**DOMAIN FINDING ALGORITHM**

CSE 549 Computational Biology

Muhammad Ali Ejaz <110559131>

Supriya Pramod Deshpande <110369866>

Kumar Anupam <110614410>

Neha Sudhakar Ellur <110468264>

Under the guidance of: Dr. Robert Patro

1. **INTRODUCTION**

A chromosome found in nucleus of a cell. It is an organized structure of DNA and protein, a single piece of coiled DNA containing many genes, regulatory elements and other nucleotide sequences. Chromosomes also contain DNA-bound proteins, which serve to package the DNA and control its functions. Chromosomes vary widely between different organisms.

The structure of the chromosomes has a deep impact on the cell development and cycle. It defines the chromosomal territories, which are known to be arranged radially around the nucleus. It is known to directly correlate with the gene density and size. Through multiple different studies, numerous spatial interactions between neighboring chromosome territories have been found, which can help in studying the influence of chromosome conformation on the functioning of the cells.

1. **PROJECT DETAILS**
2. **To run the code**
3. To load the parameters:

library(HiCseg)

1. To load the data

data(matrix)

dim=dim(matrix)

n=dim[1]

image(1:n,1:n,matrix,xlab="",ylab="")

We form a symmetric matrix of size 200 x 200 with the change points on the x-axis

And the log likelihood on the y-axis.

1. To form the link between C and R

result = HiCseg\_linkC\_R(size\_mat, nb\_change\_max, distrib, mat\_data, model)

The arguments are defined as below:

* size\_mat: Size of the matrix
* nb\_change\_max: Maximum number of change points
* distrib: The distribution of the data.

The values can be:

* **G**: Gaussian Distribution
* **P**: Poisson Distribution
* **B**: Negative Binomial Distribution
* mat\_data: Data of the matrix
* model: Type of the model
  + **D** – Block Diagonal Matrix
  + **Dplus** – Extended block diagonal Matrix

1. Output

The command that forms the link between C and R internally calls the following C function:

tmp=.C("Function\_HiC\_R",as.integer(size\_mat),as.integer(nb\_change\_max),

as.character(distrib), as.double(as.vector(mat\_data)), t\_hat=as.integer(rep(0,nb\_change\_max)), J=as.double(rep(0.0,nb\_change\_max)), t\_est=as.integer(rep(0,K)),as.character(model))

where,

t\_hat: Contains the estimated change-points

J: Gives the values of the log-likelihood for different number of change-points up to some constants

t\_est\_mat: Gives the matrix of the estimated change-points for different possible number of change-points i.e when there is no change point, one change-point, two change-points and so on.

The corresponding function called is:

int Function\_HiC\_R(int \*size, int \*maximum\_no\_change\_points,

char \*\*distribution, double \*matrix, int \*out\_t\_hat,

double \*out\_log\_likelihood, int \*out\_est\_chng\_pt, char \* \*model\_type)

and the output is populated in :

out\_t\_hat

out\_log\_likelihood

out\_est\_chng\_pt

Example for running:

library(HiCseg)

data(matrix)

dim=dim(matrix)

n=dim[1]

image(1:n,1:n,matrix,xlab="",ylab="")

result = HiCseg\_linkC\_R(200,10, “B”, matrix, “D”)

1. **Major Data Structures Used**
2. Y – The input matrix
3. T – A temporary data structure to store the means which will be used to compute the domain and non-domain region matrices.
4. D – Matrix to compute the domain regions.
5. R – Matrix to compute the non-domain regions
6. delta – Matrix to calculate data for : )
7. exterior – Matrix to calculate data for : )
8. I - The matrix which computes the maximum over the sum of delta and exterior.

Given by:

))

1. **REFERENCES**
2. Two-dimensional segmentation for analyzing Hi-C data
   * + - Celine Levy-Leduc, M. Delattre, T. Mary-Huard and S. Robin
3. Multiple change-point estimation with a total variation penalty.
   * + - Z. Harchaoui and C. Levy-Leduc (2010).
4. Optimal detection of changepoints with a linear computational cost.

- R. Killick (2012)

1. Pruned dynamic programming for optimal multiple change-point detection.

G. Rigaill (2010)