Discovering and analyzing DNA sequence motifs The rGADEM package.

Arnaud Droit *and Raphael Gottardo †

April 14, 2011

A step-by-step guide in the analysis of DNA sequence motifs using the rGADEM package in R

Contents

*arnaud.droit@ircm.qc.ca
†raphael.gottardo@ircm.qc.ca

Part I
Licensing

rGADEM is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version. We ask you to cite the following paper if you use this software for publication.

> L. Leiping. GADEM: A Genetic Algorithm Guided Formation of Spaced Dyads Coupled with an EM Algorithm for Motif Discovery. J Comput Biology

Part II
Introduction

In our guide, we include examples of code that we hope will help you when using the rGADEM package. The examples are kept at the basic level for ease of understanding. Some of the options in the functions have been set by default. To learn more about the exact parameters and usage of each function, you may type `help(FUNCTION_NAME)` of the function of interest in R after the rGADEM package is loaded.
Genome-wide analyses of protein binding sites generate large amounts of data; a ChIP data-set might contain 10,000 sites. Unbiased motif discovery in such datasets is not generally feasible using current methods that employ probabilistic models. We propose an efficient method, rGADEM, which combines spaced dyads and an expectation-maximization (EM) algorithm. Candidate words (four to six nucleotides) for constructing spaced dyads are prioritized by their degree of overrepresentation in the input sequence data. Spaced dyads are converted into starting position weight matrices (PWMs). rGADEM then employs a genetic algorithm (GA), with an embedded EM algorithm to improve starting PWMs, to guide the evolution of a population of spaced dyads toward one whose entropy scores are more statistically significant. Spaced dyads whose entropy scores reach a pre-specified significance threshold are declared motifs.
To use rGADEM, the user provide a set of coordinate include in the BED file or set of sequences in the FASTA format. The BED file contains location on chromosome, start and end position on the chromosome. For each line on

the BED file, we convert coordinate on a FASTA sequence. rGADEM has been developed in order to facilitate the discover and analysis of transcriptor factors and it is designed to works with the identification of motif package : MotIV. Thus, you can use the object returns by the rGADEM package to use MotIV (see MotIV package in Bioconductor for more detail).

The C code in rGADEM makes use of Grand Central Dispatch on Mac OS X 10.6 (Apple Inc) and openMP (openMP.org), with no user configuration, which greatly facilitates parallel processing and improve computing time on multicore machines (i.e. most modern computers). This implementation can actually reduce the computing time by a factor of 10, from several hours to a few minutes (depending on the number of input sequences).

Part III
Step-by-step Guide

1 rGADEM package and Packages

To load the rGADEM package, you should use this commande :

```
> library(rGADEM)
```

In the case of your data provide from Homo Sapiens :

```
> library(BSgenome.Hsapiens.UCSC.hg18)
```

2 Loading in the data

The next step is to load data from BED files or FASTA format in the R environment. The sequences are stored in some of the basic containers defined in the `Biostrings` package (see Biostring's document for more information). So ,the data can be manipulated in a consistent and easy way.
The maximum number of sequences allowed is 44 000. But it is possible to change the default parameters by editing the defines.h file and recompiling it.
The data used in this example are available in : extdata folder. The path for the data are : /rGADEM/inst/extdata.

2.1 From a BED file

Each line on the BED file contain the location, start and end position on the chromosome.

```
> pwd <- ""
> path <- system.file("extdata/Test_100.bed", package = "rGADEM")
> BedFile <- paste(pwd, path, sep = "")
> BED <- read.table(BedFile, header = FALSE, sep = "\t")
> BED <- data.frame(chr = as.factor(BED[, 1]), start = as.numeric(BED[,
+     2]), end = as.numeric(BED[, 3]))
```

Once the data have been read, we can create the list of 'RangedData' object (from `IRanges` package) where each element of the list corresponds to a different chromosome.

```
> rgBED <- IRanges(start = BED[, 2], end = BED[, 3])
> Sequences <- RangedData(rgBED, space = BED[, 1])
```

2.2 From a FASTA file

In the case you have a FASTA file, you can load the data as follow :

```
> pwd <- ""
> path <- system.file("extdata/Test_100.fasta", package = "rGADEM")
> FastaFile <- paste(pwd, path, sep = "")
> Sequences <- read.DNAStringSet(FastaFile, "fasta")
```

3 rGADEM analysis

At this time, we are now ready to start rGADEM analysis. If you want more details about rGADEM parameters, you may type `help(gadem)` in R environment. In this example, we have defined two parameters for rGADEM :

- verbose =1 : Print immediate results on screen.
- genome = Hsapiens : specify the genome.

We also describe two important parameters :

- P-Value cutoff: The P-Value cutoff controls the number of binding site in a motif. By default, the P-value cutoff is : 0.0002

- E-Value cutoff: The E-Value cutoff controls the number of motifs to be identified. By default, the E-value cutoff is : 0.0

```
> gadem <- GADEM(Sequences, verbose = 1, genome = Hsapiens)

Retrieving sequences... Done.
*** Start C Programm ***
```

4 Seeded analysis

In a seeded analysis rGADEM does not generate the starting PWMs through spaced dyads and optimize them through a Genetic Algorithm. This makes seeded runs much faster than unseeded. The efficiency of seeded runs makes it practical, even for sequence sets consisting of thousands of ChIP-Seq peak cores, to assess several alternative seed PWMs when prior knowledge suggests that this may be advisable (for example, when several database motifs are plausible candidate seeds).

The main advantage of a seeded analysis over an unseeded analysis is its computational efficiency.We recommend a seeded analysis whenever a reasonable starting PWM is available. However, for `de novo` motif discovery , an unseeded analysis is necessary.

First step is to prepare a text file with your PWM. It could be a general database (JASPAR, Transfac©,...). Only STAT1 have been selected in our example but it is possible to select a list of PWMs.

```
> path <- system.file("extdata/jaspar2009.txt", package = "rGADEM")
> seededPwm <- readPWMfile(path)
> grep("STAT1", names(seededPwm))
> STAT1.PWM = seededPwm[103]
```

At this step, we have two choice :

Only seeded analysis :

```
> gadem <- GADEM(Sequences, verbose = 1, genome = Hsapiens, Spwm = STAT1.PWM,
+     fixSeeded = TRUE)
```

or seeded analysis following by unseeded analysis :

```
> gadem <- GADEM(Sequences, verbose = 1, genome = Hsapiens, Spwm = STAT1.PWM)
```

5 rGADEM output

At the end of analysis, gadem object have been created in your R current session. This object contain all of your data information about your analysis (sequence consensus, pwm, chromosome, pvalue...). In fact, gadem object is a list of object.

- align : This object contains the individual motifs identified but and the location (seqID and position) of the sites in the original sequence data.
- motif : This object contains contains PWM, motif consensus, motif length and all aligned sequences for a specific motif.
- parameters : This object contains contains parameters of rGADEM analysis.

For more details, please see the RD files for each object.

5.1 Acces to pwm

To view all PWM

```
> nOccurrences(gadem)

[1] 60
```

To view pwm for the motif 1 :

```
> nOccurrences(gadem)[1]

[1] 60
```

5.2 Acces to sequence consensus

To view all sequences consensus :

```
> consensus(gadem)

[1] "wrwGTmAACAs"
```

5.3 Acces to sequence consensus

To access to the first sequence :

```
> consensus(gadem)[1]

[1] "wrwGTmAACAs"
```

5.4 Acces to chromosome position

To view start position on chromosome :

```
> startPos(gadem)

$m1
[1] 117197889
```

5.5 Acces to chromosome position

To view end position on chromosome :

```
> endPos(gadem)

$m1
[1] 117198090
```

5.6 Acces to parameters

And finaly, if you want to show parameters for this analysis :

```
> gadem@parameters
```

6 Background model

6.1 Description

It is convenient to assume that the background model is independent and identically distributed, as the exact distribution of the log likelihood ratio (llr) score can be efficiently approximated using the probability generating

function method. However, when a higher order background model is used, analytic determination of the null distribution of the llr score is nontrivial, and rGADEM generates a large number of subsequences with the same length as the motif, then approximates the null distribution from their llr scores. It generates the subsequences using the [A,C,G,T] frequencies in the input sequences, which preserves the marginal nucleotide probabilities in both the input and background datasets.

When the background model is assumed to be independent between positions, it is computationally efficient to approximate the exact distribution of the llr score using the probability generating function (pgf) method of Staden (1989). However, for a higher-order Markov background model, obtaining the null distribution of the llr score analytically is not trivial, and we adopt an empirical approach. rGADEM first generates many null background subsequences of length w by simulating them from the 0th-order Markov model estimated by GADEM from the input data.

6.2 Example of background model

The background Markov model can be estimated from the input data by GADEM or read from a file using the -fbm argument. We recommend -bOrder 0 when -fbm is not used. Otherwise, a higher order (e.g., -bOrder 3 or 4) may be reasonable.

Up to 8th order (octamer + 1nt = nonamer) is allowed.

```
monomer frequency
a 0.20850000001660
c 0.29149999998340
g 0.29149999998340
t 0.20850000001660

dimer frequency
aa 0.04800960194357
ac 0.05151030207800
ag 0.08171634323790
at 0.02720544114470
....
ta 0.03460692142891
tc 0.05361072215865
```

tg 0.07231446287687
tt 0.04800960194357

trimer frequency
aaa 0.01200480194395
aac 0.01550620248175
aag 0.01420568228200
aat 0.00630252106809
....
tta 0.01190476192858
ttc 0.01180472191322
ttg 0.01230492199004
ttt 0.01200480194395

The GADEM package includes pre-computed genome-wide frequencies data for human, mouse and Drosophila, and source code for generating such files.