Identification and quantification of DNA sequence motifs that occur within the flanking regions of a set of eQTL

Masters Project

Jackie Kiewa

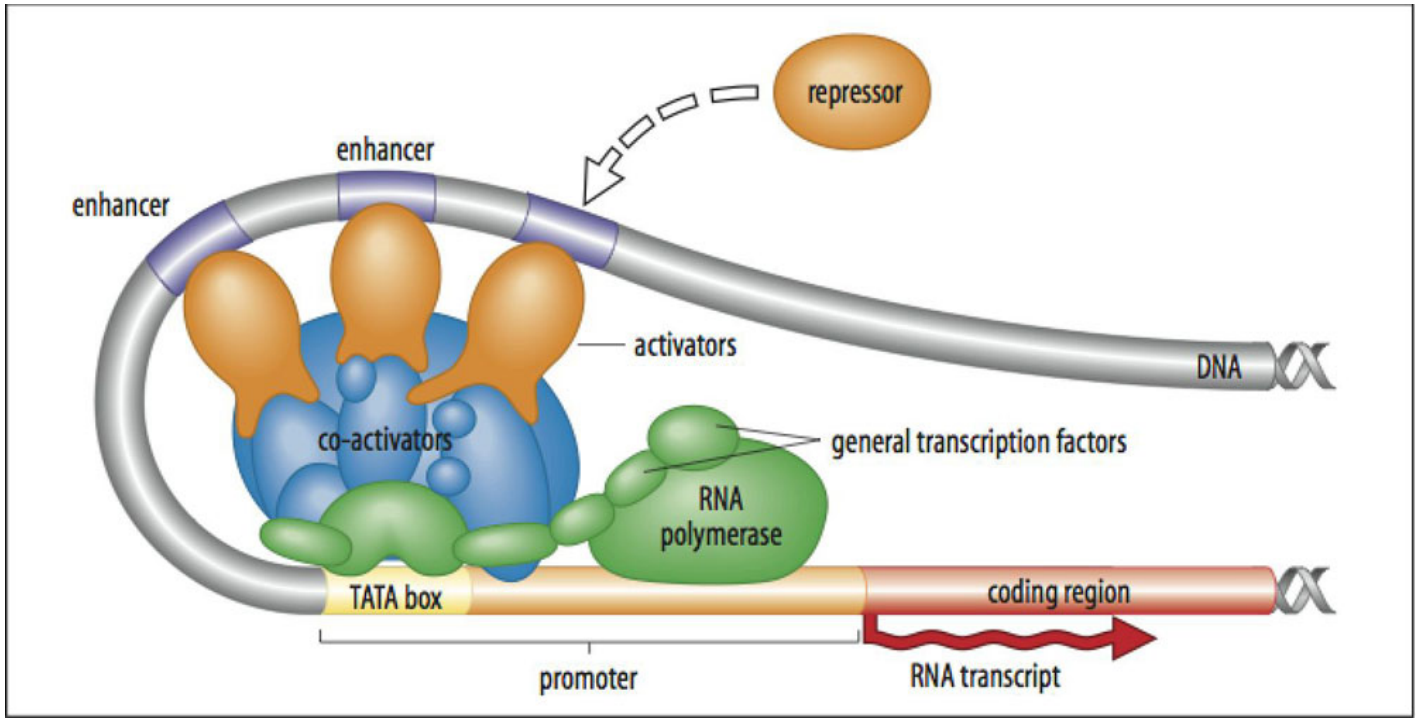
May 15, 2018

Outline

Aim of the Research

- To identify and quantify DNA sequence motifs that occur within the flanking sequences of a set of eQTL

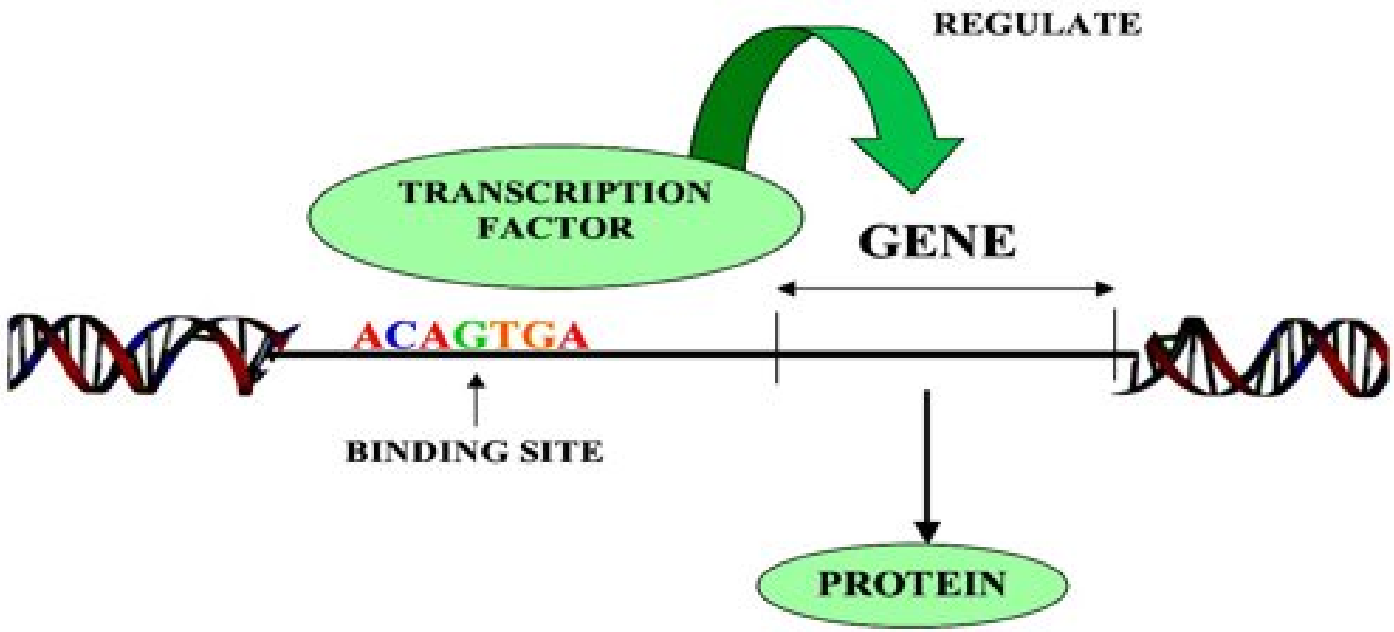
Why?



- It is hypothesised that eQTLs influence their targeted gene through regulation of gene expression
- Quantification and characterisation of enriched sequence motifs in the vicinity of eQTLs will contribute to an understanding of
  - The mechanics of regulation
  - Variation in regulation

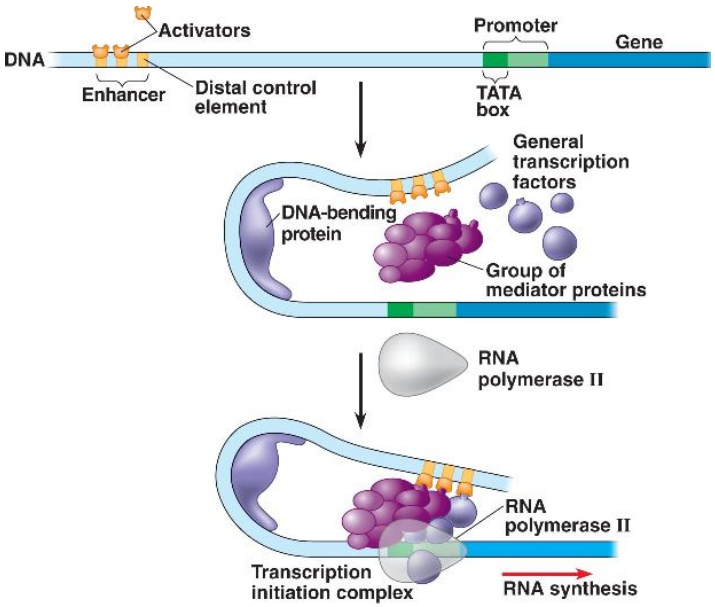
What are motifs?

- DNA motifs are short, recurring patterns in DNA that are presumed to have biologically active functions
- The most intensely studied function of DNA motifs is as enhancers, or binding sites for transcription factors.

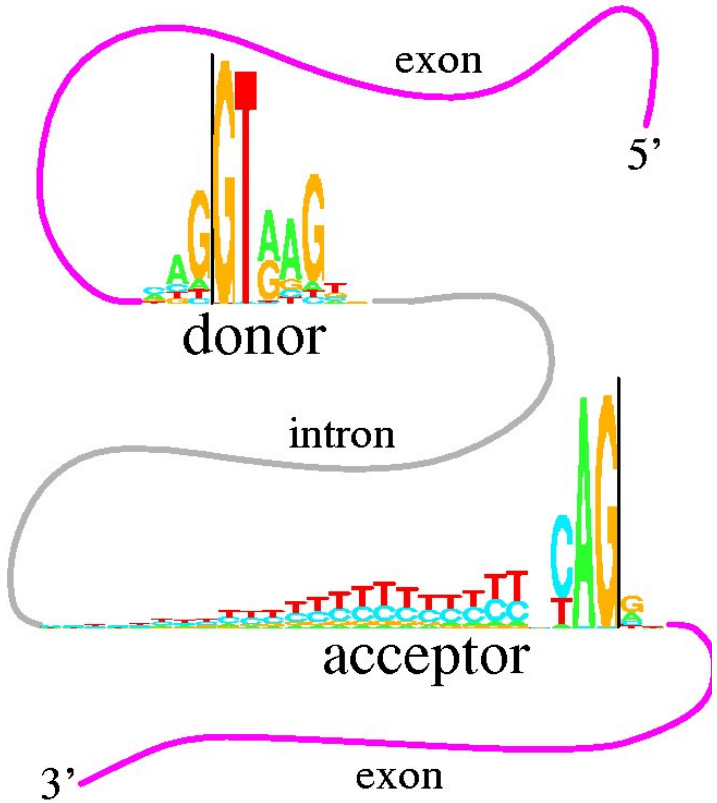


Other functions for motifs

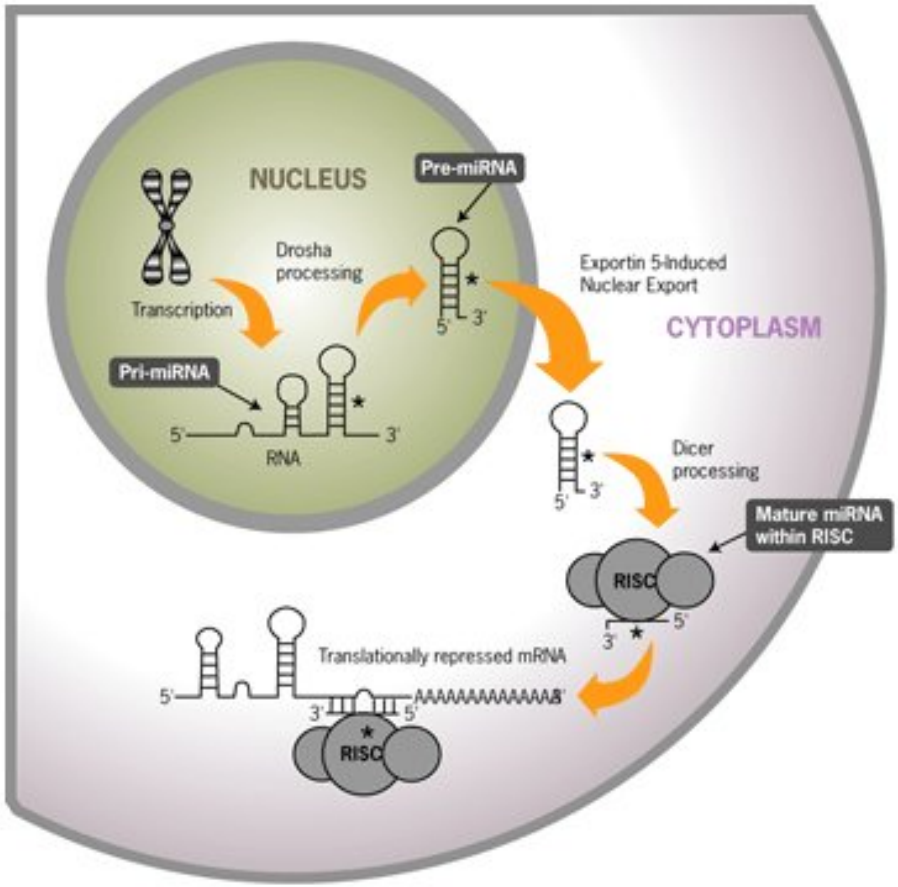
- Change the 3D structure of the genome
- Provide a section that is easy to unwind, such as the TATA box motif, which can initiate transcription



- Create splice sites for the removal of introns (post-transcriptional processing)



Another post transcriptional function: as an MiRNA binding site



My data: the set of eQTLs

- Data from the Consortium for the Architecture of Gene Expression (CAGE)
- Individual-level whole blood expression and genotype data from 2765 individuals

Target Group	
Total eQTLs	14,995
cis eQTLs	11,204
cis eQTLs without duplicates	10,499

Control Group	
Total non-eQTL snps	17,226
random set of null snps	10,499

My data: the sequences

- Sequences were created using the flanking regions of each eQTL or null snp:

Pilot study	100 nucleotides on either side of the eQTL (or null snp)
Main study	1000 nucleotides on either side of the eQTL (or null snp

Finding Motifs

Experimentally: Motifs can be found through actual binding events, measured through ChIP-seq.

Using computer algorithms: Using sequence data, find "words" or short sequences of DNA which seem to be enriched over these sequences.

The most common length of a transcription binding site is 10 nucleotides, though it can be shorter or longer.

Finding Motifs

CGGGGCTATGCAACTGGGTCGTCACATTCCCCTTTCGATA
TTTGAGGGTGCCCAATAAATGCAACTCCAAAGCGGACAAA
GGATGCAACTGATGCCGTTTGACGACCTAAATCAACGGCC
AAGGATGCAACTCCAGGAGCGCCTTTGCTGGTTCTACCTG
AATTTTCTAAAAAGATTATAATGTCGGTCCATGCAACTTC
CTGCTGTACAACTGAGATCATGCTGCATGCAACTTTCAAC
TACATGATCTTTTTGATGCAACTTGGATGAGGGAATGATGC

Finding Motifs

CGGGGCTATGCAACTGGGTCGTCACATTCCCCTTTCGATA
TTTGAGGGTGCCCAATAAATGCAACTCCAAAGCGGACAAA
GGATGCAACTGATGCCGTTTGACGACCTAAATCAACGGCC
AAGGATGCAACTCCAGGAGCGCCTTTGCTGGTTCTACCTG
AATTTTCTAAAAAGATTATAATGTCGGTCCATGCAACTTC
CTGCTGTACAACTGAGATCATGCTGCATGCAACTTTCAAC
TACATGATCTTTTGATGCAACTTGGATGAGGGAATGATGC

Finding Motifs

CGGGGCTATcCAAgTGGGTCGTCACATTCCCCTTTCGATA
TTTGAGGGTGCCCAATAAggGCAACTCCAAAGCGGACAAA
GGATGgAtCTGATGCCGTTTGACGACCTAAATCAACGGCC
AAGGAaGCAACcCCAGGAGCGCCTTTGCTGGTTCTACCTG
AATTTTCTAAAAAGATTATAATGTCGGTCCtTGgAACTTC
CTGCTGTACAACTGAGATCATGCTGCATGCcAtTTTCAAC
TACATGATCTTTTTGATGgcACTTGGATGAGGGAATGATGC

Finding Motifs: "The Planted (l,d)-Motif Search" (Buhler and Tompa, 2002)

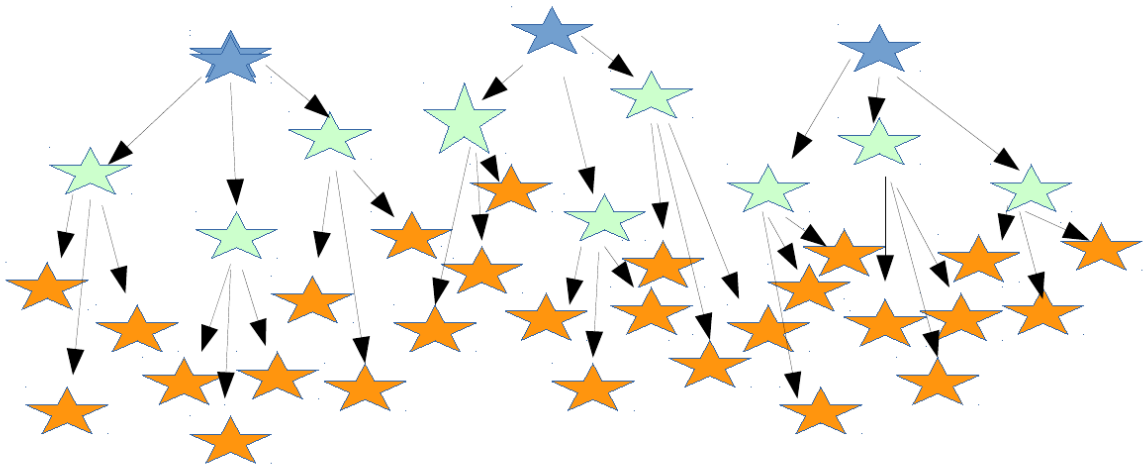
M = motif consensus
l = length of M
t = number of sequences
n = length of each sequence
d = number of point changes in each occurrence of M

Number of possible starting positions = $t(n - l + 1)$

Number of possible starting position combinations = $(n - l + 1)^t$

For my data, number of possible combinations for a 10nt motif = 1992^{10499}

Finding Motifs: An "NP-complete problem"



- NP = nondeterministic polynomial time
- Exact (pattern finding) algorithms focus on ways to reduce the search space
- Approximate algorithms use heuristic approaches such as expectation maximisation or oligo tables

Finding Motifs Example: Gibbs Sampling

- Lawrence et al. 1993 Detecting subtle sequence signals: A Gibbs sampling strategy for multiple alignment. *Science 262*: 5131 pp 208-214
- algoshareify 2012 Gibbs sampling https://www.youtube.com/watch?v=cAtCTVVqCVU

Set of 4 sequences, each 10 nucleotides long ($t = 4$; $n = 10$):

A C C A T G A C A G

G A G T A T A C C T

C A T G C T T A C T

C G G A A T G C A T

One example of the motif in each sequence (OOPS model).

Motif is 7 nucleotides long ($l = 7$)

	0	1	2	3	4	5	6	7
A								
C								
G								
T								

Choose a target sequence at random

A C C A T G A C A G target sequence

G A G T A T A C C T background

C A T G C T T A C T background

C G G A A T G C A T background

For the background sequences, a start point is chosen at random

	0	1	2	3	4	5	6	7
A								
C								
G								
T								

For background motifs only, count the number of times each nucleotide occurs in each position

A C C A T G A C A G target sequence

G A G T A T A C C T background

C A T G C T T A C T background

C G G A A T G C A T background

	0	1	2	3	4	5	6	7
A		0	1	1	2	1	0	0
C		0	1	0	0	1	2	1
G		2	1	0	0	0	1	0
T		1	0	2	1	1	0	2

To fill in the green column, just count up and add all the non-motif background nucleotides

Then count the number of times each nucleotide occurs in the non-motif background as a whole

A C C A T G A C A G target sequence
G A G T A T A C C T background
C A T G C T T A C T background
C G G A A T G C A T background

	0	1	2	3	4	5	6	7
A	3	0	1	1	2	1	0	0
C	2	0	1	0	0	1	2	1
G	2	2	1	0	0	0	1	0
T	2	1	0	2	1	1	0	2

Now need to normalize each of the counts by dividing each column element by the number of background sequences. Also need to add pseudocounts to avoid having any zero answers.

Normalize the motif counts

$$q_{i,j} = \frac{c_{ij}+b_j}{t-1+B}$$

	0	1	2	3	4	5	6	7
A	3	0	1	1	2	1	0	0
C	2	0	1	0	0	1	2	1
G	2	2	1	0	0	0	1	0
T	2	1	0	2	1	1	0	2

$$p_{1,A} = \frac{c_{1,A}+b_A}{t-1+B} = \frac{0+0.5}{4-1+2} = 0.1$$

	0	1	2	3	4	5	6	7
A		0.1						
C								
G								
T								

Normalize the non-motif counts

$$q_j = \frac{c_j + b_j}{\sum_{k=1}^{j} c_k + B}$$

	0	1	2	3	4	5	6	7
A	3	0	1	1	2	1	0	0
C	2	0	1	0	0	1	2	1
G	2	2	1	0	0	0	1	0
T	2	1	0	2	1	1	0	2

$$p_A = \frac{3+0.5}{9+2} = 0.31$$

	0	1	2	3	4	5	6	7
A	0.31	0.1	0.3	0.3	0.5	0.3	0.1	0.1
C	0.23	0.1	0.3	0.1	0.1	0.3	0.5	0.3
G	0.23	0.5	0.3	0.1	0.1	0.1	0.3	0.1
T	0.23	0.3	0.1	0.5	0.3	0.3	0.1	0.5

Back to the target sequence

A C C A T G A C A G

Since the motif length l is 7, there are 4 possible start points for the motif $(n - l + 1)$

So we now iterate through each start position, using the frequency table to calculate the probability of each resulting motif

A C C A T G A C A G

$$p(M_1) = \frac{p_{1,A} \cdot p_{2,C} \cdot p_{3,C} \cdot p_{4,A} \cdot p_{5,T} \cdot p_{6,G} \cdot p_{7,A}}{p_A \cdot p_C \cdot p_C \cdot p_A \cdot p_T \cdot p_G \cdot p_A}$$

	0	1	2	3	4	5	6	7
A	0.31	(0.1)	0.3	0.3	(0.5)	0.3	0.1	(0.1)
C	0.23	0.1	(0.3)	(0.1)	0.1	0.3	0.5	0.3
G	0.23	0.5	0.3	0.1	0.1	0.1	(0.3)	0.1
T	0.23	0.3	0.1	0.5	0.3	(0.3)	0.1	0.5

$p(M_1)$	$p(M_2)$	$p(M_3)$	$p(M_4)$
0.16	0.13	0.26	0.017

Weighted random sample

$p(M_1)$	$p(M_2)$	$p(M_3)$	$p(M_4)$
0.16	0.13	0.26	0.017

Normalize the table:

$p(M_1)$	$p(M_2)$	$p(M_3)$	$p(M_4)$
0.28	0.23	0.46	0.03

The algorithm now makes a random but weighted choice of motif start position, using the normalized weights for each start position.

Continue iterations until convergence

- This represents the first iteration of the algorithm, ending with a choice of position for the motif in the first target sequence.
- For the second iteration a different sequence is chosen as target, and the process is repeated, resulting in the choice of a possible motif position for this sequence.
- The process is repeated hundreds of times.
- At designated intervals, a log likelihood of current motifs is calculated - if it is better than the last, continue, otherwise, go back to the positions used for the previous log likelihood.
- Eventually, convergence should occur and motif positions will no longer change.

Finding Motifs: the Position Weight Matrix

TFBS Position Weight Matrix (PWM)

Sites

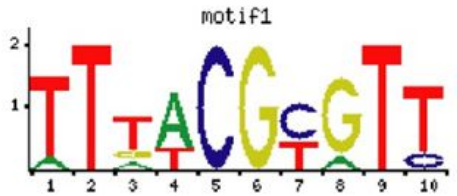
ATGGCATG
AGGGTGCG
ATCGCATG
TTGCCACG
ATGGTATT
ATTGCACG
AGGGCGTT
ATGACATG
ATGGCATG
ACTGGATG

Alignment Matrix

Pos	A	C	G	T
1	9	0	0	1
2	0	1	2	7
3	0	1	7	2
4	1	1	8	0
5	0	7	1	2
6	8	0	2	0
7	0	3	0	7
8	0	0	8	2

Frequency weight Matrix

Pos	A	C	G	T	Con
1	0.9	0	0	0.1	A
2	0	0.1	0.2	0.7	T
3	0	0.1	0.7	0.2	G
4	0.1	0.1	0.8	0	G
5	0	0.7	0.1	0.2	C
6	0.8	0	0.2	0	A
7	0	0.3	0	0.7	T
8	0	0	0.8	0.2	G




```
A [ 2  0  2 11  0  0  0  2  0  0 ]
C [ 0  0  0  0 15  0  9  0  0  3 ]
G [ 0  0  3  0  0 15  0 13  0  0 ]
T [13 15 10  4  0  0  6  0 15 12 ]
```

Note the strong independence assumption between positions.
Holds for most transcription binding profiles in the human genome.

Developing a PWM

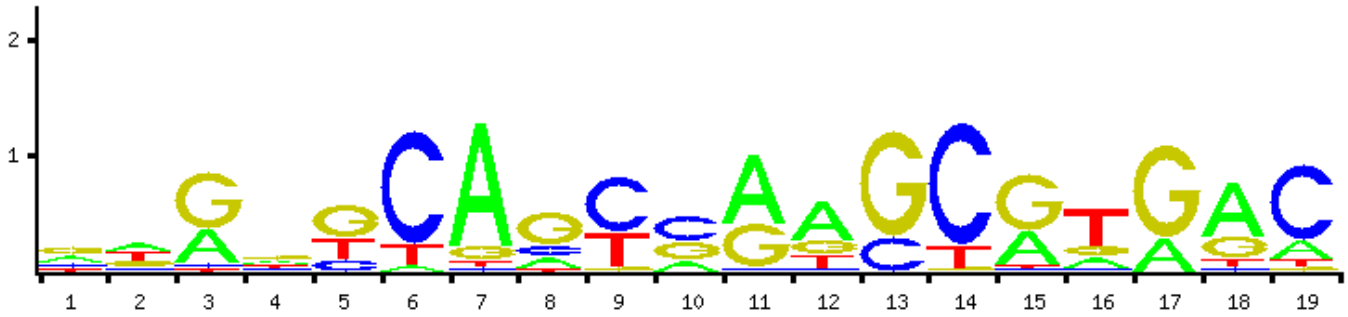
- Find all examples of the putative motif
- The weighting of each nucleotide in each position is determined by

$$\frac{q_{ij}}{p_j}$$

- Where q_{ij} is the number of times the nucleotide j appears in the i^{th} position of the motif, divided by the number of motifs
- And p_j is the number of times j appears in the background (no motifs) divided by the total number of nucleotides
- And the probability of the motif M is given by

$$p_M = \sum_{i=1}^{I} log \frac{q_{ij}}{p_j}, j = [ATCG]$$

The PWM model



- Advantages
  - The model is simple and easy to use
  - It easily invokes the visual aid of a sequence logo
  - Several databases store hundreds of PWMs for known TF binding motifs
- Disadvantages
  - The model assumes independence between each base in the motif
  - Many different PWMs exist for every known TF, with no robust way to test the quality of each PWM

The Pilot Study

- The main objective of the pilot study was to choose and evaluate algorithms that might isolate motifs enriched in the eQTL sequences compared to the null sequences

Choosing algorithms

- There are many - hundreds - of algorithms to choose from
- All use different methods to speed things up
- Often their academic papers (if one exits) are not very enlightening, so the user has very little idea of what is going on behind the scenes
- There is no "gold standard" algorithm for comparison of new algorithms

Our criteria

- Used a discriminative algorithm
- Number of citations
- Represented a cross-section of algorithms
- Ease of use
- Appropriate for eQTL sequences

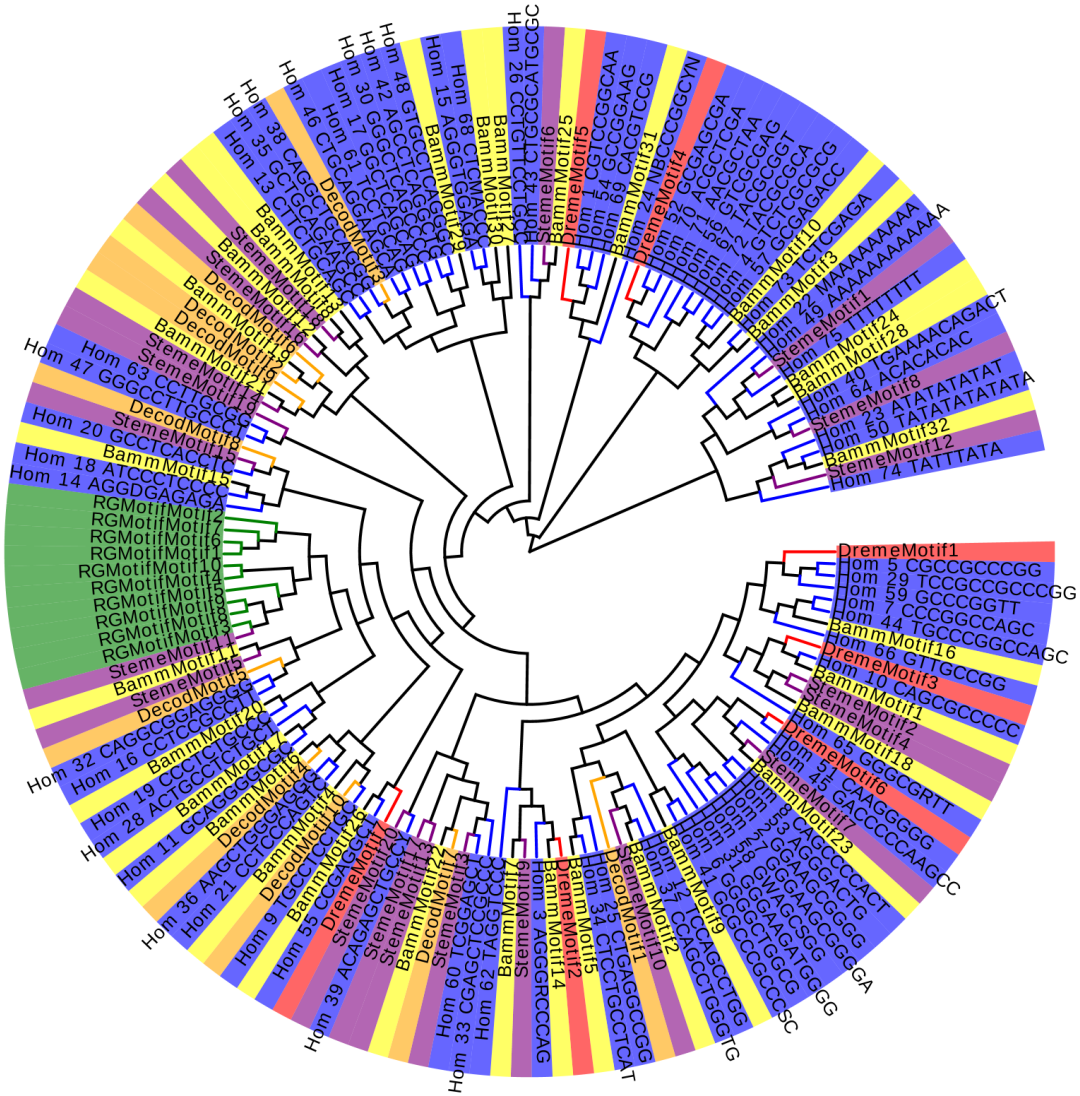
The algorithms chosen

Algorithm	Methods Used
Dreme	Combining oligos and EM
HOMER	Combining oligos and optimization
motifRG	Logistic regression
STEME	Suffix tree
BaMM!motif	Expectation maximisation
DECOD	k-mer counts table

An organising algorithm: Stamp

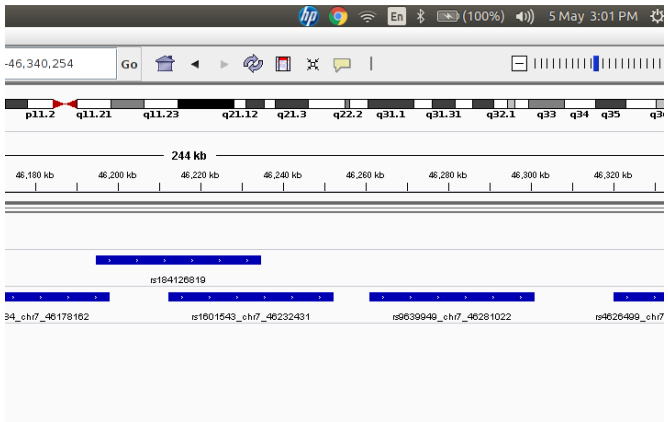
- Reviews of algorithms have recommended the use of more than one algorithm in motif searches (Makolo, 2016; Tompa, 2005)
- Algorithms have been developed to organise the results of these complex searches
- Motifs can be fed into the Stamp algorithm:
  - aligns the motifs
  - develops a "tree" of related motifs
  - clusters the motifs into known transcription factor binding sites

Results



red=DREME
blue=HOMER
yellow=BaMM
green=motifRG
purple=STEME
orange=DECOD

For the Main Study:



- Used the HOMER and the BaMM!motif algorithms
- Decided on 2001 nucleotides as the length of each sequence, centred on the cis eQTLs
  - longer sequences had too many overlaps with the null sequences
  - the algorithms could run with these sequences on my computer
    - ★ HOMER took about 5 hours
    - ★ BaMM took about 36 hours - and needed 32GB of virtual memory

The HOMER algorithm

- The bad bits:
  - No published journal paper that describes the algorithm
  - The article nominated for citation is about the use of HOMER in macrophage detetction, not about the HOMER algorithm
  - Its extensive set of webpages does not clearly describe the algorithm, nor provide any real statistical analysis

The HOMER algorithm

- How it works
  - Big time saving step: has a data bank of 922 known motifs. It matches these motifs to the target sequences and calculates an enrichment value. Enriched motifs are then masked from the sequences.
  - Then begins the de-novo motif finding:
    - ★ Counts frequencies of short oligos
    - ★ put into an oligo table
    - ★ determines relative significance
    - ★ Promising oligos combined to create larger sequences and their frequency counted
    - ★ Creates a probability matrix and scores oligos against this matrix
    - ★ performs an optimisation step, which juggles the probability scores against the number of oligos that can be included, until an optimum is reached which becomes the new motif.
    - ★ This motif is then masked from all sequences and the process begun again.

The HOMER algorithm

- The good bits
  - It is user friendly and easy to run
  - It has lots of options and ways to analyse the results
  - It has 2652 citations

The BaMM algorithm

- A credible citation: Siebert, M., and Soding, J. (2016). Bayesian Markov models consistently outperform PWMs at predicting motifs in nucleotide sequences. Nucleic Acids Research, 44(13), 60556069.
- Only 12 citations, but a relatively young algorithm, and uses a very different approach from HOMER
- Addresses the problem that PWM's assume independence of all nucleotides within the PWM

Problem with position independence

- G A T C
- G A T C
- G T A C
- G T A C

	Pos 1	Pos 2	Pos 3	Pos 4
A	0	0.5	0.5	0
C	0	0	0	1
G	1	0	0	0
T	0	0.5	0.5	0

Calculate the probability of the sequences GATC and GTTC

GATC

$1 \times 0.5 \times 0.5 \times 1 = 0.25$

GTTC

$1 \times 0.5 \times 0.5 \times 1 = 0.25$

A PWM cannot learn that an A in 2nd position must be followed by a T in 3rd position, and vice versa.

The BaMM algorithm

- BaMM uses a Markov Model to incorporate sequence position dependency
- Markov Models tend to become very complicated very quickly
- A k_{th} order Markov Model takes note of the previous k positions. For a motif of length l, there will be $l \cdot 3 \cdot 4^k$ parameters.
- The BaMM model starts with a motif sequence of 5 nucleotides and uses EM to work out optimal transition and emission probabilities, and obtain a p-value for this model.
- Then it adds one nucleotide at a time to lengthen the motif.
- The new probability is based on Bayesian joint probability, multiplying the probability of the current Markov model with the probability of the new nucleotide occurring in that position.

Initial Results

Each algorithm searched the target group of 10,499 sequences for motifs that were relatively enriched when compared with the null control group of 10, 499 sequences

HOMER suggests a stringent probability value of $p < 1e-50$ when deciding whether to investigate a found motif

	Total motifs found	motifs with $p < 1e-50$
HOMER	238	138
BaMM	99	38

Ran some checks...

- Asked both HOMER and BaMM to search for motifs in the null sequences when compared with the eQTL sequences
  - expecting that not so many motifs would be found and that these might not be very important
- Then divided all the null sequences (17,226) randomly in half and asked the algorithms to find motifs in one half compared with the other half (GroupA Vs GroupB, n=8613)
  - expecting that no or very few motifs would be found, and these purely by chance

Results of the checks:

	Null Vs eQTL	GroupA Vs GroupB
HOMER	No motifs found	No motifs found
BaMM	168 motifs found	132 motifs found

BaMM result particularly disconcerting

- The top motif found for the half-null Vs half-null was the same as the top motif found for the eQTL Vs null
  - C T A C T A A A A A T A C A A A A
- Therefore ran the half-null Vs half-null again - using GroupB as the target and comparing it to GroupA:

GroupB Vs GroupA	124 motifs found with the same top motif

Created two more random groups and tried again:

Results

Half-sequence	Motif present?
Group 1 Vs Group 2	Yes - top motif (1716 of 8613 sequences)
Group 2 Vs Group 1	Yes - top motif (1697 of 8613 sequences)
Group 3 Vs Group 4	Yes - top motif (1669 of 8613 sequences)
Group 4 Vs Group 3	Yes - top motif (1737 of 8613 sequences)

What's happening?

One explanation is that Bamm is not working as a discriminative algorithm, but is simply finding motifs that are enriched in the target sequences

However, this explanation is contradicted by the fact that the motif under scrutiny is not reported at all in the null vs eQTL analysis

So how is Bamm calculating its *eValues*? How does it decide whether a motif is significantly enriched or not?

Looking inside BaMM's black box

- What kind of comparison is significant?
- With respect to C T A C T A A A A A T A C A A A A:
  - The half-sequences have the same number of sequences with this motif. This is seen as significant in both directions (both A vs B and B vs A are significant)
  - eQTL sequences: 3519 of 10499 or 33.52 percent of the sequences have the motif
  - Null sequences: Approximately 3409 of 17226 or 19.78 percent of the sequences have the motif
  - The comparison of eQTL with null sequences is seen as significant, but not the other way round

With respect to HOMER...

Is it better at calculating genuine enrichment values so that fewer motifs are reported?

Or is it just not very good at finding motifs?

Although it did find a lot more motifs for the eQTL sequences, which is what one would expect

Further check: another algorithm

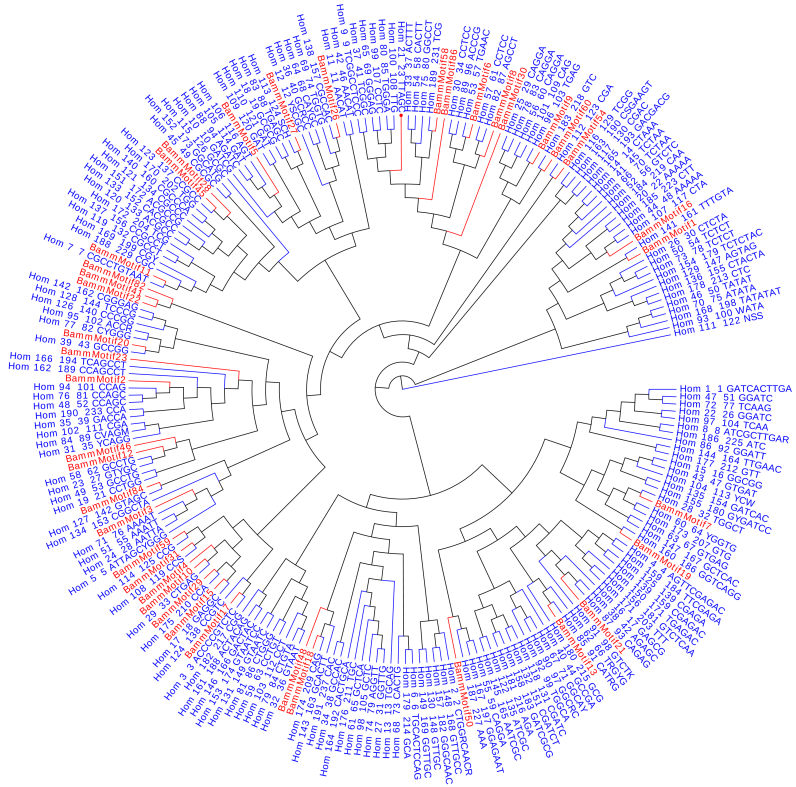
- Whilst it is generally stated that there is no gold standard when it comes to motif finding algorithms, the MEME suite of algorithms is often quoted when new algorithms are measuring their results
- The de novo motif finding algorithms in the MEME suite were generally not suitable for our study
- But the motif enrichment algorithms could be used now we had a list of motifs, and could provide a further check on the relative enrichment of the motifs found by BaMM and HOMER

AME Results

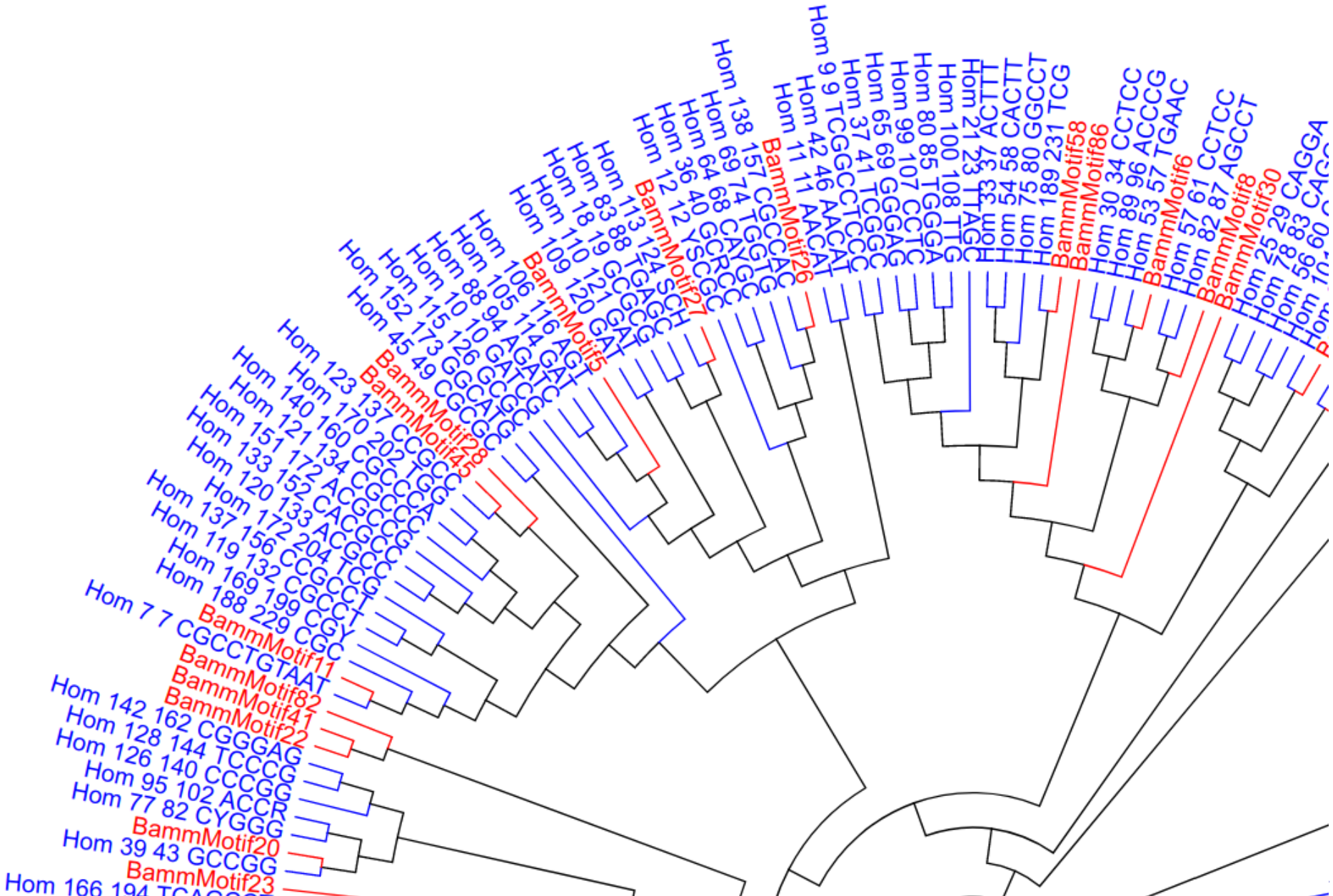
- Submitted all motifs found by BaMM and HOMER, not just those with $p < 1e-50$
- Applied the cut-off of $p < 1e-50$ to the AME motifs found

	Original set had $p < 1e-50$	Original set had $p >= 1e-50$	Total number of motifs with AME $p < 1e-50$
HOMER motifs (n=238)	124 of 138 motifs	67 of 100 motifs	191 of 238 motifs
BaMM motifs (n=99)	28 of 38 motifs	12 of 61 motifs	40 of 99 motifs

Stamp Cladogram



Stamp Cladogram detail

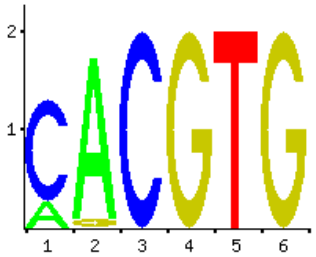


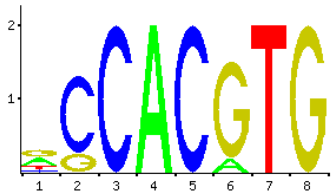
Stamp Results

Stamp provides two main sources of information

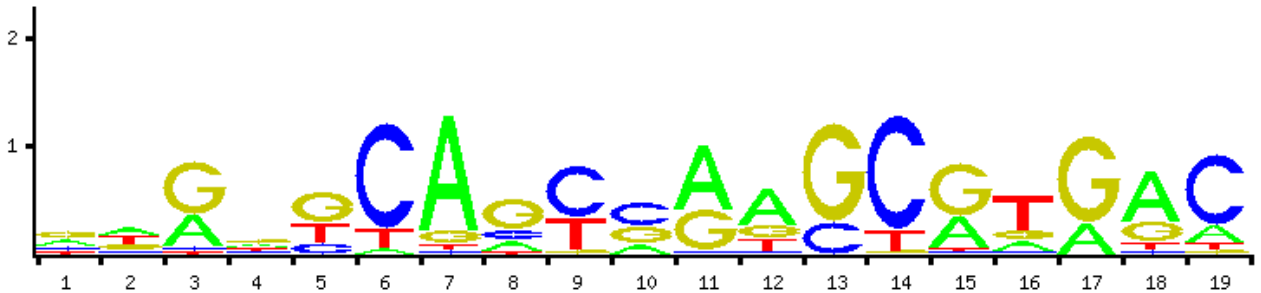
- First it aligns the motifs and uses the alignment to create a phylogenetic tree
- Secondly, it compares the pwm's with an established data base of pwm's to match motifs to transcription factors
  - There are two complications in doing this:
    - ★ Most motifs will bind to more than one transcription factor
    - ★ Most transcription factors will bind to more than one motif

Matches for "Hom1" Sequence (GATCACTTGA)





TF	Prob	Sequence
Snail	0.00001277	GATCACTTGA
Nkx2-5	0.0017017	TCAAGTGATC
Mycn	0.0018502	G ATCACTTG A
Arnt	0.0018675	GAT CACTTG A
Pax5	0.0019205	- ———TCAAGTGATC -



StampResults: The motifs were organized into a total of 117 TFs

Sheet1

TF	Freq	Homer Motifs	Best Homer Match	Bamm Motifs	Best Bamm Match
MEF2A	9	8	4.2558E-006	1	2.1202E-008
IRF1	7	6	3.2215E-009	1	0.000037414
CF2-II	4	4	6.5516E-009	0	NoMotifFound
Fos	4	3	2.6568E-006	1	0.000024565
Prrx2	4	3	9.6374E-006	1	0.000036278
Snail	4	4	3.2471E-006	0	NoMotifFound
SQUA	4	2	0.000006268	2	3.7115E-007
ID1	3	2	0.000017274	1	6.6052E-006
Broad-comple	2	2	5.4473E-007	0	NoMotifFound
CFI-USP	2	1	3.3852E-006	1	2.3691E-006

Stamp Results: Example of multiple motifs

	A	B	C	D	E
1	TF	Match Prob	Motif ID	Sequence	Enrichment Prob
2	Prrx2	9.6374E-006	Hom_24__28_AATTA	AATTAGCCGGG	6.17E-141
3	Prrx2	0.000013905	Hom_51__55_AAATT	AAATTAGCCGGG	1.99E-172
4	Prrx2	0.000018128	Hom_71__76_AAAAT	AAAATTAGCCGGG	1.53E-193
5	Prrx2	0.000036278	BammMotif3	AAATTAGCYRGGYRTGG	0
6					

Comparison with null motifs for Prrx2 transcription factor

	A	B	C	D	E
1	TF	Match Prob	Motif ID	Sequence	Enrichment Prob
2	Prrx2	0.000023162	Hom_9__114_ATGAT	CCATTAATGGG	3.52E-086
3	Prrx2	0.000038305	Hom_30__101_TGTG	TGTGAGTCAATTAA	0
4	Prrx2	0.000021283	Hom_38__87_AATTA	AATTACTTTTGCA	1.33E-282
5	Prrx2	0.00001812	Hom_45__94_AATAA	AATAATTTTAGGC	5.85E-264
6	Prrx2	0.000015199	Hom_47__61_GATTC	GATTCAATTACC	4.89E-259
7	Prrx2	0.000019201	Hom_57__84_GATTC	GATTCAATTACCT	1.43E-218
8	Prrx2	0.000036897	BammMotif129	TAAAARYAGAACTACCA	4.18E-112
9					

Searching for differences

- The eQTL sequences returned a bewildering array of motifs and a long list of transcription factors that might bind to these motifs.
- The null sequences returned different motifs, but another list of transcription factors, many of which are the same as the eQTL transcription factors.
- What differences emerge from this data, and what patterns?

Searching for differences

	Number of TFs	Number of Unique TFs
eQTL	34	18
null	46	30

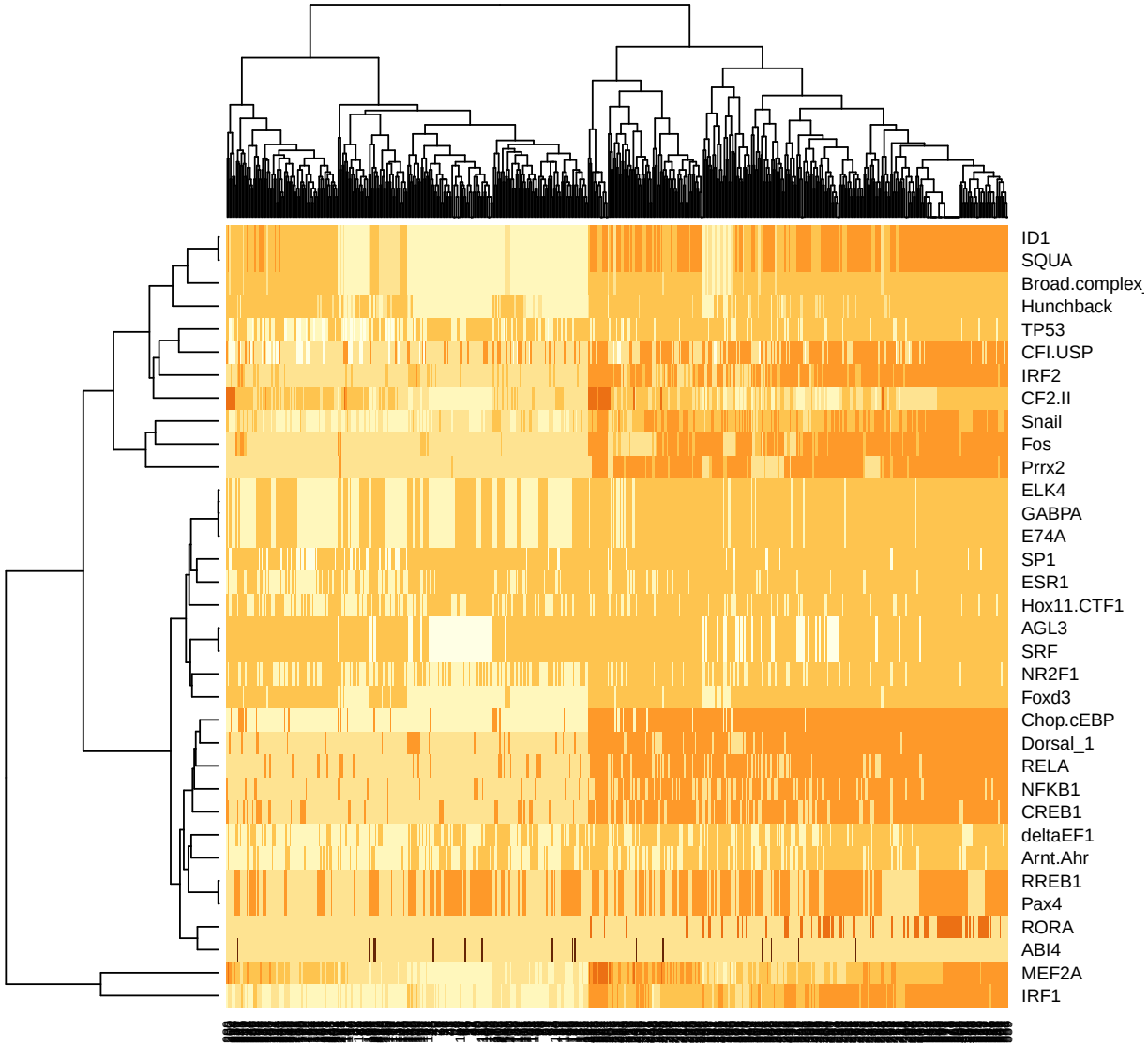
Differences...

Perhaps more important than the collection of transcription factors that are unique to the eQTLs or the nulls is the particular **combination** of transcription factors within any eQTL sequence.
"Individual genes have binding site for multiple transcription factors. These transcriptional factors bind and work in combination to control the individual genes.This is termed as combinatorial gene regulation, a common process of gene regulation in higher eukaryotes including humans" (Qidwai et.al., 2011)

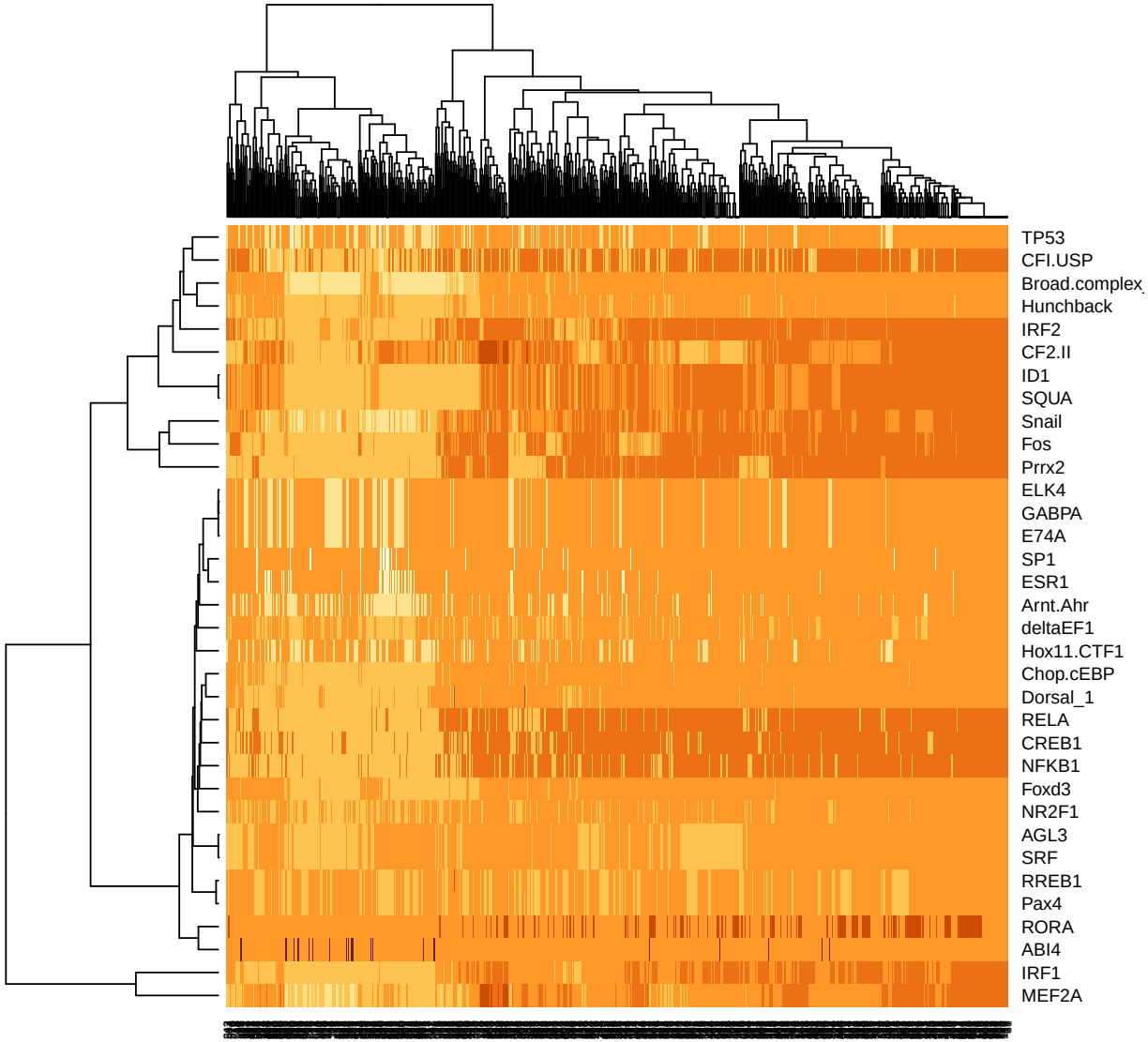
Complete list of eQTL transcription factors

ABI4	AGL3	Arnt-Ahr	Broad-complex_1	CF2-II
CFI-USP	Chop-cEBP	CREB1	deltaEF1	Dorsal_1
E74A	ELK4	ESR1	Fos	Foxd3
GABPA	Hox11-CTF1	Hunchback	ID1	
IRF1	IRF2	MEF2A	NFKB1	NR2F1
Pax4	Prrx2	RELA	RORA	RREB1
Snail	Sp1	SQUA	SRF	TP53

Combinations: Chromosome 11



Combinations: Chromosome 19



Combinations

First combination:

ID1; SQUA; Broad-complex_1; Hunchback; TP53; CFI.USP; IRF2; CF2II; Snail; Fos; Prrx2

Second combination:

ELK4; GABPA; E74A; SP1; ESR1; Hox11-CTF1; AGL3; SRF; NR2F1; Foxd3; Chop-cEBP; Dorsal_1; RELA; NFKB1; CREB1; deltaEF1; Arnt-Ahr; RREB1; Pax4, RORA; ABI4

Third combination:

MEF2A; IRF1

Stamp Results: motifs without strong matches to TFs

- The Stamp search used the JASPAR data base to match motifs to TFs.

182 out of 231 motifs did not achieve a strong match with any TF.

Resolving the unmatched motifs

- Some of the motifs were part of longer motifs:

CAGGAGTTCGA
CAGGAGTTCGAGA
CAGGAGTTCGAGAC

- The shorter motifs were removed

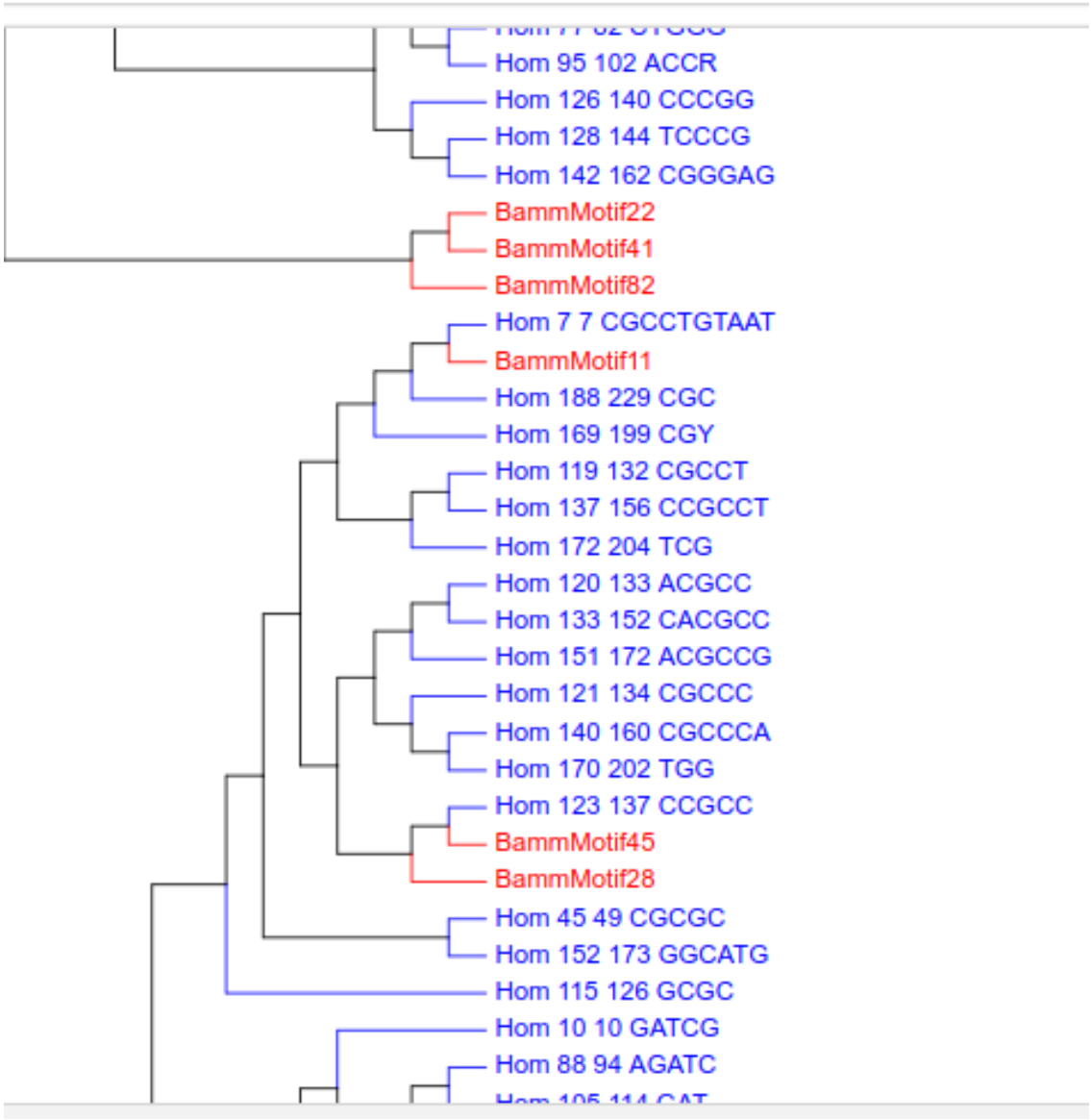
Resolving the unmatched motifs

- The "Hocomoco" and "Uniprobe" lists of TFs were searched using the "TomTom" tool in the MEME suite
- These searches identified strong matches for a further 21 motifs to 9 new TFs

This left a total of 118 unmatched motifs

AAAAAAAAGT	CCGTC	GATCCDCCYRCCTCRGC	GTTCAAGC
AACATGGYGAA	CCTCCCGRGTW	GCACTCCA	GTYGCCCAGGC*
AACTCCTGRSCTCARGT	CCTCCYGAGTAG	GCACTT	NSSGYNNVKVYHGV
AACTGAGGCHCAGAGAG	CCTGTAATCCCAGC	GCCCNGCCC	SCHGCGCCVCTGCA
AATCGC	CGAGACCAGCCTGG*	GCCGAGATYGC	TACAAGGGATGTGAAGG
ACCCCGTCTC	CGASGGCCGTSAAC	GCCGGGCGCGG	TAGTAGAGAYRGGGTTT
ACCCGGGAGGCGG	CGATCT*	GCCTGGGTGACA	TCAAGYGATYCTCC*
ACCRCRCCYGGCYA	CGCCCA*	GCGCGGTGGC*	TCAGCCTCCYRAGTAGC*
ACGCC	CGCCMCC	GCGGCGGCGGCGGC	TCCCAAAGTGCTGGGAT
ACGCCG	CGCCTGTAG*	GCRCCACCACG	TCCCG
ACTACAG	CGGCCT	GCTCCTGAAGGAAGCAC	TCGCCTGA
ACTCCAGCCTGGGYRAC	CGGCTA	GGAGAAT	TCGGGAGGC
ACTTTGGGAGG	CGGGAG	GGATCGCTTGA	TCTCTAMWAAAAA*
AGCCTCCYGAGTA	CGGGCCCHS	GGATTAAGGGCGGTGCA	TGAACCCGGGAG
AGGGAGGGAGGGAGGGA	CGGSSCCA	GGATTGCTTGAGC	TGAGAC
AGTAG	CGGTAGKC	GGCAYGATCTYR	TGAGCCACTGCRC
AGTGCAGTGGYRYRATC	CGRGGCG	GGCCGGGYRYRGTGGCT	TGAGCCCAGGAGKT
AGTGGCGCGATCTC	CGYCTCTA	GGCCTCCCAAAGT	TGCACTCCAG
AGTTCGAGAC	CTCTACTAAAA*	GGCGCRG	TGCAGTGAGC*
AGTTCTGGAGGCTGGRA	CTGAGGCRGGA	GGCGGATCAC	TGGCAMGATCTYG
ATTAGCYGGG	CTGGRCAACR	GGCRCGATCTCGG	TGGGCGAC
AYRGAGTCTYRCTCTGT	CTGTARTCCCAGC	GGCTGAGGCAGGAGAAT	TGGTGGCGGGCG
AYTACAGGYRCCYRCCA	CTTGAACCYRGGAGGYR	GGTCAGG	TTCGAGA
CACTTTGGGAGG	CVAGMCCAGCCTG	GTAATCCCAGC*	TTTGAGAC
CAGGAGTTCGAGAC*	CYGGGCGTGGTGG	GTAGC	TTTGTA
CAGGAGTTCVAG*	GACCAKCCTGGCYAACA*	GTCAGGAG	VTGTGCCAGGCACTG
CAYGCRCCACCA	GACGACG	GTCTCAAAAAAA*	YCAGGCTGGTC
CCACATCCTCTCCAGCA	GACTAC	GTCTKGCTCTGTC	YSCGCCRCCV
CCCGG	GATCACACCACTGC*	GTGAGCCGAGAT	
CCGCCT*	GATCATGCCACTGY	GTGATCYGCCC	

Motif Pairs


Hom 95 102 ACCR
Hom 126 140 CCCGG
Hom 128 144 TCCCG
Hom 142 162 CGGGAG
BammMotif22
BammMotif41
BammMotif82
Hom 7 7 CGCCTGTAAT
BammMotif11
Hom 188 229 CGC
Hom 169 199 CGY
Hom 119 132 CGCCT
Hom 137 156 CCGCCT
Hom 172 204 TCG
Hom 120 133 ACGCC
Hom 133 152 CACGCC
Hom 151 172 ACGCCG
Hom 121 134 CGCCC
Hom 140 160 CGCCCA
Hom 170 202 TGG
Hom 123 137 CCGCC
BammMotif45
BammMotif28
Hom 45 49 CGCGC
Hom 152 173 GGCATG
Hom 115 126 GCGC
Hom 10 10 GATCG
Hom 88 94 AGATC

22 Motif Pairs

Hom_108	Pax4/RREB1/SP1	BammMotif31	SP1
Hom_112	unmatched	BammMotif54	unmatched
Hom_113	unmatched	BammMotif27	unmatched
Hom_114	ABI4/SP1/SP2	BammMotif59	TAF1/SP2
Hom_123	unmatched	BammMotif45	unmatched
Hom_127	unmatched	BammMotif84	Dorsal_1/RREB1
Hom_138	Met32-9	BammMotif26	SP4
Hom_141	unmatched	BammMotif1	MEF2A/SQUA
Hom_160	unmatched	BammMotif19	RORA
Hom_17	ZFX	BammMotif15	Zbtb3_1048
Hom_174	unmatched	BammMotif18	unmatched
Hom_175	deltaEF1/Snail	BammMotif29	unmatched
Hom_183	unmatched	BammMotif60	unmatched
Hom_189	unmatched	BammMotif58	unmatched
Hom_28	Fos	BammMotif7	Fos
Hom_29	ZN770	BammMotif10	ZN770/ZSC22
Hom_3	unmatched	BammMotif17	unmatched
Hom_39	unmatched	BammMotif20	unmatched
Hom_53	unmatched	BammMotif6	unmatched
Hom_7	unmatched	BammMotif11	unmatched
Hom_85	IRF1	BammMotif13	IRF1
Hom_96	unmatched	BammMotif9	unmatched

Unmatched Sequences for Focus

Hom_108	Pax4/RREB1/SP1	CCSCCMCCMCSCCC	BammMotif31	SP1	CCMCGCCC
Hom_112	unmatched	CGASGGCCGTSAAC	BammMotif54	unmatched	TACAAGGGATGTGAAGG
Hom_113	unmatched	SCHGCGCCVCTGCA	BammMotif27	unmatched	GGCGCRG
Hom_114	ABI4/SP1/SP2	CCGGCGCCGCCCSS	BammMotif59	TAF1/SP2	GCGGCGGCGGCGGC
Hom_123	unmatched	CCGCC	BammMotif45	unmatched	GGATTAAGGGCGGTGCA
Hom_127	unmatched	GTAGC	BammMotif84	Dorsal_1/RREB1	GGAAMAACACCCGCTAC
Hom_138	Met32-9	CGCCAC	BammMotif26	SP4	CGCCMCC
Hom_141	unmatched	TTTGTA	BammMotif1	MEF2A/SQUA	CTACTAAAAATACAAAA
Hom_160	unmatched	GGTCAGG	BammMotif19	RORA	YRGATCACAAGGTCAGG
Hom_17	ZFX	GAGGCGGAGG	BammMotif15	Zbtb3_1048	CACTGCAASCTCYRCCT
Hom_174	unmatched	CAGCACTT	BammMotif18	unmatched	TCCCAAAGTGCTGGGAT
Hom_175	deltaEF1/Snail	CCACCTCG	BammMotif29	unmatched	CGRGGCG
Hom_183	unmatched	GTCAGGAG	BammMotif60	unmatched	GCTCCTGAAGGAAGCAC
Hom_189	unmatched	TCGGGAGGC	BammMotif58	unmatched	AGTTCTGGAGGCTGGRA
Hom_28	Fos	TGGCTCACGCC	BammMotif7	Fos	GTGGCTCAYRCCTGTAA
Hom_29	ZN770	CTGAGGCRGGA	BammMotif10	ZN770/ZSC22	GGCTGAGGCAGGAGAAT
Hom_3	unmatched	ACCCCGTCTC	BammMotif17	unmatched	TAGTAGAGAYRGGGTTT
Hom_39	unmatched	GCCGGGCGCGG	BammMotif20	unmatched	GGCCGGGYRYRGTGGCT
Hom_53	unmatched	TGAACCCGGGAG	BammMotif6	unmatched	CTTGAACCYRGGAGGYR
Hom_7	unmatched	CGCCTGTAAT	BammMotif11	unmatched	AYTACAGGYRCCYRCCA
Hom_85	IRF1	CATRGTGAAACCC	BammMotif13	IRF1	CYAACATGGTGAAACCC
Hom_96	unmatched	TGAGCCCAGGAGKT	BammMotif9	unmatched	AACTCCTGRSCTCARGT

Motif positioning

- An important feature of the data is the position of the motif in the sequence
- For each motif, HOMER provides a useful calculation of distance of the motif from the centre of the sequence
- The eQTL motifs show a slight tendency to cluster around the eQTL itself
- The null motifs seem to be more randomly scattered over the sequence

Motif positioning examples

Generated a random sample (5 Homer, 5 Bamm) of the paired motif list as examples

Generated a random sample (5 Homer, 5 Bamm) from all null motifs

Hom_108	Hom_183	Hom_39	Hom_96	Hom_189
Bamm 29	Bamm 54	Bamm 19	Bamm 6	Bamm 20

Table : eQTL motif selection

Hom_12	Hom_22	Hom_31	Hom_53	Hom_54
Bamm 2	Bamm 9	Bamm 89	Bamm 101	Bamm 149

Table : null motif selection

The eQTL plots

Figure :
Hom 108

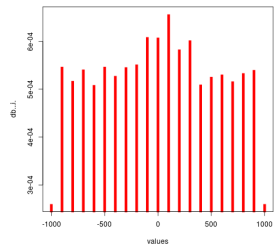


Figure :
Hom 183

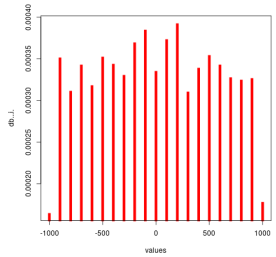


Figure :
Hom 39

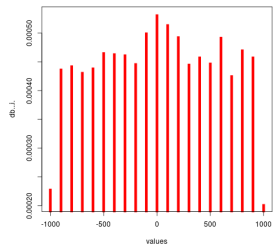


Figure :
Hom 96

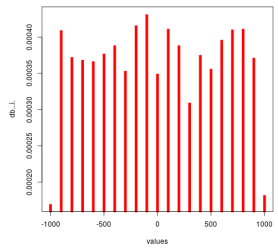


Figure :
Hom 189

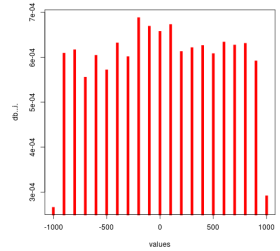


Figure :
Bamm 29

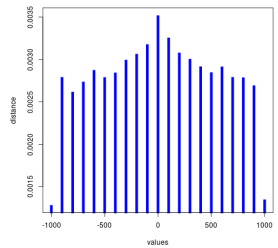


Figure :
Bamm 54

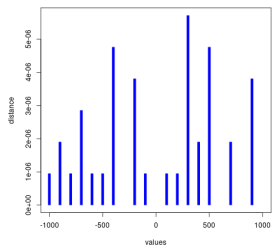


Figure :
Bamm 19

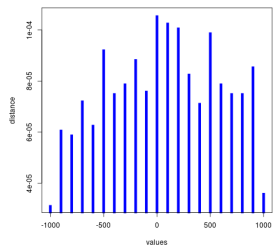


Figure :
Bamm 6

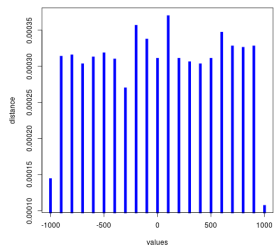
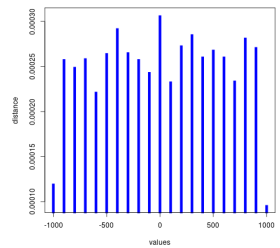


Figure :
Bamm 20



The Null Plots

Figure :
Hom 12

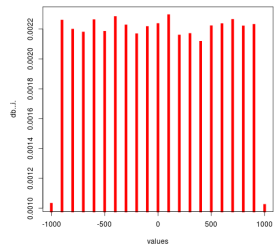


Figure :
Hom 22

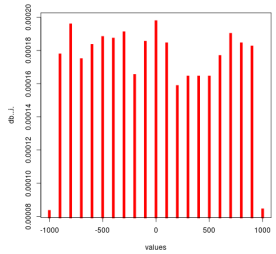


Figure :
Hom 31

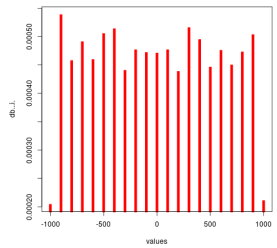


Figure :
Hom 53

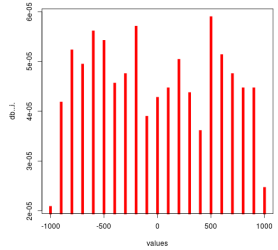


Figure :
Hom 54

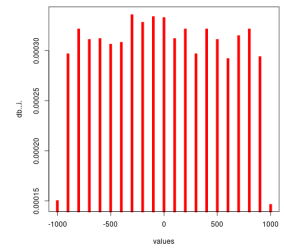


Figure :
Bamm 2

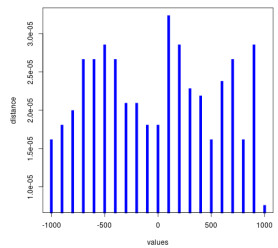


Figure :
Bamm 9

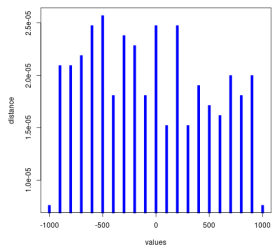


Figure :
Bamm 89

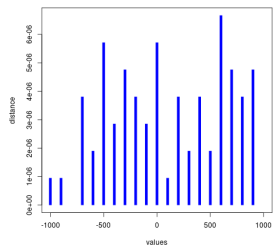


Figure :
Bamm 101

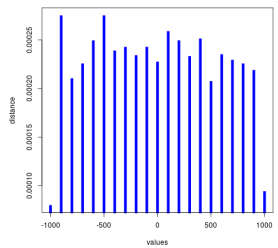
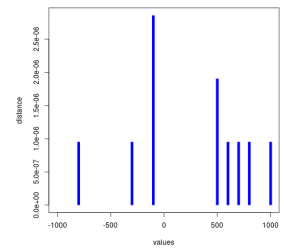
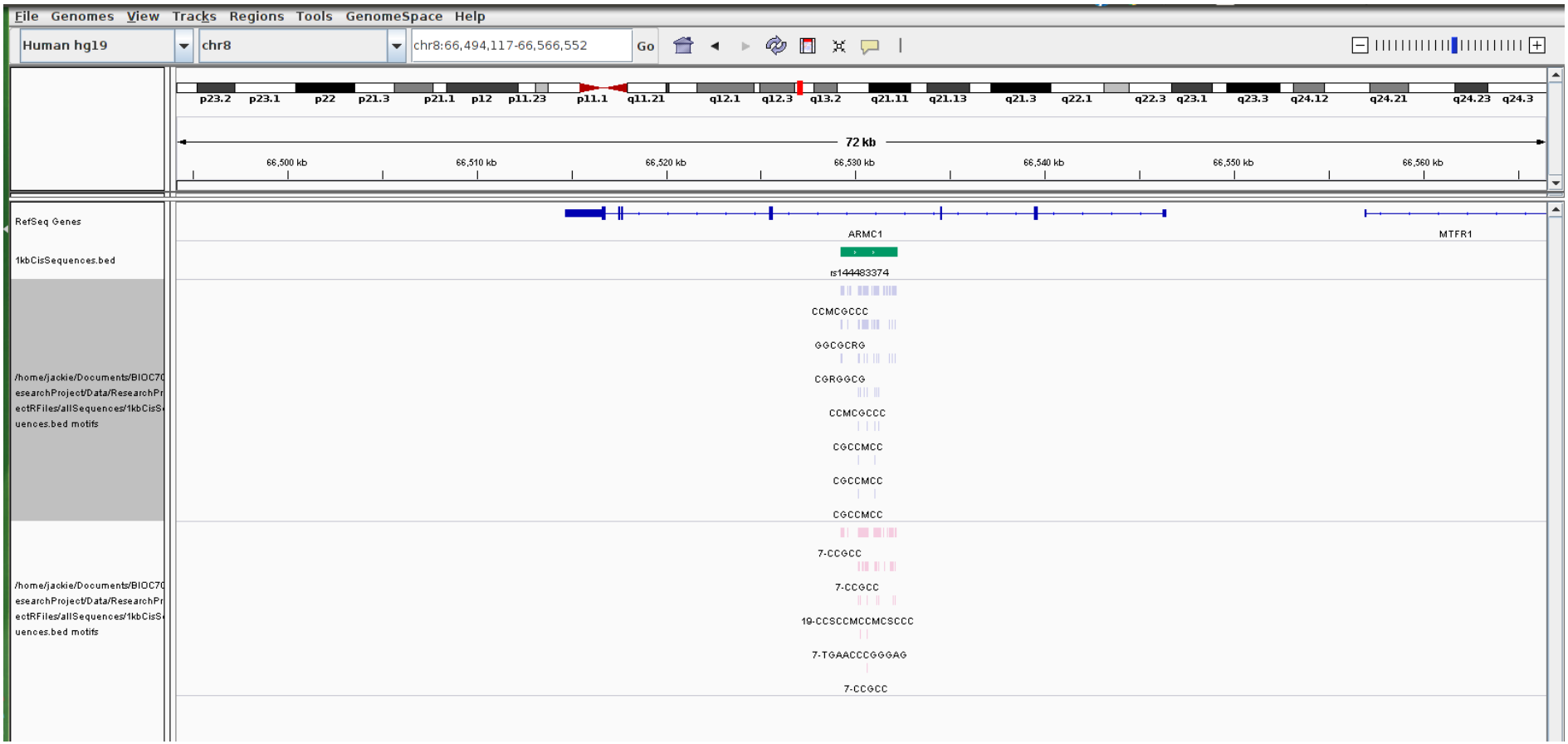


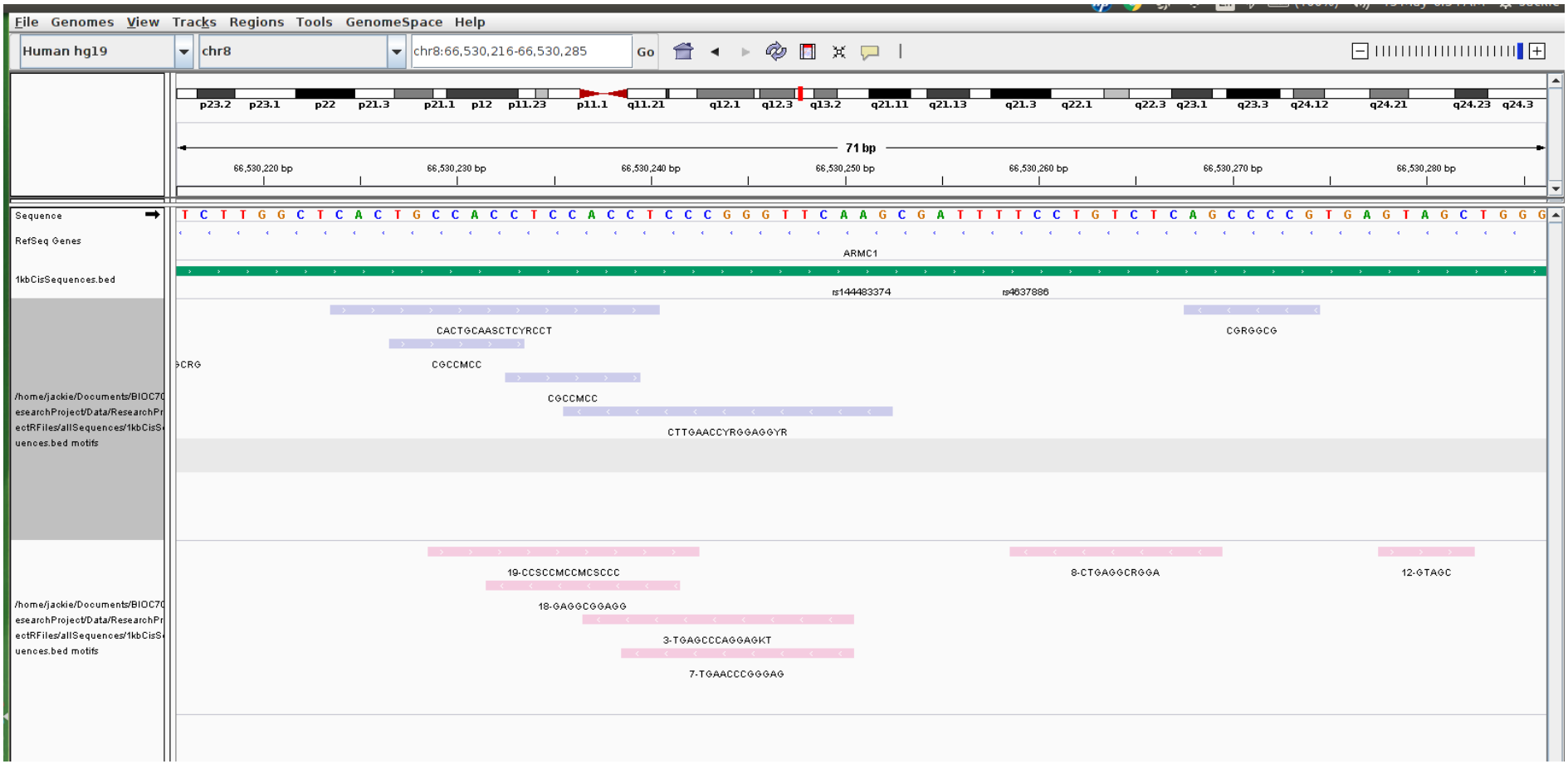
Figure :
Bamm 149



On the genome viewer



On the genome viewer



Conclusions

- In terms of the algorithms...
  - Our chosen algorithms, HOMER and BaMM!motif, were good at finding motifs within a manageable time frame.
  - The enrichment values offered by BaMM are sometimes puzzling
  - Once motifs are found, other algorithms can be used to substantiate the enrichment values
  - Despite its lack of academic credentials, the Homer algorithm is fast, reliable, its results seemed valid, and it provides a suite of extremely useful tools for annotation of data.
  - The Stamp algorithm was extremely useful in organizing the data

Conclusions

- In terms of the motifs found...
  - Approximately 30 percent of the motifs were strongly matched to TFs. No one TF emerged as important, but clusters of TFs can be identified.
  - There remains a total of 118 motifs that were not strongly matched to TFs.
  - One difference that emerged between eQTL and null motifs was a slight tendency for eQTL motifs to cluster around the eQTL.
  - Further research...Might begin with the unmatched sequences of the motifs found by both Homer and BaMM