Finding Candidate Binding Sites for Known Transcription Factors via Sequence Matching

The binding of transcription factor proteins (TFs) to DNA promoter regions upstream of gene transcription start sites (TSSs) is one of the most important mechanisms by which gene expression, and thus many cellular processes, are controlled. Though in recent years many new kinds of data have become available for identifying transcription factor binding sites (TFBSs) -- ChIP-seq and DNase I hypersensitivity regions among them -- sequence matching continues to play an important role.

In this workflow we demonstrate Bioconductor techniques for finding candidate TF binding sites in DNA sequence using the model organism Saccharomyces cerevisiae. The methods demonstrated here apply equally well to other organisms.

Introduction

Eukaryotic gene regulation can be very complex. Transcription factor binding to promoter DNA sequences is a stochastic process, and imperfect matches can be sufficient for binding. Chromatin remodeling, methylation, histone modification, chromosome interaction, distal enhancers, and the cooperative binding of transcription co-factors all play an important role. We avoid most of this complexity in this demonstration workflow in order to examine transcription factor binding sites in a small set of seven broadly co-expressed Saccharomyces cerevisiae genes of related function. These genes exhibit highly correlated mRNA expression across 200 experimental conditions, and are annotated to Nitrogen Catabolite Repression (NCR), the means by which yeast cells switch between using rich and poor nitrogen sources.

We will see, however, that even this small collection of co-regulated genes of similar function exhibits considerable regulatory complexity, with (among other things) activators and repressors competing to bind to the same DNA promoter sequence. Our case study sheds some light on this complexity, and demonstrates how several new Bioconductor packages and methods allow us to
• Search and retrieve DNA-binding motifs from the MotifDb package
• Extract the DNA sequence of the promoter regions of genes of interest
• Locate motifs in the promoter sequence

Installation and Use

To install the necessary packages and all of their dependencies, evaluate the commands

```R
source("http://bioconductor.org/biocLite.R")
biocLite(c("MotifDb", "GenomicFeatures", "TxDb.Scerevisiae.UCSC.sacCer3.sgdGene",
          "org.Sc.sgd.db", "BSgenome.Scerevisiae.UCSC.sacCer3", "motifStack", "seqLogo"))
```

Package installation is required only once per R installation. When working with an organism other than S.cerevisiae, substitute the three species-specific packages as needed.

To use these packages in an R session, evaluate these commands:

```R
## See system.file("LICENSE", package="MotifDb") for use restrictions.

library(MotifDb)
library(seqLogo)
library(motifStack)
library(Biostrings)
library(GenomicFeatures)
library(org.Sc.sgd.db)
```
library(BSgenome.Scerevisiae.UCSC.sacCer3)
library(TxDB.Scerevisiae.UCSC.sacCer3.sgdGene)
These instructions are required once in each R session.

Biological Background

The x-y plot below displays expression levels of seven genes across 200 conditions, from a compendium of yeast expression data which accompanies *Allocco et al.*, 2004, "Quantifying the relationship between co-expression, co-regulation and gene function":
Allocco et al establish that

In *S. cerevisiae*, two genes have a 50% chance of having a common transcription factor binder if the correlation between their expression profiles is equal to 0.84.

These seven highly-correlated (> 0.85) NCR genes form a connected subnetwork within the complete co-expression network derived from the compendium data (work not shown). Network edges indicate correlated expression of the two connected genes across all 200 conditions. The edges are colored as a function of that correlation: red for perfect correlation, white indicating correlation of 0.85, and intermediate colors for intermediate values. DAL80 is rendered as an octagon to indicate its special status as a transcription factor. We presume, following Allocco, that such correlation among genes, including one transcription factor, is a plausible place to look for shared transcription factor binding sites.
Some insight into the co-regulation of these seven genes is obtained from Georis et al, 2009, "The Yeast GATA Factor Gat1 Occupies a Central Position in Nitrogen Catabolite Repression-Sensitive Gene Activation".
Saccharomyces cerevisiae cells are able to adapt their metabolism according to the quality of the nitrogen sources available in the environment. Nitrogen catabolite repression (NCR) restrains the yeast's capacity to use poor nitrogen sources when rich ones are available. NCR-sensitive expression is modulated by the synchronized action of four DNA-binding GATA factors. Although the first identified GATA factor, Gln3, was considered the major activator of NCR-sensitive gene expression, our work positions Gat1 as a key factor for the integrated control of NCR in yeast for the following reasons: (i) Gat1 appeared to be the limiting factor for NCR gene expression, (ii) GAT1 expression was regulated by the four GATA factors in response to nitrogen availability, (iii) the two negative GATA factors Dal80 and Gzf3 interfered with Gat1 binding to DNA, and (iv) Gln3 binding to some NCR promoters required Gat1. Our study also provides mechanistic insights into the mode of action of the two negative GATA factors. Gzf3 interfered with Gat1 by nuclear sequestration and by competition at its own promoter. Dal80-dependent repression of NCR-sensitive gene expression occurred at three possible levels: Dal80 represses GAT1 expression, it competes with Gat1 for binding, and it directly represses NCR gene transcription. (emphasis added)

Thus DAL80 is but one of four interacting transcription factors which all bind the GATA motif. We will see below that DAL80 lacks the GATA sequence in its own promoter, but that the motif is well-represented in the promoters of the other six.

In order to demonstrate Bioconductor capabilities for finding binding sites for known transcription factors via sequence matching, we will use the shared DNA-binding GATA sequence as retrieved from one of those factors from MotifDb, DAL80.

Sequence Search

Sequence-based transcription factor binding site search methods answer two questions:

- For a given TF, what DNA sequence pattern/s does it preferentially bind to?
- Are these patterns present in the promoter region of some gene X?
A gene's *promoter region* is traditionally (if loosely) defined with respect to its transcription start site (TSS): 1000-3000 base pairs upstream, and 100-300 basepairs downstream. For the purposes of this workflow, we will focus only on these cis-regulatory regions, ignoring enhancer regions, which are also protein/DNA binding sites, but typically at a much greater distance from the TSS. An alternative and more inclusive "proximal regulatory region" may be appropriate for metazoans: 5000 base pairs up- and downstream of the TSS. Promoter length statistics for yeast are available from Kristiansson et al, 2009: "Evolutionary Forces Act on Promoter Length: Identification of Enriched Cis-Regulatory Elements"

Histogram of the 5,735 Saccharomyces cerevisiae promoters used in this study. The median promoter length is 455 bp and the distribution is asymmetric with a right tail. Roughly, 5% of the promoters are longer than 2,000 bp and thus not shown in this figure.

- The "normal" location of a promoter is strictly and simply upstream of a gene transcript's TSS.
• Other regulatory structures are not uncommon, so a comprehensive search for TFBSs, especially in mammalian genomes, should include downstream sequence as well. For simplicity’s sake we will use a uniform upstream distance of 1000 bp, and 0 bp downstream in the analyses below.

Sample Workflow

We begin by visualizing DAL80’s TF binding motif using either of two Bioconductor packages: seqLogo, and motifStack. First, query MotifDb for the PFM (position frequency matrix):

query(MotifDb, “DAL80”)

## MotifDb object of length 2
## | Created from downloaded public sources: 2012-Nov-01
## | 2 position frequency matrices from 2 sources:
## | JASPAR_CORE: 1
## | ScerTF: 1
## | 1 organism/s
## | Scerevisiae: 2
## Scerevisiae-JASPAR_CORE-DAL80-MA0289.1
## Scerevisiae-ScerTF-DAL80-harbison

There are two motifs. How do they compare? The seqlogo package has been the standard tool for viewing sequence logos, but can only portray one logo at a time.
dal80.jaspar <- query(MotifDb, "DAL80")[[1]]
dal80.scerf <- query(MotifDb, "DAL80")[[2]]
seqLogo(dal80.jaspar)
seqLogo(dal80.scertf)
With a little preparation, the new (October 2012) package motifStack can plot both motifs together. First, create instances of the `pfm` class:

```r
pfm.dal80.jaspar <- new("pfm", mat = query(MotifDb, "dal80")[[1]], name = "DAL80-JASPAR")
pfm.dal80.scertf <- new("pfm", mat = query(MotifDb, "dal80")[[2]], name = "DAL80-ScerTF")
plotMotifLogoStack(DNAmotifAlignment(c(pfm.dal80.scertf, pfm.dal80.jaspar)))
```
Of these two, the JASPAR motif has more detail, but the ScerTF motif is more recently published. ScerTF has a reputation for careful yeast-specific curation. We will use the ScerTF version.

Georis et al mention that DAL80 "competes with Gat1 for binding" -- suggesting that they would have highly similar motifs. MotifDb has 3 entries for GAT1:

```r
query(MotifDb, "gat1")
```

```r
## MotifDb object of length 3
## | Created from downloaded public sources: 2012-Nov-01
## | 3 position frequency matrices from 3 sources:
## |    JASPAR_CORE:  1
## |        ScerTF:  1
## |        UniPROBE:  1
## | 1 organism/s
## | Scerevisiae:  3
## Scerevisiae-JASPAR_CORE-GAT1-MA0300.1
## Scerevisiae-ScerTF-GAT1-zhu
## Scerevisiae-UniPROBE-Gat1.UP00287
```

Plot the three together:

```r
pfm.gat1.jaspar = new("pfm", mat = query(MotifDb, "gat1"))[1], name = "GAT1-JASPAR")
pfm.gat1.scertf = new("pfm", mat = query(MotifDb, "gat1"))[2], name = "GAT1-ScerTF")
pfm.gat1.uniprobe = new("pfm", mat = query(MotifDb, "gat1"))[3], name = "GAT1-UniPROBE")
```
plotMotifLogoStack(c(pfm.gat1.uniprobe, pfm.gat1.scertf, pfm.gat1.jaspar))
The GAT1-JASPAR motif is very similar to DAL80's GATAA motif, and thus consistent with the Georis claim that GAT1 and DAL80 compete for binding. The GAT1-ScerTF and GAT1-UniPROBE motifs are similar, but differ in length. They are reverse complements of the canonical GATAA motif.

To match motifs in a promoter, these steps are required:

- Retrieve the binding motif (the position frequency matrix, or PFM) of a given transcription factor
- Retrieve the promoter regions for a set of candidate targets
- Identify the sequence matches of the binding motif in the genes' promoter regions

The three search motifs, one for DAL80, and two for GAT1, must be transformed before they can be matched to DNA sequence. MotifDb returns a position frequency matrix (a PFM) with all columns summing to 1.0, but the Biostrings matchPWM method expects a position count matrix (a PCM) with integer values. Transform the frequency matrix into a count matrix using the somewhat arbitrary but functionally reliable scaling factor of 100:

```r
pfm.dal80.scertf <- query(MotifDb, "dal80")[[2]]
pcm.dal80.scertf <- round(100 * pfm.dal80.scertf)

pfm.gat1.jaspar <- query(MotifDb, "gat1")[[1]]
pcm.gat1.jaspar <- round(100 * pfm.gat1.jaspar)

pfm.gat1.scertf <- query(MotifDb, "gat1")[[2]]
pcm.gat1.scertf <- round(100 * pfm.gat1.scertf)
```

Create a list of the seven genes from the DAL80 co-regulated subnetwork (displayed above). Lookup their
systematic names, which will be needed immediately below.

```r
genes <- c("DAL1", "DAL2", "DAL4", "DAL5", "DAL7", "DAL80", "GAP1")
orfs <- as.character(mget(genes, org.Sc.sgdCOMMON2ORF))
```

Obtain the coordinates of the transcripts for the orfs.

```r
grl <- transcriptsBy(TxDB.Scerevisiae.UCSC.sacCer3.sgdGene, by = "gene")[orfs]
```

These coordinates, returned in a GRangesList object, specify the start location (chromosome and base pair) of every known transcript for each gene. With this information, GenomicFeatures::getPromoterSeq calculates and returns the DNA sequence of the promoter:

```r
promoter.seqs <- getPromoterSeq(grl, Scerevisiae, upstream = 1000, downstream = 0)
```

We will next search for a match of the motif to the first of these sequences, the promoter for DAL1. Note that here, and below, we use a 90% "min.score" when we call matchPWM. This high minimum match score seems reasonable given the relative absence of variability in DAL80's PFM:

```r
pfm.dal80.scertf
```
The **GATAA** pattern is a very strong signal in this motif.

Note that some restructuring is needed for us to use the results of `getPromoterSeqs` as an argument to `matchPWM`. We call the `getPromoterSeq` method with a GRangesList, which contains a unique set of genomic ranges, representing transcript coordinates, for each of several genes. The corresponding result is a `DNAStringSetList` in which there is one DNAStringSet (essentially a list of DNAStrings) for each gene in the input list. Both of these variables are therefore lists of lists, in which the outer list is named with genes, and the inner lists are per-transcript coordinates or DNA strings.

Since we need DNA strings without that overarching by-gene-name structure, we call `unlist` to strip off that structure, leaving us with the desired DNAStringSet:

```r
print(class(promoter.seqs))
## [1] "DNAStringSetList"
## attr(,"package")
## [1] "Biostrings"

promoter.seqs <- unlist(promoter.seqs)
print(class(promoter.seqs))
```
The close proximity of these two GATAA hits suggests that dimeric DAL80, or some other GATAA-binding dimer, may bind DAL1.

All of the matches in the promoters of all seven genes to one binding motif may be found at once with this command:

```r
pwm.hits <- sapply(promoter.seqs, function(pseq) matchPWM(pcm.dal80.scertf, pseq, min.score = "90%"))
```

And we can summarize the motif hits for each of the three motifs (dal80.scertf, gat1.jaspar, gat1.scertf) by creating a data.frame of counts, by gene and motif. First, determine the hits:
dal80.scertf.hits <- sapply(promoter.seqs, function(pseq) matchPWM(pcm.dal80.scertf, pseq, min.score = "90%"))
gat1.scertf.hits <- sapply(promoter.seqs, function(pseq) matchPWM(pcm.gat1.scertf, pseq, min.score = "90%"))
gat1.jaspar.hits <- sapply(promoter.seqs, function(pseq) matchPWM(pcm.gat1.jaspar, pseq, min.score = "90%"))

Now count their lengths:

dal80.scertf <- sapply(dal80.scertf.hits, length)
gat1.jaspar <- sapply(gat1.jaspar.hits, length)
gat1.scertf <- sapply(gat1.scertf.hits, length)

Finally, create a data.frame from this information:

tbl.gata <- data.frame(gene = genes, dal80.scertf, gat1.jaspar, gat1.scertf)

The simple dag80.scertf 5-base motif has the most hits. The more complex 8-base gat1.jaspar mtoif has fewer hits: perhaps it is over-specified. The 'other'(non-GATAA) motif of GAT1 obtained from ScerTF has fewer matches in the promoters of these genes than do the GATA motifs. The non-GATAA motif hits may in fact, be not much different from chance, as could be revealed by sampling the distribution of motif hits in the
promoters of randomly selected genes. Such analyses will be left as an exercise for the reader.

Biological Summary

This dataset and our exploration has revealed a number of GATAA binding sites within these tightly co-regulated NCR genes, but leaves unanswered questions, some of which are:

- GAT1 is reported to have two (or more) quite different binding motifs. Is this due to its having two or more distinct binding domains? Are they each functional, but only in different conditions?
- The gene expression of the negative regulator DAL80 is highly correlated with the expression of genes it is known to repress. We would expect the opposite relationship between a negative regulator and its targets. Why doesn't abundant DAL80 prevent the expression of the other six genes?
- The DAL80/ScerTF motif and GAT1/JASPAR motif are very closely related. The match table, just above, shows quite different totals for the two motifs. Does the find structure of the motif explain the difference?
- One speculative explanation for the counter-intuitive DAL80 expression is "nuclear sequestration", a mechanism by which a gene is expressed but the mRNA is held in reserve for later use. See Lavut A, Raveh D 2012.

That GAT1 has multiple binding motifs (we show two, SGD four is yet another indication of the incompletely understood complexity of gene regulation.

The four GATAA-binding regulators, two positive and two negative, and their many downstream targets, some of whose binding sequences we have studied here, can thus be seen to be parts of complex regulatory circuits whose full elucidation has not yet been worked out. Judicious integration of many other kinds of data, careful laboratory work, and the right computational tools, will eventually clarify them.
TASK 2

Clustering Copy file capitals.txt (https://courses.cs.ut.ee/2011/Bioinformatics/uploads/Main/capitals.txt) to your directory

1. Run R, input the data there using
capitals=read.table("capitals.txt")

1. Run K-means with different values of K (kmeans)
2. Calculate all pairwise distances, try different measures (dist)
3. Cluster with hierarchical clustering, try different methods (hclust)
4. Plot the result to PNG (png, piclust)