Enhancing motif finding models using multiple sources of genome-wide data

Heejung Shim[1], Oliver Bembom [2], Sündüz Keleş[3,4]

[1] Department of Human Genetics, University of Chicago,
[2] Division of Biostatistics, University of California, Berkeley,
[3] Department of Statistics,
[4] Department of Biostatistics and Medical Informatics,
University of Wisconsin.

April 21, 2011

1 Overview

The `SUCcESS` package implements the CTCM model (particularly logistic regression model) proposed by Shim and Keleş (2008) for integrating quantitative information into motif finding as well as its extension to use multiple data sources at a time (e.g., ChIP-seq, nucleosome occupancy, or conservation score). We implemented them as an extended module for `cosmo`, developed by Bembom *et al.* (2007), implementing an algorithm which supervises detection of motifs using a set of constraints for a position weight matrix. Note that although this package provides all functions implemented in `cosmo`, we ask that you instead use the latest version of `cosmo` for the supervised motif detection algorithm. This package is for running `SUCcESS` option only. This vignette just provides the instructions for how to run `SUCcESS`: you will need to consult Shim and Keleş (2008) for methodological details and the `cosmo` vignette (you can find it in the document directory of this package) for detailed options.

2 Brief Background

In addition to ChIP-seq (chromatin immunoprecipitation followed by sequencing), various sources of genomic data have been available to improve detection of transcription factor binding sites. For example, functional regions are prone to be conserved across related species (Kellis *et al.*, 2003), and thus conservation scores obtained by multi-species sequencing can provide information about which regions are more likely to have motifs. As another example, nucleosomes are know to interact with transcription factor bindings and less likely occupy in active regulatory regions (Segal *et al.*, 2006)(Lee *et al.*, 2004). Thus, nucleosome occupancy measured by either ChIP-seq experiments for histone modifications (Lee *et al.*, 2004) or MNase-seq (micrococcal nuclease digestion followed by sequencing) (Shivaswamy *et al.*, 2008)(Albert *et al.*, 2007) can be used as an additional genomic data. Any other data type which provides information on transcription factor binding sites can be integrated into our approach.

Our previous paper (Shim and Keleş, 2008) introduced the method for using ChIP-chip only. Here we extend one of the proposed three models to use those multiple data sources at a time. Let $T_{ik} = (T_{ik}^1, \ldots, T_{ik}^M)$ denote M genomic data for kth base pair in sequence i. Then, the logistic regression model in Shim and Keleş (2008) can be extended to

$$\log\left(\frac{Pr(Z_{ik}=1 \mid T_{ik})}{1-Pr(Z_{ik}=1 \mid T_{ik})}\right) \;=\; \beta_0 + \beta_1 T_{ik}^1 + \ldots + \beta_M T_{ik}^M, \qquad (1)$$

where Z_{ik} is the indicator variable denoting whether a motif starts at position k in sequence i and $\beta_j \geq 0$ for $j = 1, \ldots, M$. When multiple sources of genomic data are provided by users, `SUCcESS` will use the extended model.

3 Input Data

`SUCcESS` requires two types of input files: for (1) sequence data and (2) additional genomic data. The sequence data should be provided in FASTA format. Here is an example with two sequences.

```
>seq1
tcgtagcccaa
>seq2
tcgtagggcccgt
```

Two types of input formats can be used for the genomic data depending on the resolution of data. In both formats, each row contains information on each sequence, with one row per sequence. The sequence-level input format lists one number at each row as shown in an example below.

```
0.96
1.51
```

In the base pair-level input format, each row includes a list of numbers each of which corresponds to each base-pair in the sequence. The list should have the same length as the corresponding sequence. The following example corresponds to the sequence data in FASTA format above.

```
0.18 0.96 0.64 0.15 0.09 2.06 0 0 2.18 0.61 3.68
0.36 6.44 3.18 0.02 3.28 1.94 0.03 0.28 0.03 0.74 0.17 0.29 2.87
```

For neither sequence-level nor base pair-level genomic data (e.g., probe-level), the value of a measurement unit could be assigned to all bases within the unit or interpolation with simple averaging could be employed to obtain values at the bases between units if necessary. We assume that the genomic data at a particular position of a given sequence have a direct relationship with the probability that the position of the sequence is a motif start site. For the data who do not have a direct relationship with the motif start site probability (e.g., nucleosome occupancy), you will need to transform them. When multiple sources genomic data are available, each genomic data should be provided separately in an appropriate format.

4 run SUCcESS

This section will describe how to run `SUCcESS` using simulated data. The package can be loaded using the command

```
>library(success)
```

and SUCcESS can be run using for example

```
>seqFile = system.file("Exfiles/seqdata", package="success")
>genomicFile1 = system.file("Exfiles/genomicdata1", package="success")
>genomicFile2 = system.file("Exfiles/genomicdata2", package="success")
>resS = success(seqs=seqFile, ChIP=c(genomicFile1, genomicFile2),
models = c("TCM"), maxW = 10, minW= 9, starts = 2, numMotifs = 2)
```

Sequence data and multiple genomic data can be provided using the arguments `seqs` and `ChIP`, respectively. `SUCcESS` allows users to specify the model types to be considered among the three possible candidates (OOPS ZOOPS and TCM) using the argument `models` (`models = c("ZOOPS")` by default). The arguments `minW` and `maxW` can be used to provide the minimum and maximum motif widths to be considered (`minW = 6` and `maxW = 15` by default), respectively. `SUCcESS` sets the number of starting values to be two times of the number given by the argument `starts` (`starts = 5` by default). SUCcESS can detect multiple motifs whose number is given by `numMotifs` (1 by default) as `MEME` (Bailey and Elkan, 1995) does. Please consult the description of `cosmo` function in the `cosmo` vignette for the other arguments.

The `print` function shows the estimated position weight matrices

```
> print(resS)

Motif 1:
      1 2      3 4      5 6      7     8      9     10
A 0.857 1 0.1427 1 0.0000 0 0.0000 0.857 0.0000 0.2856
C 0.000 0 0.1429 0 0.8571 0 0.1428 0.143 0.7142 0.7144
G 0.000 0 0.5715 0 0.0000 0 0.0000 0.000 0.0000 0.0000
T 0.143 0 0.1429 0 0.1429 1 0.8572 0.000 0.2858 0.0000


Motif 2:
       1      2      3      4      5      6      7      8      9     10
A 0.2325 0.1858 0.3324 0.0000 0.4207 0.1865 0.0000 0.9036 0.0000 0.0000
C 0.3333 0.2752 0.0396 0.6472 0.2253 0.2394 0.2863 0.0000 0.5587 0.2741
G 0.4342 0.3326 0.6280 0.2948 0.1262 0.0000 0.0000 0.0964 0.3992 0.5599
T 0.0000 0.2064 0.0000 0.0580 0.2279 0.5741 0.7137 0.0000 0.0420 0.1660
```

and the estimated coefficients in the logistic regression model are listed in a more detailed summery of the results (slot `Estimated beta`).

```
> summary(resS)
Input dataset:
  Sequence Length
1     seq1    100
2     seq2    100
```

```
3     seq3    100
4     seq4    100
5     seq5    100


Candidate orders for background Markov model:
  order         klDiv
1     0  1.387680e+02
2     1  1.390599e+02
3     2 1.797693e+308
4     3           Inf
5     4           Inf
6     5           Inf
7     6           Inf

Motif 1:


Candidate models considered:
  conSet model width    wCrit modCrit conCrit
1      1   TCM     9 1353.436      NA      NA
2      1   TCM    10 1332.809      NA      NA


Selected model:
             choice  crit   critVal
Constraint        1 likCV        NA
Model           TCM   lik        NA
Width            10   bic 1332.8086
NumSites          8   lik -639.8486
Markov Order      0 likCV  138.7680


Estimated position weight matrix:
      1 2      3 4      5 6      7     8      9     10
A 0.857 1 0.1427 1 0.0000 0 0.0000 0.857 0.0000 0.2856
C 0.000 0 0.1429 0 0.8571 0 0.1428 0.143 0.7142 0.7144
G 0.000 0 0.5715 0 0.0000 0 0.0000 0.000 0.0000 0.0000
T 0.143 0 0.1429 0 0.1429 1 0.8572 0.000 0.2858 0.0000
```

```
Estimated beta:
         beta
1 -15.6601542
2   3.4721915
3   0.5722299

Motif occurrences:

E-value: 2301.455

   seq pos orient      motif         prob
1 seq3  25      1 AACACTTATC 1.000000e+00
2 seq3  35      1 TAGATTTCCA 1.000000e+00
3 seq3  45      1 AAGACTTATC 1.000000e+00
4 seq4  32      1 AAGACTTACC 9.999972e-01
5 seq2  23      1 AATACTTACC 9.999385e-01
6 seq2  38     -1 TGTAAGTCTT 9.998418e-01
7 seq4  43      1 AAAACTCACC 9.994337e-01
8 seq4  54      1 AAAATGCATC 1.628794e-19

Motif 2:


Candidate models considered:
  conSet model width    wCrit modCrit conCrit
1      1   TCM     9 1374.087      NA       NA
2      1   TCM    10 1367.573      NA       NA


Selected model:
             choice  crit   critVal
Constraint        1 likCV        NA
Model           TCM   lik        NA
Width            10   bic 1367.5731
NumSites          8   lik -657.2308
Markov Order      0 likCV  138.7680


Estimated position weight matrix:
```

```
       1      2      3      4      5      6      7      8      9     10
A 0.2325 0.1858 0.3324 0.0000 0.4207 0.1865 0.0000 0.9036 0.0000 0.0000
C 0.3333 0.2752 0.0396 0.6472 0.2253 0.2394 0.2863 0.0000 0.5587 0.2741
G 0.4342 0.3326 0.6280 0.2948 0.1262 0.0000 0.0000 0.0964 0.3992 0.5599
T 0.0000 0.2064 0.0000 0.0580 0.2279 0.5741 0.7137 0.0000 0.0420 0.1660


Estimated beta:
        beta
1 -3.2120516
2  0.1670445
3  0.4271306

Motif occurrences:

E-value: 5412862734

   seq pos orient      motif      prob
1 seq4  76      1 GGGCCTTACT 0.9999139
2 seq2   2     -1 CGTAGGGCCC 0.9991418
3 seq5  48     -1 CGTGAAGCGG 0.9974598
4 seq4  86     -1 CGTAATCCAC 0.9904049
5 seq5  32      1 CTGCCATAGC 0.9849302
6 seq2  86      1 CAGTATTACG 0.9834779
7 seq4  20      1 ATACCTTACG 0.9694659
8 seq4  65     -1 GGTGATGTGG 0.9633239
```

Sequence logo of the estimated motifs can be generated one at a time using the command:

```
> plot(resS, index = 1)
```

The argument `index` indicates which motif is used to make a plot. The sequence logo of the first motif (`index = 1`) is shown in Figure 1. The argument `type = "prob"` makes the `plot` function produce a plot of the posterior probabilities along each sequence. The plot in Figure 2 is created by the command:

```
> plot(resS, type = "prob", index = 1)
```

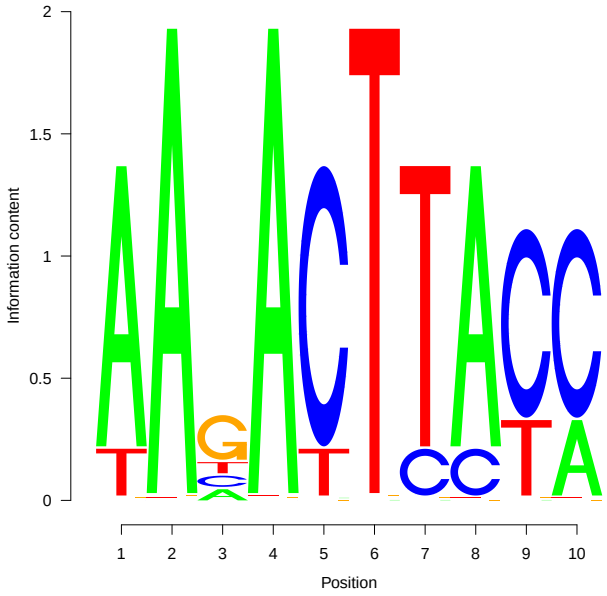

Figure 1: Sequence logo (motif 1)

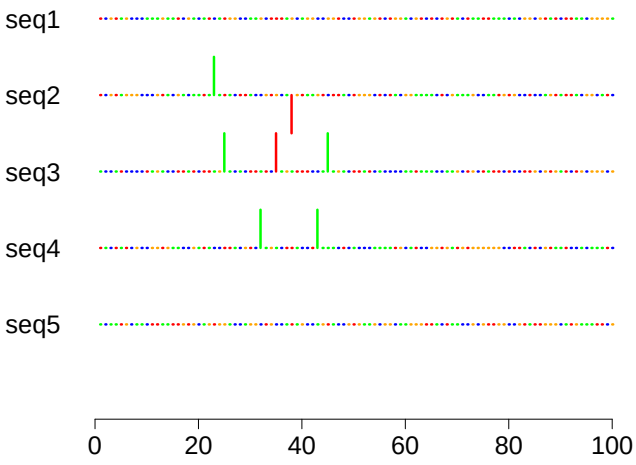

Figure 2: Posterior probabilities (motif 1)

References

Albert, I., Mavrich, T. N., Tomsho, L. P., Qi, J., Zanton, S. J., Schuster, S. C. and Pugh, B. F. (2007) Translational and rotational settings of h2a.z nucleosomes across the saccharomycescerevisiae genome. *Nature*, **446**, 572–576.

Bailey, T. and Elkan, C. (1995) Unsupervised learning of multiple motifs in biopolymers using expectation maximization. *Machine Learning*, **21**, 51–80.

Bembom, O., Keleş, S. and van der Laan, M. J. (2007) Supervised detection of conserved motifs in DNA sequences with `cosmo`. *Statistical Applications in Genetics and Molecular Biology*, **6**, Article 8. `http://www.bepress.com/sagmb/vol6/iss1/art8`.

Kellis, M., Patterson, N., Endrizzi, M., Birren, B. and Lander, E. S. (2003) Sequencing and comparison of yeast species to identify genes and regulatory elements. *Nature*, **423**, 241–254.

Lee, C.-K., Shibata, Y., Rao, B., Strahl, B. D. and Lieb, J. D. (2004) Evidence for nucleosome depletion at active regulatory regions genomewide. *Nature Genetics*, **36**, 900–905.

Segal, E., Fondufe-Mittendorf, Y., Chen, L., Thastrom, A., Field, Y., Moore, I. K., Wang, J. Z. and Widom, J. (2006) A genomic code for nucleosome positioning. *Nature*, **442**, 772–778.

Shim, H. and Keleş, S. (2008) Integrating quantitative information from ChIP-chip experiments into motif finding. *Biostatistics*, **9**, 51–65.

Shivaswamy, S., Bhinge, A., Zhao, Y., Jones, S., Hirst, M. and Iyer, V. R. (2008) Dynamic remodeling of individual nucleosomes across a eukaryotic genome in response to transcriptional perturbation. *PLoS Biology*, **6**, e65.