An ensemble of Bayesian networks to predict *in vivo* transcription factor binding sites

Alexandra Essebier [*]   Mikael Bodén [*]

A submission to the DREAM-ENCODE *in vivo* transcription factor binding site prediction challenge.

1 Introduction

1.1 Motivation

Predicting transcription factor (TF) binding sites *in vivo* is an essential advance to better understand regulation in the genome. Though chromatin immunoprecipitation followed by high throughput sequencing (ChIP-seq) has allowed detailed exploration of a handful of TFs across a number of cell types, tissues and conditions; for many TFs, a specific antibody does not exist limiting the application of ChIP-seq. ChIP-seq also has requirements for large numbers of cells to correctly perform the experimental procedure as well as cases where isolating pure cell cultures at such high levels is not possible.

Previous tools and approaches to predict TF binding *in vivo* have explored a wide variety of features that can assist in identifying binding sites. These include evolutionary conservation, DNA shape, chromatin accessibility data, DNase I hypersensitivity footprints, and detailed sequence features [Hesselberth et al., 2009, Ernst et al., 2010, Won et al., 2010, Chen et al., 2010, Pique-Regi et al., 2011, Neph et al., 2012, Arvey et al., 2012, Yip et al., 2012, Rohr et al., 2013, Sherwood et al., Raj et al., 2015, Parra et al., 2016, Liu et al., 2016].

A number of the tools apply statistical approaches to generating predictions including Poisson distributions, Bayesian mixture models, or linear discriminative analysis (LDA) [Sherwood et al., 2014, Rohr et al., 2013, Liu et al., 2016]. A second theme across the existing approaches is the use of machine learning in the form of classifiers to generate predictions for *in vivo* binding. This includes logistic regression, support vector machines (SVMs) and hidden Markov models (HMMs) [Ernst et al., 2010, Arvey et al., 2012, Mathelier and Wasserman, 2013].

Although these statistic and machine learning approaches have provided significant improvements to binding site prediction, their main limitation is the inability to explore the contributing features and their influence on binding outcomes. For example, a SVM is able to classify a site as bound or unbound but cannot provide any information on why it has made that decision. By exploring the feature space around TF binding sites *in vivo* using a Bayesian network approach, not only will predictions be made about binding, but the features that determine the environment for a binding event to occur can be queried and explored. A Bayesian network is able to capture relationships and patterns stored in the feature space making it a powerful tool for better understanding TF binding *in vivo*. This is particularly important when dealing with a complex system like the human regulome.

1.2 Novelty

Although Bayesian mixture models have been applied in the context of TF binding site prediction, Bayesian networks have not. Exploring their efficacy in capturing the feature space and making predictions around binding sites using Bayesian networks is novel work. Further improving the power of the model by building an ensemble of Bayesian networks to generating binding site predictions is also a technique that has not been attempted previously in this context.

[*] School of Chemistry and Molecular Biosciences, The University of Queensland, Australia.

2 Methods

2.1 Data selection

The consortium provided binding data for 109 ChIP-seq data sets across 31 transcription factors and 13 cell types. Each ChIP-seq data set was broken into conserved and relaxed peaks. Conserved peaks were present across multiple replicates and had an irreproducible discovery rate (IDR) score of $<$ %5 [Li et al., 2011], while relaxed peaks were present in multiple replicates but had an IDR score $>$ %5. Each ChIP-seq dataset was represented as 200bp windows at intervals of 50bp across the Human genome using three labels (see Figure 1a). A 200bp window was allocated a label 'U' (unbound) if there was no evidence of any ChIP-seq peak at that location. A label of 'A' (ambiguous) was given to 200bp windows overlapping a relaxed peak or the edges of conserved peaks. 'B' (bound) was assigned to windows overlapping conserved peaks. A set of blacklisted regions which have been shown to contain artificially high signal were excluded from the set of locations across the genome [ENCODE Project Consortium, 2012]. In total, the consortium provided 60,519,747 labels for each ChIP-seq data set.

It was not feasible to explore all available locations due to size, time and computational complexity. A maximum of %0.5 of locations across all ChIP-seq label files were annotated as 'B' indicating many of the windows did not contain sites of interest and would therefore not be informative in identifying patterns associated with TF binding. To reduce the size of the datasets, only locations which fell under the union of all conservative ChIP-seq peaks were explored. This reduced the number of labels to 8,324,886 and raised the maximum occurrence of 'B' to %5. Using the union of all ChIP-seq peaks ensures that individual TF/cell type combinations would contain examples of no binding.

2.2 Feature selection

TF binding events occur at DNA sequence specific locations in the genome and a TF motif, a key feature in identifying TF binding, is generally <30bps. Each location in the genome was represented by a window of 50bps to capture the sequence, genetic and epigenetic environment around each binding event. As binding events are dependent on the cell type specific epigenetic environment as well as the sequence features and motif specific to the TF, for each TF and cell type combination a dataset was generated.

2.2.1 ChIP-seq labels

Four labels were available from the overlapping 200bp windows provided by the consortium for each 50bp location as seen in Figure 1b. In round 1, ChIP-seq data was represented as the count of 'B', from 0-4, occurring above a 50bp window. In round 2, a state was assigned from 0-13 depending on the count of each label (U, A and B) , as shown in Table 1, to represent the binding event.

2.2.2 Sequence

The DNA sequence, including variants, to which a TF is known to bind is recorded as a motif. 29 of the 31 TFs being studied have a DNA motif to which they are known to bind (see Table 2). TAF1 and EP300 do not bind directly to DNA but rather to other TFs and therefore do not have an associated motif. A motif can be represented by a position weight matrix (PWM) which can be used to assign a score to a query sequence quantifying how well the sequence matches the motif. If a 50bp window contains a high scoring match to the motif, this would increase the probability of binding. Therefore, each 50bp window ± motif length, is scanned to identify the max score matching the TF motif of interest.

It is possible that a TF motif cannot contain a certain base pair/position combination in the PWM resulting in a probability of 0.0. The PWM scoring method used to explore query sequences operates in log space and cannot handle values of 0.0 but instead applies a pseudo count of 0.000001. In the case where a query sequence contains the base pair/position combination with a value of 0.0, the resulting score is an extreme negative value and an outlier. It also indicates the query sequence is a very poor match for the motif. In round 1, a threshold was applied and PWM scores lower than -15 were ignored. In round 2, each set of locations was split into 50bp windows which contained a valid sequence, where all base pairs in the query string matched with a non-zero probability in the PWM, and those that do not (invalid sequences).

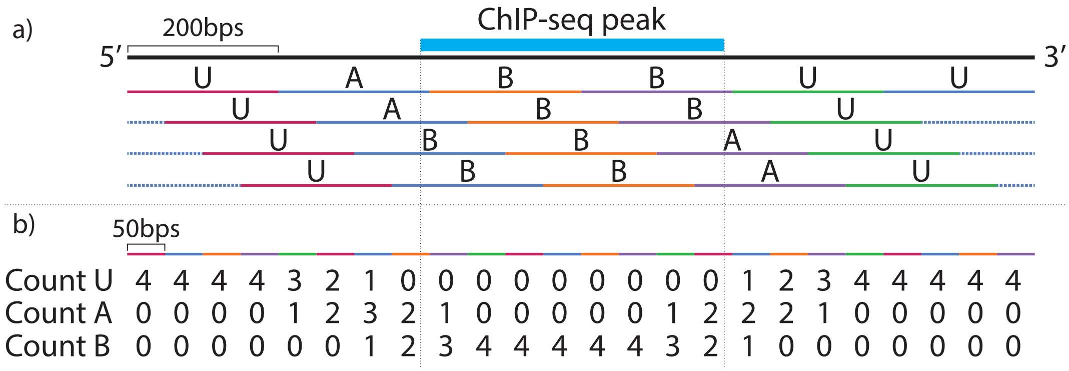
a)
200bps
ChIP-seq peak
5′
3′
U A B B U U
U A B B U
U B B A U
U B B A U
b)
50bps
Count U 4 4 4 4 3 2 1 0 0 0 0 0 0 0 0 0 1 2 3 4 4 4 4 4
Count A 0 0 0 0 1 2 3 2 1 0 0 0 0 0 1 2 2 2 1 0 0 0 0 0
Count B 0 0 0 0 0 0 1 2 3 4 4 4 4 4 3 2 1 0 0 0 0 0 0 0

Figure 1: a) The consortium provided labels (U, A or B) for every 200bp window at 50bp intervals across the Human genome for each ChIP-seq dataset as shown here. U for unbound regions which did not overlap a ChIP-seq peak. A for ambiguous regions which overlapped relaxed peaks (present in multiple replicates but had an IDR score > %5) or to represent low confidence edges of a ChIP-seq peak. B for bound regions which overlap conservative ChIP-seq peaks (IDR score of < %5). b) A 50bp window was deemed more acceptable for exploring TF binding sites requiring transformation of the data. Given the count of each label above a 50bp window, a state was assigned to represent the binding event (see Table 1).

U	A	B	State
0	0	4	0
0	4	0	1
4	0	0	2
0	1	3	3
0	2	2	4
0	3	1	5
1	2	1	6
1	3	0	7
2	2	0	8
2	1	1	9
3	1	0	10
1	1	2	11
1	0	3	12
2	0	2	13

Table 1: Given the count of each label above a 50bp window (see Figure 1b), a state was assigned to represent the binding event leading to 14 possible binding states.

GC content around TF binding motifs has previously been shown to contain differences compared to unbound sequences [Dror et al., 2015]. It has also previously been used to improve predictions of locations bound by a TF[Liu et al., 2016]. The percentage of G and C across each 50bp window was used to represent GC content.

2.2.3 Cell types for training

Not all cell types are equally informative and the relationships between cell types can vary significantly. To explore the relationships in the 13 cell types of interest, clustering based on DNase-seq peaks was performed. Given the locations based on the union of all ChIP-seq labels, a boolean dataset was created describing whether or not the location overlapped a DNase-seq peak. For each location, 13 states were recorded given the 13 cell types and stored in a tab separated file. Using R, a correlation matrix was built based on this dataset and then clustered.

```
mat <- cor(dataSet)
clusters <- hclust(dist(mat))
plot(clusters)
```

For round 1, in an attempt to further reduce data size and complexity, the only cell types that were explored in training were those most closely related to the three final submission cell types: PC-3, induced pluripotent stem cells and liver.

TF	Motif	Cell type/s	Source
ARID3A	MA0151.1	HepG2	[Mathelier et al., 2016]
ATF3	MA0605.1	H1-hESC, HepG2, K562	[Mathelier et al., 2016]
ATF7	MA0834.1	GM12878, HepG2, K562	[Mathelier et al., 2016]
CEBPB	MA0466.2	H1-hESC, HeLa-S3, HepG2, K562	[Mathelier et al., 2016]
CREB1	MA0018.2	GM12878, H1-hESC, HepG2, K562	[Mathelier et al., 2016]
CTCF	MA0139.1	H1-hESC, HeLa-S3, HepG2, K562	[Mathelier et al., 2016]
E2F1	MA0024.3	GM12878, HeLa-S3	[Mathelier et al., 2016]
E2F6	MA0471.1	H1-hESC, HeLa-S3	[Mathelier et al., 2016]
EGR1	MA0162.2	GM12878, H1-hESC	[Mathelier et al., 2016]
EP300	-	GM12878, H1-hESC, HeLa-S3, HepG2, K562	
FOXA1	MA0148.3	HepG2	[Mathelier et al., 2016]
FOXA2	MA0047.2	HepG2	[Mathelier et al., 2016]
GABPA	MA0062.2	GM12878, H1-hESC, HeLa-S3, HepG2	[Mathelier et al., 2016]
HNF4A	MA0114.3	HepG2	[Mathelier et al., 2016]
JUND	MA0491.1	HeLa-S3, HepG2, K562	[Mathelier et al., 2016]
MAFK	MA0496.1	GM12878, H1-hESC, HeLa-S3, HepG2	[Mathelier et al., 2016]
MAX	MA0058.3	GM12878, H1-hESC, HeLa-S3, HepG2	[Mathelier et al., 2016]
MYC	MA0147.2	HeLa-S3, K562	[Mathelier et al., 2016]
NANOG	NANOG_HUMAN.H10MO.A	H1-hESC	[Kulakovskiy et al., 2016]
REST	MA0138.2	H1-hESC, HeLa-S3, HepG2	[Mathelier et al., 2016]
RFX5	MA0510.2	GM12878, HeLa-S3	[Mathelier et al., 2016]
SPI1	MA0080.4	GM12878	[Mathelier et al., 2016]
SRF	MA0083.3	GM12878, H1-hESC, HepG2, K562	[Mathelier et al., 2016]
STAT3	MA0144.2	HeLa-S3	[Mathelier et al., 2016]
TAF1	-	GM12878, H1-hESC, HeLa-S3, K562	
TCF12	MA0521.1	GM12878, H1-hESC	[Mathelier et al., 2016]
TCF7L2	MA0523.1	HeLa-S3	[Mathelier et al., 2016]
TEAD4	MA0809.1	H1-hESC, HepG2, K562	[Mathelier et al., 2016]
YY1	MA0095.2	GM12878, H1-hESC, HepG2	[Mathelier et al., 2016]
ZNF143	MA0088.2	GM12878, H1-hESC, HeLa-S3, HepG2	[Mathelier et al., 2016]

Table 2: Motif sources

Figure 2 shows the results of clustering with the final submission cells in blue boxes and the cells used for training in orange boxes.

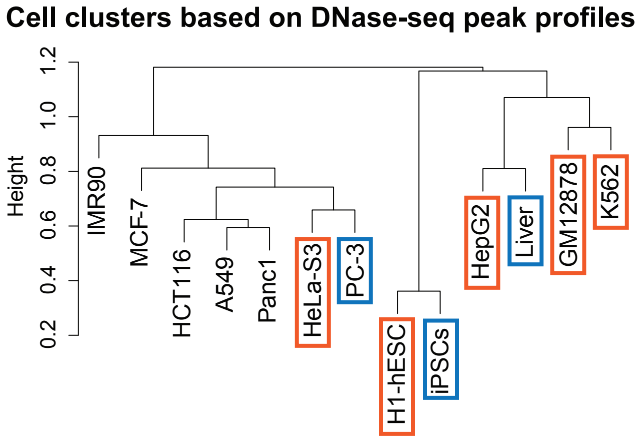

Figure 2: Cell clusters

2.2.4 DNase-seq

For each cell type under consideration, a DNase-seq dataset was provided. Conserved DNase-seq peaks were defined as having an IDR threshold of < %10 while relaxed peaks were identified in pseudo-replicates but had no IDR threshold applied [Li et al., 2011]. DNase-seq peaks represent accessible regions of chromatin and, in most cases, a TF requires the DNA to be accessible to allow binding. The minimum distance was calculated between the edge of each 50bp window and the nearest DNase-seq peak, with overlaps having a distance of zero, to describe the general chromatin environment around a binding event.

Where a protein is bound to DNA, e.g. a TF binding event, the activity of DNase I can be blocked leaving evidence of binding in the form of a DNase hypersensitivity (DHS) footprint. DHS footprints can be found where a TF binding event has occurred making them valuable in predicting binding events [Hesselberth et al., 2009, Neph et al., 2012, Sherwood et al., 2014]. There exist situations where a TF will not bind directly to the DNA but will instead bind to another TF. In this case, knowing that there is a footprint at a location, even if it is not linked to the TF of interest through a motif, is informative [Neph et al., 2012].

PIQ, the tool used to identify DHS footprints, uses motif scanning to identify putative TF binding sites then uses the read distributions from the DNase-seq bam file to identify patterns marking DHS footprints. Footprints were identified by running PIQ on each of the 512 motifs in the JASPAR core 2016 database in every available cell type to identify as many TF binding sites as possible [Mathelier et al., 2016]. The bam files used were generated by merging all available replicate bam files from each cell type (K562, PC-3 and MCF-7 were exceptions due to download issues, see Appendix .1). Each 50bp window can then be assigned a distance to the nearest footprint regardless of the motif it is linked to, as well as the distance to the nearest footprint specific to the current TF.

To better define the environment around a binding site, the number of footprints occurring in a 500bp window is also recorded. A high number of DHS footprints at a location indicates increased binding activity as they are enriched at promoter and 5'UTR regions [Neph et al., 2012]. This can mark active promoter or enhancer regions and boost the likelihood that a TF is bound.

PIQ parameters

The common.r file downloaded from PIQ was not edited. The following commands were used across all JASPAR core 2016 motifs (with the addition of NANOG from HOCOMOCO) for each cell type under consideration.

```
Rscript pwmmatch.exact.r common.r jaspar_CORE_2016_vertebrates.txt \\
motif_number motif_out_dir
Rscript bam2rdata.r DHS_out_file.RData DHS_in_file.bam
Rscript pertf.r common.r motif_out_dir scratch/motif footprints_out_dir \\
DHS_out_file.RData motif_number
```

The thresholded -calls.csv file in both the forward and reverse direction from each set of called footprints for each motif in each cell type contains the start coordinate for each footprint. Using the known motif length and the start coordinate, a bed file was created to describe the location of footprints. A single bed file was created for each cell type to describe the landscape of identified footprints by merging footprints of individual motifs.

2.2.5 Location and gene expression

TFs can bind in promoter regions, within genes (exonic and intronic) or at enhancer regions which can be 500kbps or even up to 1mbps from the gene they are regulating [Corradin et al., 2014, Roy et al., 2015]. A promoter region is defined as ±2000bps around a TSS. Locations outside the promoter region but overlapping a gene are labelled as genic while all remaining locations are labelled as distal. Promoters and enhancers (i.e. distal locations) contain variations in their epigenetic environment and are likely to demonstrate different binding patterns. Not only was each 50bp window labelled, but an absolute distance to the nearest TSS was recorded. Enhancers are known to operate over limited distances therefore, distance can act as a filter for true binding sites.

The expression pattern of a putative target gene can help identify whether a TF binding site is occupied *in vivo*. A 50bp window occurring in a promoter region or within a gene was annotated to the nearest gene. For each cell-type-specific RNA-seq dataset, replicates were merged and the average of the replicate TPM values was assigned to the gene. Gene symbols were used to match expression values to genomic locations of genes using GenCode annotations (v19) [Harrow et al., 2012]. The gene's expression value, as transcripts per million (TPM), was also recorded. When a 50bp window occurs distally, annotating it to a target gene is more challenging. In this case, the window is assigned the highest expression value of all genes occurring in a 500kbp window.

To provide more information about the environment around a 50bp window, the count of genes in a 500kbp was recorded. The human genome has previously been broken down into 3-5 groups of isochores ranging from GC-poor to GC-rich [Mouchiroud et al., 1991, Zoubak et al., 1996]. GC-rich isochores have been shown to have a higher gene density as well as an abundance of open chromatin regions compared to the GC-poor, gene-poor isochores [Mouchiroud et al., 1991, Zoubak et al., 1996, Bernardi, 2005]. While gene deserts fall into GC-poor isochores with general features of inaccessibility they have been shown to contain enhancer regions and therefore TF binding sites [Nobrega et al., 2003, Sotelo et al., 2010].

2.3 Model construction and training

A Bayesian network is composed of a number of vertices associated with variables (1:1). The variables represent a range of features that are either observed or latent, in relation to a 50bp window. These variables can be discrete, meaning that they take on one of a finite number of values, or continuous, meaning that they take on a real value. For each location in our training data set, we assign observed values to these variables, or leave them unspecified when not known.

Generally, the joint probability of all variables, $X_1, \ldots, X_N$, in the Bayesian network is given by

$$P(X_1 = x_1, \ldots, X_N = x_N) = \prod_{i=1}^{N} P(X_i = x_i \mid pa(X_i))$$

where $pa(X_i)$ is the set of parents of the ith variable, as indicated by the acyclic graph formed from directed edges between vertices (parent-to-child).

The parameters of the network define the conditional probabilities associated with each vertex, i.e. the probability of the variable associated with it, conditioned on the variable's parents. In this study they are *learned* from observations in the data sets using standard Expectation-Maximization to cope with the absence of explicit evidence for variables. The structure is fixed to accommodate causal relationships that are evident from the literature.

Some child variables are continuous and are here represented by a Gaussian density, meaning that EM finds a mean and a variance for each possible combination of parent variable assignments. In several cases, the value is first log-transformed to suit a Gaussian density.

Node	Feature	Type	State(s)
DistPeak	Distance to nearest DNase-seq peak	Discrete	0-7: 0, ≤ 50, ≤ 200, ≤ 500, $\leq 1,000$, $\leq 20,000$, $\leq 100,000$, $> 100,000$
DistGFoot	Distance to nearest non-specific DHS footprint	Discrete	
DistSFoot	Distance to nearest DHS footprint matching TF of interest	Discrete	
Location	Location of window relative to nearest gene	Discrete	Promoter, Genic or Distal
MotifScore	Score from PWM scan of window	Real	Not transformed
GCcontent	Proportion of G and C in window	Real	Not transformed
GFCount	Count of DHS footprints in 500bp window	Real	log-transformed
Expression	Expression value in TPM of target gene	Real	log-transformed
Overlaps	Number of genes in 500kbp window	Real	log-transformed
Distance	Distance to nearest TSS	Real	log-transformed

Table 3: A record of the nodes in the Bayesian network, the features they represent and their data type.

Inference of $P(X \mid E = e)$ where X is the (uninstantiated) query variable(s), and $E = e$ is the assigned evidence, is based on the full joint probability. In order to obtain this value, we marginalise over the set of unobserved variables Y, which take values y,

$$P(X \mid E = e) = \eta \sum_{y} P(X, E = e, Y = y)$$

where η is a normalising constant that ensures that conditional probabilities of X's possible values sum to 1. Any variable can be inferred in the Bayesian network. When estimating the ability of the model to predict TF binding, we infer the posterior probability of the variable representing this variability given evidence available for that locus, for instance sequence composition, and number of repeat copies.

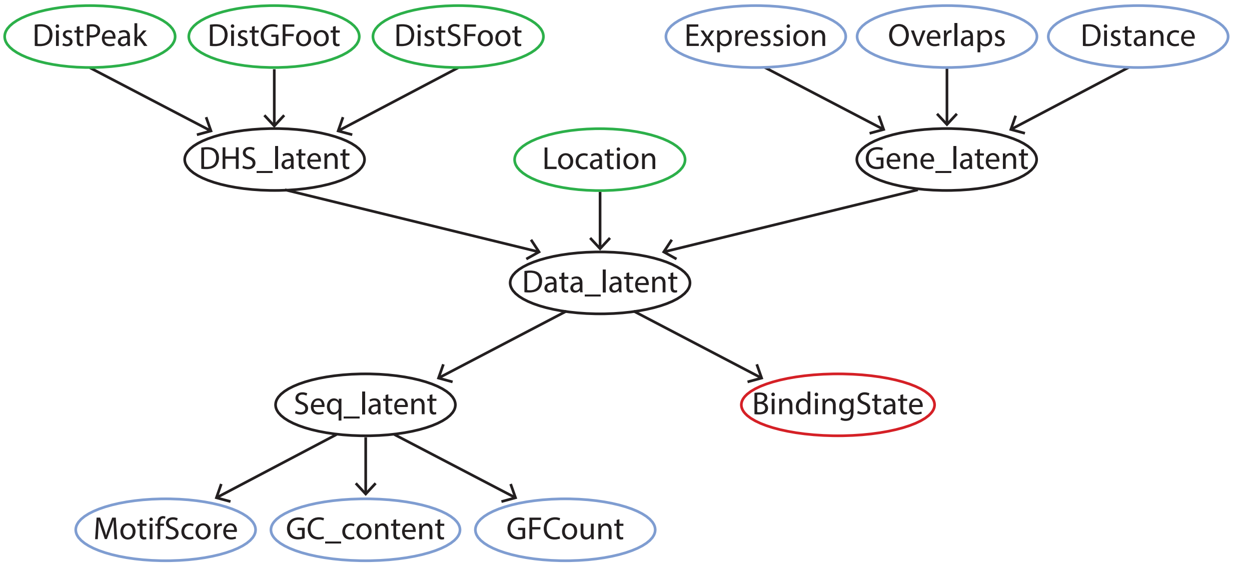

Figure 3: The network structure of the model that will be used to capture patterns and relationships from the data. Nodes are coloured by their input with green being discrete, blue being continuous or real, black as latent (no input) and red representing the node which will capture binding probabilities (also discrete). Nodes are described in detail in Table 3 and in 'Feature selection' above. Arrows represent the node edges and conditional dependency between a parent and child node.

2.4 Ensemble

Combining multiple models can improve overall performance when compared to the performance of individual models. It also allows the data to be broken into smaller pieces with each model learning features and patterns from independent datasets. In round 1, averaging across model predictions was used to improve overall performance. In round 2, as shown by [Garg et al., 2002], a Bayesian network was used to combine predictions from individual Bayesian network models using the network configuration shown in Figure 4. This is achieved using the same theory as reported above for Bayesian network training and querying.

Given a set of training data, each individual model is used to generate a prediction for each data point. The predictions, in this case probabilities from $0-1$, are then fed into the network in Figure 4 with the required number of child nodes to match the number of input models. The ensemble is trained on data where the binding state is known.

The same process is applied to test data, where a prediction is generated from each individual model then fed into the ensemble. In this case, the binding state is unknown, and therefore inferred to obtain a final probability of binding for each test data point.

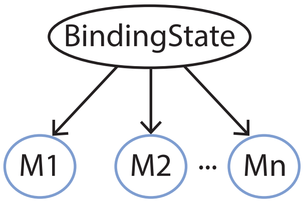

Figure 4: The ensemble network structure.

2.5 Training and testing

In round 1, one million data points were generated for each TF. A minimum of ten networks and training cell type specific datasets were created for each TF. A TF with three training cell types would have twelve models with four from each different cell type while a TF with two training cell types would have a total of ten models with five from each cell type.

In round 2, three million training data points were generated for each TF with equal numbers of each cell type explored (e.g. if a TF has two training cell types, each would be used to generate data for 1.5 of the 3 million data points). Each set of data is then broken into valid and invalid given the results of the PWM scan of the sequence in the 50bp window. A total of ten individual datasets were created from the valid and invalid sequences with number and size determined by the split of sequences (e.g. where the majority of data points were valid, eight datasets would be used to capture the valid sequences while only two would represent the invalid). A valid or invalid model is used to train the broken down data and the resulting models form the input into the ensemble.

3 Running the code

3.1 Data generation

The features defined above were extracted using a Python script (gen_data_9_py) which iterated over each 50bp location, identified distances to DNase-seq peaks, DHS sites, genes and subsequent features based on these distances (e.g. counts for footprints and genes, and expression patterns). The script also calculated PWM scores and assigned a binding state to each location. There are four versions of this script: two for training data which does include a binding state and two for testing data where this information is not available. Within each, one script is available for TFs containing a motif while a second version (*_nm.py) is adjusted to handle TFs with no motif.

To run the training data generation scripts, the following files are required:

- DNASE.cell.conservative.narrowPeak - as provided by consortium

- Bed file containing all footprints for cell - as described in 2.2.4
- Bed file containing the footprints for the TF of interest and cell type of interest - as described in 2.2.4
- File containing locations of genes and associated expression values in TPM *
- PWM file representing motif
- ChIP-seq labels for single TF-cell type combination (original 200bp format) **

* The columns in this file are: chromosome, start, end, strand, Ensembl ID, gene symbol, artifact, expression value, artifact. The two artifact columns must exist but are no longer utilised and can be set to all 1's.

** The columns in this file are: chromosome, start, end, binding state (U, A or B).

The DNASE peaks, footprint files and label file are all indexed using Tabix and must be bgzip compressed and indexed using tabix -p bed format [Li, 2011]. The testing scripts do not require the ChIP-seq label file while the scripts for TFs with no motifs do not require the footprints for a specific TF or the PWM file.

3.2 BNkit

BNkit is currently provided to the organisers as jar files designed to do different tasks relating to training and testing existing networks and datasets. This Java project is not user ready and exists as an in-house tool. It is challenging to provide instructions which will allow the organisers to run all required steps, in particularly, generation of networks.

- training.jar will train individual Bayesian networks given a network and dataset and outputs a trained network. This is used for training all individual models that will be built into ensembles.
    - **Input**: Bayesian network (example provided: dream_net_CTCF_valid_0.out)
    - **Input**: Data set generated by gen_data_9_training.py or gen_data_9_training_nm.py
    - **Output**: Bayesian network (example provided: dream_net_CTCF_valid_0.out_trained.new)
- ensemble.jar will identify models that form part of the ensemble, use training data to generate predictions across all models, use predictions and known binding state to train ensemble, generate predictions across all models for testing data, then infer binding state using trained ensemble and the predictions from testing data. Much of this is hardcoded and requires specific file structure. It is not in a state that would be considered 'user friendly'. I am happy to discuss in more detail but instructions at this stage are not easy to put into words.

4 Discussion

Insights gained during challenge

Many parameters, features and variables were explored, tested, altered, discarded and added throughout the course of this challenge. In every case, it was in an attempt to provide the model more information that would assist in predicting true binding sites. To report all would be excessive. What appeared to be most informative was the combination of the three features describing the DHS peak and footprint data. A significant improvement in performance was gained when these features were switched from boolean (overlapping window of interest or not) to a distance measure.

Two different Bayesian network approaches were applied through the two rounds of this challenge. Unfortunately, time ran out to submit during the final submission queue of the first round. When exploring the leaderboard submissions that were completed using a Bayesian network approach in round 1, no single situation or set of variables could be isolated that explained why the model performed well on some TFs and poorly on others. All features were explored in an attempt to correlate certain states that would influence performance in line with the auPRC results obtained from the submitted data.

The complexity and layers of detailed information stored in the genetic and epigenetic environments of TF binding sites made determining changes in performance difficult. Some theories include: frequency of TF binding to sites with no

TF	Proportion valid	Proportion invalid
ARID3A	0.013	0.011
ATF3	0.015	0.012
ATF7	0.026	0.000
CEBPB	0.023	0.000
CREB1	0.015	0.010
CTCF	0.023	0.000
E2F1	0.007	0.000
E2F6	0.019	0.010
EGR1	0.007	0.004
FOXA1	0.029	0.019
FOXA2	0.035	0.000
GABPA	0.004	0.000
HNF4A	0.011	0.000
JUND	0.082	0.016
MAFK	0.012	0.008
MAX	0.023	0.000
MYC	0.021	0.012
NANOG	0.003	0.000
REST	0.006	0.000
RFX5	0.008	0.000
SPI1	0.030	0.000
SRF	0.003	0.001
STAT3	0.010	0.008
TCF12	0.013	0.006
TCF7L2	0.015	0.015
TEAD4	0.013	0.000
YY1	0.013	0.027
ZNF143	0.015	0.010

Table 4: Given a sample of three million data points across a range of cell types for each TF, broken into locations containing valid or invalid sequence, what is the proportion of locations mapped to a true binding event? TFs in bold contained fewer than 15 examples of binding at locations with invalid sequence.

specific motif, information content of the motif, epigenetic environment preferred by a TF and cell type specificity of a TF. Many attempts to improve on the model were made for round 2 but no submissions were made to the leaderboard round so no results are available to gauge performance improvements.

The main limitation of a Bayesian network approach is its generative nature. In trying to recreate the observed data, it can struggle when presented sets of variables that may not necessarily have been observed together previously. An occurrence that turned out to be quite frequent when dealing with data on the scale presented in this challenge. As described above, the training data was broken into valid and invalid sequences. Very few of the invalid models were successfully trained using the invalid sequence data. It is likely that the invalid sequence data was sparse in terms of the relationships and patterns that it captured. Of interest was the fact that many TFs did not contain any instances of binding at the locations which did not contain valid sequence (see Table 4).

5 DREAM Results

The results included here are preliminary and relate to observations made when participating in the DREAM challenge rather than outcomes. The focus in this section is identification of areas where improvements could be made.

One method of exploring the relationships modelled by a Bayesian network is to create a test dataset where different features are specified. For example, one data point may only contain a motif score as input while another may only look at DHS peak distance. From the controlled test data, a number of predictions are made to observe which features or feature combinations resulted in the highest predictions. Such tests were performed on models trained using different TFs and cell types generated for round 1.

Different TF and cell type model combinations led to vastly different outcomes in binding probability across all features tested. For example, a model trained on CTCF data in cell type A549 would assign a higher probability of binding than a model trained in cell type H1-hESC when the input data was a distance of '0' to all three DHS features. One general observation across all models was that including more features led to better performance when the features were set to favour a binding event given prior biological knowledge.

5.1 Binding is dependent on sequence

A higher motif score generally led to a higher probability of binding across all tested models however, a CTCF model trained on H1-hESC data assigned a probability of 0.9 to the highest motif, a REST-H1-hESC model predicted 0.78 probability and an E2F6-HeLa model assigned probabilies as low as 0.03. This is an example of the variation of features observed across TFs and cell types as well as an example of the challenges faced when attempted to extract biological relevant patterns.

5.2 Chromatin accessibility improves model performance

A significant improvement in performance was gained when the three features describing the DHS peak and footprint data were switched from boolean (overlapping window of interest or not) to a distance measure. This indicates that the wider epigenetic environment around a binding site is more informative than the environment immediately over a binding site.

5.3 Cell-type-specificity influences model performance

It has already been noted that models trained on different cell types exhibit different outcomes. It was also observed that some cell type models were able to make predictions across cell types with comparable accuracy while others could not. For example, H1-hESC models can make predictions in most other cell types while A549 models cannot do so and maintain accuracy. This is most likely due to cell-type-specific patterns which influence TF binding and result in the model identifying different features as associated with increased probability of binding.

In round 2, models were trained on data from multiple cell types rather than individual. This is suspected to be the main cause for the decrease in performance observed in results from round 2. Merging cell types led to poorer performance, testing cell-type specific model on other cell type led to generally poorer result. Some cell types are more informative than others.

5.4 Bayesian networks are sensitive to large, sparse datasets

The main limitation of a Bayesian network approach is its generative nature. In trying to recreate the observed data, it can struggle when presented sets of variables that may not necessarily have been observed together previously. An occurrence that turned out to be quite frequent when dealing with data on the scale presented in this challenge. As described above, the training data was broken into valid and invalid sequences. Very few of the invalid models were successfully trained using the invalid sequence data. It is likely that the invalid sequence data was sparse in terms of the relationships and patterns that it captured.

5.5 Bayesian network ensemble provides moderate improvement to performance

The ensemble model always showed better performance than the worst performing model in the set. It did not, however, provide significant improvements to prediction accuracy and, at best, was a moderate improvement over averaging predictions from the contributing models.

5.6 Round 1 performance

The performance of a number of predictions submitted to the first round of the challenge compared to the best performing team's outcome can be seen in Figure 5.

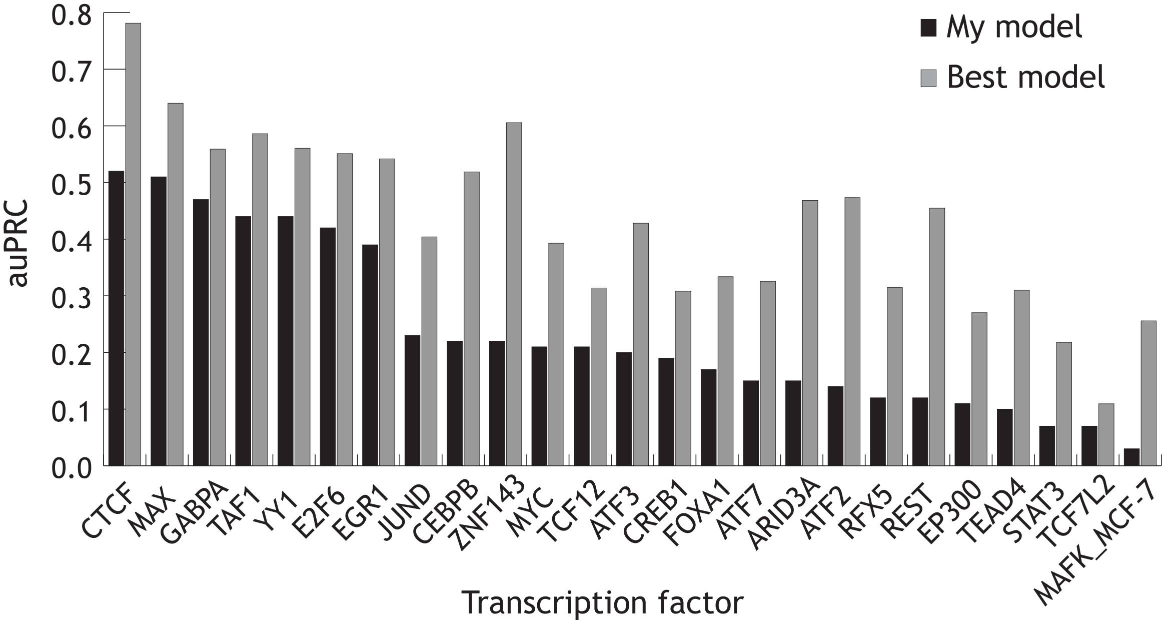

Figure 5: The auPRC scores across TFs explored by the DREAM challenge for my model compared to the best performing model.

5.7 Round 2 performance

At this stage, final performance outcomes for round 2 have not been released. There is evidence that the final models submitted here performed worse than those submitted for round 1. However, the main goal was to be able to generate the datasets and predictions that were required for the final submission. This was an achievement unto itself as a submission of 60,519,747 predictions for each of 13 datasets was made requiring significant compute time, coordination of code and elements of parallel processing.

6 Author's Statement

AE was responsible for construction of code, generation of data, feature identification, model design and submission to the challenge. MB constructed the Bayesian network code and was essential in idea development as well as providing general support and guidance. AE and MB both contributed to writing this document.

7 Acknowledgements

This work was supported through an Australian Government Research Training Program Scholarship

Cell type	DNase-seq bam files
K562	DNASE.K562.biorep2.techrep12.bam DNASE.K562.biorep2.techrep18.bam DNASE.K562.biorep2.techrep3.bam DNASE.K562.biorep2.techrep14.bam DNASE.K562.biorep2.techrep1.bam DNASE.K562.biorep2.techrep5.bam DNASE.K562.biorep2.techrep16.bam DNASE.K562.biorep2.techrep2.bam
MCF-7	DNASE.MCF-7.biorep1.techrep1.bam DNASE.MCF-7.biorep1.techrep2.bam DNASE.MCF-7.biorep1.techrep4.bam DNASE.MCF-7.biorep1.techrep5.bam
PC-3	DNASE.PC-3.biorep1.techrep2.bam DNASE.PC-3.biorep1.techrep3.bam

Table 5: Where download issues were encountered, not all bam files were merged into a single bam file to be passed to PIQ. Recorded here are the three cell types where problems occurred and the bam files that were successfully merged and used in the analysis.

Appendices

.1 DNase-seq downloads

See Table 5.

References

[Arvey et al., 2012] Arvey, A., Agius, P., Noble, W. S. and Leslie, C. (2012). Sequence and chromatin determinants of cell-type–specific transcription factor binding. Genome research *22*, 1723–1734.

[Bernardi, 2005] Bernardi, G. (2005). Structural and evolutionary genomics: natural selection in genome evolution, vol. 37,. Elsevier.

[Chen et al., 2010] Chen, X., Hoffman, M. M., Bilmes, J. A., Hesselberth, J. R. and Noble, W. S. (2010). A dynamic Bayesian network for identifying protein-binding footprints from single molecule-based sequencing data. Bioinformatics *26*, i334–i342.

[Corradin et al., 2014] Corradin, O., Saiakhova, A., Akhtar-Zaidi, B., Myeroff, L., Willis, J., Cowper-Sal, R., Lupien, M., Markowitz, S., Scacheri, P. C. et al. (2014). Combinatorial effects of multiple enhancer variants in linkage disequilibrium dictate levels of gene expression to confer susceptibility to common traits. Genome research *24*, 1–13.

[Dror et al., 2015] Dror, I., Golan, T., Levy, C., Rohs, R. and Mandel-Gutfreund, Y. (2015). A widespread role of the motif environment in transcription factor binding across diverse protein families. Genome research *25*, 1268–1280.

[ENCODE Project Consortium, 2012] ENCODE Project Consortium (2012). An integrated encyclopedia of DNA elements in the human genome. Nature *489*, 57–74.

[Ernst et al., 2010] Ernst, J., Plasterer, H. L., Simon, I. and Bar-Joseph, Z. (2010). Integrating multiple evidence sources to predict transcription factor binding in the human genome. Genome research *20*, 526–536.

[Garg et al., 2002] Garg, A., Pavlovic, V. and Huang, T. S. (2002). Bayesian networks as ensemble of classifiers. In Pattern Recognition, 2002. Proceedings. 16th International Conference on vol. 2, pp. 779–784, IEEE IEEE.

[Harrow et al., 2012] Harrow, J., Frankish, A., Gonzalez, J. M., Tapanari, E., Diekhans, M., Kokocinski, F., Aken, B. L., Barrell, D., Zadissa, A., Searle, S. et al. (2012). GENCODE: the reference human genome annotation for The ENCODE Project. Genome research *22*, 1760–1774.

[Hesselberth et al., 2009] Hesselberth, J. R., Chen, X., Zhang, Z., Sabo, P. J., Sandstrom, R., Reynolds, A. P., Thurman, R. E., Neph, S., Kuehn, M. S., Noble, W. S. et al. (2009). Global mapping of protein-DNA interactions in vivo by digital genomic footprinting. Nature methods *6*, 283–289.

[Kulakovskiy et al., 2016] Kulakovskiy, I. V., Vorontsov, I. E., Yevshin, I. S., Soboleva, A. V., Kasianov, A. S., Ashoor, H., Ba-Alawi, W., Bajic, V. B., Medvedeva, Y. A., Kolpakov, F. A. et al. (2016). HOCOMOCO: expansion and enhancement of the collection of transcription factor binding sites models. Nucleic acids research *44*, D116–D125.

[Li, 2011] Li, H. (2011). Tabix: fast retrieval of sequence features from generic TAB-delimited files. Bioinformatics *27*, 718–719.

[Li et al., 2011] Li, Q., Brown, J. B., Huang, H. and Bickel, P. J. (2011). Measuring reproducibility of high-throughput experiments. The annals of applied statistics *5*, 1752–1779.

[Liu et al., 2016] Liu, L., Zhao, W. and Zhou, X. (2016). Modeling co-occupancy of transcription factors using chromatin features. Nucleic acids research *44*, e49–e49.

[Mathelier et al., 2016] Mathelier, A., Fornes, O., Arenillas, D. J., Chen, C.-y., Denay, G., Lee, J., Shi, W., Shyr, C., Tan, G., Worsley-Hunt, R. et al. (2016). JASPAR 2016: a major expansion and update of the open-access database of transcription factor binding profiles. Nucleic acids research *44*, D110–D115.

[Mathelier and Wasserman, 2013] Mathelier, A. and Wasserman, W. W. (2013). The next generation of transcription factor binding site prediction. PLoS Comput Biol *9*, e1003214.

[Mouchiroud et al., 1991] Mouchiroud, D., D'Onofrio, G., Aïssani, B., Macaya, G., Gautier, C. and Bernardi, G. (1991). The distribution of genes in the human genome. Gene *100*, 181–187.

[Neph et al., 2012] Neph, S., Vierstra, J., Stergachis, A. B., Reynolds, A. P., Haugen, E., Vernot, B., Thurman, R. E., John, S., Sandstrom, R., Johnson, A. K. et al. (2012). An expansive human regulatory lexicon encoded in transcription factor footprints. Nature *489*, 83–90.

[Nobrega et al., 2003] Nobrega, M. A., Ovcharenko, I., Afzal, V. and Rubin, E. M. (2003). Scanning human gene deserts for long-range enhancers. Science *302*, 413–413.

[Parra et al., 2016] Parra, R. G., Rohr, C. O., Koile, D., Perez-Castro, C. and Yankilevich, P. (2016). INSECT 2.0: a web-server for genome-wide cis-regulatory modules prediction. Bioinformatics *32*, 1229–1231.

[Pique-Regi et al., 2011] Pique-Regi, R., Degner, J. F., Pai, A. A., Gaffney, D. J., Gilad, Y. and Pritchard, J. K. (2011). Accurate inference of transcription factor binding from DNA sequence and chromatin accessibility data. Genome research *21*, 447–455.

[Raj et al., 2015] Raj, A., Shim, H., Gilad, Y., Pritchard, J. K. and Stephens, M. (2015). msCentipede: modeling heterogeneity across genomic sites and replicates improves accuracy in the inference of transcription factor binding. PloS one *10*, e0138030.

[Rohr et al., 2013] Rohr, C. O., Parra, R. G., Yankilevich, P. and Perez-Castro, C. (2013). INSECT: IN-silico SEarch for Co-occurring Transcription factors. Bioinformatics *29*, 2852–2858.

[Roy et al., 2015] Roy, S., Siahpirani, A. F., Chasman, D., Knaack, S., Ay, F., Stewart, R., Wilson, M. and Sridharan, R. (2015). A predictive modeling approach for cell line-specific long-range regulatory interactions. Nucleic acids research *43*, 8694–8712.

[Sherwood et al., 2014] Sherwood, R. I., Hashimoto, T., O'Donnell, C. W., Lewis, S., Barkal, A. A., van Hoff, J. P., Karun, V., Jaakkola, T. and Gifford, D. K. (2014). Discovery of directional and nondirectional pioneer transcription factors by modeling DNase profile magnitude and shape. Nature biotechnology *32*, 171–178.

[Sotelo et al., 2010] Sotelo, J., Esposito, D., Duhagon, M. A., Banfield, K., Mehalko, J., Liao, H., Stephens, R. M., Harris, T. J., Munroe, D. J. and Wu, X. (2010). Long-range enhancers on 8q24 regulate c-Myc. Proceedings of the National Academy of Sciences *107*, 3001–3005.

[Won et al., 2010] Won, K.-J., Ren, B. and Wang, W. (2010). Genome-wide prediction of transcription factor binding sites using an integrated model. Genome biology *11*, R7.

[Yip et al., 2012] Yip, K. Y., Cheng, C., Bhardwaj, N., Brown, J. B., Leng, J., Kundaje, A., Rozowsky, J., Birney, E., Bickel, P., Snyder, M. et al. (2012). Classification of human genomic regions based on experimentally determined binding sites of more than 100 transcription-related factors. Genome biology *13*, 1.

[Zoubak et al., 1996] Zoubak, S., Clay, O. and Bernardi, G. (1996). The gene distribution of the human genome. Gene *174*, 95–102.