a topology based manner. This also sheds light on the evolutionary relationships between various components of the system. Modelling biological systems into a graph gives rise to a huge class of networks termed as 'Biological Networks'.
According to the theories of evolution, conserved sequences reoccur as network motifs or sub-graphs. This conservation of certain structures is an evolutionary mechanism adopted to achieve a specific functionality important for the survival of an organism in the changing environment. Detection of these motifs in a given network is therefore a vital task and it can also be used as a critical feature if we wish to train a classifier that deals with network identification. Sampling sub-graphs poses a problem because it is unable to perform exhaustive search for lower number of random iterations, the run time of algorithms involved in detection of motifs with higher number of nodes is usually very high for enumeration based strategies. The fact that adds up to the difficulty is that there does not exist any polynomial-time algorithm that could validate topological equivalence. The intention of the project comprehensively explores the two types of algorithms based on sampling methods and enumeration methods.

2 Methodology[1] [2]

Three algorithms were implemented in this project. Their explanation is as follows:

2.1 Brute force algorithm

Brute force is mainly implemented to set a standard metric for the other two algorithms. It uses a method of exhaustive search, systematically enumerating all the possible motifs of size 4 and categorizing them to their respective configurations. Being a fairly straight forward approach to find the solution of the problem, it gives an exact frequency for all the possible configurations but the only problem is that being an exponential time complexity problem it requires great amount of compute.

2.2 Enumeration algorithm - Kavosh

This algorithm is heavily dominated by the number of sub-graphs and its run time appears to be fairly higher than the sampling method but lesser than the brute force strategy. KAVSOH was implemented in Python which makes use of an efficient enumeration strategy by the name of 'Revolving door ordering'. All possible motifs of a given size are generated as patterns and then the entire graph is spanned for those patterns. Frequency was obtained for each and every pattern.

2.3 Sampling algorithm

This strategy is independent of the network size and the degree of the hubs and it allows the detection of the rarest motifs in both hub and non-hub sub graphs. The algorithm involves random selection of edges until the sub graph size equals the size of the motif. Once the size becomes equal, it is then added to the list of sub graphs. 1000 such graphs were sampled and added to the list. NetworkX library was used to check graph isomorphism.

For validation purpose, the size of the motif was set to 4, as the time complexity grew exponentially with the increase in the motif size. The possible 4 sized motif configurations are shown in Fig 1. Results were compiled and a histogram was plotted as shown in Fig. 2, Fig. 3, Fig. 4 and Fig.5.

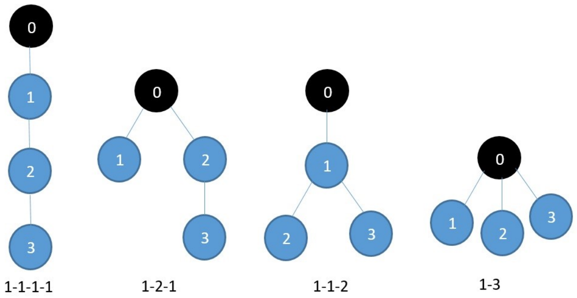

Figure 1: All configurations for motif size 4, which was used for validation

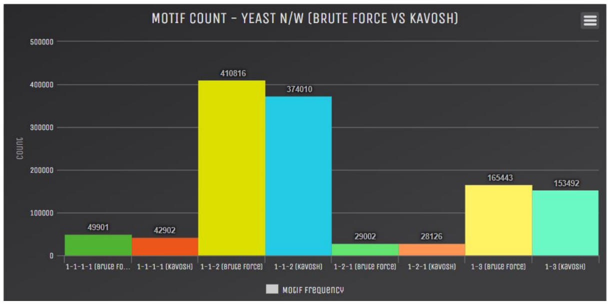

Figure 2: Motif count comparison between brute force and Kavosh for all 4 configurations-Yeast network.

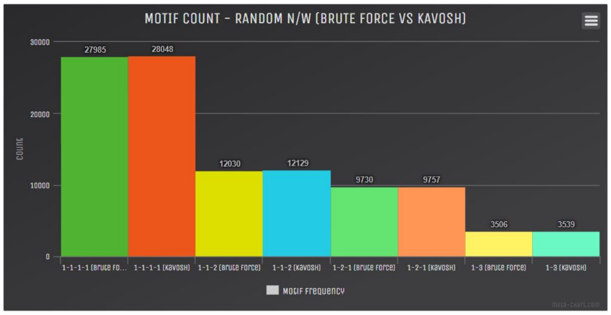

Figure 3: Motif count comparison between brute force and Kavosh for all 4 configurations-Random network.

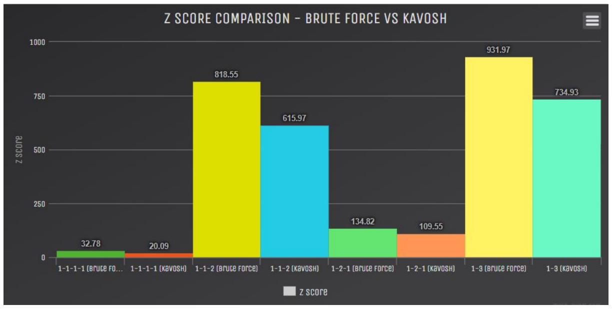

Figure 4: Z-Score comparison between brute force and Kavosh for all 4 configurations.

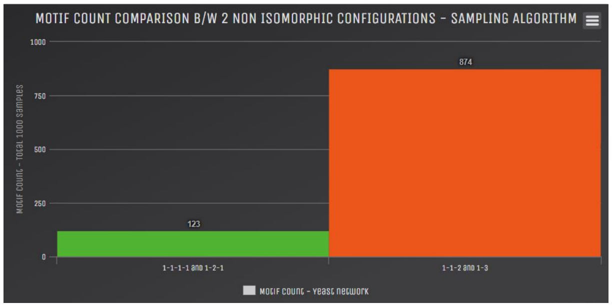

Figure 5: Relative count comparison between two distinct identifiable sets as per sampling strategy.

3 Inference

Using the histograms, we have tried to measure the spread of a set of data and how our algorithms have helped us infer the theories of evolution that we have known. By comparing Fig 2, Fig 3 and Fig 5 we can clearly observe that for all the configurations the number of motifs in biological networks is greater than in the random networks. This observation also helps us conclude our intent of assigning relative importance to the motifs in the biological networks. In Fig 4 we have used Z-scores to build a graphical representation of how the various configurations are distributed around the mean. A positive Z-score value indicates that the number of motifs are more significant in the biological networks. So overall from the graph it can be concluded that since all Z-scores are positive, all the configurations are comparatively more important in biological networks. Also we can see that the sum of z-score of config 1-1-2 and config 1-3 is more than the sum of config 1-1-1-1 and config 1-2-1 which again in turn helps conclude our intent to achieve a specific functionality important for the survival of an organism in the changing environment. Also the results of brute force versus the other two algorithms very evidently support our above observations.

4 Comparative Analysis

The implementation of Kavosh's enumeration strategy is done in four basic steps namely : class label identification, enumeration, generation of random graphs and motif identification. Each step has its own time complexity in terms of the network size and sub graph identification(as per motif size). Whereas, the sampling algorithm makes use of the randomized approach, and samples out sub graphs from the network. Thus, sampling of random sub graphs makes it highly time efficient with respect to Kavosh.

Elaborating more with respect to time complexity, as the motif size increases, the number of possible sub-graphs and thus; isomorphs generated increases.

The 'Revolving door' strategy makes the process of generation of all possible sub graphs efficient. The sampling strategy behaves in a contrasting manner with respect to Kavosh as the time is loosely dependent on the size of the motif, thus it does not require any optimization strategy like the 'Revolving door'. Brute force on the other hand does not work on any optimization heuristic and thereby is the most time consuming one.

Also, for networks with sub-graphs connected via hubs, the estimated runtime is supposed to be much smaller for sampling algorithm with respect to the enumeration strategy. If a graph lacks hubs; sampling works better. This is so because sampling method calculates relative frequencies while sampling sub graphs, whereas, enumeration computes the significance of all motifs extrapolated via frequency and *Z- score* calculation. With respect to accuracy, brute force is the most trusted one, followed by enumeration which is more accurate with respect to sampling.

Thus, the ordering is as follows:
Brute force > Enumeration > Sampling (Accuracy)
Sampling > Enumeration > Brute force (Time complexity)
The trade off between run time and and accuracy make the enumerative strategy our preferred candidate algorithm.

5 Problems faced and their solutions

Following problems were faced during the course of this project:
1. Implementing the revolving door strategy(enumeration algorithm) was challenging as it was not clearly specified in the paper. In order to implement that we had to go through its mathematics and a recursive method was written for the same.
2. Sampling algorithm involved a concept of maintaining the ordering for visited nodes and the incident edges over those visited nodes and this was handled using lists and strategic update of lists.
3. Also, while checking graph isomorphism in sampling, differentiating between 1-1-1-1 and 1-2-1 configuration along with 1-1-2 and 1-3 configuration was not possible due to the nature of the model sub graphs.

6 Contribution in the project post Milestone 1

Brute force strategy was implemented by Akshita, result generation and statistical analysis was taken care of by Ankit and Sana compared the three motif finding strategies analytically.

7 Code repository details

Private git repositories by the name of Brute Force, Kavosh and Sampling algorithm have been shared with ayalurarvind@gmail.com.

References

[1] Zahra Razaghi Moghadam Kashani, Hayedeh Ahrabian, Elahe Elahi, Abbas Nowzari-Dalini, Elnaz Saberi Ansari, Sahar Asadi, Shahin Mohammadi, Falk Schreiber, and Ali Masoudi-Nejad. Kavosh: a new algorithm for finding network motifs. *BMC Bioinformatics*, 10(1):318, Oct 2009.

[2] N. Kashtan, S. Itzkovitz, R. Milo, and U. Alon. Efficient sampling algorithm for estimating subgraph concentrations and detecting network motifs. *Bioinformatics*, 20(11):1746–1758, 2004.

[3] Elisabeth Wong, Brittany Baur, Saad Quader, and Chun-Hsi Huang. Biological network motif detection: principles and practice. *Briefings in Bioinformatics*, 13(2):202–215, 2012.