IITM-CS6024 : Algorithmic Approaches to Computational Biology Release Date: Oct 8

Assignment 2 Due Date : Oct 18, 23:55

Roll No: BS17B011 Name: G PRASHANT

- Use LaTeX to write-up your solutions, and submit the resulting single pdf file at GradeScope by the due date (Note: As before, **no late submissions** will be allowed, other than one-day late submission with 10% penalty! Within GradeScope, indicate the page number where your solution to each question starts, else we won't be able to grade it!).

- Collaboration is encouraged, but all write-ups must be done individually and independently, and mention your collaborator(s) if any. Same rules apply for codes uploaded to HackerRank (i.e., write your own code; we will run plagiarism checks on codes).

- If you have referred a book or any other online material for obtaining a solution, please cite the source. Again don't copy the source *as is* - you may use the source to understand the solution, but write-up the solution in your own words.

1. (6 points) [Biological significance] If a pattern of length k appears exactly in every sequence, a simple enumeration of all k-letter patterns that appear in the sequences gives the solution to finding the pattern that occurs frequently.

   (a) (2 points) Why is finding DNA motifs more complex than just matching all k-length patterns in the input DNA sequences?

   SOLUTION

   A naive approach to finding the DNA motifs would be to match all k-mer patterns in the input DNA sequences and find the k-mer occurring in all the sequences. However, there are inherent challenges with this approach, as stated below:

   - In reality, the binding specificity of transcription factors is tolerant to a few mutations in the DNA sequence motifs. Hence, the naive approach does not take into account the number of mismatches to be tolerated while finding the DNA motifs.

   - Even if this approach is modified to finding DNA motifs of length k with at most d mismatches, the algorithm will be extremely slow for large values of n (length of each DNA sequence), k and d, as the complexity of searching for all possible k-mers is $O(nkt)$, where t is the total number of sequences.

   (b) (1 point) State two biological reasons, apart from transcription factor (TF) protein binding to DNA, for why motifs occur in DNA sequences.

SOLUTION

Any short repeatedly occurring pattern in DNA that have a potential biological significance can be considered as DNA sequence motif. While transcription factor binding motifs serve as popular examples, there are other biological reasons for motifs to occur in DNA sequences, as given below:

- DNA sequence motifs can serve as binding sites for **nucleases**, which are proteins that cleave nucleic acids into smaller units.[4]
- DNA motifs in the exonic regions of the genome can sometimes be associated with encoding **structural motifs in proteins**.[3]

There are other DNA motifs that are involved in processes at the level of RNA, such as mRNA preprocessing, ribosome binding and transcription termination.

(c) (1 point) Do mutations in TF binding motifs always lead to changes in the expression of the downstream gene?

SOLUTION

In general, transcription factors are tolerant to a few mutations in the DNA sequence motifs. However, in some cases, mutations in TF binding motifs can result in increase or decrease in binding affinity of transcription factors, which causes modified expression levels of the downstream gene. Hence, such mutations do not always lead to changes in expression levels of the downstream gene.

(d) (2 points) Why are some DNA motifs the same across many species?

SOLUTION

Evolutionary conservation results in some DNA motifs being same, or conserved across many species. Important DNA motifs that are required for sustenance of life organisms are more likely to be conserved across species. For example, the ***glial cells missing*** **(gcm)** motif, is shown to be conserved in *Drosophila* and mammals including humans and mice.[1]

2. (6 points) [Solve or search!] Answer YES/NO to these questions on the Euclidean-distance-based clustering problems seen in class. Justify each answer very briefly (one/two sentences either with your own argument or by citing a properly-formatted reference paper/book). Each of the n datapoints is of dimension m, and we seek k clusters.

(a) (1 point) For all $m > 1$, is k-Means problem NP-hard for general k and polynomial-time solvable for $k = 1$?

SOLUTION

YES. Determining the centroid (optimal solution) in the case when $k = 1$ is easily solvable in polynomial time. When $k \geq 2$, the solution is dependent the initialization of the centroids, resulting in infinite number of possibilities. Hence, finding the global optimal solution is not efficiently solvable in polynomial time in a deterministic manner, making the problem NP-hard.

(b) (1 point) For all $m > 1$, is k-Centers problem NP-hard for general k and **not** polynomial-time solvable for $k = 1$?

SOLUTION

NO. Although k-Centers problem is NP-hard for a general value of k, finding the optimal center when $k = 1$ boils down to finding the smallest hypersphere enclosing all the data points, which is solvable in polynomial time.

(c) (1 point) For $m = 1$ (i.e., all data points lie on a line), is k-Means problem polynomial-time solvable for general k?

SOLUTION

YES. The problem of finding the optimal set of $k-$Means can be solved in polynomial-time provided $m = 1$, using a **Dynamic Programming** approach[8].

(d) (1 point) For $m = 1$, is k-Centers problem polynomial-time solvable for general k?

SOLUTION

YES. The problem of finding the optimal set of $k-$centers can also be solved in polynomial-time provided $m = 1$, using a **Dynamic Programming** approach, with a recurrence relation as follows:

$$D[i, 1] = (x_1 + x_i)/2$$

$$D[i + 1, k'] = \min(D[i, k' - 1], D[i + 1, k' - 1])$$

where $D[i, k']$ represents the cost of the sub-problem of clustering data points $x_1, x_2, \cdots, x_i$ into k' clusters.

(e) (1 point) For $m = 1$ and $k = 1$, does the optimal solution of the k-Centers problem always cost half the distance between the two farthest datapoints?

SOLUTION

YES. In the case when $m = 1$ and $k = 1$, the smallest enclosing circle will have the two farthest datapoints on the periphery, representing the diameter.

Hence, the optimal center will be the **midpoint** of the two farthest point.

(f) (1 point) The cost of a solution (SOLN) returned by the FarthestFirstTraversal heuristic seen in class is guaranteed to be at most twice the cost of an optimal solution (OPT) i.e., $SOLN \leq 2\,OPT$. Are there input instances (sets of datapoints) where the solution cost can be the worst value of **exactly** twice the optimal cost i.e., $SOLN = 2\,OPT$?

SOLUTION

YES. $SOLN = 2\,OPT$ will occur when any two centers determined by the heuristic are separated by a distance of $SOLN$ and are a part of the same cluster in the optimal solution, where the corresponding optimal center is the **midpoint** of the two centers.

Note that polynomial-time solvable means the running time is polynomial in n, k and m, and "general k" refers to a k that is not fixed to any value (such as 1) but instead provided as part of the input.

3. (11 points) [Random coding] You are about to implement a variant (Gibbs sampling) of one of the top 10 algorithms of the 20th century (Metropolis algorithm for Monte Carlo).

(a) (8 points) Solve the HackerRank Challenge:
`https://www.hackerrank.com/iitm-cs6024aacb-assignment-2` that asks you to apply Gibbs sampling technique to find motifs in a set of sequences. Please follow random seed, pseudocount, and other instructions carefully to produce the exact same output as the test cases.

SOLUTION

The required code written using Python was submitted on HackerRank. Although the code produced the expected result, it was not successful in passing all the test cases as it exceeded the time limit of 20 seconds. With 40 random starts and 2000 iterations, there are 40000 steps in total, and each step scales linearly with k and the length of the DNA sequence. The code takes 0.36 seconds for parameter values $\{k = 8,\ t = 5,\ N = 100\}$ and around 80 seconds for parameter values $\{k = 15,\ t = 20,\ N = 2000\}$. I have designed the code to be as efficient as possible to the best of my knowledge, and I am looking forward to learning new ways to further improve the efficiency to as good as 4 times faster than the current algorithm - which would identify motifs in under 20 seconds even for large input values. Another way to make the code run faster is to decrease the number of random starts ($rStart$). For example, by decreasing the number of random starts to 11, the code was successful with 7 out of the 11 test cases on HackerRank.

(b) (3 points) Can you find a better-scoring motif for “Sample Input 1” than the one that your current algorithm outputs? What parameters of your algorithm did you tune to achieve this better-scoring motif? Feel free to run your code outside of HackerRank to answer this subquestion.

SOLUTION

The score obtained upon finding the best-scoring motifs for Sample Input 1 with the default value of parameters ($N = 2000,\ rStart = 40,\ k = 15$) is **63**. Meanwhile, I achieved a lower (hence better) score of **42** when I implemented the algorithm with parameters $\{N = 3000,\ rStart = 50,\ k = 10\}$. The improvement of the score is not only attributed to the higher number of iterations and random starts, but also to the lower value of k. When k is lower, the probability that each sequence's motif being similar to the Consensus sequence is much higher than when k is higher.

4. (11 points) p53 is a transcription factor that suppresses tumor growth through regulation of dozens of target genes with diverse biological functions. This master regulator is inactivated in nearly all tumors. Let's try to identify the DNA-binding motif onto which p53 transcription factor binds using a motif-finding tool called MEME (available via Galaxy or stand-alone).

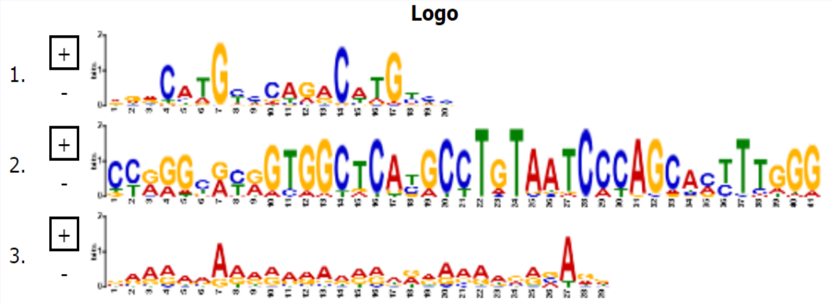
(a) (7 points) Run MEME on this input fasta file to get 3 candidate motifs. Of these three, which one do you think actually binds to p53 and why? The input fasta file contains a sample of ∼200-500bp sequences bound by p53 in a ChIP-seq experiment[1]. MEME may also have output two other motifs - argue by giving specific reasons whether those motifs could be true binding elements of p53?

SOLUTION

The input file `p53.fa` that contains 721 DNA sequences was given as input to the ”Multiple EM for Motif Elicitation” (MEME) tool in order to find 3 potential candidate motifs that the transcription factor p53 is likely to bind to. The program was run with default arguments (options), allowing zero or one motifs per sequence. The statistical factors of the 3 candidate motifs that were generated by the tool and the corresponding motif logos are tabulated below:

Consensus Sequence	**E-value**	**Sites**	**Width**
AGACATGCCCAGACATGCCC	9.1e-185	528	20
CCGGGCGCGGTGGCTCATGCCTGTAATCCCAGCACTTTGGG	9.9e-067	19	41
GAAAAAAAAAAAAAAAAGAAAAAGAGG	1.1e-052	288	29

[1] Our initial search for motifs in the 2000-bp promoter sequences of known target genes of the p53 didn't yield expected results, so we resorted to ∼200-500bp p53-bound sequences obtained from the peaks of a genome-wide p53 ChIP-seq experiment described at the p53 BAER resource



Here, the E-value is defined as the the expected number of sequences in a random database of the same size that would match the motifs as well as the sequence does and is equal to the combined p-value of the sequence times the number of sequences in the database (definition as stated in the documentation of MEME). It can be noticed that the E-value is the lowest for the first motif. Moreover, the first motif has the highest number of matches (528). Hence, **the first motif (AGACATGCCCAGACATGCCC) is the most likely to bind to p53** as the occurrence of this motif is the most statistically significant with the highest number of matches.

In addition to the reasons stated above, it is very fascinating to note that several studies have determined that the length of the DNA sequence motifs that the transcription factor p53 binds to is approximately 20 base pairs [2][6], and thousands of binding sites have been mapped so far in the human genome. These binding sites are extensively studied in order to derive insights into the molecular mechanisms leading to the progression of various types of cancers.

(b) (4 points) Run your Gibbs Sampler code from last question on the sequences provided. Report the motif you found, and did it match the one output by MEME? Report the parameter values you tried and chose for motif width (k), number of iterations (N), and the number of random restarts.

SOLUTION

In order to perform a systematic comparison between the results of MEME and the Gibbs Sampler code, I tried different k values corresponding to the width of the motifs generated by MEME $(20, 41, 29)$. The number of random starts was chosen to be either of 20 or 40. The number of iterations N was chosen to be 2000 for all the cases. The results obtained are tabulated below:

Consensus Sequence	k	$(rStart, N, t)$	Score
TCTCCTGCTTTTTCTTTTCA	20	(20,2000,721)	7614
TCTCCTGCTTTTTCTTTTCA	20	(40,2000,721)	7614
TCCTTTCTTCTTTTTCTTTCTTCAGGAAATTTTTTACACTG	41	(20,2000,721)	17857
TCCTTTCTTCTTTTTCTTTCTTCAGGAAATTTTTTACACTG	41	(40,2000,721)	17857
TTCTTTTTCCTTTTCTTCAGAGATGAAAT	29	(20,2000,721)	11934
TTTTTTTTCTTCTCTTTCCAAATCTGGCA	29	(40,2000,721)	11896

It can be noticed that for all values of k, the Consensus motifs generated by the Gibbs Sampler code is very different from that generated by MEME. This difference is attributed to the difference in the mathematical and algorithmic approaches used by two methods. MEME uses a **Mixture-Model Expectation Maximization** approach, allowing at most 1 motif per sequence ('zero or one motifs per sequence'), while the Gibbs Sampler code uses a **Markov Chain Montè Carlo** based approach, making sure that exactly one motif is taken into consideration from each of the 721 sequences.

5. (6 points) [Research exploratorium]: Provide properly-formatted references for papers in this solution.

   (a) (3 points) Read this paper on how to read a paper https://web.stanford.edu/class/ee384m/Handouts/HowtoReadPaper.pdf. What new tip/trick did you learn from this paper that you didn't already know before?

SOLUTION

Surprisingly, I have come across the same article earlier when I was doing the 'Current Topics in Synthetic Biology' (BT4310) Course during my 5^{th} semester. I wish to answer this question by explaining my learning outcome obtained through this paper after I had first read it.

- Prior to reading this article, I found it challenging to read research papers and get a concise understanding of them. I usually attempt to read the whole paper at once, but often get intimidated by the high usage of technical terms.
- This resulted in a lot of ambiguity despite spending significant amount of time in reading research papers.
- To be specific, learning about the First Pass Approach completely changed my perspective on how to quickly read through multiple research papers in the process of doing a literature survey.
- Learning about the Second and Third Pass approaches helped me to extract finer details like proofs, methods, diagrams and experimental techniques from research papers.

- Altogether, this article helped me enhance my performance to a great extent in the BT4310 course - the objective of which is to expose students to a variety of research avenues related to Synthetic Biology and enable them to make good research presentations.

(b) (3 points) What are some latest research publications you could find on *de novo* motif finding? Report one such paper based on Gibbs-sampling-like heuristics and another based on machine learning approaches such as clustering, SVM or deep learning. Try to answer this question by looking only at the papers' title/abstract.

SOLUTION

One of the recent and interesting studies on *de novo* motif identification that I came across is **Motif identification method based on Gibbs sampling and genetic algorithm**[7]. This research paper was published in 2017 in the Cluster Computing Journal. This method constructs a position-weight matrix based on Gibbs Sampling, and uses a genetic algorithm based approach for motif recognition.

The other research paper that I happened to read while looking for novel approaches for motif identification is **DeepFinder: An integration of feature-based and deep learning approach for DNA motif discovery**[5], which was published in 2018 in the Biotechnology & Biotechnological Equipment Journal (Taylor & Francis). The authors of this paper propose a three-staged approach for motif prediction, that employs neural networks and deep learning to construct a motif model, where the features are associated with the binding sites.

References

[1] Y. Akiyama, T. Hosoya, A. M. Poole, and Y. Hotta. The gcm-motif: a novel dna-binding motif conserved in drosophila and mammals. *Proceedings of the National Academy of Sciences of the United States of America*, 93(25):14912–14916, Dec 1996. 8962155[pmid].

[2] Feifei Bao, Peter R. LoVerso, Jeffrey N. Fisk, Victor B. Zhurkin, and Feng Cui. p53 binding sites in normal and cancer cells are characterized by distinct chromatin context. *Cell cycle (Georgetown, Tex.)*, 16(21):2073–2085, 2017. 28820292[pmid].

[3] Modan K. Das and Ho-Kwok Dai. A survey of dna motif finding algorithms. *BMC bioinformatics*, 8 Suppl 7(Suppl 7):S21–S21, Nov 2007. 18047721[pmid].

[4] Patrik D'haeseleer. What are dna sequence motifs? *Nature biotechnology*, 24(4):423–425, 2006.

[5] Nung Kion Lee, Farah Liyana Azizan, Yu Shiong Wong, and Norshafarina Omar. Deepfinder: An integration of feature-based and deep learning approach for dna motif discovery. *Biotechnology & Biotechnological Equipment*, 32(3):759–768, 2018.

[6] Ji-Hyun Lim, Natasha S. Latysheva, Richard D. Iggo, and Daniel Barker. Cluster analysis of p53 binding site sequences reveals subsets with different functions. *Cancer informatics*, 15:199–209, Oct 2016. 27812278[pmid].

[7] Xiaochun Sheng and Kefeng Wang. Motif identification method based on gibbs sampling and genetic algorithm. *Cluster Computing*, 20(1):33–41, Mar 2017.

[8] Haizhou Wang and Mingzhou Song. Ckmeans.1d.dp: Optimal k-means clustering in one dimension by dynamic programming. *The R journal*, 3(2):29–33, Dec 2011. 27942416[pmid].