Supplementary Information

Motivation for Synergistic Chromatin Models (SCMs)

Accessible chromatin has been found to correlate strongly with active gene regulatory regions. Genome-scale DNase I hypersensitivity (DNase-seq) analysis has confirmed that nearly all enhancers and promoters that can be defined through independent methods occur in accessible chromatin (Thurman et al., 2012). Additionally, transcription factor (TF) binding sites are almost always associated with increased accessibility (Boyle et al., 2008; Hesselberth et al., 2009; Neph et al., 2012). Thus, active gene regulatory regions populated by TFs and transcriptional machinery occur in accessible chromatin.

There is evidence that the accessibility of chromatin helps to determine the activity of genomic regions. In addition to providing structure, nucleosomes inhibit the gene regulatory function of DNA through tightly winding the DNA and thus competing with TFs and the transcriptional machinery (Richmond & Davey, 2003; Zaret & Carroll, 2011). Some TFs, known as settler TFs, depend directly on prior chromatin accessibility in their binding decisions (Sherwood et al., 2014). Chromatin remodeling enzymes that decrease nucleosome-DNA contact are required for proper gene regulation (Ho & Crabtree, 2010), and an array of histone modifications and DNA methylation have been reported to regulate nucleosome-DNA contact (Ernst & Kellis, 2013; Heintzman et al., 2009; D. Lee, Karchin, & Beer, 2011; Meissner et al., 2008; Ram et al., 2011; Zhou, Goren, & Bernstein, 2011). Altogether, chromatin accessibility is a tightly regulated genomic feature that not only correlates with gene regulatory activity but also helps to govern the gene regulatory status of a cell through controlling which genomic regions are available to TFs and transcriptional machinery.

Our aim is to establish how DNA encodes chromatin accessibility. DNA, as the chief source of heritable information in a cell, almost surely does encode chromatin accessibility; however, it has thus far not been possible to predict cellular chromatin accessibility accurately from genomic DNA sequence. A class of pioneer TFs has been shown to open chromatin at previously closed sites (Gualdi et al., 1996; Sherwood et al., 2014; Soufi, Donahue, & Zaret, 2012; Zaret & Carroll, 2011), providing a paradigm that TF-DNA interactions can directly modulate chromatin accessibility. Promoters have characteristic patterns of chromatin accessibility (Thurman et al., 2012), and several canonical promoter-enriched sequences have been identified (Frith et al., 2008; Lenhard, Sandelin, & Carninci, 2012; Sandelin et al., 2007; Valen & Sandelin, 2011). Additionally, CpG sequences have been shown to signal chromatin accessibility at some promoter sequences (Thomson et al., 2010), and GC content is known to affect chromatin state(Wang et al., 2012; White, Myers, Corbo, & Cohen, 2013). A code based on periodic spacing of dinucleotide DNA sequence motifs has also been revealed to predict nucleosome positions in some instances, although its genome-wide accuracy is only modest (Hughes & Rando, 2014; Kaplan et al., 2009; Peckham et al., 2007; Segal et al., 2006).

However, no current model based on these codes can explain genome-wide chromatin accessibility. Pioneer TF binding has only been shown to causally influence accessibility at a small number of genomic loci, and pioneer TFs do not bind to every instance of their binding motif in the genome as might be expected by their imperviousness to prior chromatin state(Sherwood et al., 2014),indicating that a code positing chromatin opening at pioneer TF motifs would be replete with false positives. Promoter motifs and CpG islands are also highly degenerate(Siebert & Söding, 2014), thus a code positing accessibility at every instance of these motifs would have poor predictive accuracy. And the nucleosome positioning code performs most poorly at predicting nucleosome-depleted regions that typify accessible chromatin (Hughes & Rando, 2014; Kaplan et al., 2009; Peckham et al., 2007; Segal et al., 2006). Therefore, prior research has not established how chromatin accessibility is encoded.

In this work, we generate a computational algorithm, the Synergistic Chromatin Model (SCM), that uses machine learning techniques to predict genome-wide chromatin accessibility. The goal of SCMs is to provide insight into the mechanisms that underlie chromatin accessibility. SCMs use logic operations based on assumptions about how cells encode chromatin accessibility to generate predictions of the chromatin accessibility of every DNA base in the genome. For this reason, we refer to a DNA logic underlying chromatin accessibility. We posit that a model that is accurate enough to predict accessibility under a wide array of natural and artificial conditions will improve our understanding of the actual logic used by cells.

Because the inputs to chromatin accessibility may not all be known, we have tried to minimize bias in our approach to uncovering its logic. Nonetheless, we do make several assumptions that we outline below:

1. DNA encodes chromatin accessibility. This is the foundational hypothesis of this work.

2. DNase-seq data reflects chromatin accessibility. It is known that the DNase I enzyme has sequence preferences in its cutting(Koohy, Down, & Hubbard, 2013) which can lead to erroneous conclusions about chromatin state (He et al., 2013). We do not directly correct for such enzyme preferences, so these may be incorporated in the model. However, we have been careful to compare results against naked DNase I digestion as well as focus on DNase I hypersensitive sites. Decades of work have shown that DNase-I hypersensitivity analysis does reflect true biological states, and in fact all enhancers and promoters defined through separate methods occur in DNase-I hypersensitive regions (Thurman et al., 2012). Thus, we believe this to be a sound assumption.

3. The DNA “code words” encoding chromatin accessibility can be represented as k-mers 8 bp or smaller. Most DNA coding elements including the majority of TF binding motifs, nucleosome positioning signals, splicing elements(Barash et al., 2010), and codons are short stretches of sequence. Our method can also use bridging k-mers to piece together longer code words provided they occur at short and stereotyped distances (see Supplementary Fig. 1). The k ≤ 8 bp cutoff is a technical compromise to enable manageable algorithm complexity, yet it is reasonable to assume that most of the coding information in the genome can be learned by 8 bp k-mers. While TFs act at motifs, or thermodynamically related collections of k-mers, modeling motifs as k-mers are statistically equivalent. Once the SCM has been learned, these k-mers can be constructed into motifs to aid biological interpretation.

4. K-mers affect chromatin accessibility locally, within +/- 1 kb from their occurrence. Our previous work has shown that pioneer TFs alter chromatin state within 1 kb of their occurrence (Sherwood et al., 2014), and Ctcf, the most powerful chromatin-shaping TF known, also affects accessibility within 1 kb (Boyle et al., 2011). There are known cases in which chromatin states spread over long distances (Hathaway et al., 2012; J. T. Lee & Bartolomei, 2013), most notably in the inactivation of the X-chromosome; however, these are likely to be exceptional. Three-dimensional chromatin interactions have also been well documented(Dostie et al., 2006; Fullwood et al., 2009; Lieberman-Aiden et al., 2009; Simonis et al., 2006), yet the specificity of these interactions makes it likely that they are governed through local DNA sequences and thus could be modeled through a local logic.

5. A small number of k-mers determine chromatin accessibility. There are around 2,000 transcription factors in the genome, and the vast majority do not play roles in chromatin accessibility. Proteins need not be the only readers of the accessibility logic, yet it seems

reasonable to assume that the code words for accessibility number in the hundreds or thousands. Transcription factors act through thermodynamically related collections of k-mers, meaning that there are more active k-mers than there are TFs. Nonetheless, it is still safe to assume that a minority of the 87,380 $k \leq 8$ k-mers play roles in chromatin accessibility.

6. A particular k-mer produces the same effect on chromatin accessibility wherever it occurs. This assumption implies a mechanistically simple logic in which the effectors, be they TFs or thermodynamic properties of the DNA itself, act in a stereotyped way at every occurrence genome-wide. This assumption does not allow for conditional TF interactions(Mullen et al., 2011; Trompouki et al., 2011) (i.e. a pioneer TF only opens chromatin when adjacent to a specific cofactor k-mer), as these would imply that k-mer effects are dependent on surrounding k-mers. Similarly, a logic in which TF access to DNA is dominantly blocked by the surrounding chromatin state, as in X-chromosome heterochromatin spreading, is inconsistent with this assumption. However, conditional models are harder to learn computationally, as there are only a small number of examples of each k-mer with each potential conditional interacting partner. Thus, we have chosen to gauge the accuracy of a model that excludes the complications of conditionality in an attempt to determine whether the chromatin accessibility logic predominantly relies upon unconditional logic.

7. K-mer effects on chromatin accessibility non-specifically synergize such that the chromatin accessibility at any DNA base is the multiplicative product of the effects of all nearby chromatin accessibility-affecting k-mers. This assumption follows from the apparent non-linearity of genomic functionalization. DNase-seq data and ChIP-seq data both reveal a small number of genomic loci with strong activity (DHS, ChIP peaks) surrounded by a vast majority of genomic space with no activity above background. This all-or-nothing architecture is more consistent with a non-linear than a linear underlying logic. Our assumption of non-specific synergy could be explained biologically if TFs influence chromatin accessibility by acting synergisticly to displace nucleosomes as has been proposed previously(Mirny, 2010). Our model does not take into account specific cofactor interactions that might enhance synergy for the same reason above that conditionality is substantially more difficult to learn.

Using the set of assumptions outlined above, we have designed SCMs to learn the logic underlying chromatin accessibility. But how do we determine whether our SCMs are good representations of the underlying logic? No prior genome-wide models of chromatin accessibility have been published, so there is no benchmark of accuracy for comparison. We have chosen a multi-part logical framework to gauge the accuracy of our model:

1. Accuracy at predicting held-out DNase-seq data. We always separate chromosomes used in learning and in testing to avoid overfitting that could occur if a SCM “remembers” specific stretches of DNA sequence. SCMs are tested over a range of cell types and DNase-seq protocols to gauge how consistent their performance is. These tests lead to a correlation coefficient of how well the SCM predicts genome-wide variance in DNase-seq signal. To evaluate the meaning of this correlation coefficient, variant models that alter the SCM’s assumptions are compared to determine whether the SCM’s parameters are improving predictive accuracy. Thus, if SCM predicts chromatin accessibility well, and its accuracy is diminished if parameters are changed, then the included parameters may well reflect true aspects of the chromatin accessibility logic.

2. Accuracy at predicting held-out ATAC-seq data when a SCM model is trained on ether ATAC-seq or DNase-seq data. Since ATAC-seq data is an alternate assay for chromatin accessibility, the ability to predict this data type as well as do cross-prediction reflects the ability of the model to accurately capture accessibility information.

3. Accuracy at predicting chromatin accessibility over a wide variety of sequence types. Using genomic data alone to gauge SCM accuracy has several limitations. (i) Genomic regions can be copied across chromosomes in certain instances, so held-out chromosomes may resemble the chromosomes used in learning. (ii) Evolutionary selection may have over-specified regions of accessible chromatin, as these are often vital gene regulatory regions. Adding redundancy to accessible regions would mitigate against deleterious gene regulatory consequences of a single mutation, and mutations that add accessibility-promoting motifs in inaccessible chromatin might be selected against. Thus, a SCM that accurately predicts genomic accessibility may be learning redundancies that would not enable generalization to any DNA sequence. To gauge true accuracy of a SCM, we have devised a method, SLOT, that enables us to test DNase-I hypersensitivity of a large number of arbitrarily designed DNA sequences in any defined genomic context. By testing SCM accuracy on a wide variety of sequences that bear no resemblance to the genomic sequences used in training, we can more accurately gauge the generalized accuracy of SCMs. Thus, if the SCM accurately predicts sequence-dependent chromatin accessibility in a controlled context over a wide range of sequences that bear no resemblance to the training data, then we can conclude that the SCM is modeling the actual logic of chromatin accessibility.

4. Ability to make additional predictions about the biology underlying chromatin accessibility. A good model should not only accurately model data but should yield insights into the underlying biological paradigms. We have chosen to follow up one specific prediction, that genome-wide binding patterns of the pioneer TF Nrf1 can be predicted using the same synergistic logic governing chromatin accessibility, because it is one of the more surprising implications of the SCM. However, with time, we expect our group and others to test other SCM predictions, which will either lend more credence to the model of chromatin accessibility underlying the SCM or may identify model deficiencies that prompt model refinements.

Interpreting the parameters of the SCM should pave the way for an integrated understanding of the cellular systems that have evolved to regulate chromatin accessibility, leading to specific predictions about how to alter accessibility through altering the protein and DNA components of this system.

However, we acknowledge that SCMs as currently formulated cannot be fully accurate representations of underlying biology. The functionalization of DNA into chromatin through the action of sequence-specific regulators is probabilistic and dynamic, whereas our current models are deterministic and static. Thus, our models can be understood as statements of the equilibrium tendencies of the system governing chromatin accessibility. How chromatin accessibility is buffered to maintain consistent function in spite of stochastic changes in conditions and how accessibility is dynamically regulated to enable changes in cellular function are fascinating questions ripe for future modeling efforts.

Distinction from discriminative motif discovery

A class of methods with similar methodological ideas to SCMs is discriminative motif discovery, which seeks to identify the k-mer sequences that constitute a sequence motif for an underlying transcription factor (or functional element). In this approach the user identifies a set of regions of interest and uses a discriminative motif finder to construct a model that can distinguish these regions from background based upon k-mer frequencies. While our approach is similar in that it uses short k-mer sequences as a underlying predictor, it is quite distinct in goals.

We also note that the SCM is also distinct from variant prioritization techniques which seek to discover functionally relevant bases in the genome. The SCM seeks to simply model the relationship between sequence and high-thoughtput sequencing reads without claims about the underlying causal mechanism or phenotype.

- **Binary vs Quantitative:** Discriminative motif discovery has focused upon binary features such as regions with transcription factor binding. While discriminative motif finding is very well suited for detection of motifs underneath a ChIP-seq peak, our goals have been to ask whether DNase-seq can be quantified in its entirety from a spatially synergistic sequence model. Much of our results rely on quantitative measurements of DNase-seq.
- **Understanding spatial effects:** Our goals were to understand the spatial effects and interactions involved in chromatin accessibility rather than just purely predicting DNase-seq data. Existing methods for discriminative motif discovery have treated each region as an exchangeable bag, discarding relative positional information of k-mers. While this is suitable for motif detection, we wanted to understand if spatial interactions among k-mers would be able to predict DNase-seq. For example in Figure 4 we show that the parameters learned by our model closely match the DNase-seq footprints observed for transcription factor binding motifs in other datasets. Our model is not only designed simply as a way to predict; it is meant as a computational realization of our current understanding of chromatin accessibility.
- **Genome-wide vs focus on function:** One final goal of our model was to remove the uncertainty associated with selecting a functional region or parameter. The SCM can be run without any parameter tuning. It takes an aligned set of sequences and outputs k-mer profiles. The reason for this approach is to minimize the biased selection of functional regions. Unlike ChIP-seq, DNase-seq signals can be quite broad and vary in strength, leading to questions of whether the final results depend on the peak selection methods used.

Since it is possible to use SCM to perform classification of a set of DNase-seq peak regions, we have performed a set of comparisons against a state-of-the-art discriminative motif finders to show that even on a discriminative motif-detection task, SCM is competitive (Supplementary Fig. 2).

Comparisons to classification models

We construct training and test sets on ENCODE K562 DNase-seq datasets using the same peak definitions as Figure 1d with a 300bp window around each peak as a positive example. Negative examples are drawn uniformly at random from the genome.

For the gapped kmer-model (Ghandi et al. 2014) we use a 300 bp window as suggested in (Zhou and Troyanskaya 2015) for training and test, using the default execution flag of '-d 3' as suggested in the README.

Additionally, we trained the gapped kmer-model using the ENCODE hotspots with the same parameters, this training method performs on-par with the 300bp window model despite the training test set mismatch.

For SeqGL (Setty and Leslie 2015), we use the same 300bp train/test windows as the gapped kmer model, and train the model using the function 'run.seqgl.wrapper' with default parameters.

For DeepSEA(Zhou and Troyanskaya 2015), we took the pre-trained model available at the authors' website, and used a 1kb window generated by adding flanking bases to the 300bp train/test sets of the gapped kmer-model and extract the 'K562.DNase.None' column (no parameters exist for running DeepSEA).

We use the best transformed output from each method for our regression comparisons. For SeqGL and GKMSVM these are the outputs of the linear SVM and group lasso respectively, while for DeepSEA this was the probability outputs.

Our goal with these comparisons was not to show that SCM is suited for discriminative motif detection; on the contrary we expect these methods to perform quite well in tasks like ChIP-seq for which they were designed. Our goal is instead to show that our method is robust and flexible at modeling DNase-seq read counts over the genome despite having no parameters and being biologically driven.

Implementation of SCMs

Our goal is to produce a predictive model of sequence to a quantitative, integer-valued trait measured per base on the genome.

The design of our algorithm is guided by several goals:

- **Predictive model:** our model should predict trait that can be held-out and evaluated for goodness of fit. This makes the overall problem well-defined and easy to evaluate.
- **Parameter independence:** the model should not have any performance-influencing parameters. All parameters that can be set should be set as large as memory and computation time allows.
- **Tractable runtime:** the model should run in less than several days for any number of experiments on the human genome.
- **Interpretable parameters:** the output parameters should be interpretable as the local effects of an *K*-mer.
- **Theoretical grounding:** the model should provide reasonable theoretical guarantees on model recovery and prediction capacity.

These requirements naturally lead us to construct a genome-wide Poisson regression, where the variables are *K*-mer indicators that act log-linearly. The technical innovation in this paper is the introduction of a tractable method for fitting L_1 regularized linear models over the genome. Note that while a negative binomial regression would have the advantage of allowing us to fit overdispersed count data, it has the drawback that the overdispersion parameter makes the overall objective function nonconvex, and makes comparisons between separate samples impossible due to different variances. We instead use count truncation at ten reads per base to control the effective overdisperson uniformly over all samples.

In the paper we use a maximum K-mer length of 8 which was the maximum that would fit in memory in an Amazon EC2 c3.8xlarge instance. Larger K-mers tested on a larger memory machine did not perform substantially better than 8-mers.

Notation and genome representation

Throughout, we assume that the genome consists of one large chromosome with coordinate 0 to *N*. In practice we will construct this by concatenating chromosomes with the telomeres acting as spacers. The variable *K* represents the maximum *k*-mer length considered, the model fits all *k*-mers from 1…K. The variable *M* represents the influence of each *k*-mer.

The regularization parameter η is a scalar representing our belief about the sparsity of the problem.

Whenever possible, we will use *i* for genomic coordinate, *k* for *k*-mer length, and *j* for coordinate offset from the start of a *k*-mer.

The input variable *c* is a vector of length *N* representing counts and c_i represents the read-count observed at base *i*.

The latent variable λ is a vector of length *N* representing the current estimate for *c* using θ.

θ^k is the parameter matrix of size $4^k \times 2M$ associated with the set of all *k*-mers.

The variable g^k is a mapping from genomic coordinate *i* to the *k* mer starting at *i*. The *k*-mer for g^k is represented as an integer that maps to rows of θ such that the g^kth row of θ^k is the effect of a *k*-mer starting at coordinate *i*.

For instance, g_i^4 is the 4-mer starting at coordinate *i*. If this is **ATCG**, then the row $\theta^k_{g_i^4}$ must be the effect that **ATCG** exerts on its neighbors.

The special parameter θ_0 is used to set the average read rate of the genome globally.

Model setup

The problem we solve is a regularized Poisson regression. We would like to maximize the following:

$$\max_{\theta}\left(\sum_i c_i log(\lambda_i) - \lambda_i\right) - \eta \sum \left|\theta^k\right|_1$$

The intermediate variables λ are defined by:

$$\lambda_i = exp\left(\left(\sum_{k \in [1..K]} \sum_{j \in [-M,M-1]} \theta^k_{\left(g^k_{i+j}, -j\right)}\right) - \theta_0\right).$$

Naive inference algorithm

Naively, we would attempt batch proximal gradient descent on this objective function, which would involve the following steps:

1. Given current iterate θ, calculate current λ for all bases $i \in [0, N]$ by
$$\lambda_i = exp\left(\left(\sum_{k \in [1..K]} \sum_{j \in [-M,M-1]} \theta^k_{\left(g^k_{i+j}, -j\right)}\right) - \theta_0\right).$$
2. Given current λ calculate the per base gradient vector
$$dlog(\lambda_i) = err_i = c_i - \lambda_i.$$
3. Propagate the errors back to the parameter θ. Let s be the integer index corresponding to a *k*-mer. Then the gradient of this kmer s with off set j is
$$d\theta^k_{s,j} = \sum_{\{i: g_i^k = s\}} err_{\{i+j\}}$$
and
$$d\theta_0 = \sum_{i=1}^{N} err_i.$$
4. Update the current parameter with stepsize alpha.
$$\theta^k = \theta^k + \alpha d\theta^k$$
5. Update the constant offset

$$\theta_0 = \theta_0 - \alpha d\theta_0$$

6. Apply the proximal operator for L_1 regularization

$$\theta^k_{\{s,j\}} = \begin{cases} \theta^k_{\{s,j\}} - \alpha\eta \ \ if\left|\theta^k_{\{s,j\}}\right| > \alpha\eta \\ 0 \qquad otherwise \end{cases}$$

This algorithm is prohibitively slow, with an iteration runtime of $O(NMK + 4^K M)$. In practice, contribution from NMK dwarfs that of $4^K M$ since the gradient computation is cache incoherent and $N = 3 \times 10^9$ which is much greater than $4^K M = 6 \times 10^4$

There are two free parameters (α and η). The value for η is set via grid-search over values of η using held-out sets starting with the maximal feasible η. This maximum is calculated analytically as the maximal η for which all Kmers are nonzero. We will discuss setting α below.

Understanding the model

The k-mer model is a standard generalized linear model cast for a particular problem, but the role of regularization and the model class represented by the model may be confusing for some. The next sections make clear the action of the model.

Convexity of loss

The overall objective function of our Poisson regression is convex, this has a variety of important theoretical and practical properties:

1. The algorithm does not depend on initial condition of the optimizer
2. The optimizer converges and at fast rates.
3. We can understand the algorithm's behavior by analyzing its gradient.

Role of L_1 regularization

The model self-tunes its complexity through L_1 regularization. Consider the following toy example: the dinucleotide AT is a causal pioneer signature and we hope to detect this is the key k-mer.

Consider two possible solutions: the optimal one where only AT models the DNase effects and a suboptimal one in which 3-mers ATA,ATC,ATG,ATT together model the effect of AT. Then note that the L_1 penalty penalizes the latter model four times as much, making our algorithm strongly prefer the true model.

This argument can be generalized is a straightforward way to understand the way in which L_1 regularization determines which k-mers are set to zero and which others are set to nonzero values. By the proximal update equation, it is clear that in order for a k-mer at zero to become nonzero, the following has to be true:

$$d\theta^k_{\{s,j\}} = \sum_{\{i:g^k_i = s\}} err_{i+j} \geq \eta$$

For a given k-mer, s, consider the following thought experiment: take the optimal solution θ and set the effect of s to zero. The above equation claims that if the sum of errors around s is less than eta, s should be set to zero and θ could not be an optimal solution due to convexity.

Therefore a k-mer has two paths to becoming nonzero: it can have large effect whenever it appears, or it can appear many times and cause a consistent small (log) effect.

Role of exponential link function

The exponential link function allows the model to capture nonspecific interactions between k-mers. As an example let us consider the case of modeling a CTCF motif consisting of CCACCAGGGG using 4-mers. Consider upweighting the first base of CCAC in the length M vector, the second base of CACC, the third base of ACCA ... this allows us to construct a regression that has some activation c with a single 4-mer, but has c^7 activity when beneath a CTCF motif.

The combination of exponential link and modeling all local effects allows for very powerful combinatorial expressions that would not be possible with any model ignoring spatial relationships between k-mers. This expressivity of the model is the reason why both advanced optimization and regularization techniques are necessary to keep the model under control.

Accelerating the algorithm

Prefix compression

Note that due to our k-mer matching scheme, a 3-mer can be represented as the sum of the 4 possible 4-mers whose prefix matches the 3-mer. Utilizing this fact, we can obtain runtimes of $O(NM + 4^K M + 4^K K)$ and also reduce cache incoherence substantially.

We maintain a matrix ϕ of size $4^k \times 2M$which represents only the longest k-mers. We then modify the first through fourth steps to use ϕ instead of θ. Since every k-mer has a unique prefix match, this reduction maintains correctness of the algorithm.

Finally before step 6 we apply a decoding step. Let $g(s,k)$ be a set-valued function consisting of all k-mers whose first k-1 characters match s.

$$d\theta_s^k = \sum_{s' \in g(s,k)} d\phi_s'$$

The use of dynamic programming (generate θ_s^{k-1} followed by θ_s^{k-2}) gives a runtime of $O(4^K)$ to decode the compressed representation.

After step 6 we re-encode the parameter matrix into the compressed representation. Given a k-mer s, let $f(s,k)$ be the set valued function returning the k character prefix of s.

$$\phi_s = \sum_{\{k=1\}}^{K} \theta^k_{f(s,k)}$$

This takes runtime $O(4^K K)$.

Representing the k-mers as bitstrings where each two bits represents a base allows for the query operations to be done nearly entirely bitshifts and cache-coherent additions, which allows for fast encoding and decoding for typical values of K=8 and M=500.

More efficient proximal operators

We derive a provably correct and more efficient proximal operator for the gradient descent algorithm.

The basic algorithm uses the standard L_1 soft-threshold prox operator:

$$\theta^k_{\{s,j\}} = \begin{cases} \theta^k_{\{s,j\}} - \alpha\eta \ \ if\left|\theta^k_{\{s,j\}}\right| > \alpha\eta \\ 0 \qquad\qquad otherwise \end{cases}$$

However, we note that this solution can be strictly improved with little extra effort. Using the same insight as our prefix compression scheme, note that adding a constant c to a k-mer and adding the same constant to the 4 possible k-1 mer prefix matches returns the same predicted λ values but have different L_1 penalty values. Using this idea we can decrease the L_1 penalty without affecting the goodness of fit.

This algorithmically captures the intuition that if {ATA,ATC,ATG,ATT} all have similar and positive effects, we can better represent the effect using just AT.

Define the median of a k-mer as the median parameter value of the prefix matching k+1-mers and the negative of itself. Define g(s,k) as before as the set-valued function returning the four possible one character continuations of s (for example, given AT, g(s,3)={ATA,ATC,ATG,ATT}) and the function $m(s,j)$ as:

$$m(s,j) = median\left(\theta^{k+1}_{g(s,k),j} - \theta^{k}_{s,j}\right).$$

Then the parameters for any kmers $s' \in h(s)$ can be updated as

$$\theta^{k}_{s',j} = \theta^{k}_{s',j} - m(s,j)$$

and

$$\theta^{k}_{s,j} = \theta^{k}_{s,j} + m(s,j).$$

This is a dynamic programming algorithm starting at K-1 and stopping at k=1. This procedure is guaranteed to not change the likelihood term depending on λ while strictly shrinking the L_1 norm of θ.

Stochastic gradient descent

We find that gradient descent is still far too slow to run on a single 32-core machine in less than a week. We achieve nearly ten-fold speedup by utilizing stochastic rather than batch gradient descent.

The variant of gradient descent we use is a minibatch-gradient, where we calculate the gradient and error over a smaller subregion of the genome. We use twenty million bases as our minibatch size (which we will refer to as B).

To control the step-size more intelligently, we also use a variant of stochastic gradient descent known as Adagrad(Duchi, Hazan, & Singer, 2011). We maintain a separate history δ_s for every k-mer which we increment with the norm of the gradient.

In our variant, we cut the genome into twenty-million base chunks called minibatches. Let $l \in$ and $\sigma(l)$ be a permutation of l.

The steps one to three in the previous algorithm becomes:

1. At the beginning of every pass over the full genome, we generate a new random permutation $\sigma(l)$
2. Pick a global step size α by doing a line-search along the region of size B with largest number of reads.
3. For $i \in \left[1 \dots \lfloor\frac{N}{B}\rfloor\right]$ do a full update (naïve algorithm) on the subset of bases $(\sigma(l)_{i-1})B + [0, B]$.
4. For each k-mer s, update its value with
$$\theta^{k}_{s} = \theta^{k}_{s} + \alpha d\, \theta^{k}_{s} / \sqrt{\delta_s}$$
5. If the average function value of all minibatches is more than 10% greater than the previous iteration, set $\alpha = \alpha/2$, reset parameters and redo the loop
6. Else return the averaged iterates over the whole pass.

While this algorithm gives no asymptotic performance improvement over the batch gradient, in practice it returns a solution equivalent to the batch gradient in time that is 10-20 times faster than the batch method. This is a well-documented effect in the literature.

Non-synergistic additive models

Synergistic interaction between transcription factors are often defined as non-linear effects of sequence elements on a phenotype of interest, such as gene expression(Veitia 2003). In the case of chromatin accessibility, we are considering the question of whether a log-linear model for accessibility is appropriate compared to an additive, linear model. Such model comparisons were performed in prior work (He et al. 2010) and used as evidence for a logistic link between transcription factor binding and gene expression.

We therefore propose and fit a null linear additive model of k-mer effects which acts as a way to test whether DNase accessibility fits a synergistic sequence model better. To do this, we modify our objective function, defined as

$$\max_{\theta}\left(\sum_{i} c_i log(\lambda_i) - \lambda_i\right) - \eta \sum \left|\theta^k\right|_1$$

With intermediate variables λ are defined by:

$$\lambda_i = exp\left(\left(\sum_{k\in[1..K]}\ \sum_{j\in[-M,M-1]} \theta^k_{\left(g^k_{i+j},-j\right)}\right) - \theta_0\right).$$

By replacing the intermediate variable λdefined with an additive link,

$$\lambda_i = \left(\sum_{k\in[1..K]}\ \sum_{j\in[-M,M-1]} \theta^k_{\left(g^k_{i+j},-j\right)}\right) - \theta_0.$$

With the additional constraint, $\theta^k_{i,j} > 0$and $\theta_0 > 0.001$where the latter constraint is required to stabilize the objective function.

The resulting model has a similar expressive power as the log-linear model with identical parameters and convolutional assumptions on the k-mers, but with an additive rather than exponential effect of k-mers on chromatin accessibility.

Methods

Cell culture

Mouse embryonic stem cell culture was performed according to previously published protocols[3(Sherwood et al., 2014)]. Undifferentiated 129P2/OlaHsd mouse ES cells were maintained on gelatin-coated plates feeder-free in mES media composed of Knockout DMEM (Life Technologies) supplemented with 15% defined fetal bovine serum (FBS) (HyClone), 0.1mM nonessential amino acids (Life Technologies), Glutamax (Life Technologies), 0.55mM 2-mercaptoethanol (Sigma), 1X ESGRO LIF (Millipore), 5 nM GSK-3 inhibitor XV and 500 nM UO126. Cells were regularly tested for mycoplasma. Genetic manipulations to stem cell lines are described below.

DNase-seq

DNase-seq was performed as described previously(Sherwood et al., 2014). 10-100 million cells were digested with 60-100 units of DNase I (Promega) per 10^7 nuclei. 50-125 bp hypersensitive DNA was collected using E-Gel SizeSelect Agarose 2% gels (Life Technologies). Library preparation and Illumina HiSeq were performed by the MIT BioMicroCenter.

ChIP-seq

ChIP was performed according to the “Mammalian ChIP-on-chip” protocol (Agilent) using a polyclonal antibody against Nrf1 antibody (ab34682, Abcam) and Protein G Dynabeads (Life Technologies). 10-100 million cells were used for each experiment. qPCR using positive and negative control primers was performed to ensure ChIP enrichment. Library preparation and Illumina HiSeq were performed by the MIT BioMicroCenter.

Single Locus Oligonucleotide Transfer (SLOT)

To begin optimizing SLOT, we ordered a library of 175 bp oligonucleotide DNA sequences containing 100 bp variable phrases with the following common features: flanking primer sequences distinct from any genomic DNA sequence, a unique DNA barcode distinct from all other barcodes at Levenshtein distance = 2, and a common internal primer past the barcode (see Supplementary Fig. 8) from Broad Technology Services. This library was amplified using primers that add 67 bp homology arms to each end using NEBNext High-Fidelity 2X PCR Master mix (New England Biolabs), as we found that this polymerase minimized library amplification bias. Homology arms were designed to flank two genomic CrispR guide RNA sequences in genomic regions with no surrounding DNase-seq activity in mESC.

In order to utilize CrispR-mediated homologous recombination, we cloned the required guide RNA and Cas9 components into convenient vectors. We cloned the U6 promoter guide RNA hairpin construct from the dual Cas9/guide RNA expression plasmid pX330(Cong et al., 2013) (Addgene) into the Tol2 transposon vector p2TAL200R175(Kawakami & Noda, 2004) along with either a Hyromycin resistance cassette or a Blasticidin resistance cassette to form p2T U6sgRNA HygroR and p2T U6sgRNA BlastR. In later tests, we modified the hairpin structure to incorporate the “FE” alterations shown to improve guide RNA hairpin stability(Chen et al., 2013), creating p2T U6sgRNA-FE HygroR and p2T U6sgRNA-FE BlastR. We cloned the CBh promoter Cas9 construct from pX330 into the p2Lox vector designed to integrate expression constructs into the HPRT locus of the p2Lox mESC line(Iacovino et al., 2009; Mazzoni et al., 2011) to form p2Lox CBh Cas9. In later tests, we cloned CBh Cas9 into the p2TAL200R175 vector along with a Blasticidin resistance cassette to form p2T CBh Cas9 BlastR. We cloned guide RNAs targeting two closed chromatin loci into p2T U6sgRNA HygroR, p2T U6sgRNA BlastR, p2T U6sgRNA-FE HygroR, and p2T U6sgRNA-FE BlastR (denoted p2T U6sgLocusA HygroR etc.).

We then tested homologous recombination frequency by introducing 5 ug PCR-amplified library + 5 ug CBh Cas9 + 5 ug p2T U6sgLocusA HygroR into 10^6 mESCs by co-electroporation. Transient antibiotic selection was performed for 72 hours at 24-96 hours post-electroporation. We achieved 0.6% integrated allele frequency as assessed by comparing qPCR cycle counts of a locus-specific primer and a phrase-specific primer with control locus cycle counts (see Supplementary Fig. 8). We then asked whether constitutive expression of either Cas9 or guide RNA could improve homologous recombination frequency. We used the p2Lox system (p2Lox CBh Cas9) to constitutively express Cas9 and the Tol2 transposon system (p2T U6sgLocusA HygroR) to constitutively express either Cas9 or guide RNA in p2Lox mESCs. We then co-electroporated 5 ug PCR-amplified library + 5 ug CBh Cas9 + 5 ug p2T U6sgLocusA BlastR into 10^6 mESCs (we found that additional Cas9 and guide RNA improved homologous recombination even when constitutively expressed). We found that constitutive Cas9 or guide RNA improved homologous recombination frequency, yet constitutive guide RNA (6.3% allele frequency) was more efficient than constitutive Cas9 (1.7% allele frequency). We then repeated the constitutive guide RNA expression experiment but using the FE versions for constitutive (p2T U6sgLocusA-FE HygroR) and additional transient (p2T U6sgLocusA-FE BlastR) expression, achieving significantly more efficient homologous recombination (15.9% allele frequency). We then reasoned that transient selection for Cas9 electroporation might be more effective than selection for guide RNA electroporation. So, in the context of constitutive guide RNA expression (p2T U6sgLocusA-FE HygroR), we co-electroporated 5 ug PCR-amplified library + 5 ug p2T CBh Cas9 BlastR + 5 ug p2T U6sgLocusA HygroR into 10^6 mESCs, achieving even more efficient homologous recombination (30.9% allele frequency). Results from this set of experiments are summarized in Supplementary Fig. 8.

We thus standardized the SLOT protocol: construct a stable cell line constitutively expressing p2T U6sgRNA HygroR, then co-electroporate PCR-amplified library + p2T CBh Cas9 BlastR + p2T U6sgRNA HygroR. For the experiments described in this work, we electroporated 10^7 mESC with 20 ug of each component DNA, achieving 20-50% allele frequency in all three loci. Library-integrated mESCs were grown for 7-21 days after electroporation before DNase-I hypersensitivity analysis, and care was taken to maintain high pool complexity by splitting at high density.

DNase-I hypersensitivity analysis was performed mostly according to our previously published protocol(Sherwood et al., 2014) with several differences. Immediately after nuclear extraction, 5-10% of nuclei were reserved for genomic DNA isolation to serve as a control. The remaining nuclei were treated with 70-90 units of DNase per 10^7 cells. After DNA purification, E-gel size-selection was performed to isolate 125-275 bp DNA, a size range that accommodates the minimal size required to amplify with locus-specific and internal primers (see Supplementary Fig. 8). qPCR using positive and negative control primers was performed to ensure enrichment of DNase-hypersensitive DNA.

Then, we performed a three-step library preparation to allow Illumina deep sequencing analysis of barcode representation (see Supplementary Fig. 8). First, we specifically amplified locus-integrated DNA from either genomic DNA or DNase-I hypersensitive DNA for 16-20 PCR cycles using NEBNext. For genomic DNA, 16 ug of input DNA was used in an 800 uL reaction to ensure library diversity; for DNase-I hypersensitive DNA, ½ the product was used in a 400 uL reaction. This step serves two purposes: (i) it separates locus-integrated phrases from unintegrated phrases which remain in cells in substantial numbers even several weeks after electroporation; (ii) it highly enriches for phrases at the expense of other genomic regions, simplifying the template for the subsequent tailed primer PCR steps. We performed QiaQuick PCR purification (Qiagen) to purify the products. We then performed qPCR on a small aliquot of each sample to calculate how many PCR cycles to perform for the subsequent two PCR steps, typically performing 5-10 cycles of each PCR. The second PCR step amplifies the barcode using the flanking and internal primers while adding half of the Illumina paired-end primer sequences as well as 5-6 bp multiplexing sample barcodes to allow sequencing of multiple samples per MiSeq/HiSeq lane. The third PCR step adds the full Illumina paired-end primer sequences. For the experiments reported in this work, we used 70 bp single-end Illumina MiSeq, performed by the MIT BioMicroCenter. Full phrases were also sequenced from genomic DNA using a similar library preparation strategy as above but using the flanking primer instead of the internal primer to amplify locus-integrated phrases. These samples were sequenced using 150 + 150 bp paired-end MiSeq.

Oligonucleotides

Oligonucleotides used in this work are listed in Supplementary Table 2.

Mapping and data preprocessing

We use a unified alignment, error-control and bias reduction pipeline from bcbio-nextgen built as a docker container. The aligner used was BWA, with a map quality cutoff of at least 1 (unique mapping). The mapped reads were then filtered, retaining at most ten reads per base per strand to remove anomalous towers of reads.

Construction of a DeBrujin graph based SLOT test phrase set

Traditional phrase library construction methods have relied primarily on hand-constructing a scaffold of control sequences with a replaceable positive phrase component that accepts a motif. This approach has the problem that only a few parts of the phrase are actually activating, and that the positive phrase components must be known quite precisely.

Instead, we demonstrate a method that can construct a phrase library where each phrase is constructed solely of positive or control sequences which can model more complex structures compared to traditional methods. For example, our algorithm is able to identify that if a phrase begins with CCACCA, then the most DNase-seq positive completion is to continue the CTCF motif as CCACCAGGGGC.

Using a pre-trained DNase-seq K-mer model we construct the compressed phrase representation ϕ and separate the k-mers into three classes: DNase-hypersensitive, control, and DNase-closing.

For the control-phrases, we construct a deBrujin graph over the K-mers in the control group and perform a random walk of length 100-K (our desired phrase length). This produces a set of 100-base phrases that are predicted to have no effect at all bases.

For the positive-phrases, we construct a deBrujin graph over both the control and positive groups. Random walking over this graph could result in a phrase that is entirely control, so we construct the phrase in the following way:

1. begin at a positive phrase randomly chosen from the positive phrase set.
2. if there exists any positive phrases in the out-neighborhood, then jump randomly amongst them
3. otherwise, jump deterministically to the most positive phrase.

This process has the property that it random walks until it can no-longer continue a positive phrase, and then hill-climbs to the next closest phrase. We find that this process produces substantially higher enrichment compared to randomly inserting K-mers into a scaffold.

Collapsing k-mers into motifs

The motif-construction algorithm is for visualization purposes and so optimizes for speed rather than performance. We use a hierarchical clustering scheme combined with pairwise alignment.

1. Filter the overall K-mer set into enriched K-mers by thresholding on the sum of effects ϕ.
2. Calculate the pairwise Levenstein distance amongst the candidate K-mer set.
3. Use complete linkage clustering to obtain clusters with distance at least 2 amongst clusters.
4. Pairwise align each cluster to the most activating k-mer in the cluster

Raw data sources

All sequencing data were mapped using the BWA aligner, and use the quality score filter mapq>20.

Sequencing datasets used in this paper from other publications are:

NAME	UCSC Acession
HEK293T	wgEncodeEH002565
H7ES	wgEncodeEH002554
GM12878	wgEncodeEH000534
GM10248	wgEncodeEH003487
Frontal cortex OC	wgEncodeEH003471
FibroP_AG20443	wgEncodeEH002569
FibroP_AG08395	wgEncodeEH002568
Fibrobl_GM03348 lenti-Myod	wgEncodeEH003473
ECC1 DMSO	wgEncodeEH002555
Cerebrum frontal OC	wgEncodeEH003480
K562	wgEncodeEH000530 wgEncodeEH003489

In addition we have produced and analyzed the following datasets generated in-house:

mES DNase-Seq 130801 50-100bp	
mES DNase-Seq 130801 175-400bp	
mES DNase-Seq 120710 replicate 1	
mES DNase-Seq 120710 replicate 2	
mES DNase-Seq NFya 50-100bp	
mES DNase-Seq NFya 175-400bp	
mES Nrf1 rep 1 no dox	
mES Nrf1 rep 1 dox	
mES Nrf1 rep 1 sr3	
mES Nrf1 rep 1 sr8	
Hues8 S6 DALT 50-100	
Hues8 S6 DALT 175-400	

SCM scoring and determination of correlation coefficient with actual data.

The k-mer model generates predictions according to the formula for λ above. We then deduplicate both prediction and raw reads and measure Pearson's correlation after a 2-kilobase smoothing. Variance stabilizing transforms (square-root) are used whenever analysis require Homoscedasticity such as in Figure1c.

Certain analyses are done after mean-subtraction and normalization by the standard deviation (Z-score).

For details see the Rmarkdown documents used to generate the figures which are made available with the code.

K-mer effect size determination

We measure a K-mer's importance by its summed absolute log-contributions. In the notation described previously, this is equivalent to summing $|\theta|$ over a row.

SLOT computational analysis

In order to analyze the integrated oligonucleiotides we follow the following procedure:

1. Reconstruct the set of genomically integrated phrases by paired-end sequencing and overlapping sequences using FLASH (http://ccb.jhu.edu/software/FLASH/)
2. Use the 20bp variable region past the primer as a barcode
3. Predict per-base number of reads using formula for λ
4. Take the average number of reads over each oligo
5. For each barcode, assign each barcode a kmer-score according to the max over the set of mapping oligos

Measuring sequence fidelity: we calculate the error rate by taking each unique barcode generated by 20bp region and calculating the fraction of sequenced oligos that are not the consensus sequence. We then average this over all barcodes to get an expected error rate per base.

Mappability was measured by taking each oligo design and mapping it using bwa-mem to the genome.

Determination of enrichment between k-mers and pioneer, settler, migrant motifs.

We used a master list of JASPAR motifs assigned to the three categories following our PIQ paper(Sherwood et al., 2014). The data and analysis are all available as a reproducible markdown document with our code. The top 500 ranked 8-mers are mapped to sequence with a log-likelihood cutoff of 5. We calculate the random background rate of matching motifs by randomly sampling from the set of all 8-mers. The enrichment is then calculated as a ratio.

Binarization

For both DNase-seq and Nrf1 binding we consider binarized region calling. For this we use the same smoothing as above. Each base is assigned the average number of reads within a $\pm 100bp$ region. Then we call as hypersensitive any base with averaged read count 5 standard deviations greater than the mean. Hypersensitive bases within 10 bases are merged together.

For AUC calculations we rank randomly sampled bases using the SCM prediction. If the base lies within a hypersensitive region, it receives a positive label, otherwise it receives a negative label.

Segmentation

In the analysis of classes of regulatory regions, we use segmentations derived from ENCODE with CpG taken from 450k Bead arrays (http://genome.ucsc.edu/cgi-bin/hgTrackUi?db=hg19&g=wgEncodeHaibMethyl450) and segmentations from chromHMM (http://genome.ucsc.edu/cgi-bin/hgTrackUi?db=hg19&g=wgEncodeBroadHmm).

For calculating correlations we simply subset over each category of segmentation and CpG island and calculate correlation only over the segment.

Performance of SCM model as a function of cell-type specificity

i. Regions of DNase-I enrichment

We smoothed the real and SCM-predicted read counts using a 100bp window for the held-out chromosomes chr15 to chr22. We randomly shuffled the real read counts across bases in each chromosome and then smoothed these random read counts using a 100bp window as well. For each held-out chromosome, we computed the mean and standard deviation of the smoothed read counts and transformed the read counts to Z-scores by subtracting the mean and dividing by the standard deviation.

In order to find putative regions of enrichment, we selected all continuous regions of the chromosome with Z-score >=3.1 which would correspond to a p-value of 0.001, in case of a normal distribution. Now we discretized the regions of enrichment in the real data into 100bp blocks and computed the sum of Z-scores within each of these blocks. We simultaneously selected a set of random 100000 bins of length 100bp across the chromosome and computed the sum of the randomized Z-scores in these bins. Using these scores, we were able to determine a z-score sum that corresponded to a 5% FDR threshold. We used this cutoff for sum of Z-scores to determine the true regions of enrichment in the real data.

ii. Computing performance metrics

Calculating sensitivity: Our true positive set was defined as the set of enriched 100bp bins in the real data that met our FDR threshold. We calculated sum of Z-scores for the predicted data for each of these bins and compared it to the previously computed 5% FDR threshold using randomized real data. Sensitivity was calculated as the ratio of the number of enriched bins that are also enriched in predicted data to the total number of enriched bins in real data.

Calculating specificity: As true negatives we selected a random set of 100bp bins of equal number to the true positive set that are not found to be DNase-I enriched in the real data .We then calculated the sum of Z-scores for each of these bins using the predicted data and counted the number of bins that are not enriched based on the FDR threshold. Specificity was calculated as the ratio of the number of non-enriched real bins that are also not enriched in predicted data to the number of non-enriched real bins.

Balanced accuracy was then measured as an average of the sensitivity and specificity. As a control, we used the real data in one cell type to determine true positives and negatives and compared it to the predicted data from the other 10 cell-types.

SCM code is available at http://scm.csail.mit.edu.

References

Barash, Y., Calarco, J. A., Gao, W., Pan, Q., Wang, X., Shai, O., . . . Frey, B. J. (2010). Deciphering the splicing code. *Nature, 465*(7294), 53-59. doi:10.1038/nature09000

Boyle, A. P., Davis, S., Shulha, H. P., Meltzer, P., Margulies, E. H., Weng, Z., . . . Crawford, G. E. (2008). High-resolution mapping and characterization of open chromatin across the genome. *Cell, 132*(2), 311-322.

Boyle, A. P., Song, L., Lee, B. K., London, D., Keefe, D., Birney, E., . . . Furey, T. S. (2011). High-resolution genome-wide in vivo footprinting of diverse transcription factors in human cells. *Genome Res, 21*(3), 456-464.

Chen, B., Gilbert, L. A., Cimini, B. A., Schnitzbauer, J., Zhang, W., Li, G. W., . . . Huang, B. (2013). Dynamic imaging of genomic loci in living human cells by an optimized CRISPR/Cas system. *Cell, 155*(7), 1479-1491. doi:10.1016/j.cell.2013.12.001

Cong, L., Ran, F. A., Cox, D., Lin, S., Barretto, R., Habib, N., . . . Zhang, F. (2013). Multiplex genome engineering using CRISPR/Cas systems. *Science, 339*(6121), 819-823. doi:10.1126/science.1231143

Dostie, J., Richmond, T. A., Arnaout, R. A., Selzer, R. R., Lee, W. L., Honan, T. A., . . . Dekker, J. (2006). Chromosome Conformation Capture Carbon Copy (5C): a massively parallel solution for mapping interactions between genomic elements. *Genome Res, 16*(10), 1299-1309. doi:10.1101/gr.5571506

Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research, 12*(Jul), 2121-2159.

Ernst, J., & Kellis, M. (2013). Interplay between chromatin state, regulator binding, and regulatory motifs in six human cell types. *Genome research, 23*(7), 1142-1154. doi:10.1101/gr.144840.112

Frith, M. C., Valen, E., Krogh, A., Hayashizaki, Y., Carninci, P., & Sandelin, A. (2008). A code for transcription initiation in mammalian genomes. *Genome Res, 18*(1), 1-12. doi:10.1101/gr.6831208

Fullwood, M. J., Liu, M. H., Pan, Y. F., Liu, J., Xu, H., Mohamed, Y. B., . . . Ruan, Y. (2009). An oestrogen-receptor-alpha-bound human chromatin interactome. *Nature, 462*(7269), 58-64.

Gualdi, R., Bossard, P., Zheng, M., Hamada, Y., Coleman, J. R., & Zaret, K. S. (1996). Hepatic specification of the gut endoderm in vitro: cell signaling and transcriptional control. *Genes Dev, 10*(13), 1670-1682.

Hathaway, N. A., Bell, O., Hodges, C., Miller, E. L., Neel, D. S., & Crabtree, G. R. (2012). Dynamics and memory of heterochromatin in living cells. *Cell, 149*(7), 1447-1460. doi:10.1016/j.cell.2012.03.052

He, H. H., Meyer, C. A., Hu, S. S., Chen, M. W., Zang, C., Liu, Y., . . . Brown, M. (2013). Refined DNase-seq protocol and data analysis reveals intrinsic bias in transcription factor footprint identification. *Nat Methods*. doi:10.1038/nmeth.2762

Heintzman, N. D., Hon, G. C., Hawkins, R. D., Kheradpour, P., Stark, A., Harp, L. F., . . . Ren, B. (2009). Histone modifications at human enhancers reflect global cell-type-specific gene expression. *Nature, 459*(7243), 108-112.

Hesselberth, J. R., Chen, X., Zhang, Z., Sabo, P. J., Sandstrom, R., Reynolds, A. P., . . . Stamatoyannopoulos, J. A. (2009). Global mapping of protein-DNA interactions in vivo by digital genomic footprinting. *Nat Methods, 6*(4), 283-289. doi:10.1038/nmeth.1313

Ho, L., & Crabtree, G. R. (2010). Chromatin remodelling during development. *Nature, 463*(7280), 474-484. doi:10.1038/nature08911

Hughes, A. L., & Rando, O. J. (2014). Mechanisms underlying nucleosome positioning in vivo. *Annu Rev Biophys, 43*, 41-63. doi:10.1146/annurev-biophys-051013-023114

Iacovino, M., Hernandez, C., Xu, Z., Bajwa, G., Prather, M., & Kyba, M. (2009). A conserved role for Hox paralog group 4 in regulation of hematopoietic progenitors. *Stem Cells Dev, 18*(5), 783-792.

Kaplan, N., Moore, I. K., Fondufe-Mittendorf, Y., Gossett, A. J., Tillo, D., Field, Y., . . . Segal, E. (2009). The DNA-encoded nucleosome organization of a eukaryotic genome. *Nature, 458*(7236), 362-366. doi:10.1038/nature07667

Kawakami, K., & Noda, T. (2004). Transposition of the Tol2 element, an Ac-like element from the Japanese medaka fish Oryzias latipes, in mouse embryonic stem cells. *Genetics, 166*(2), 895-899.

Koohy, H., Down, T. A., & Hubbard, T. J. (2013). Chromatin accessibility data sets show bias due to sequence specificity of the DNase I enzyme. *PLoS One, 8*(7), e69853. doi:10.1371/journal.pone.0069853

Lee, D., Karchin, R., & Beer, M. A. (2011). Discriminative prediction of mammalian enhancers from DNA sequence. *Genome Res, 21*(12), 2167-2180. doi:10.1101/gr.121905.111

Lee, J. T., & Bartolomei, M. S. (2013). X-inactivation, imprinting, and long noncoding RNAs in health and disease. *Cell, 152*(6), 1308-1323. doi:10.1016/j.cell.2013.02.016

Lenhard, B., Sandelin, A., & Carninci, P. (2012). Metazoan promoters: emerging characteristics and insights into transcriptional regulation. *Nat Rev Genet, 13*(4), 233-245. doi:10.1038/nrg3163

Lieberman-Aiden, E., van Berkum, N. L., Williams, L., Imakaev, M., Ragoczy, T., Telling, A., . . . Dekker, J. (2009). Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science, 326*(5950), 289-293. doi:10.1126/science.1181369

Mazzoni, E. O., Mahony, S., Iacovino, M., Morrison, C. A., Mountoufaris, G., Closser, M., . . . Wichterle, H. (2011). Embryonic stem cell-based mapping of developmental transcriptional programs. *Nat Methods, 8*(12), 1056-1058.

Meissner, A., Mikkelsen, T. S., Gu, H., Wernig, M., Hanna, J., Sivachenko, A., . . . Lander, E. S. (2008). Genome-scale DNA methylation maps of pluripotent and differentiated cells. *Nature, 454*(7205), 766-770.

Mirny, L. A. (2010). Nucleosome-mediated cooperativity between transcription factors. *Proc Natl Acad Sci U S A, 107*(52), 22534-22539. doi:10.1073/pnas.0913805107

Mullen, A. C., Orlando, D. A., Newman, J. J., Loven, J., Kumar, R. M., Bilodeau, S., . . . Young, R. A. (2011). Master transcription factors determine cell-type-specific responses to TGF-beta signaling. *Cell, 147*(3), 565-576.

Neph, S., Vierstra, J., Stergachis, A. B., Reynolds, A. P., Haugen, E., Vernot, B., . . . Stamatoyannopoulos, J. A. (2012). An expansive human regulatory lexicon encoded in transcription factor footprints. *Nature, 489*(7414), 83-90.

Peckham, H. E., Thurman, R. E., Fu, Y., Stamatoyannopoulos, J. A., Noble, W. S., Struhl, K., & Weng, Z. (2007). Nucleosome positioning signals in genomic DNA. *Genome Res, 17*(8), 1170-1177. doi:10.1101/gr.6101007

Ram, O., Goren, A., Amit, I., Shoresh, N., Yosef, N., Ernst, J., . . . Bernstein, Bradley E. (2011). Combinatorial Patterning of Chromatin Regulators Uncovered by Genome-wide Location Analysis in Human Cells. *Cell, 147*(7), 1628-1639. doi:10.1016/j.cell.2011.09.057

Richmond, T. J., & Davey, C. A. (2003). The structure of DNA in the nucleosome core. *Nature, 423*(6936), 145-150. doi:10.1038/nature01595

Sandelin, A., Carninci, P., Lenhard, B., Ponjavic, J., Hayashizaki, Y., & Hume, D. A. (2007). Mammalian RNA polymerase II core promoters: insights from genome-wide studies. *Nat Rev Genet, 8*(6), 424-436. doi:10.1038/nrg2026

Segal, E., Fondufe-Mittendorf, Y., Chen, L., Thåström, A., Field, Y., Moore, I. K., . . . Widom, J. (2006). A genomic code for nucleosome positioning. *Nature, 442*(7104), 772-778. doi:10.1038/nature04979

Sherwood, R. I., Hashimoto, T., O'Donnell, C. W., Lewis, S., Barkal, A. A., van Hoff, J. P., . . . Gifford, D. K. (2014). Discovery of directional and nondirectional pioneer transcription factors by modeling DNase profile magnitude and shape. *Nat Biotechnol, 32*(2), 171-178. doi:10.1038/nbt.2798

Siebert, M., & Söding, J. (2014). Universality of core promoter elements? *Nature, 511*(7510), E11-12. doi:10.1038/nature13587

Simonis, M., Klous, P., Splinter, E., Moshkin, Y., Willemsen, R., de Wit, E., . . . de Laat, W. (2006). Nuclear organization of active and inactive chromatin domains uncovered by chromosome conformation capture-on-chip (4C). *Nat Genet, 38*(11), 1348-1354. doi:10.1038/ng1896

Soufi, A., Donahue, G., & Zaret, K. S. (2012). Facilitators and impediments of the pluripotency reprogramming factors' initial engagement with the genome. *Cell, 151*(5), 994-1004. doi:10.1016/j.cell.2012.09.045

Thomson, J. P., Skene, P. J., Selfridge, J., Clouaire, T., Guy, J., Webb, S., . . . Bird, A. (2010). CpG islands influence chromatin structure via the CpG-binding protein Cfp1. *Nature, 464*(7291), 1082-1086. doi:10.1038/nature08924

Thurman, R. E., Rynes, E., Humbert, R., Vierstra, J., Maurano, M. T., Haugen, E., . . . Stamatoyannopoulos, J. A. (2012). The accessible chromatin landscape of the human genome. *Nature, 489*(7414), 75-82.

Trompouki, E., Bowman, T. V., Lawton, L. N., Fan, Z. P., Wu, D. C., DiBiase, A., . . . Zon, L. I. (2011). Lineage regulators direct BMP and Wnt pathways to cell-specific programs during differentiation and regeneration. *Cell, 147*(3), 577-589.

Valen, E., & Sandelin, A. (2011). Genomic and chromatin signals underlying transcription start-site selection. *Trends Genet, 27*(11), 475-485. doi:10.1016/j.tig.2011.08.001

Wang, J., Zhuang, J., Iyer, S., Lin, X., Whitfield, T. W., Greven, M. C., . . . Weng, Z. (2012). Sequence features and chromatin structure around the genomic regions bound by 119 human transcription factors. *Genome research, 22*(9), 1798-1812. doi:10.1101/gr.139105.112

White, M. A., Myers, C. A., Corbo, J. C., & Cohen, B. A. (2013). Massively parallel in vivo enhancer assay reveals that highly local features determine the cis-regulatory function of ChIP-seq peaks. *Proceedings of the National Academy of Sciences*. doi:10.1073/pnas.1307449110

Zaret, K. S., & Carroll, J. S. (2011). Pioneer transcription factors: establishing competence for gene expression. *Genes Dev, 25*(21), 2227-2241.

Zhou, V. W., Goren, A., & Bernstein, B. E. (2011). Charting histone modifications and the functional organization of mammalian genomes. *Nat Rev Genet, 12*(1), 7-18. doi:10.1038/nrg2905

Ghandi M, Lee D, Mohammad-Noori M, Beer MA. 2014. Enhanced regulatory sequence prediction using gapped k-mer features. *PLoS Comput Biol* **10**: e1003711.

He X, Samee MAH, Blatti C, Sinha S. 2010. Thermodynamics-based models of transcriptional regulation by enhancers: the roles of synergistic activation, cooperative binding and short-range repression. *PLoS Comput Biol* **6**.

Setty M, Leslie CS. 2015. SeqGL Identifies Context-Dependent Binding Signals in Genome-Wide Regulatory Element Maps. *PLoS Comput Biol* **11**: e1004271.

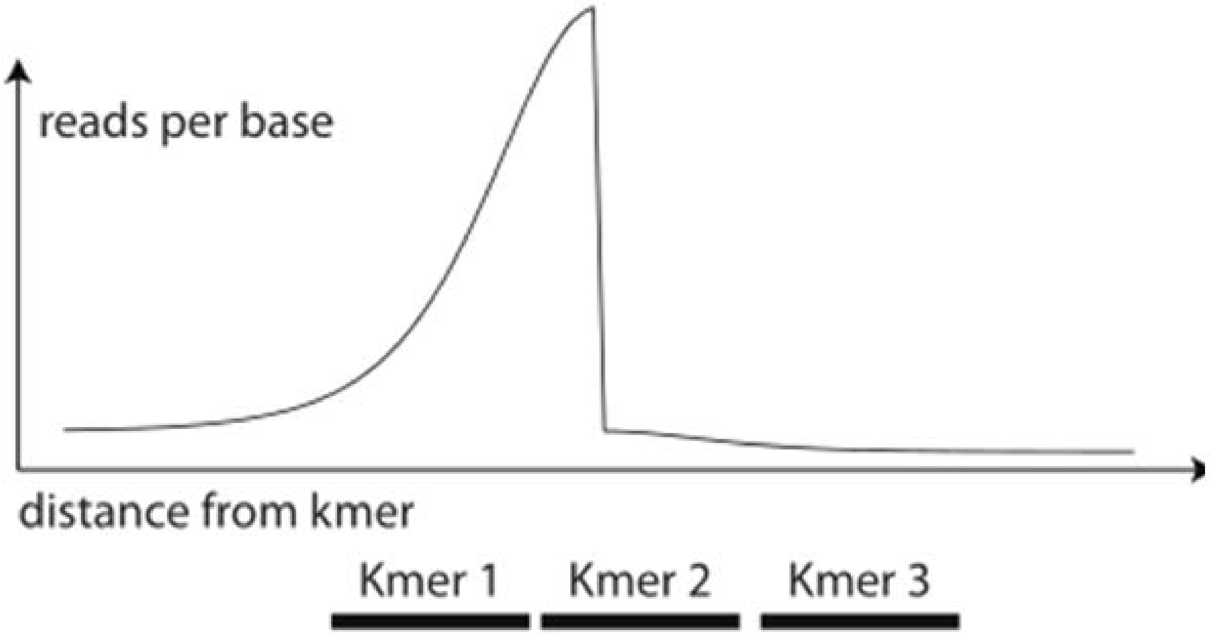
Veitia RA. 2003. A sigmoidal transcriptional response: cooperativity, synergy and dosage effects. *Biol Rev Camb Philos Soc* **78**: 149–170.

Zhou J, Troyanskaya OG. 2015. Predicting effects of noncoding variants with deep learning-based sequence model. *Nat Methods* **12**: 931–934.

Supplementary Figures

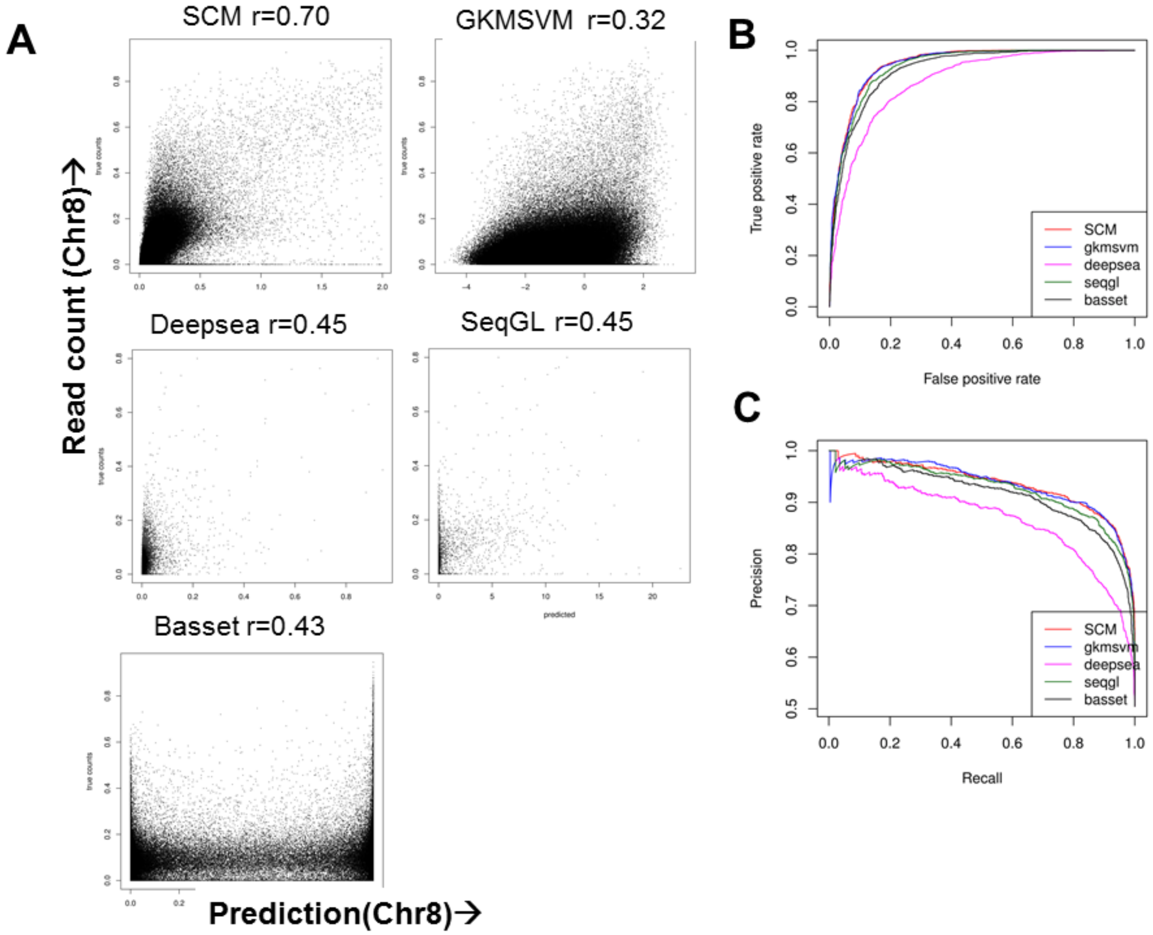
Supplementary Figure 1: SCM parameters allow modeling of fixed-distance k-mer interaction

Nearby k-mers can interact non-linearly to produce defined logic functions. Below is a diagram of how three k-mers produce a NOT function where Kmer3 is present. Compositions for AND and OR have similar constructions.



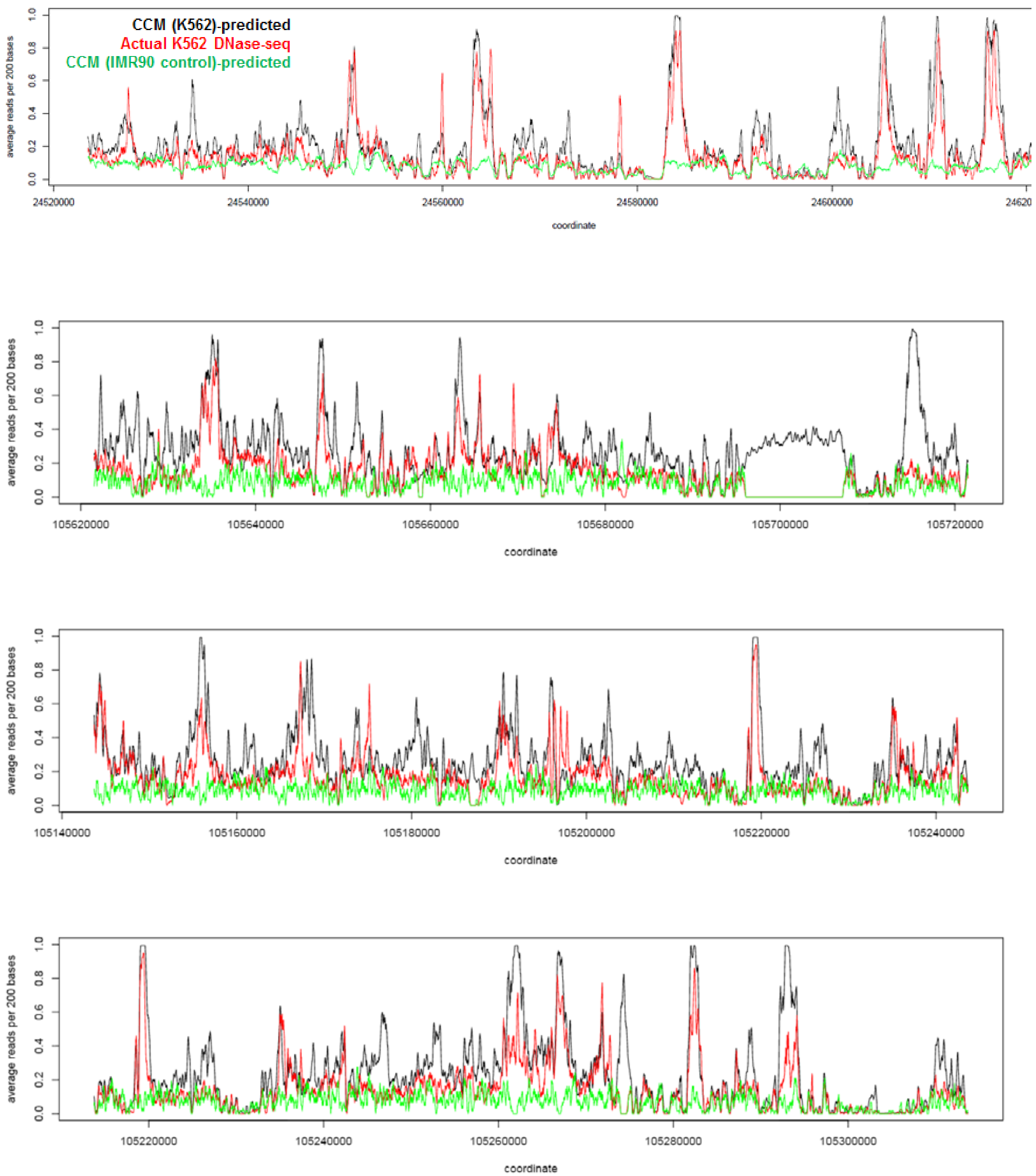
Supplementary Fig. 2: SCM is equivalent to the state-of-the-art classifiers on classification tasks and better at read count prediction.

Comparison of held out classification accuracy between SCM and classification methods. The comparison in panel A favors SCM, as the classification methods are not designed to optimize correlations. For GKMSVM and SeqGL we use the linear predictions from the SVM and group lasso directly, while for DeepSEA we use the probability predictions. The comparison in panel B favors the classifiers, as the classification methods can model the definition of a peak, while the SCM must model the entire genome. In both cases, DeepSEA is at a disadvantage because the DeepSEA model is pre-trained against ENCODE calls rather than our DNase peak calls over 300bp windows. In Panel A). we find that the SCM substantially outperforms the competitors at predicting the expected read count over 1kb smoothed windows in the genome. Spearman correlations for each method are (SCM:0.74, GKMSVM:0.27, Deepsea:0.43, Seqgl: 0.48, Basset:0.50) In Panel B), we find that the SCM still performs best compared to 3 classification methods despite the fact that the SCM is not designed for classification.



Supplementary Figure 3: Additional examples of SCM prediction of DNase-seq data

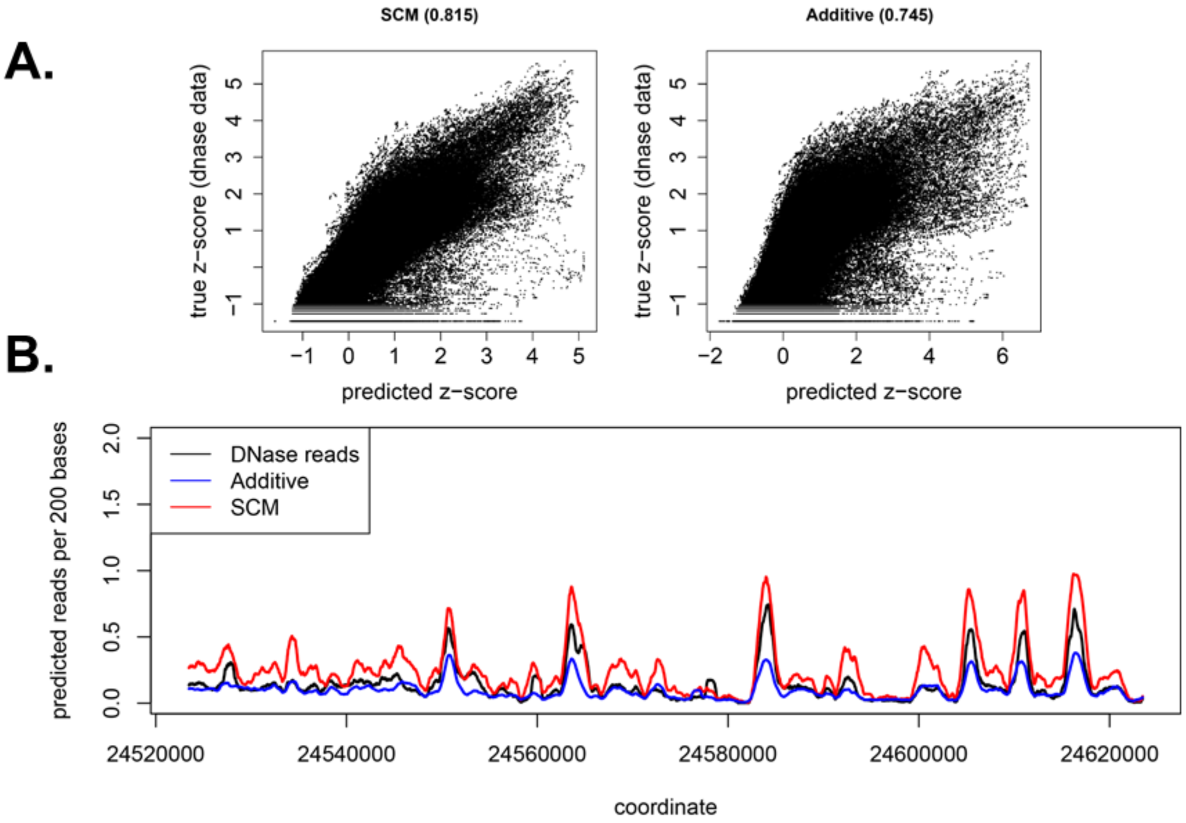
Example human K562 held-out genomic regions showing DNase-seq reads (red), SCM-predicted reads (black), and reads from a control model trained on IMR-90 naked DNA DNase-seq data (green), all smoothed at 200 bp.



Supplementary figure 4: Comparison to a non-synergistic baseline.

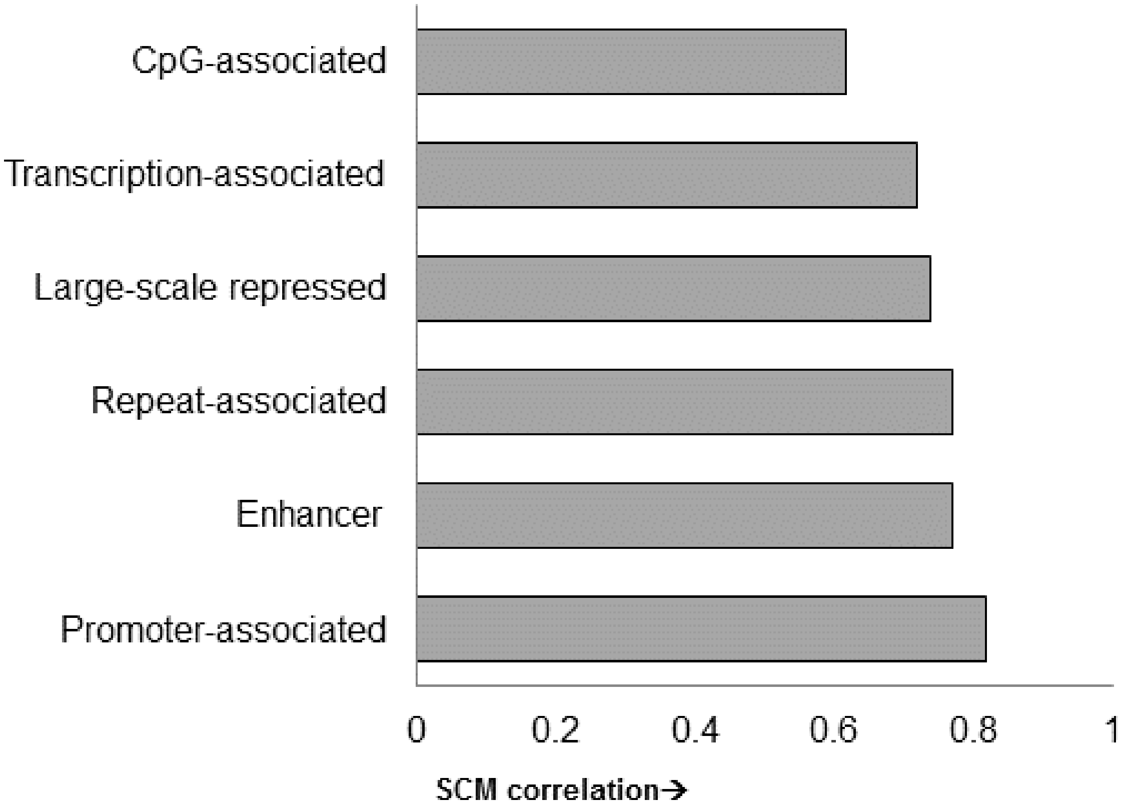
An additive, non-synergistic model of DNase was produced by training the model according to the section “Non-additive synergistic models”. We then examined whether an additive model using the same training method and model capacity performed as well as the synergistic model.

A. The SCM shows better linear correlation than the baseline method in the same comparison as figure 1c.
B. Example plot shows qualitatively better correlation of SCM to observed DNase data compared to the non-synergistic model. The additive model cannot reproduce the observed dynamic range of the peaks due to the lack of synergy.



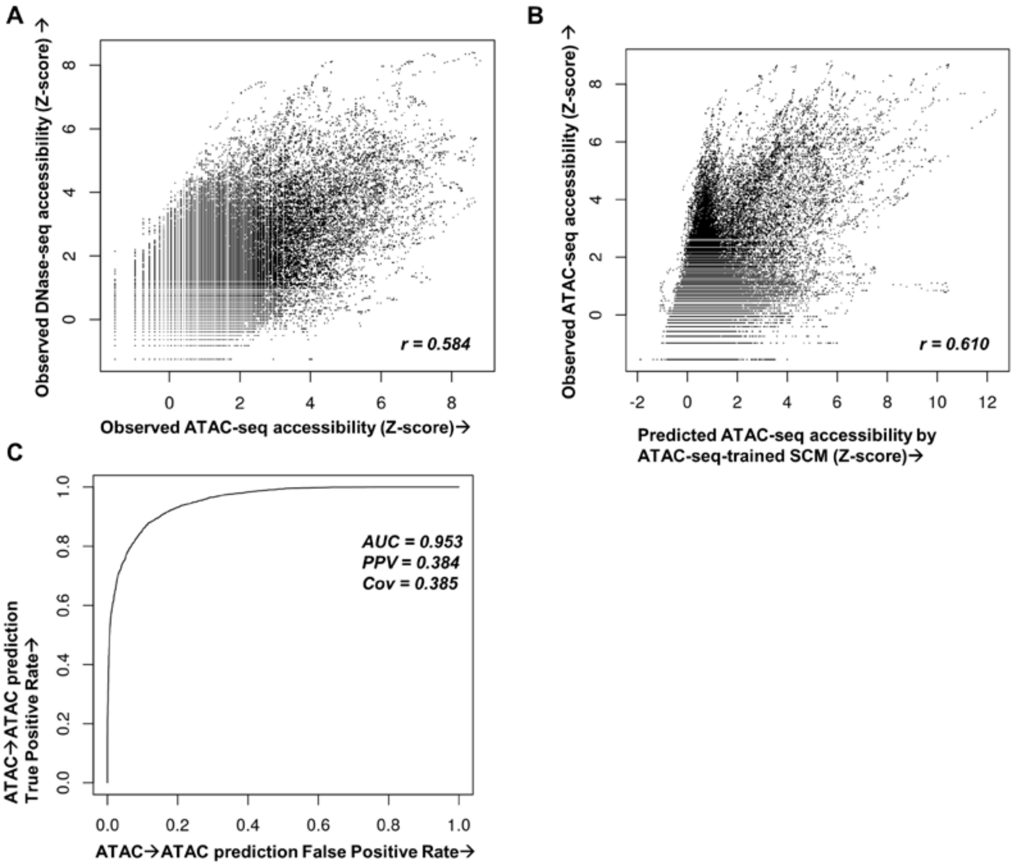
Supplementary Figure 5: SCM predicts the vast majority of genomic enhancers and promoters

Pearson's correlation coefficients measuring accuracy of SCM on specific chromatin types as defined by ChromHMM. This figures captures intra-category variability, resulting in CpG annotations receiving relatively low correlations. This is partially due to the fact that while it is straightforward to distinguish CpG vs non-CpG regions through DNase-seq, it is difficult to quantitatively predict the DNase accessibility of any CpG region.



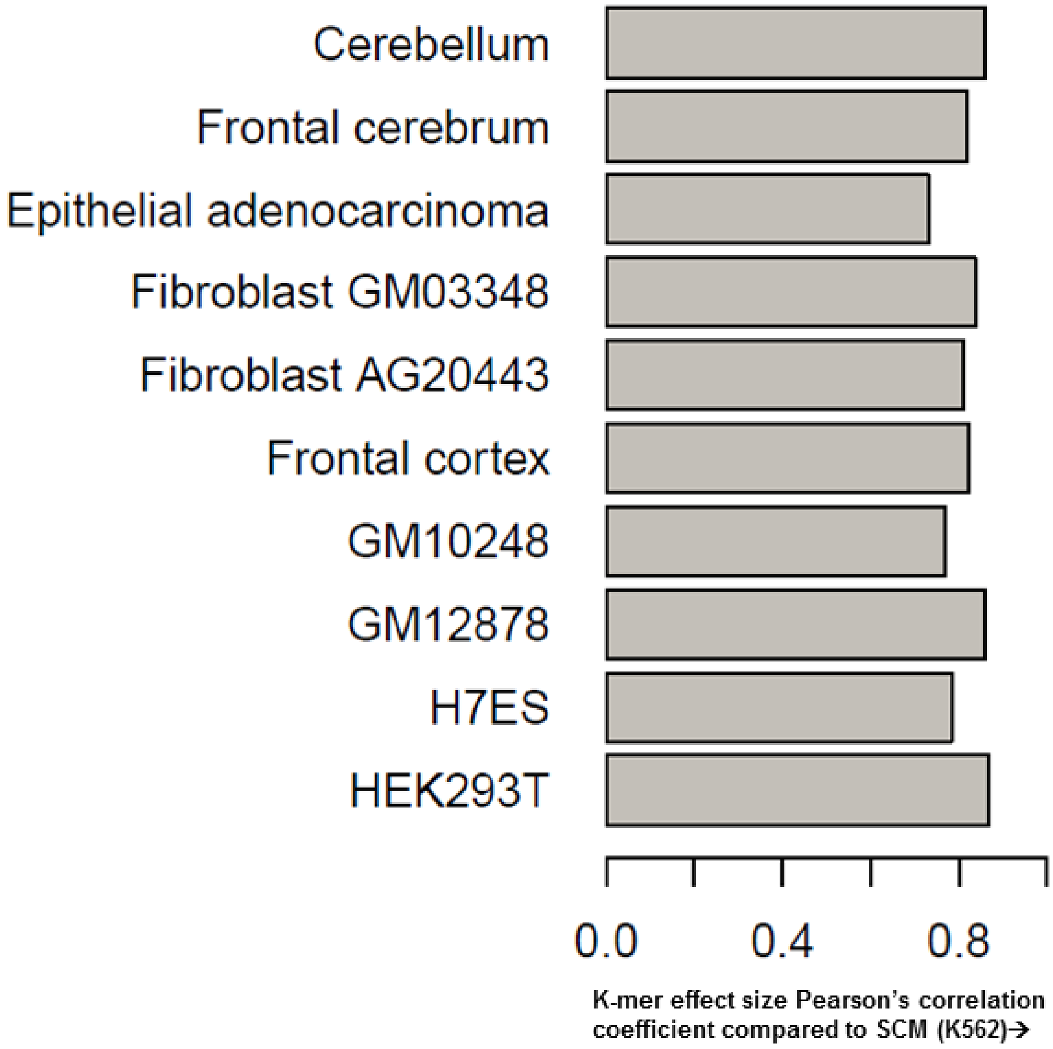
Supplementary Figure 6: ATAC-seq raw signal diverges from DNase-seq signal and can be predicted by a SCM with decent accuracy

a. Comparison of observed ATAC-seq (x-axis) and observed DNase-seq (y-axis) reads in 2 kb binned regions of GM12878 held-out chromosome 14. **b.** Comparison of SCM (ATAC-seq GM12878)-predicted (x-axis) and observed (y-axis) ATAC-seq reads in 2 kb binned regions of GM12878 held-out chromosome 14. **c.** Receiver-operator curve (ROC) showing predictive accuracy of a SCM trained on GM12878 ATAC-seq data at predicting held-out GM12878 ATAC-seq peaks.



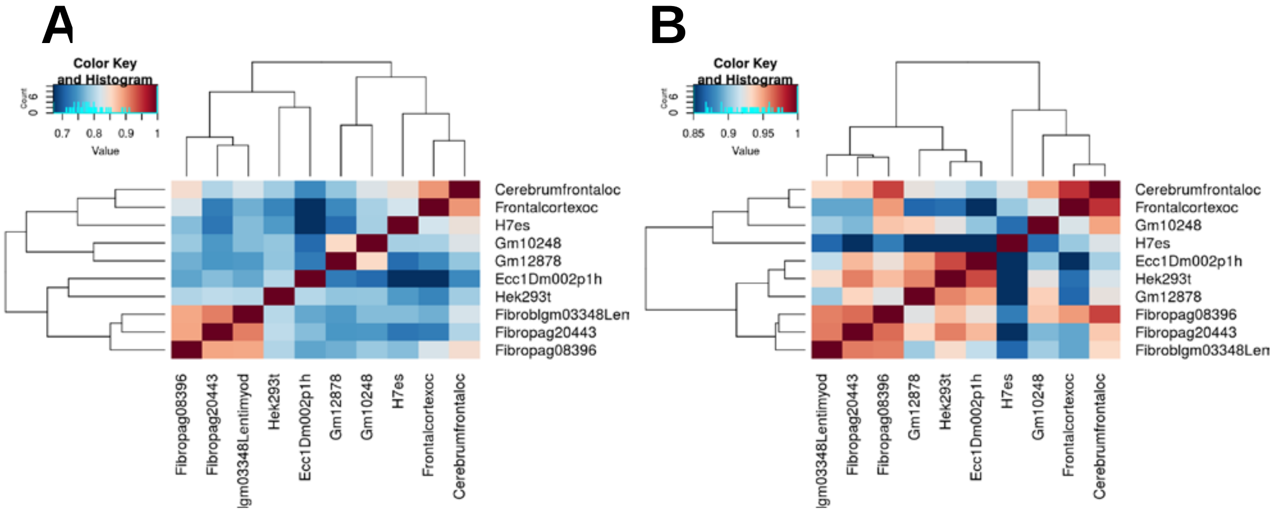
Supplementary Figure 7: The strongest SCM k-mers are conserved across a wide range of human cell types

Pearson's correlation coefficients measuring the similarity of k-mer effect sizes in SCM trained on the listed cell types as compared to the k-mer effect sizes in SCM (K562).



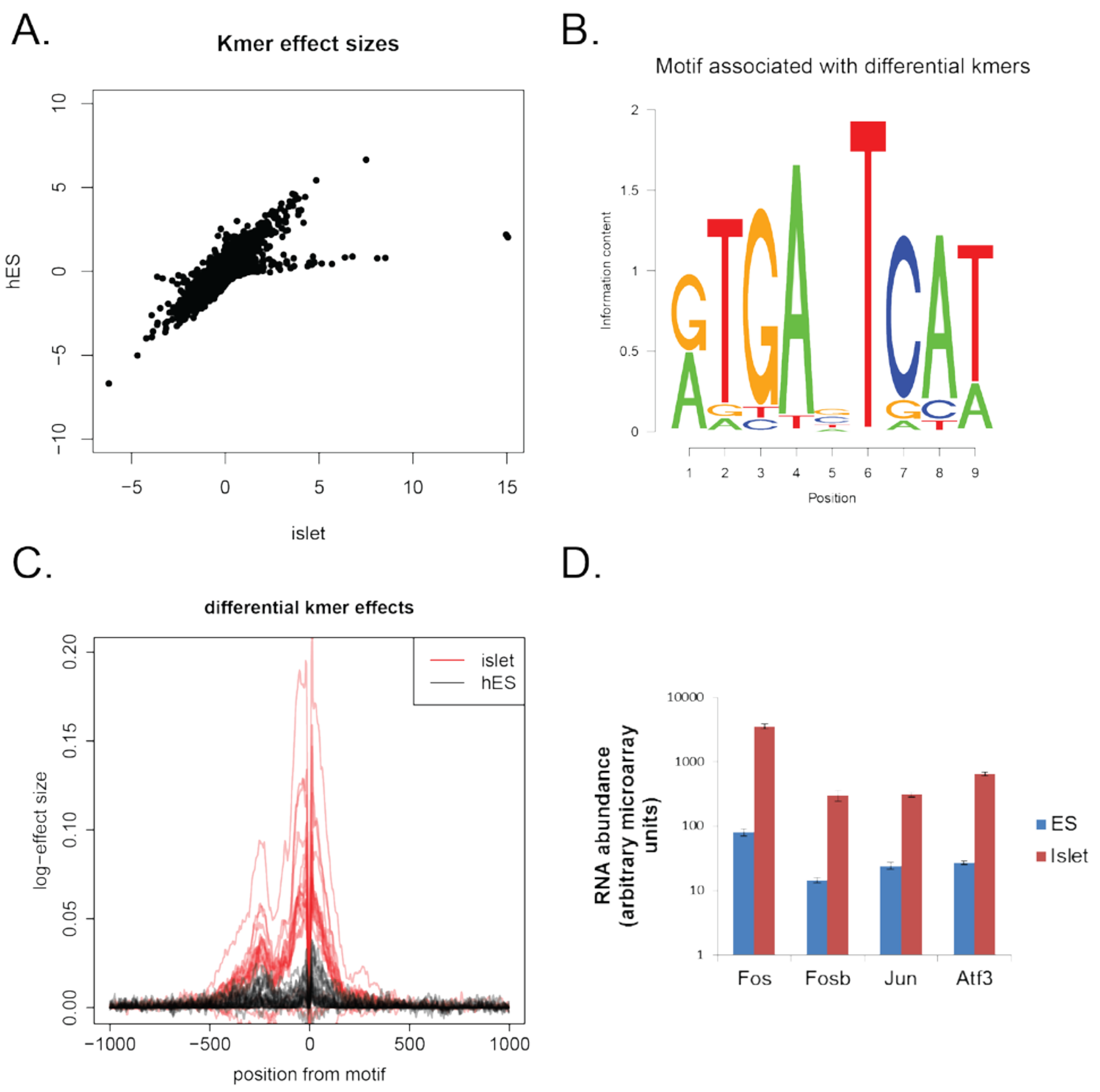
Supplementary figure 8

Correlations between cell types on chromosome 14 (held-out chromosome) across varying cell types for 200-base pair smoothed read counts in real data (A) and predicted reads from the SCM (B). The two panels show similar clustering across similar cell types (Fibroblast cells and Neuronal cells form distinct clusters in each case).



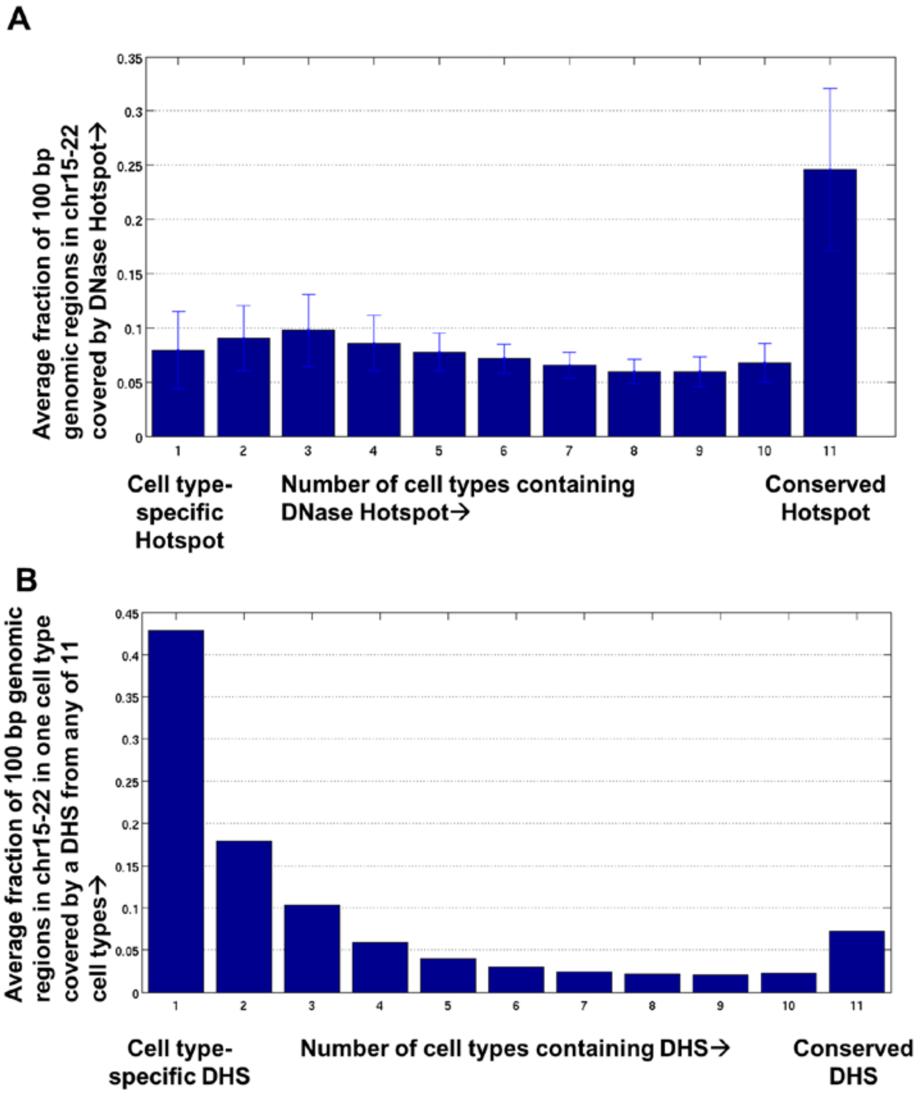
Supplementary figure 9: SCM captures aspects of cell type specific k-mer behavior.

- **A.** Scatterplot of kmer effect sizes (identical to those shown in Figure 3) on SCM trained on human ES cells (y-axis) and Islet cells (x-axis). There is a clear set of enriched k-mers active in islet.
- **B.** Motif generated from the 20 extremal k-mers that are active in Islet and not in ES. The Motif is nearly exactly the AP-1 complex motif.
- **C.** The log-effects of the top 20 extremal k-mers shown in panel A shown in red (islet) and black (human ES). It is clear that there is a strong activation of AP-1 associated chromatin opening in the islet, but not in the ES cell case.
- **D.** The components of the AP-1 complex are more highly expressed in Islet compared to human ES cells.



Supplementary Figure 10: The computed cell type-specificity of DNase-I hypersensitive regions depends on how the problem is defined.

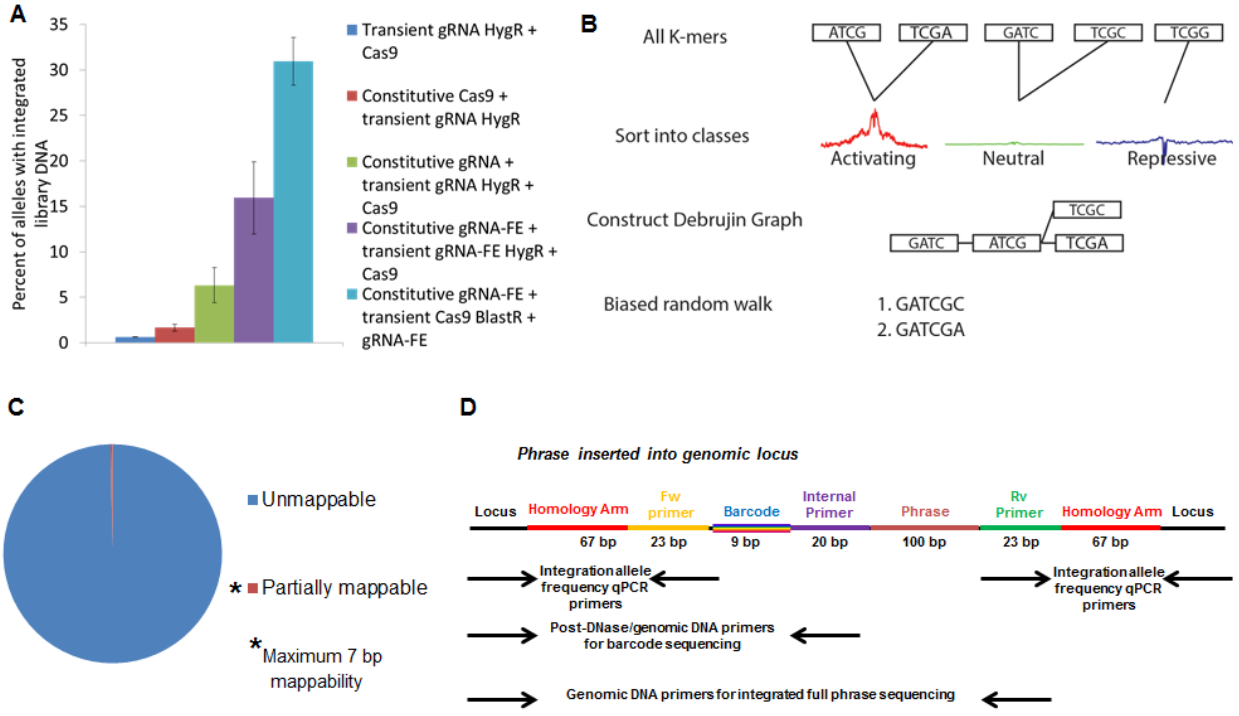
a. Histogram showing the fraction of the genomic space covered by DNase-I hypersensitive Hotspots in a single cell type that is covered by a Hotspot in ten other human cell types. **b.** Histogram showing the fraction of the total genomic space covered by a DNase-I hypersensitive site (DHS) in the superset of 11 human cell types that is covered by a DHS in the ten other human cell types. In this plot, all DHS from the 11 cell types are considered as opposed to Figure 3b and the above panel which compute cell type-specificity using DHS calls from a single cell type. For the purposes of our work, we believe the overlap of DHS space from one dataset to all others (Figure 3b and the above panel) to be the most relevant statistic, as SCMs are each trained on one specific dataset and asked to predict the entirety of genomic chromatin accessibility

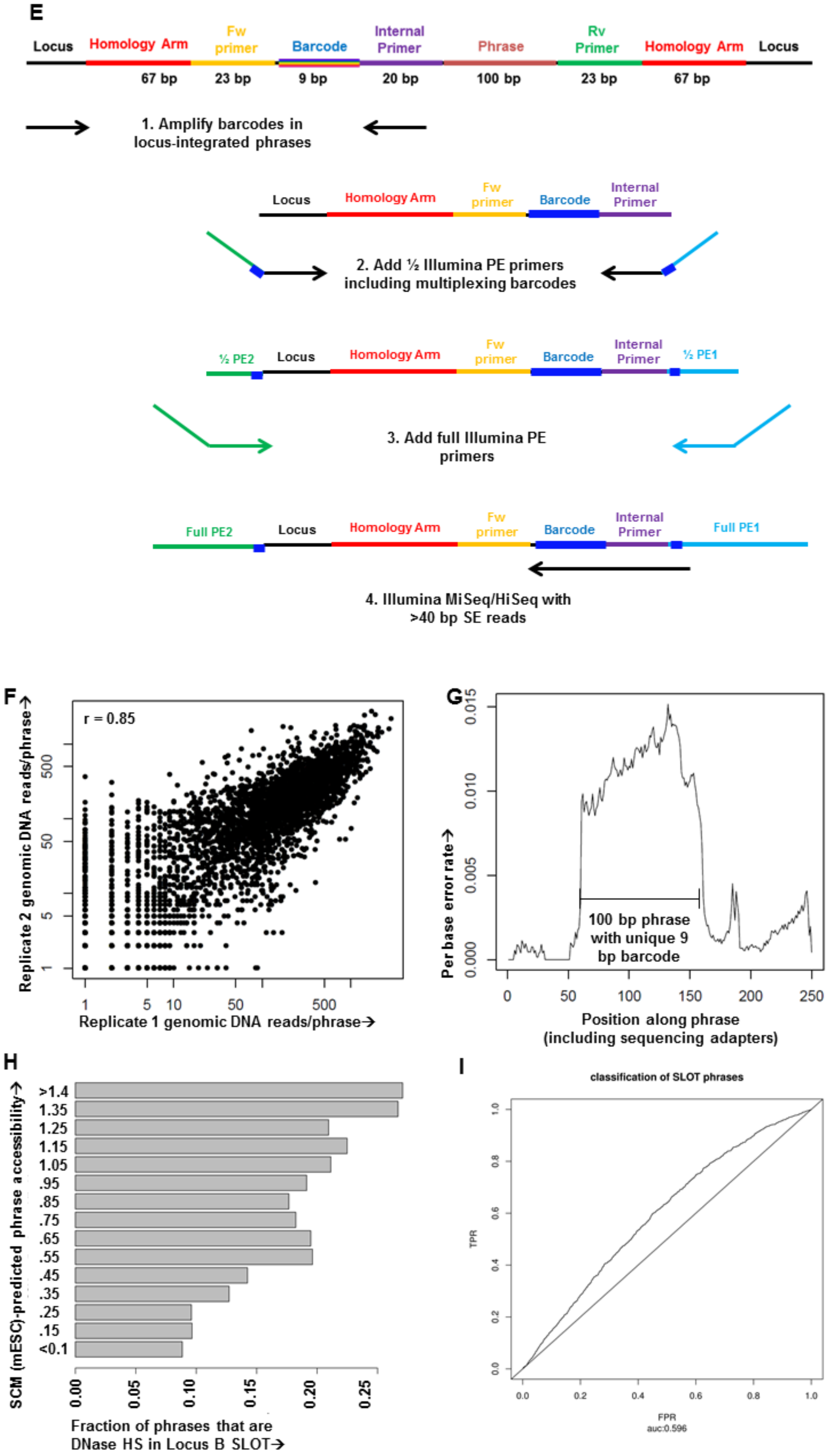


.

Supplementary figure 11: SLOT confirms SCM accuracy in predicting sequence-dependent chromatin accessibility

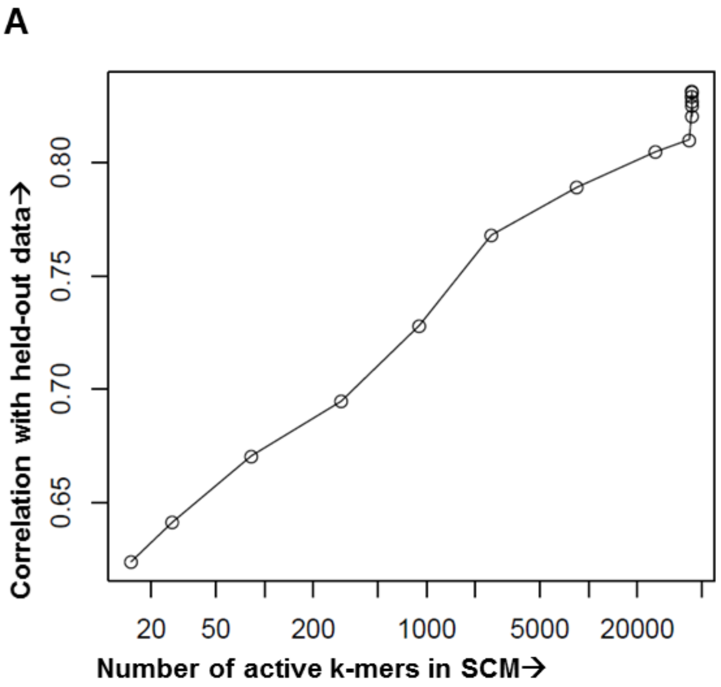
a. SLOT was optimized to yield the highest phrase integration frequency. From left to right on the x-axis are successive optimization steps and on the y-axis is the allele frequency of integrated phrases. Transient co-electroporation of homology-arm tailed phrase library DNA, Cas9 enzyme, and guide RNA followed by transient antibiotic selection for guide RNA expression yields <1% integration, whereas stable constitutive expression of FE-modified guide RNA followed by transient co-electroporation of homology-arm tailed phrase library DNA, Cas9 enzyme, and guide RNA followed by transient antibiotic selection for Cas9 expression yields 30% integration. **b**. A de Bruijn graph-based algorithm is used to generate a library of phrases with varying predicted chromatin accessibility and little resemblance to native genomic DNA sequence. **c**. Over 99% of the phrases have no similarity to mouse genomic DNA as determined by mappability, and the maximum mappable fragment is 7 bp. **d**. The components of a SLOT library phrase inserted into a genomic locus are presented. Locus-specific and phrase-specific primers are used in qPCR to determine integration frequency, locus-specific and internal phrase primers are used to amplify integrated barcodes from DNase-I hypersensitive and genomic DNA for deep sequencing, and locus-specific and phrase-flanking primers are used to amplify integrated phrases for full-phrase deep sequencing. **e**. The SLOT PCR-based library preparation method is diagrammed. Three steps of PCR are used to generate libraries for deep sequencing. **f**. Technical replicate library preparations of the same SLOT genomic DNA library using barcode-only and full-phrase methods yield highly concordant barcode readouts, suggesting highly reproducible library preparation. **g**. Per-base error rates in full-phrase deep sequencing sorted by barcode are less than 2% at every base, suggesting that barcodes can be reliably used to estimate readouts of specific phrases. **h**. SCM-predicted and SLOT-measured barcode reads are associated after integration in an additional genomic loci, indicating that SCM predictions robustly predict sequence-dependent chromatin accessibility. **i.** ROC curve for prediction of individual SLOT barcodes.

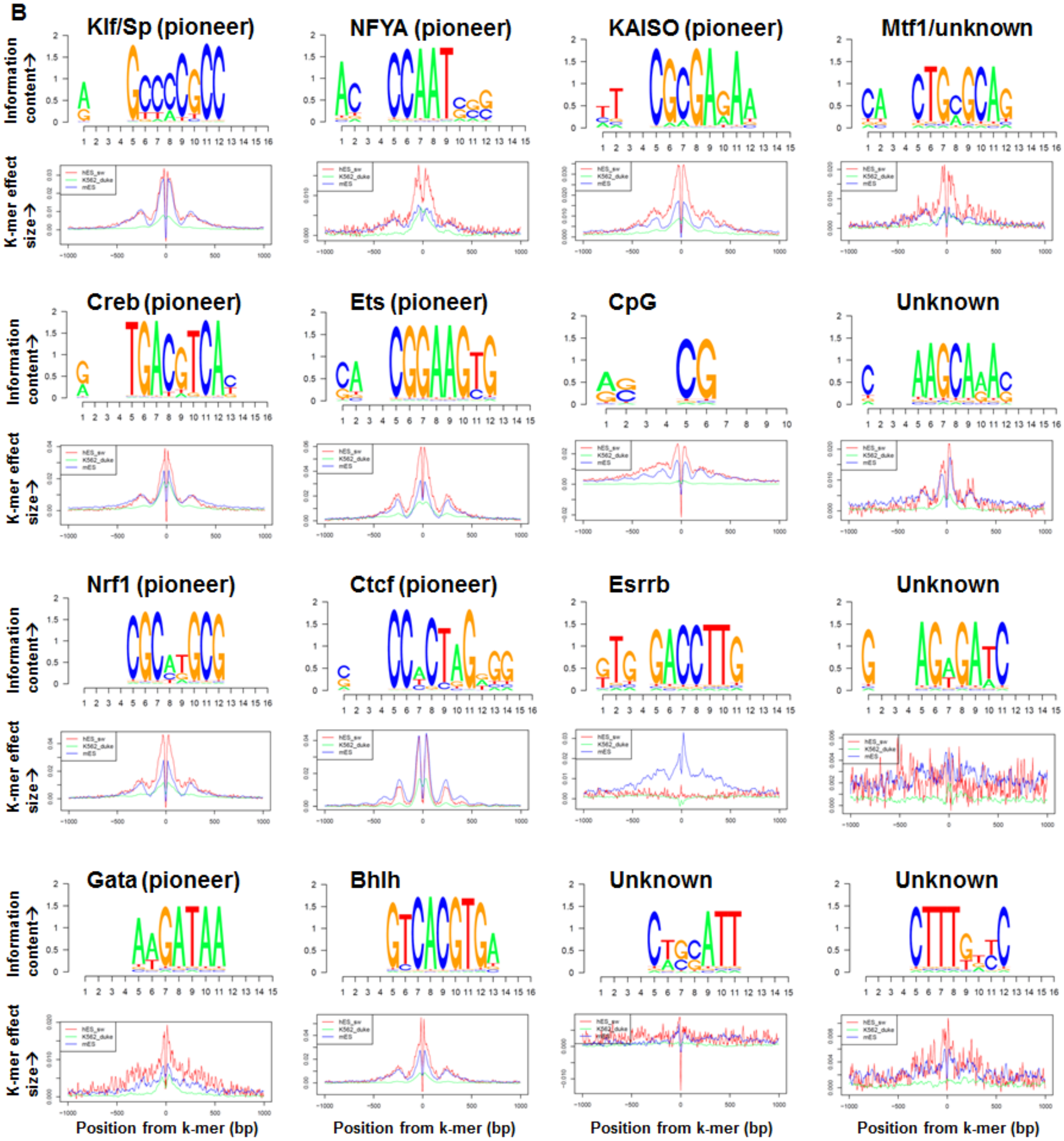



E
Locus
Homology Arm
Fw primer
Barcode
Internal Primer
Phrase
Rv Primer
Homology Arm
Locus
67 bp
23 bp
9 bp
20 bp
100 bp
23 bp
67 bp
1. Amplify barcodes in locus-integrated phrases
Locus
Homology Arm
Fw primer
Barcode
Internal Primer
2. Add ½ Illumina PE primers including multiplexing barcodes
½ PE2
Locus
Homology Arm
Fw primer
Barcode
Internal Primer
½ PE1
3. Add full Illumina PE primers
Full PE2
Locus
Homology Arm
Fw primer
Barcode
Internal Primer
Full PE1
4. Illumina MiSeq/HiSeq with >40 bp SE reads
F
r = 0.85
Replicate 2 genomic DNA reads/phrase→
1
5
50
500
1
5
10
50
500
Replicate 1 genomic DNA reads/phrase→
G
Per base error rate→
0.000
0.005
0.010
0.015
100 bp phrase with unique 9 bp barcode
0
50
100
150
200
250
Position along phrase (including sequencing adapters)
H
SCM (mESC)-predicted phrase accessibility→
>1.4
1.35
1.25
1.15
1.05
.95
.85
.75
.65
.55
.45
.35
.25
.15
<0.1
0.00
0.05
0.10
0.15
0.20
0.25
Fraction of phrases that are DNase HS in Locus B SLOT→
I
classification of SLOT phrases
TPR
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
FPR
auc:0.596

Supplementary figure 12: Properties of SCM k-mers

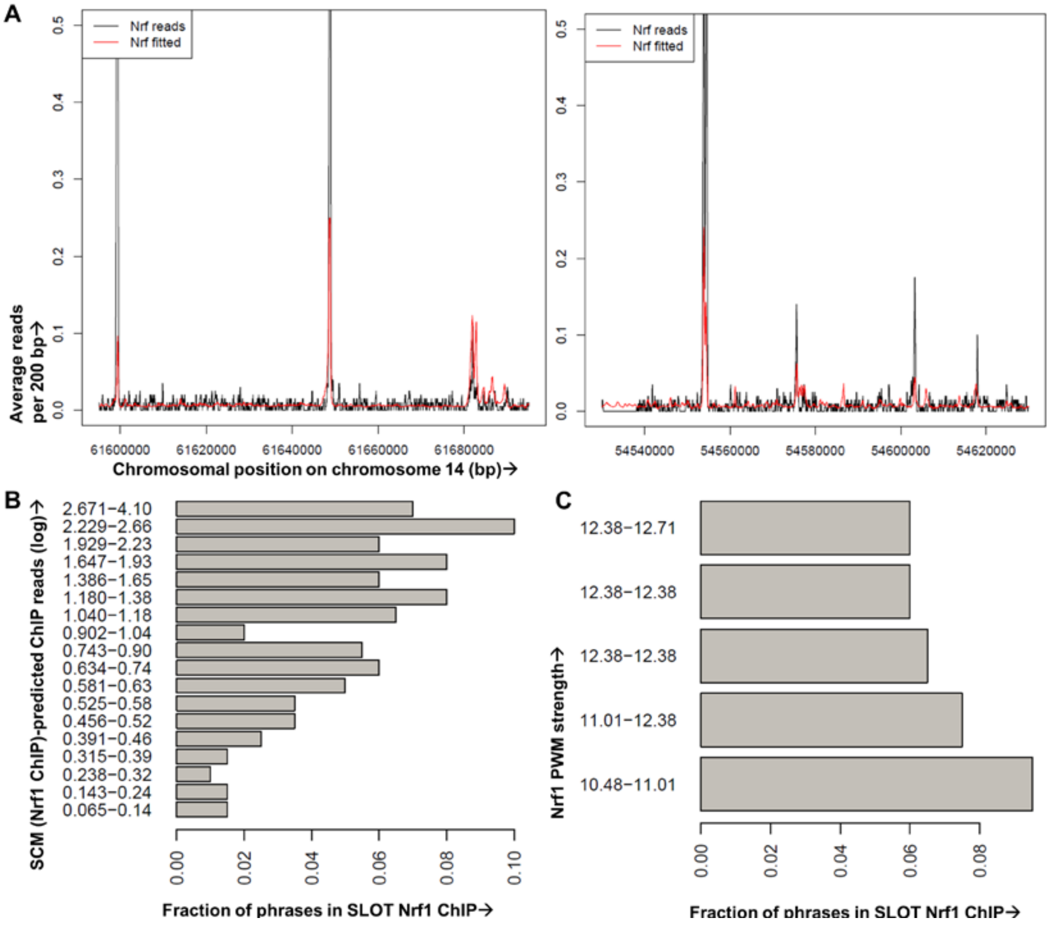
a. SCMs constrained to allow fewer than 20,000 active k-mers have substantially lower correlation with held-out DNase-seq data than constrained models with over 20,000 k-mers or an unconstrained model. **b**. Example PWM motifs derived from clustering the 500 k-mers with strongest mESC SCM effect size. Below the PWM are merged spatial k-mer effect sizes for all k-mers contributing to the motif within +/-1000 bp of the k-mer in hESC (red), mESC (blue), and K562 (green), showing the common effects of k-mers in these cell types. Names above correspond to high-confidence database matches with TF motifs when known, and known pioneer TFs are denoted. Several PWMs do not strongly correspond to known motifs.




B
Klf/Sp (pioneer)
NFYA (pioneer)
KAISO (pioneer)
Mtf1/unknown
Creb (pioneer)
Ets (pioneer)
CpG
Unknown
Nrf1 (pioneer)
Ctcf (pioneer)
Esrrb
Unknown
Gata (pioneer)
Bhlh
Unknown
Unknown
Information content→
K-mer effect size→
hES_sw
K562_duke
mES
Position from k-mer (bp)

Supplementary figure 13: Nrf1 binding is better predicted by synergistic logic in genome-wide binding prediction and in a SLOT assay

a. Example mouse ESC held-out genomic regions showing Nrf1 ChIP-seq reads (black) and reads predicted from a Nrf1 ChIP-trained SCM (red), both smoothed at 200 bp. **b-c**. Fraction of phrases, binned into 100 phrase bins by their overall SCM-predicted chromatin accessibility (**b**, x-axis) or Nrf1 PWM strength (**c**, x-axis), that are pulled down by Nrf1 ChIP in a SLOT assay (y-axis). Linear correlation is seen when phrases are ranked by SCM-predicted ChIP reads (**b**) but not PWM strength (**c**).



Supplementary Tables

Supplementary Table 1: Summary of CMM and control CMM held-out chromosome performance

Training data	Held-out data	Notes	Pearson's correlation coefficient
K562 DNase-seq	K562 DNase-seq		0.800
IMR90 naked DNA	K562 DNase-seq		0.468
mESC DNase-seq	mESC DNase-seq		0.791
Genomic DNA	mESC DNase-seq		0.482
mESC DNase-seq	mESC DNase-seq	Window size=2, max K=6	0.410
mESC Nrf ChIP	mESC DNase-seq		0.589
HEK293T DNase-seq	HEK293T DNase-seq		0.857
H7ES DNase-seq	H7ES DNase-seq		0.881
GM12878 DNase-seq	GM12878 DNase-seq		0.807
GM10248 DNase-seq	GM10248 DNase-seq		0.842
Frontal cortex DNase-seq	Frontal cortex DNase-seq		0.831
Fibroblast AG20443 DNase-seq	Fibroblast AG20443 DNase-seq		0.836
Fibroblast GM03348 DNase-seq	Fibroblast GM03348 DNase-seq		0.839
Epithelial adenocarcinoma DNase-seq	Epithelial adenocarcinoma DNase-seq		0.868
Frontal Cerebrum DNase-seq	Frontal Cerebrum DNase-seq		0.792
Cerebellum DNase-seq	Cerebellum DNase-seq		0.838

Supplementary Table 2: Oligonucleotides used in this work

CrispR guide RNA cloning oligos	
sgRNALocusA_fwoligo	CACC GTAGCCCAGGTGTGCAGGCT
sgRNALocusA_rvoligo	AAAC AGCCTGCACACCTGGGCTAC
sgRNALocusB_fwoligo	CACC GAGCAGGTGACAATTTCAGA
sgRNALocusB_rvoligo	AAAC TCTGAAATTGTCACCTGCTC
Homology-directed repair tailed primers	
LocusA_HDR_fw	TTCGAATCACTCCATGTGAGTATCACAGAACGGGTGCAGGAGATCAGTTGCTGTGATGGATAGAC CGAAAGGATGGGAGTACTAAGCT
LocusA_HDR_rv	ACCACAGTGACATCCGCCCTGAAGCAGGCAGCAGAGCAGATGCTCTGAGATGCTTGCTTTCTGT CTCAGTACTTTGTCCGTGCTGAC
LocusB_HDR_fw	GTGAGGCTGGTGGAAGACCACAAACAGGGGAGGGTCATGGAGAGGTCAGGGGTTGCCAACAAAA CGAAAGGATGGGAGTACTAAGCT
LocusB_HDR_rv	CCTGGTCCAGACACTCATTCTCAAGCTTCCTCATGCTCTTGTGGGAAGCATAGATGCTTTCAGAG CTCAGTACTTTGTCCGTGCTGAC
SLOT library prep and qPCR validation primers	
LocusA_upstream_fw	CCGGTGGGGTCTCAGTGTTA
LocusA_downstream_rv	CACTGTTCTTTGTGCCATCCCTTTA
LocusB_upstream_fw	TCTACCACACTTCCAGCAGG
LocusB_downstream_rv	CAGATGTGAGGTCAAGGCTGGG
Library_rv_reversecomplement	CGTCAGCACGGACAAAGTACTGAG
Library_fw_reversecomplement	AAGCTTAGTACTCCCATCCTTTCG
Library_fw_extension	GCCGAAAGGATGGGAGTACTAAGCT
Library_rv_extension	CTCAGTACTTTGTCCGTGCTGACG
InternalPrimer_extension	AGGCCTTTCGACCTGCATCCA
PhrPE1_BcO	CTCTTTCCCTACACGACGCTCTTCCGATCTaactc GCCGAAAGGATGGGAGTACTAAGCT
PhrPE1_BcP	CTCTTTCCCTACACGACGCTCTTCCGATCTctgga GCCGAAAGGATGGGAGTACTAAGCT
PhrPE1_BcQ	CTCTTTCCCTACACGACGCTCTTCCGATCTggact GCCGAAAGGATGGGAGTACTAAGCT
PhrPE1_BcR	CTCTTTCCCTACACGACGCTCTTCCGATCTtctgc GCCGAAAGGATGGGAGTACTAAGCT
PhrPE1_BcS	CTCTTTCCCTACACGACGCTCTTCCGATCTaaccg GCCGAAAGGATGGGAGTACTAAGCT
PhrPE1_BcT	CTCTTTCCCTACACGACGCTCTTCCGATCTctctg GCCGAAAGGATGGGAGTACTAAGCT
PhrPE1_BcU	CTCTTTCCCTACACGACGCTCTTCCGATCTggtaa GCCGAAAGGATGGGAGTACTAAGCT
PhrPE1_BcV	CTCTTTCCCTACACGACGCTCTTCCGATCTaagct GCCGAAAGGATGGGAGTACTAAGCT
PhrPE1_BcW	CTCTTTCCCTACACGACGCTCTTCCGATCTtcgtc GCCGAAAGGATGGGAGTACTAAGCT
PhrPE1_BcX	CTCTTTCCCTACACGACGCTCTTCCGATCTccaat GCCGAAAGGATGGGAGTACTAAGCT
PhrPE1_BcY	CTCTTTCCCTACACGACGCTCTTCCGATCTgcgta GCCGAAAGGATGGGAGTACTAAGCT
PhrPE1_BcZ	CTCTTTCCCTACACGACGCTCTTCCGATCTtgagc GCCGAAAGGATGGGAGTACTAAGCT
SSPBMPE2_BcA	CATTCCTGCTGAACCGCTCTTCCGATCT ACATCGCTCAGTACTTTGTCCGTGCTGACG
SSPBMPE2_BcB	CATTCCTGCTGAACCGCTCTTCCGATCT GCCTAACTCAGTACTTTGTCCGTGCTGACG
SSPBMPE2_BcC	CATTCCTGCTGAACCGCTCTTCCGATCT TGGTCACTCAGTACTTTGTCCGTGCTGACG
SSPBMPE2_BcD	CATTCCTGCTGAACCGCTCTTCCGATCT CACTGTCTCAGTACTTTGTCCGTGCTGACG
IntPriPE2_BcA	CATTCCTGCTGAACCGCTCTTCCGATCT ACATCAGGCCTTTCGACCTGCATCCA

IntPriPE2_BcB	CATTCCTGCTGAACCGCTCTTCCGATCT GCCTAAGGCCTTTCGACCTGCATCCA
IntPriPE2_BcC	CATTCCTGCTGAACCGCTCTTCCGATCT TGGTCAGGCCTTTCGACCTGCATCCA
IntPriPE2_BcD	CATTCCTGCTGAACCGCTCTTCCGATCT CACTGAGGCCTTTCGACCTGCATCCA
PE1	AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGCTCTTCCGATCT
PE2	CAAGCAGAAGACGGCATACGAGATCGGTCTCGGCATTCCTGCTGAACCGCTCTTCCGATCT
DNase control primers	
NegDNase50b_1_fw	TTGACTGCTCCCAGGTAGAGA
NegDNase50b_1_rv	TCTTGGTGATTTCATTCATAGGC
NegDNase50b_2_fw	TCCATAATGATTTGGGGAAAG
NegDNase50b_2_rv	GAAAGTTCTGGAAGACAGTGCAT
NegDNase50b_3_fw	CCAACTGCCTCCATTAGAGC
NegDNase50b_3_rv	TGCATGCTTGTGAATGTCAA
PosDNase50b_2_fw	TTTGGAAACAACCACAGTGC
PosDNase50b_2_rv	CAATACGCAGCTTTGACCAG
PosDNase50b_4_fw	GTTAAACCCAGCCTCAGTGG
PosDNase50b_4_rv	CTTCCAGGGCCTTCTTTGAT
PosDNase50b_5_fw	TTCAGGGTCCAAATAGCAGTC
PosDNase50b_5_rv	TGTTGTTAGAATGGCCACCA
Nrf1 ChIP control primers	
NrfPos_1_fw	GGAGCCGCGAGACTATGTG
NrfPos_1_rv	GCAATGCCGCTTCCAC
NrfPos_2_fw	CTGCGCAGCACAGTGGAC
NrfPos_2_rv	GCGGGACTTCCTGTCTCAG
NrfPos_3_fw	CATGTCCGCTTGTAGGTGTG
NrfPos_3_rv	TGCGCACAGGTTTTCTACTG