Bees? DNA Motif Discovery with Alternating Global-Local Search

- CSC 530: Group 2 Project Proposal -

Grant Billings and Karthik Sanka

09/29/2022

Abbreviations

(l, d): a planted motif of length l with d random changes; **DNA**: Deoxyribonucleic Acid; **HMC**: Hamiltonian Monte Carlo; **MEME**: Multiple Expectation Maximization for Motif Elicitation; **PSO**: Particle Swarm Optimization;

1 Executive Summary

Living organisms have genomes that evolve randomly over time, with natural selection working to increase the frequency of functionally beneficial sequences over generations. Motifs are non-random nucleotide sequences that in many cases have been shown to have biological function in gene regulation. Detection of motifs in sets of sequences is challenging because random mutations make exact matching of sequences ineffective, and brute force methods are very slow. The most popular software package for motif discovery is MEME, which works well but slows down significantly if many query sequences are provided. Motif discovery across large data sets has become an important step in genome analysis. *Software that can efficiently mine these sequences for motifs is needed.*

Nature-inspired algorithms have promise for DNA motif discovery since they broadly allow for efficient exploration of potential motifs while allowing good solutions to learn from each other. We propose use of Particle Swarm Optimization with Hamiltonian Monte Carlo (PSO-HMC) in alternating cycles of global and local search to quickly find motifs. Our algorithm will be tuned using implanted motifs in simulated data, tested on previously characterized benchmarking data, and finally applied to discover new sequence motifs in a cotton promoter sequence dataset. Sensitivity, specificity, and running time will be used to compare performance between our software and other widely used alternatives. At the end of the semester, we will present our findings in comparison to other available software in a poster. We will also create an animation showing the algorithm running and share it upload it to Wikipedia so others can gain a visual intuition for how PSO-HMC works. *Our work will contribute to the rapid characterization of large genomic datasets.*

2 Abstract

Biologists are interested in detecting motifs from DNA sequencing data because of their role in gene expression and chromatin architecture. The (l, d) planted motif problem is NP-complete, so heuristics are usually employed to find motifs. Non-probabilistic scoring functions for potential motifs and their positions in sequences are discrete, making the non-convex, non-smooth solution space very difficult to work with using traditional optimization techniques. Nature-inspired algorithms tend to excel in problems of this type due to the ability for the algorithm to exchange information on potential solutions. Here, we propose a novel method for motif discovery using 1) alternating rounds of Particle Swarm Optimization for efficient global exploration of the solution space; and 2) Hamiltonian Monte Carlo for detailed local search to avoid poor outcomes due to local optima. We will implement our algorithm in Julia, and benchmark on synthetic and real datasets. Key deliverables include a poster presentation, as well as release of a graphical representation

of the algorithm and code into the public domain. We hope the speed and quality of the predicted motifs will help researchers generate hypotheses for motif sequences that can then be functionally validated through wet lab experiments.

3 Prior Work

Example citation [1]

4 Project Description

4.1 Data

- Simulated (l,d) implanted motifs
- Actual annotated *cis* element plant dataset from Sabastian and Contreras-Moreira, Bioinformatics, Nov 2013
- Upstream sequence from fiber-specific genes in Ando et al, BMC Genomics, April 2021

4.2 The Algorithm

Algorithm 1 Motif Detection with PSO-HMC

for all motif lengths $k \in 5..15$ **do** ▷ repeat algorithm for each plausible motif length
  Initialize a set M of particle position vectors and velocities m containing p particles in $\mathbb{Z}^n$
  Initialize a dictionary for the 10 best motif starting positions M_{best} and their scores
  Initialize a vector V for storing the scoring distribution information near each m
  $i \leftarrow 1$
  while not converged or $i <$ iteration limit **do** ▷ search until all particles are very close
    for all particles $m_i \in 1..p$ **do** ▷ do local search near each particle
      Evaluate the current score with $Score(m_i)$ ▷ score is hamming dist. against consensus
      if $Score(m_i) > \min(M_{\text{best}})$ **then**
        Add the score and position $M_{\text{best}}[m_i] \leftarrow Score(m_i)$ ▷ store for update step and output
      end if
      Initialize a dictionary O for the q motif starting positions and their scores
      for all sampled particles $o_j \in 1..q$ **do** ▷ local search step
        Allow the particle to roll in the solution space near m_i
        Add the resulting motif starting positions and scores $O[o_j] \leftarrow Score(o_j)$
        if $Score(o_i) > \min(M_{\text{best}})$ **then**
          Add the score and position $M_{\text{best}}[o_i] \leftarrow Score(o_i)$ ▷ store for update step and output
        end if
      end for
      $V[i] \leftarrow O$ ▷ store local search results for update step
    end for
    for all particles $m_i \in 1..p$ **do** ▷ global search step to pull particles towards best particle
      Use $V[i]$ and argmax(M_{best}) to propose a new position and direction for m_i
    end for
    $i \leftarrow i + 1$
  end while
  return M_{best}
end for

Sample new particle positions mi+1 from HMC results, with extra probability in the direction of the "best" particle
Report the best z non-overlapping particle positions

4.3 Implementation

- Implementation in Julia
- Particle Swarm Optimization with *inertia* and *social attraction* parameters
- Hamiltonian Monte Carlo

4.4 Experiments

4.5 Evaluation and Statistics

4.6 Deliverables

- Poster
- Graphical demonstration of algorithm
- Public availability of code on GitHub

4.7 Anticipated Problems and Solutions

5 Timeline

References

[1] Iztok Fister Jr et al. "A brief review of nature-inspired algorithms for optimization". In: *arXiv preprint arXiv:1307.4186* (2013).