nature portfolio

https://doi.org/10.1038/s41588-021-00947-3

Supplementary information

The dynamic, combinatorial *cis*-regulatory lexicon of epidermal differentiation

In the format provided by the authors and unedited

SUPPLEMENTARY INFORMATION

Table of contents

MATERIALS AND METHODS

Lead Contacts

Further information and requests for resources and reagents should be directed to and will be fulfilled by Lead Contacts Anshul Kundaje (akundaje@stanford.edu) and Paul A. Khavari (khavari@stanford.edu).

Materials Availability

This study did not generate new unique reagents.

Data Availability

ATAC-seq, ChIP-seq, PAS-seq, HiChIP, and MPRA experiments can all be found on the Gene Expression Omnibus (GEO): GSE181416. There are no restrictions on the datasets. hg19 annotations can be found at https://hgdownload.soe.ucsc.edu/downloads.html, and GENCODE annotations can be found at https://www.gencodegenes.org/human/release_19.html. FANTOM5 transcription factors can be found at https://fantom.gsc.riken.jp/5/sstar/Browse_Transcription_Factors_hg19. The HOCOMOCO database can be found at https://hocomoco11.autosome.ru/.

Code Availability

Integrative analysis code and scripts can be found at https://github.com/vervacity/ggr-project (DOI: https://doi.org/10.5281/zenodo.5161189), and the deep learning code can be found at https://github.com/kundajelab/tronn (DOI: https://doi.org/10.5281/zenodo.5160998).

Experiments and data processing

Cell culture

Primary human keratinocytes were isolated from fresh surgically discarded neonatal foreskin and cultured in Keratinocyte-SFM (Life Technologies 17005-142) and Medium 154 (Life Technologies M-154-500). Pen/Strep (Life Technologies 15140-122) and Anti-mycotic (Life Technologies 15240-062) were also added to the culture. Keratinocytes were induced to differentiate by addition of 1.2 mM calcium (added 12 hours after seeding at confluence) for 6 days in full confluence. Cells were harvested every 12 hours for a total of 13 timepoints and banked into cell pellets, viable batches (10% DMSO in media), or cross-linked with 1% formaldehyde and frozen down at -80 deg C. Further details can be found on the ENCODE portal under GGR experiment accessions (**Supplementary Table S1**).

ATAC-seq experiments

ATAC-seq[1] was performed on all 13 timepoints. Detailed methods can be found on the ENCODE portal under GGR experiment accessions (**Supplementary Table S1**). ATAC-seq read alignment, quality filtering, duplicate removal, transposase shifting, peak calling, and signal generation were all performed through thes ENCODE ATAC-seq pipeline (https://github.com/ENCODE-DCC/atac-seq-pipeline). Briefly, adapter sequences were trimmed, sequences were mapped to the hg19 reference genome using Bowtie2[2] (-X2000), poor quality reads were removed[3], PCR duplicates were removed[4] (Picard Tools MarkDuplicates), chrM reads were removed, reads with MAPQ > 30 were retained and read ends were shifted +4 on the positive strand or -5 on the negative strand to produce a set of filtered high quality reads. These reads were put through MACS2[5] to get peak calls and signal files. Finally, IDR analysis was run on the two replicate peak files to produce an IDR peak file that is the reproducible set of peaks across both replicates[6].

ChIP-seq experiments

ChIP-seq for H3K27ac, H3K4me1, H3K27me3, and CTCF were performed on 3 timepoints (days 0.0, 3.0, and 6.0). Detailed methods can be found on the ENCODE portal under GGR experiment accessions (**Supplementary Table S1**). ChIP-seq read alignment, quality filtering, duplicate removal, peak calling, and signal generation were all performed through the ENCODE ChIP-seq pipeline. Briefly, sequences were mapped to the hg19 reference genome using BWA[7], and poor quality reads were removed, PCR

duplicates were removed (Picard Tools MarkDuplicates) to produce a set of filtered high quality reads with high mapping scores (MAPQ > 30). These reads were put through MACS2 to get peak calls and signal files. Finally, reproducible sets of peaks across both replicates (naïve overlap peaks) were used for all downstream analysis. The full pipeline can be found at https://github.com/ENCODE-DCC/chip-seq-pipeline2.

PAS-seq experiments

PAS-seq was performed on all 13 timepoints. Detailed methods can be found on the ENCODE portal under GGR experimental accessions (**Supplementary Table S1**). PAS-seq read alignment and quantification were performed using the ENCODE RNA-seq pipeline v2.3.1 (https://github.com/ENCODE-DCC/long-rna-seq-pipeline). Briefly, sequences were mapped to the hg19 reference genome with GENCODE V19 annotations using STAR aligner[8] (v2.4.1d), quantification was performed with RSEM[9] (v1.2.21), and signal files were produced with STAR and ucsc tools (v3.0.9, http://hgdownload.soe.ucsc.edu/admin/exe).

DESeq2[10] was used to identify gene sets that were significantly differentially expressed (adjusted *p*-value < 0.05) in each time point relative to timepoint day 0. We used GSEA[11] (v3.0) to identify enriched functional terms for each differential gene set. We used the GseaPreranked tool and classic scoring scheme, to determine the GSEA normalized enrichment score (NES) for skin-relevant gene sets from MSigDB[11], specifically CORNIFIED_ENVELOPE, KERATINIZATION, and KERATINOCYTE_DIFFERENTIATION.

HiChIP experiments

The HiChIP protocol was performed as previously described[12] using antibody H3K27ac (Abcam, ab4729) on 10 million cells per sample with the following modifications. Samples were sheared using a Covaris E220 using the following parameters: Fill Level = 10, Duty Cycle = 5, PIP = 140, Cycles/Burst = 200, Time = 4 minutes and then clarified by centrifugation for 15 minutes at 16100 rcf at 4° C. We used 4 ug of antibody to H3K27ac and captured the chromatin-antibody complex with 34 uL Protein A beads (Thermo Fisher). Qubit quantification following ChIP ranged from 125-150 ng. The amount of Tn5 used and number of PCR cycles performed were based on the post-ChIP Qubit amounts, as previously described[12]. HiChIP samples were size selected by PAGE purification (300-700 bp) for effective paired-end tag mapping and where therefore removed of all primer contamination. All libraries were sequenced on the Illumina HiSeq 4000 instrument to an average read depth of 300 million total reads.

HiChIP paired-end reads were aligned to the hg19 genome using the HiC-Pro pipeline[13]. Default settings were used to remove duplicate reads, assign reads to MboI restriction fragments, filter for valid interactions, and generate binned interaction matrices. HiC-Pro filtered reads were then processed using hichipper[14] using the {EACH, ALL} settings to call HiChIP peaks to MboI restriction fragments. HiC-Pro valid interaction pairs and hichipper HiChIP peaks were then processed using FitHiChIP[15] to call significant chromatin contacts using the default settings except for the following:

MappSize=500, IntType=3, BINSIZE=5000, QVALUE=0.01, UseP2PBackgrnd=0, Draw=1, TimeProf=1.

Analysis of epigenomic and transcriptomic landscapes

Genome annotations: reference genome, transcription factors, known motifs

We use reference genome hg19 and GENCODE v19[16]. For conversions between Ensemble IDs and HGNC, we use the biomaRt package[17] in R. For transcription factors, we use the FANTOM5 list of transcription factors[18] (**Supplementary Table S11**). For conversion of Entrez IDs to Ensembl IDs, we use the biomaRt package in R.

For our known motif compendium, we use the HOCOMOCO resource[19]. To improve the quality of the motifs, we first remove non-informative bases on the ends of all position weight matrices (PWMs) in the database by clipping positions with information content (IC) < 0.4 from the ends in, until we hit a position with IC > 0.4. We reduce redundancy in this database using the RSAT matrix clustering methodology[20]. In brief, we cross correlate all motifs to all other motifs in the database, getting both the max raw cross correlation (cor) and the max normalized cross correlation (Ncor). The Ncor is the max cross correlation normalized by a width metric (divide the length of the best cross correlated alignment of the two PWMs by the number of overlapping base pairs between the two PWMs). We use 1 - Ncor as a distance metric to build a hierarchical clustering of the PWMs. We then merge PWMs from the leaves of the hierarchical clustering tree towards the root, stopping at each branch when cutoffs for

cor and Ncor (cor < 0.8, Ncor < 0.65) are passed. These cutoffs are the ones empirically derived in the RSAT matrix clustering study. We track all PWMs that were merged as well as associated Ensembl IDs for the corresponding transcription factors.

Determining a keratinocyte atlas of cis-regulatory elements

To determine the landscape of accessible regulatory elements across keratinocyte differentiation, we take the union set of the ATAC-seq peaks across all timepoints, using bedtools merge[21], to determine an atlas of cis-regulatory elements (CREs). We use the IDR peak files for each timepoint as the peak set for that timepoint. This CRE atlas consists of 225,996 accessible regions that are accessible at some timepoint in differentiation. At this point in the analysis it was noted that days 3.5, 4.0, and 5.5 had small differences that could be attributed to a growth response from media changes, which were not noted to significantly change the regions included in the CRE atlas but could have important effects on the accessibility signals and downstream quantitative analyses. These timepoints were therefore removed for all downstream analyses. With our valid timepoints we generated a signal coverage matrix with the following computational pipeline. At the biological replicate level, we determined the transposase-corrected cut sites (the single base pair locations of transposase binding events on genomic DNA) from the sequencing reads by taking the read ends and correcting the positions to be +4 on the 5' end and -5 on the 3' end. We then count the number of cut sites that fall into each element of our CRE atlas to get transposase events per biological replicate sample. This gives us a count-based matrix of (regions,

samples). This count matrix with replicate information can be appropriately analyzed with DESeq2 with its underlying assumptions[10]. We thus use DESeq2 on all pairs of timepoints to get all CREs that have differential signal between any pair of timepoints, using an FDR of 0.0005 to give us a post-analysis Bonferroni corrected FDR of 0.05 across all tests. Under this analysis framework, 47,835 CREs (21% of the CRE atlas) were found to be dynamically accessible across differentiation.

To determine homogeneity and purity of the cell cultures used for data generation across keratinocyte differentiation, we utilized available single cell ATAC-seq (scATAC-seq) data[22], which was provided as counts in regions. We used the master list of regions derived from the scATAC-seq data and obtained read counts per replicate per timepoint within each region from our ATAC-seq data. This count matrix, along with the scATAC-seq count matrix, were normalized with the DESeq2 regularized log transform to obtain normalized signal matrices. The signal matrices were then analyzed with UMAP[23] with nearest neighbors n=15 and all other settings as defaults, and our ATAC-seq replicate timepoints were projected into the reduced dimensionality space.

Time series clustering of dynamic CREs with replicate reproducibility

To group the dynamically accessible CREs into defined trajectories across time, we utilized Dirichlet Process-Gaussian Process (DP-GP) time series clustering with replicate reproducibility. This analysis framework extends DP-GP time series clustering[24] to consider replicates and to determine which clusters are reproducible across replicates. First, we sum the transposase event counts for each biological

replicate into a pooled count for each timepoint. This pooled count matrix is then used to calculate the DESeq2 regularized log transform to get a normalized signal matrix, where each CRE has a normalized value across timepoints. This same regularized log transform is applied to count matrices for replicate 1 and replicate 2, to generate similar normalized signal matrices for each replicate that are all normalized to the same transform. Then, the signal matrix with the pooled data is subsampled (n=5000 for speed, since the algorithm was originally built to run effectively at the scale of thousands of genes, not tens of thousands of regions) with the default parameters, providing the initial set of time series clusters. The cluster set is filtered for cluster size such that any cluster that has a total membership of CREs < 2% of all dynamically accessible CREs is removed. The cluster set is further filtered to remove non-dynamic trajectories, which are the clusters whose multivariate Gaussian process does not reject the null hypothesis of no change across time (in other words, the 0-vector falls in the 99.9% multivariate confidence interval). We then run reproducibility in the following manner. For each CRE, with its corresponding signal trajectory across time, we assign the CRE to each cluster that it could match, and this is also done for the pooled signal trajectory of that CRE as well as the signal trajectories in the separated replicates. The CRE matches a cluster if it's in the multivariate confidence interval (CI 0.95) for the trajectory, and is correlated by Spearman and Pearson correlation (p<0.05). We then only keep cluster matches for that CRE if all three trajectories – pooled, replicate 1, and replicate 2 – were matches in that cluster. If there is more than one matched cluster, the CRE is assigned to the cluster for which it is the least Euclidean distance away from the

mean trajectory. If there are no matched cluster, the CRE is considered irreproducible across time and discarded. After this is done for all CREs, any clusters that do not have matched CREs are discarded. This framework thus allows for utilizing replicate information within a time series framework to improve clustering as well as cluster membership. Under this analysis, 15 time series patterns of accessibility were found in keratinocyte differentiation, comprising 40,103 dynamically accessible and time series reproducible CREs.

Analysis of histone modifications in the CRE atlas

To characterize the diversity of CREs by histone modification, histone marks were analyzed with an accessibility-centric approach. For each histone mark (H3K27ac, H3K4me1, and H3K27me3), a union set of regions was generated by taking the CREs, extending the flanks on either side by 1kbp, and keeping any CREs that overlapped peaks for that mark across any of the timepoints. This analysis finds 83,785 CREs marked by H3K27ac, 122,395 CREs marked by H3K4me1, and 36,084 CREs marked by H3K27me3. We then generated count matrices for each set of CREs in the following manner. At the biological replicate level, we determined the midpoints from the paired sequencing reads as estimated positions where the histone was present on genomic DNA. We then count the number of read midpoints that fall into each flank-extended element of each CRE to get marked histone events per biological replicate sample. This gives us a count-based matrix of (regions, samples). This count matrix with replicate information can be appropriately analyzed with DESeq2 with its underlying

assumptions[10]. We thus use DESeq2 on sequential pairs of timepoints to get differentially marked CREs across time. Given three possible transitions (increase, no change, decrease in histone mark signal), and three timepoints (day 0, 3, and 6) for which histone mark data was collected, we enumerate 9 possible patterns for histone marks across time.

Analysis of chromatin states in the CRE atlas

To characterize the diversity of CREs by chromatin state, the histone mark analysis from above was used to consider all histone marks together. Chromatin states were generated by enumeration. With 9 possible patterns for each histone mark and three assayed marks, the total possible chromatin states is 729. However, most of the possible states do not appear, demonstrating a much more limited set of states.

Determining the transcriptomic atlas of keratinocyte differentiation

To determine the landscape of transcripts across keratinocyte differentiation, we first determine the set of expressed genes at each timepoint. We do this by first normalizing the full matrix of protein-coding transcripts across timepoints using the rlog function from DESeq2[10], and then setting an empirical threshold based on the best separation of a Gaussian mixture model on the rlog normalized values (threshold = 4.0). We then take the union of all expressed genes across timepoints to determine the transcriptomic atlas, which consists of 12,190 genes. We then use DESeq2 on all pairs of timepoints to get all genes that have differential signal between any pair of

timepoints, using an FDR of 0.0005 to give us a post-analysis Bonferroni corrected FDR of 0.05 across all tests. Under this analysis framework, 5,046 genes (41% of the transcriptome atlas) were found to be dynamically accessible across differentiation.

Time series clustering of dynamic genes with replicate reproducibility

To group the dynamic genes into defined trajectories across time, the same framework used for the dynamic CREs was also utilized for the dynamic genes (see above section, “Time series clustering of dynamic CREs with replicate reproducibility”). Under this analysis, 11 time series patterns of expression were found in keratinocyte differentiation, comprising 3,610 genes (29% of the transcriptomic atlas) that are dynamic and time-series reproducible.

Analysis of chromatin conformation

To determine a set of loops for downstream analyses, a replicate-based analysis was run to get replicate reproducible loops. For each timepoint, loops were generated for the pooled data (aggregated across both replicates), replicate 1, and replicate 2. Consensus loops were generated by getting the consensus endpoints from the union merge across the pooled, replicate 1, and replicate 2 endpoints. These were filtered such that each loop had a non-zero value for the pooled version, replicate 1 version, and replicate 2 version. These values were run through IDR (p<0.05) to keep loops that were replicate consistent[6]. These loops were then merged across timepoints to get the

union set of replicate consistent loops across differentiation. Under this analysis, 101,884 loops were replicate consistent.

Linking by proximity

We utilize an exponential decay function $e^{(-\lambda d)}$ to compute a linkage score between each ATAC-seq peaks and each expressed genes separated by distance d. Since previous work has shown that the median distance for functional distal regulatory elements to gene TSSs is 25kb[25], we fit the exponential decay function such that the median score is at 25kb (i.e., $\lambda = \ln(2)/25000$). We then keep all peak-gene links that are within 100kb of each other. For curating a gene set linked to a region set, we use the above links to get genes that are proximally linked to the regions where 1) the genes are expressed at some point in the timecourse, 2) the gene TSS is within 100kb upstream or downstream of a region, 3) the summed score for the gene is > 0.5. For example, if two regions are within 100kb of a gene TSS and the sum of the link scores for the two regions is 0.51, then the gene is kept as part of the downstream gene set, with the corresponding summed score. We then use this summed score to rank the genes so that a ranked enrichment tools can be used. This linking and scoring strategy is the main strategy used to find gene sets to use in gene set enrichment analyses.

Deep learning on dynamic regulatory DNA sequence

Convolutional neural networks on DNA sequence

We trained multi-task convolutional neural networks (CNNs) to accurately map 1 kbp DNA sequence regions across the genome to quantitative read outs of chromatin accessibility and multiple histone marks in each time point of keratinocyte differentiation. CNNs can learn complex sequence patterns that are predictive of genome-wide chromatin accessibility and histone mark profiles. We use a multi-stage, transfer learning training regimen to maximize prediction performance and model stability by leveraging large compendia of chromatin accessibility data across 100s of diverse tissues.

Architecture

We used the previously optimized multi-task Basset CNN architecture for predicting genome-wide chromatin accessibility from DNA sequence across multiple samples[26]. The inputs to the model are 1 kbp long DNA sequences that are one-hot encoded (A=[1,0,0,0], C=[0,1,0,0], G=[0,0,1,0], T=[0,0,0,1]). The Basset model has three convolutional layers with the following parameters: the first layer has 300 filters of size (1, 19) and stride (1, 1) followed by batch normalization, a ReLU non-linearity, and max-pooling with size (1, 3) and stride (1, 3); the second layer has 200 filters of size (1, 11) and stride (1, 1) followed by batch normalization, a ReLU non-linearity, and max-pooling with size (1, 4) and stride (1, 4); the third layer has 200 filters of size (1, 7) and stride (1, 1) followed by batch normalization, a ReLU non-linearity, and max pooling with size (1, 4) and stride (1, 4). After the convolutional layers there are two fully connected

layers, each with 1000 neurons, followed by batch normalization, a ReLU non-linearity, and dropout where the keep probability is 0.7. The final layer mapped to multiple outputs (multi-task output) spanning the time points and each of the different types of molecular read outs (chromatin accessibility or histone marks). We use binary or continuous output labels and associated loss functions in the multi-stage training (see below). When training on binary labels (accessible vs. not accessible or bound vs. unbound), we use the binary cross-entropy loss function with logistic outputs. When training on continuous, quantitative measures of accessibility or histone marks, we use the mean-squared error loss function with linear outputs. The multi-task loss is the sum of the loss over all tasks.

Multi-stage transfer learning regimen

We bin the genome into 1 kbp windows with a stride of 50 bp. Each bin can serve as an example in a training, validation/tuning or test set. We divide chromosomes into 10 folds (**Supplementary Table S7**). We use a cross-validation set up where we use 8 folds for training, 1 for validation/tuning, 1 for testing.

We use a multi-stage training regimen to maximize performance and model stability. In stage 1, we train a 'reference' multi-task CNN model with randomly initialized parameters (variance scaling initialization, ie Xavier intialization) on DNase-seq and TF ChIP-seq data from a large collection of biosamples from the ENCODE and Roadmap Epigenomics Project[27,28]. All datasets used are detailed in **Supplementary Tables S8, S9**. In this stage, the labels associated with each input sequence are binary. A 1 kbp

sequence in the genome is assigned a positive label for a particular task (DNase-seq or TF ChIP-seq in a specific biosample), if the central 200bp of the sequence overlaps a DNase-seq or TF ChIP-seq peak in the biosample by at least 50%. All other bins in the genome are assigned negative labels for that task. The possible negative labeled bins significantly outnumber the positive labeled bins, since much of the genome is not accessible (or not TF bound). Hence, we use a subset of informative negative examples from the training chromosomes to train the models. For each task, we include negative labeled bins flanking every positive labeled bin (3 flanks, stride 50bp, on either side of the region). We further sample negatively labeled bins (half as many positive bins in the task). Finally, we include bins that overlap a comprehensive catalog of DNase-seq peaks[28]. This generates a dataset with reasonable class imbalances per task while maintaining diversity in negative examples[29].

In stage 2, we initialize a multi-task CNN model with the parameters derived from the reference ENCODE/Roadmap model[30] and then train it to map DNA sequence bins to binary labels corresponding to important region sets as derived in the characterization of the epigenomic landscape. These important region sets include: ATAC-seq, H3K27ac, H3K4me1, and H3K27me3 region sets by timepoint, the region sets defined by accessibility time series clustering, region sets defined by dynamic and static histone modifications, region sets defined by dynamic and static chromatin states, and region sets from TF ChIP-seq experiments for CTCF, TP63, ZNF750, POL2, and KLF4. In total these region sets comprise 119 binary label sets used for multitask

training. The genomic bins for training and their associated binary labels for each of the tasks are constructed as described above[31].

In the final stage 3, we initialize a multi-task CNN with the parameters of the binary keratinocyte model from stage 2[32]. We then train the model CNN with the mean-squared error regression loss function to map DNA sequence bins to continuous, quantitative measures of ATAC-seq, H3K27ac ChIP-seq and H3K4me1 ChIP-seq in our keratinocyte differentiation time course (19 tasks). The genomic bins used for training cover the union of peaks across all time points. For each 1 kbp sequence bin, we compute the average of the log of the smoothed depth-normalized read coverage (log of the MACS2 fold-enrichment of smoothed observed 5' end counts relative to expected local Poisson background) over the central 200 bp of the bin for ATAC-seq or over the entire 1 kbp for histone marks. The average is computed using bigWigAverageOverBed (column mean0). The average signal scores are normalized using quantile normalization across all time points for each of the assays (ATAC, H3K27ac, and H3K4me1). These normalized scores are used as quantitative labels for each bin.

We use the same cross-validation folds for training, tuning and testing across all stages. The model parameters are transferred across stages to exactly match the cross-validation fold structure. Hence, for each fold, the test sets are completely held-out across all stages of training. This multi-stage training set up allows the model to utilize larger sets of existing data to improve its understanding of DNA sequence and regulatory logic encoded in the human genome.

Training hyperparameters

The following hyperparameters were used for all models at all stages. We train for a maximum of 30 epochs with early stopping, where the patience (number of epochs of nonimproved performance before stopping) is 3 and the metric considered is average AUPRC across all tasks on the validation set. The loss function for classification models is binary cross entropy, and the loss function for regression is mean squared error (MSE). The optimizer used is RMSprop with a learning rate of 0.002, a decay of 0.98, and a momentum of 0.0.

Performance evaluation

We evaluate on the held-out test chromosomes of each fold, calculating our performance metrics across the entire length of the chromosomes (genome-wide evaluation). For each task, we use the area under the precision-recall curve (AUPRC) to measure performance of the binary models and Spearman's R and Pearson's R to measure performance of the regression models.

Prediction calibration through quantile normalization

Using MSE loss on regression models provides effective ranking across predictions in the same task, but the prediction outputs may not be well calibrated to match the observed output labels. As such, we rescale the model's continuous output predictions by quantile normalizing the distribution of the model predictions with respect to the distribution of the ground truth measured labels. We obtain prediction scores for a

random set of 1000 examples, which then provides us with a distribution of predicted scores and a corresponding distribution of the labels. We can then use those distributions to quantile match a prediction score value to a label score (for example, we can determine that a prediction score is in the 90th percentile of the distribution of prediction scores and should be matched to the 90th percentile of the distribution of label scores). Importantly this re-scaling does not actually change the performance of the model, it simply re-calibrates the output. Additionally, given that the continuous signal labels across tasks are quantile normalized relative to each other, the re-calibration of the prediction scores also normalizes the prediction scores across tasks.

Inference of predictive motif instances

Overview

The multi-task CNNs, described above, map every candidate regulatory DNA sequence to quantitative measures of chromatin accessibility at each time point in the differentiation time course. We developed an interpretation framework to interrogate the model and decipher motif instances in each candidate element that are predictive of chromatin accessibility at each time point. First, we use gradient based feature attribution methods to decompose the predicted output (at each time point) for an input sequence in terms of contribution scores of each nucleotide in the sequence. We develop methods to stabilize and normalize the scores. We develop stringent null models to identify statistically significant contribution scores. We then use a large

compendium of pre-compiled TF motifs to scan and score the sequences as well as the contribution score profiles. We develop stringent null models to infer predictive motif instances that have statistically significant contribution scores and sequence match scores. The following sections provide details for each of these steps.

Estimating nucleotide-resolution contribution scores

The gradient of the predicted output with respect to each base at each position in the input DNA sequence, gated by the observed base, estimates the sensitivity of the output to infinitesimal changes in the input[33]. This measure of importance is often referred to as input-gated gradients. The method is efficient since a single backpropagation pass can be used to estimate the contribution of all nucleotides in an input DNA sequence to a specific output prediction.

We compared the input-gated gradient scores to contribution scores derived from another related approach called DeepLIFT[34] on a subset of the time points. DeepLIFT backpropagates a score, analogous to gradients, which is based on comparing the activations of all the neurons in the network for the input sequence to those obtained from neutral 'reference' sequences. We use 12 dinucleotide-shuffled versions of each input sequence as reference sequences. We used the DeepSHAP implementation of DeepLIFT (https://github.com/slundberg/shap/blob/0.28.5/shap/explainers/deep/deep_tf.py) to obtain contribution scores for all observed bases in each sequence.

We estimated input-gated gradient and DeepLIFT contribution scores for all nucleotides in all sequences with respect to quantitative chromatin accessibility predictions for three time points (0h-early, 3h-mid and 6h-late), using each of the models for the 10 folds of cross-validation. For each method, we averaged the scores for each sequence in each time point across all the 10 folds. For each sequence, we used cosine similarity to compare the average input-gated gradient and DeepLIFT score profiles separately. We observed high similarity between input-gated gradients and DeepLIFT scores (median cosine similarity across all sequences and all the 3 time points = 0.8736). While gradient based scores are often more unstable and less accurate than DeepLIFT scores, the regularization of our models via the multi-stage transfer learning and averaging over folds, greatly stabilizes the gradient based scores. Hence, we decided to use input-gated gradient scores as contribution scores for all downstream analyses, since it is more efficient than DeepLIFT and produces very similar contribution score profiles with respect to motif instance discovery.

Estimating statistically significant contribution scores

For each input sequence, we compute input-gated gradient score profiles from dinucleotide shuffled versions of the sequence. We use these scores to construct an empirical null distribution of contribution scores for that sequence. We use that empirical null distribution to derive empirical statistical significance of the observed contribution scores. We use a threshold of $p < 0.01$ to call statistically significant scores. The scores of all positions that do not pass the significance threshold are set to 0.

Normalization of contribution scores

We normalize the contribution score profile of each sequence by dividing the score of each position by the sum of the absolute value of contribution scores across the entire sequence and multiplying them by the predicted output.

Trimming contribution scores

We observed that statistically significant contribution scores peaked within 160 bp for the peak summit. Hence, we trim the DNA sequences from its original 1000bp context to the central 160bp for downstream analyses. We further eliminate the trimmed sequences from downstream analyses that have less than 10 base pairs of significant scores. These shorter sequences are also compatible with testing in reporter constructs.

Average contribution score profiles across all folds for each sequence in each time points

We estimate contribution score profiles for each sequence with respect to predictions in each of the time points using models from each of the 10 folds. These score profiles are filtered for statistical significance, normalized and trimmed as described above. We average the contribution scores of each position in each sequence for each time point, across the 10 folds. We compute the 99% confidence interval for each position using the scores from the 10 folds. If the confidence interval

includes 0, the score of the position is set to 0, else it is set to be the average score across the 10 folds.

Validation of contribution scores by Allele-sensitive ATAC (asATAC) analysis

We utilized our ATAC-seq data to determine allele-sensitive accessible sites. Since the data was collected from primary samples, we were able to utilize an allele-sensitive ATAC-seq analysis to determine a number of single nucleotide polymorphisms (SNPs) that exhibited significant allelic imbalance of ATAC-seq reads. Utilizing QuaSAR[35], a computational framework for calling genotypes and allelic imbalanced sites, we were able to call 16,686 heterozygous SNPs and capture 283 SNPs with statistically significant (FDR < 10%) allelic sensitivity across our two patient samples (and across all ATAC timepoints). We estimated contribution score profiles for the sequences containing each of these 16,686 SNPs, to determine if the SNP locations overlapped statistically significant contribution scores and whether the models predicted differential accessibility prediction for the two alleles.

Identifying dynamic predictive motif instances using sequence match and contribution scores

We identify dynamic predictive motif instances in each input sequence across time points, for each of the known motifs in the motif compendium, by scanning and scoring the sequence as well as the dynamic the contribution score profiles derived from the model.

First, for each PWM motif, we compute sequence match scores at every position in each sequence. The scanning and scoring can be implemented as a convolution operation. Hence, we use the deep learning framework to implement a single convolutional layer with filters corresponding to each of the PWMs in the deep learning framework. When loading the PWM weights into the filters, we pad the weights to get all filters to be the same size, normalize by the length of the nonzero weights (divide by the length), and convert the weights to a unit vector (divide by L1 norm). We use the convolutional layer to scan and score all PWMs across the forward and reverse complement of each one-hot encoded sequence. We also use the same operation to scan and score dinucleotide shuffled versions of each of the genomic sequences. We thus obtain an empirical null distribution of match scores for each PWM for each sequence. We identify positions with significant sequence match scores as those that pass $p < 0.05$ based on the empirical distributions. For any sequence, the significant positions based on sequence match scores will be identical across all time points.

Next, we use the PWMs to scan and score the dynamic contribution score profiles for each sequence in each time point. Essentially, we repeat the same convolution operation using PWM filters but using the contribution score profiles to weight the one-hot encoded sequences and their reverse complements. Hence, we obtain contribution weighted match scores to the PWMs. We once again retain statistically significant contributed weighted match scores, using a $p < 0.05$ threshold, based on null distributions of the contribution weighted match scores for dinucleotide shuffled sequences. We compute these contribution weighted match scores and

significance separately for the positive contribution scores and negative contribution scores, as negative scores can influence PWM weights detrimentally (e.g. negative PWM weight value that now contributes positively because of a negative contribution score.)

Our final set of predictive motif instances for each sequence in each time point correspond to positions that have significant sequence match scores and significant contribution weighted match scores. Since the contribution score profiles for each sequence can change across time points, the predictive motif instances are dynamic across time points.

Identifying significant differential motifs between two sets of sequences

We developed an approach to identify significant differential motifs between any foreground set of sequences relative to a background set of sequences. We use the sequences belonging to each dynamic trajectory as foreground region sets. First, we identify predictive motif instances for all PWMs in both sets using the method described above. We then use bootstraps of GC-content matched background genomic sequences (n=1000) that do not overlap any accessible peaks to estimate a null distribution of the number of PWM hits (average across the sequences in the bootstrapped background set). We then estimate an empirical *p*-value for each PWM in the foreground relative to these bootstrapped backgrounds. We use Storey's *q*-value method to perform a multiple hypothesis correction. We use an *q*-value threshold of 10% to identify statistically significant differential PWMs.

Comparison to conventional motif discovery using HOMER

To determine the utility of using neural net motifs for motif discovery, we compared the predictive motif inferred from our interpretation framework that leverages neural network derived contribution scores to an exemplar conventional motif discovery and enrichment method called HOMER (Hypergeometric Optimization of Motif EnRichment)[36]. HOMER was run with default parameters on foreground set of sequences underlying ATAC-seq peaks of CREs belong to each dynamic trajectory cluster against the background set of all CREs across the entire differentiation time course. The union set of motifs discovered by HOMER across the dynamic trajectories was compared to the union of predictive motifs inferred using our framework.

Comparison to predictive motif instances to all motif instances based on activity correlation to TF expression

We also compared our predictive motif instances to all motif instances identified solely based on sequence motif match scores by computing correlations of the motif activity across time points to RNA expression of TFs annotated to bind the motifs. Specifically, for the predictive motif instances, we use the average of the contribution-weighted motif match scores over all predictive instances of each motif at each time point as the measure of motif activity. For all motif instances based on sequence-only motif match scores, we used the average chromatin accessibility signal across peaks overlapping all instances of each motif as the activity scores.

Validation of predictive motif sites by TF ChIP-seq

The predictive motif instances are a subset of all sequence-based motif instances that also have significant contribution scores. We hypothesized that predictive motif instances are more likely to distinguish those that are bound by TFs from unbound motif instances. Hence, we used publicly available TF ChIP-seq to analyze occupancy over these motif sites. First, for each TF with available ChIP-seq data, we used our models to obtain all predictive motif instances of the PWM for the TF. We also collated a control set of motif instances with significant sequence match scores that are not marked as predictive and are matched for accessibility to the set of predictive instances. We matched for accessibility to account for confounding effects of differentially accessible regions. Using pre-processed normalized (MACS2 derived fold enrichment of smoothed 5'-end read coverage relative to local Poisson background) bigwigs from Cistrome[37], we contrasted the average the ChIP-seq signal profile over predictive motif instances versus control motif instances using a +/- 1 kbp window (20 p bins) around the instances[38].

Validation of predictive motif sites by ATAC-seq footprinting analysis

As above, we separated motif instances of each PWM into two sets depending on whether they were marked as predictive or not, matched for GC content of the surround sequencing context. We then utilized the HINT footprinting tool[39] to generate average ATAC-seq bias-corrected cut site coverage profiles over the two sets of motif

instances using a +/-250 bp window around the motif. We normalized the average footprint profile by computing the fold enrichment of average footprint signal at each position relative to a reference. The reference was computed by averaging the footprint signal in the 50bp flanks on either side of the 250 bp footprint window. We computed the 'average footprint height' as the area under the normalized footprint profile in a +/- 100 bp window around the motif center, excluding the central +/15 bp around the motif center. We computed the 'average footprint depth' as the area under the local maximum of the normalized average footprint profile within +/- 10bps on either side of the motif center.

Identifying putative TFs binding the motifs based on correlation of weighted PWM scores and TF expression

Since the contribution score profiles for each sequence are dynamic across the time points, the predictive motif instances within each sequence also have dynamic contribution scores across the time points. For each PWM, we identify the locations of all predictive motif instances across all the time points. Each instance is represented by an instance activity vector across time consisting of contribution scores (sum over all positions in the instance) for each of the time points. For a pre-defined set of sequences, we first identify all significant differential motifs relative to a background set (as described above). For each motif, we obtain a motif activity vector for the set of sequences as the average of the activity vectors over all its predictive instances in those sequences. We identify all candidate TFs associated with the motifs (TFs of the same

family often bind similar motifs). For each candidate TF, we extract the RNA-seq expression profile (variance stabilized rlog transformed counts from DESeq2) across the time points. We compute the Pearson correlation between the TF expression profile and its motif activity vector. We retain TFs that are expressed in keratinocytes (> 1 TPM) and exhibit a correlation of at least 0.75 with the activity vectors of associated motifs.

Homotypic motif syntax analyses

To estimate the effects of density (number of instances) of a motif of interest on accessibility at a specific time point, we designed synthetic DNA sequences by embedding varying number (from 1 to 6) of motif instances (the best matching sequence to the PWM of the motif) in the central 200 bp of 1 kb sequences randomly sampled from the genome avoiding the union of ATAC-seq peaks across the entire time course. We then used the models to predict accessibility for each sequence.

To characterize motif affinity rules in genomic regions, we used a surrogate score for affinity as the log odds scores of predictive motif instances. To determine functional enrichments of homotypic motif cluster region sets, proximity linking as described above was used to link regions to gene sets, and gProfiler[40] was used to determine enrichments.

Estimating interaction effects between motifs

In silico mutagenesis scores for motif instances

We use each set of candidate regulatory elements that belong to each trajectory of accessibility dynamics as a foreground set of sequences to identify significant differential motifs relative to the background set of all peaks. As described above, we identify predictive motif instances of differential motifs by scanning sequence and gradient based contribution score profiles with the known motif compendium. We use an *in-silico* motif mutagenesis approach to further corroborate and filter high-confidence predictive motif instances. Specifically, we expect *in-silico* perturbation of predictive motif instances to induce (1) a significant change in the predicted output of the model, (2) a significant change in the contribution scores across the sequence.

We use two complementary approaches to perturb motif instances. The first approach called the “motif scramble” method, randomly scrambles the 10 bp sequence around the center of a predictive motif instance. The scramble maintains the sequence composition of the window while destroying the precise sequence of the motif instance. The second approach, called the “point mutation” method, mutates the most influential base pair with the highest contribution score in the 10bp window around the center of the motif instance. The position is mutated to the base with the most detrimental predicted effect i.e. the base with the most negative gradient score at that position. For both types of mutations, we compute the ‘mutagenesis effect size of a motif instance’ as the difference in the predicted output (units of log depth normalized coverage of ATAC-seq signal) of the model for the mutated sequence relative to the wild type. We also recompute the contribution score profile of the mutated sequence and record the

difference in contribution score ('delta contribution score') of each position in the sequence relative to the wild type contribution score profile.

We generate null control distributions for the mutagenesis effect sizes and change in contribution scores as follows. First, we identify 10 randomly chosen positions that have non-significant contribution scores (set to 0 after thresholding for significance) in the central 200 bp of each wild type sequence. We expect mutations to these positions to have no significant effect on the output or contribution scores of other positions in the sequence. We mutate (point mutation or scramble around) each of these 10 positions and compute the difference in the output prediction as well the delta contribution scores for all the other positions in the sequence. We fit separate Gaussian null distributions to the mutagenesis scores and the delta contribution scores from these 10 expected null mutations. The use these null distributions to estimate the statistical significance (p-value < 0.1) of mutagenesis scores and delta contribution score profiles for each predictive motif instances.

We identify candidate epistatic partners of a motif instance in a sequence, as all other predictive motif instances in the sequence that overlap positions with significant delta contribution scores of a target motif instances in the sequence. We have previously described this approach as Deep Feature Interaction Maps[41].

Functional enrichment of co-occurring pairs of predictive motifs

We associate each of the regulatory sequences of accessible peaks supporting a combinatorial motif set to proximal genes as follows. For sequence, we first identify up

to two closest candidate genes (based on distance from TSS) within a +/- 500 kb of the sequence, such that the genes are expressed in at least one of the time points in our differentiation time course. We then restrict all peak-gene associations to those that exhibit significant correlation between the ATAC-seq enrichment of the peaks (log fold enrichment) and the RNA-seq (TPM) expression levels of the genes across the time course. We thus obtain a gene set that is putatively regulated by any combinatorial motif set. We then test the gene sets for enrichment of functional annotations using gProfiler[40]. We use a background set of all genes expressed at any timepoint in differentiation time course) to get functional enrichments. We keep combinatorial rules that are functionally enriched for skin-related terms. We also combine combinatorial rules that were discovered in different trajectories but marked with the motif combination to create a set of rules that are all distinct motif combinations.

Testing interaction effects between pairs of motifs with combinatorial *in silico* mutagenesis

Co-occurring pairs of predictive motifs in a regulatory sequence can have different types of quantitative joint effects on chromatin accessibility (depth normalized ATAC-seq read coverage). We explore three types of joint effects. Lack of motif interactions would manifest as independent, additive effects on coverage. Interactions between motifs learned by the model would manifest as multiplicative (additive in log space) or super-multiplicative effects (multiplicative in log space) on coverage. For all pairs of functionally enriched pairs of co-occurring motifs, we identified all the

sequences containing predictive instances of the pair (as described above). We then used two complementary approaches to test each instance of a pair of motifs for epistatic interactions.

First, we used the Deep Feature Interaction Map method[41] to score epistatic interactions between pairs of candidate predictive motif instances (say A and B) in a sequence. Specifically, as described in the motif ISM section, we infer the positions in the sequence that exhibit statistically significant delta contribution scores due to *in silico* mutations to motif A. If motif instance B overlaps any positions with significant delta contribution scores then it is estimated to have an interaction effect with motif A on ATAC-seq read coverage.

Next, we corroborate the DFIM scores, with an explicit combinatorial *in silico* motif mutagenesis approach using both the 'scramble' and 'point mutation' approach. Assume we have two motif instance A and B in a sequence that we would like to test for epistatic interactions using the model.

- We record the model's output with both motif instances intact in the sequence = *o*.
- We record the output after 'mutating' only motif A i.e. the sequence only contains an intact motif B = *b*.
- We record the output after mutating only motif B, i.e. the sequence contains an intact motif A = *a*.
- Finally, we record the output after mutating both motifs A and B, which is a baseline = *n*.

- We compute the marginal effect size of adding motif A relative to a null sequence that does not contain either of the motifs = *(a – n)*.
- We compute the marginal effect size of adding motif B relative to a null sequence that does not contain either of the motifs = *(b – n)*.
- We compute the joint effect of adding motif A and B relative to the sequence that does not contain either of the motifs = *(o – n)*

We then compare the joint effect size *(o – n)* to the sum of the marginal effect sizes *(a – n) + (b – n) = (a + b – 2n).* We run a Wilcoxon signed-rank test on the paired values (joint vs. sum of marginals) across all instances of a motif pair to determine whether the joint effects on the motif pair instances is significantly greater or less than the sum of the marginal effects.

Since the output predictions are in units of log depth normalized coverage, additivity in log units translates to multiplicative effects in units of coverage. If the joint effect is significantly larger than the sum of the marginal effects, motifs A and B have super-multiplicative effect on coverage. If the joint effect is significantly lower than the sum of the joint effects, motifs A and B exhibit a sub-multiplicative effect on coverage. A non-significant difference between the joint and sum of marginals indicates a multiplicative effect of motif A and B on coverage.

Design of Massively Parallel Reporter Assay (MPRA) to test intrinsic activity dynamics of combinatorial motif rules

MPRA design

We designed MPRA constructs guided by the combinatorial motif sets that have positive motif interaction scores using the 'motif scramble' and 'point mutation' motif perturbations. For each rule of interacting motif pairs, we randomly select 19 genomic subsequences of length 160 bp within accessible peaks, contain predictive instances of both motifs in the rule and exhibit positive interaction scores. We test the wild-type (genomic) sequence and all versions of the sequences in which the motifs are combinatorially mutated.

This sampling design allows us to test the following hypotheses:

1. Trajectory: does the motif combination produce a reporter activation pattern across time points (days 0, 3, and 6 in the in vitro model) that was predicted by the trajectory it was derived from?
2. Interactions: do the motif pairs exhibit multiplicative or super-multiplicative interaction effects on intrinsic reporter activity?

We include the following positive and negative controls. As positive controls, we use 316 TSSs of the highest expressed genes (at any time point in skin differentiation). As negative controls, we generate dinucleotide shuffled versions of 50 randomly selected

genomic test sequences selected above. We also select 50 negative controls from the genome that are not found in the master list of accessible regions across keratinocyte differentiation. The list of all constructs are in **Supplementary Table S12**.

Library cloning

The MPRA oligo library was synthesized using Agilent's oligo library synthesis platform. Each oligo sequence consisted of: 5'-FWD primer binding site-[ACTGGCCGCTTCACTG]-176 nt insert-XhoI-NheI-20 nt oligo barcode-REV primer binding site-[AGATCGGAAGAGCGTCG]-3'. The oligo library was amplified using the FWD primer 5'-GCTAAGGAATTCACTGGCCGCTTCACTG-3' and REV primer 5'-GCTAAGGGATCCCGACGCTCTTCCGATC-3', which add EcoRI and BamHI restriction sites, respectively. The resulting PCR product was gel purified, digested with EcoRI and BamHI, and ligated into the pGreenFire1 lentivector backbone. Takara Stellar competent cells were transformed with the plasmid library and a fraction of the bacteria were plated to ensure a library coverage of at least ten-fold. The remainder of the transformation was incubated overnight in Luria broth. Plasmids were isolated using the Qiagen Plasmid Plus Maxi kit. If insufficient colonies were obtained to ensure a library coverage of at least ten-fold, additional transformations were performed and plasmid preps were pooled. In the second cloning step, the plasmid library was digested with XhoI and NheI and ligated with an insert containing a minimal promoter and a short stuffer sequence consisting of the first 100 bp of luciferase. The luciferase sequence is not functional and merely provides a transcript sequence linked to each oligo barcode,

which is necessary for downstream sequencing library construction. The ligated plasmids were used to transform Stellar competent cells as described above. The final plasmid library pool was sequenced on an Illumina MiSeq to ensure an oligo library coverage greater than 90%.

Cell culture

Lentivirus was produced in 293T cells (Takara 632180) in 10cm plates. Cells were transfected with 3.75ug pUC MDG, 7.5ug pCMV Δ8.91, and 7.5ug plasmid library using Lipofectamine 2000 (Life Technologies). Viral particles were collected 48 hours post-transfection and concentrated using Lenti-X Concentrator (Takara). Lentivirus was titrated in primary keratinocytes to maximize viral transduction while minimizing lentiviral toxicity. For each MPRA biological replicate, 12 million keratinocytes were transduced in 15cm plates containing 5ug/mL polybrene. Cells were selected in 0.8ug/mL puromycin 24 hours post transduction. Once selected, cells were seeded for day 0, 3, and 6 timepoints of differentiation. At each timepoint, total RNA was isolated using Qiagen's RNeasy Plus kit and then used to generate MPRA sequencing libraries.

MPRA sequencing library construction

cDNA was synthesized from total RNA using SuperScript IV (ThermoFisher Scientific) using a gene specific primer that anneals to the MPRA transcript. The primer also contains a 15 nt degenerate sequence that serves as a transcript UMI. cDNA synthesis reactions were cleaned up using SPRIselect beads (Beckman) and amplified

using PrimeSTAR Max DNA Polymerase (Takara) for five PCR cycles to add Illumina sequencing adapters. Sequencing indexes were added in a second PCR step, which was monitored on a Stratagene MX3005P quantitative PCR machine to avoid library over-amplification. Final sequencing libraries were gel purified in a 2% agarose gel. Library concentration was determined using a KAPA Library Quantification Kit (Roche). Deep sequencing was performed on an Illumina NovaSeq 6000.

MPRA analysis

The DNA plasmid library was sequenced to capture the baseline fractions of each sequence in the library. Since the UMI is on read 1 and the barcode is on read 2 based on the primer locations, we perform paired ended sequencing. The reads are then trimmed (only the 20bps after the first 17 bps in read 2 constitute the barcode) and the UMI is associated with the read such that downstream analysis can proceed as single-ended data. These adjusted reads are aligned to the barcode sequences using bwa aln/samse with default parameters, and the aligned reads are then reduced by UMI to get unique read counts per barcode. These counts are then divided by the total to get the fractional value for each barcode in the library.

The MPRA library reads were sequenced and analyzed in the same fashion as the DNA plasmid library. The counts were then renormalized using the plasmid fractions by multiplying the MPRA counts by the plasmid fractions, converting to fractions, and multiplying by the total count across the MPRA library. In other words, the renormalization provides the counts per MPRA barcode assuming a uniform distribution

of barcodes in the library at the same sequencing depth as was actually performed. These counts are then run through regularized log transform from DESeq2 to get a normalized signal matrix. This normalized matrix is then used in downstream analyses.

To test trajectory patterns, the normalized MPRA signal for all sequences belonging to the pattern are collected for days 0, 3 and 6. Then day 3 and 6 read outs are compared to day 0 by a Wilcoxon signed rank test ($p < 0.05$) to determine differential signal between timepoints. If the measurements show differential signal for any of the two days, the trajectory is considered to have dynamic activity across the time course. Then, the mean (across all sequences) pattern of the MPRA signal across the three time points is compared to the corresponding average ATAC trajectory to determine a correlative match (Spearman rank correlation $p < 0.05$) in terms of the dynamics.

To estimate interaction scores for motif pairs tested in the MPRAs, we compare the distribution of normalized MPRA signal (log scale) of wild-type sequences containing both motifs to the expected log-additive effect of each individual motif. When motif a is scrambled, we note the MPRA signal = a. When motif b is scrambled, we note the MPRA signal = b. When both motif a and b are scrambled, we note the MPRA signal = n. Then, the expected log-additive signal for the wild-type sequence containing both motifs = (a – n) + (b – n). We then utilize the Wilcoxon signed rank test ($p < 0.10$) to determine whether there is a significant difference between the observed wild-type signal and the log-additive expected signal. A significantly positive score indicates a super-multiplicative effect of the motif pair. A non-significant score indicates a

multiplicative (log-additive) effect of the motif pair. A significant negative score indicates a sub-multiplicative effect of the motif pair.

Biochemical characterization of combinatorial rules

For the following assays, the genomic sequences tested can be found in **Supplementary Table S13**, within the sequence metadata as the “active” region.

Luciferase reporter assay

A lentiviral reporter construct was designed that contains a minimal promoter driving the expression of destabilized copGFP and luciferase separated by a T2A sequence. The construct also contains a CMV driven blasticidin S deaminase gene. Genomic sequences synthesized by IDT were inserted upstream of the minimal promoter by digestion of the vector with NheI, followed by a Gibson assembly using NEBuilder. Constructs were Sanger sequenced to confirm correct cloning. Lentivirus was made as described above and concentrated 50X. 125,000 primary progenitor keratinocytes were transduced with 15uL of concentrated lentivirus in media containing 5ug/mL polybrene and seeded in 6-well plates. 1-2 days after transduction, cells were plated onto 10cm plates in media containing 4ug/mL blasticidin HCl and selected for at least 3 days. In experiments in which a transcription factor knockout was performed, keratinocytes were co-transduced with 15uL of 50X concentrated lentivirus containing a LentiCRISPR v2 (Addgene #52961) construct, in addition to the reporter construct. The original construct

was modified such that a CMV promoter drives the expression of Cas9 instead of EF-1a. Guide oligos were cloned in by digestion with BsmBI, followed by a Gibson assembly using NEBuilder. 1-2 days after transduction, cells were plated onto 10cm plates in media containing 4ug/mL blasticidin HCl and 0.8ug/mL puromycin, selected for at least 3 days, and further cultured for at least 2 more days to allow guide cutting to occur. Cells were then seeded for day 0, 3, and 6 timepoints of differentiation. Lysate was collected in 1X PLB (Promega) and stored at -80°C prior to performing the luciferase assay. Genomic DNA was also isolated to determine lentiviral integration copy number for luciferase signal normalization. Luciferase assays were performed using a Tecan Infinite M1000 plate reader. Relative luciferase units (RLU) were normalized by both genomic lentiviral copy number and cell lysate. To determine lentiviral copy number, a qPCR was performed using primers that amplify part of the luciferase gene. A standard curve was obtained using a plasmid dilution series. Genomic DNA input was normalized using primers to the intron of LIPC. Cell lysate concentrations were determined using a Pierce Microplate BCA Protein Assay Kit – Reducing Agent Compatible (Thermo Fisher Scientific).

Chromatin immunoprecipitation

Human keratinocytes were cross-linked with 1% formaldehyde and chromatin was sonicated to an average fragment length of 150-500 bp. Chromatin was immunoprecipitated overnight at 4°C. Following cross-link reversal, samples were treated with RNaseA and the DNA was purified using a ChIP DNA Purification Kit (Zymo

Research). The following antibodies were used: CREB1 (Millipore), ETV5 (Proteintech), KLF4 (Sigma), ZNF750 (Sigma), CEBPD (ThermoFisher). For re-ChIP, samples were eluted in ChIP elution buffer (1% SDS, 50mM $NaHCO_3$) then diluted 10-fold in modified RIPA buffer without SDS (1% NP-40, 1% sodium deoxycholate, 1mM EDTA in PBS) for immunoprecipitation with second antibody. All ChIP used 2ug Ab/40ug chromatin, and the re-ChIP used 1 ug Ab. The qPCR primers can be found in **Supplementary Table S14**.

Engineered human epidermal organoids

Primary human keratinocytes were isolated from fresh surgically discarded skin and cultured in Keratinocyte-SFM (Life Technologies #17005-142) and Medium 154 (Life Technologies #M-154-500). Organotypic regeneration of human epidermis was performed as previously described[42]. Briefly, cells were first transduced with lentivirus containing pGreenfire reporter constructs and selected with puromycin for 2d post-transduction. After selection, 500,000 cells were seeded onto devitalized dermis, cultured for 7d and harvested. Biologic replicates were performed in all cases.

Tissue immunofluorescence

For immunofluorescence staining, tissue sections (7um thick) were fixed using 4% paraformaldehyde. Primary antibodies GFP (ThermoFisher, dilution 1:500) and filaggrin (Abcam, dilution 1:200) were incubated overnight at 4°C and secondary antibodies (AlexaFluor 488 or 555, ThermoFisher, dilution 1:1000) were incubated at room

temperature for 1h. Tissue samples were mounted with Duolink In Situ mouting media with DAPI (Sigma). Images were taken using a Zeiss Axio Observer Z1 fluorescence microscope and Zeiss Axiovision software.

Analysis of genetic variation and heritability

We utilized LD-score regression software (https://github.com/bulik/ldsc) to determine genome-wide significant variants. From UK Biobank (http://www.nealelab.is/uk-biobank/), we utilized the GWAs results for the following phenotypes (codes in parentheses): basal cell carcinoma (20001_1061), eczema (200002_1452), psoriasis (20002_1453, L12_PSORIASIS, L12_PSORI_NAS, L40), non-melanoma malignant neoplasms of skin (C3_OTHER_SKIN, C44, C_OTHER_SKIN), actinic keratosis (L12_ACTINKERA), rosacea (L12_ROSACEA, L71), seborrheic keratosis (L82), diseases of skin and subcutaneous tissue (XII_SKIN_SUBCUTAN), and other/unspecified disorders of skin and subcutaneous tissue (L12_SKINSUCUTISNAS). We additionally utilized two other GWAS studies for dermatitis (GWAS catalog: GCST003184) and acne (GWAS catalog: GCST006640) as cited in the main text.

To characterize putative TF subnetworks for each phenotype, we utilized our putative TF network derived from the validated combinatorial rules and only kept the rules that had significantly enriched LDSC coefficients ($p < 0.05$). The sum of mean value for the coefficient for each motif was used to generate the relative size of each TF/motif node.

Limitations and future enhancements

Our study has several limitations, providing scope for future enhancements. Our cis-regulatory lexicon is biased towards activators due to our modeling focus on active CREs in each time point. Models trained on markers of repression as well as differential effects across time points could reveal cis-regulatory sequences associated with dynamic repression. The current work also does not model the combinatorial effects of multiple CREs on gene expression. However, our chromatin-based models serve as a foundation for higher-order predictive models of gene expression which could be interpreted using similar *in silico* combinatorial perturbation strategies to decipher the distributed cis-regulatory code. We do not directly model the combinatorial influence of trans-acting factors on chromatin state and gene expression. Future modeling efforts, however, could be designed to jointly learn cis and trans regulatory logic from multimodal perturbation experiments[43,44]. Finally, extensions of these models to continuous cell state transitions from multi-modal single cell readouts of chromatin state and expression are exciting avenues for future research.

REFERENCES

1. Buenrostro, J. D., Giresi, P. G., Zaba, L. C., Chang, H. Y. & Greenleaf, W. J. Transposition of native chromatin for fast and sensitive epigenomic profiling of open chromatin, DNA-binding proteins and nucleosome position. *Nat Methods* **10**, 1213–1218 (2013).

2. Langmead, B. & Salzberg, S. L. Fast gapped-read alignment with Bowtie 2. *Nat Methods* **9**, 357–359 (2012).

3. Li, H. *et al.* The Sequence Alignment/Map format and SAMtools. *Bioinformatics* **25**, 2078–2079 (2009).

4. *Picard Toolkit*. (Broad Institute, 2020).

5. Feng, J., Liu, T., Qin, B., Zhang, Y. & Liu, X. S. Identifying ChIP-seq enrichment using MACS. *Nat Protoc* **7**, 1728–1740 (2012).

6. Li, Q., Brown, J. B., Huang, H. & Bickel, P. J. Measuring reproducibility of high-throughput experiments. *Ann. Appl. Stat.* **5**, 1752–1779 (2011).

7. Li, H. & Durbin, R. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics* **25**, 1754–1760 (2009).

8. Dobin, A. *et al.* STAR: ultrafast universal RNA-seq aligner. *Bioinformatics* **29**, 15–21 (2013).

9. Li, B. & Dewey, C. N. RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC Bioinformatics* **12**, 323 (2011).

10. Love, M. I., Huber, W. & Anders, S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology* **15**, 550 (2014).

11. Subramanian, A. *et al.* Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences* **102**, 15545–15550 (2005).
12. Mumbach, M. R. *et al.* HiChIP: efficient and sensitive analysis of protein-directed genome architecture. *Nat. Methods* **13**, 919–922 (2016).
13. Servant, N. *et al.* HiC-Pro: an optimized and flexible pipeline for Hi-C data processing. *Genome Biol.* **16**, 259 (2015).
14. Lareau, C. A. & Aryee, M. J. hichipper: a preprocessing pipeline for calling DNA loops from HiChIP data. *Nat. Methods* **15**, 155–156 (2018).
15. Bhattacharyya, S., Chandra, V., Vijayanand, P. & Ay, F. Identification of significant chromatin contacts from HiChIP data by FitHiChIP. *Nat Commun* **10**, 4221 (2019).
16. Frankish, A. *et al.* GENCODE reference annotation for the human and mouse genomes. *Nucleic Acids Res.* **47**, D766–D773 (2019).
17. Durinck, S., Spellman, P. T., Birney, E. & Huber, W. Mapping identifiers for the integration of genomic datasets with the R/Bioconductor package biomaRt. *Nat Protoc* **4**, 1184–1191 (2009).
18. Lizio, M. *et al.* Update of the FANTOM web resource: high resolution transcriptome of diverse cell types in mammals. *Nucleic Acids Res* **45**, D737–D743 (2017).
19. Kulakovskiy, I. V. *et al.* HOCOMOCO: towards a complete collection of transcription factor binding models for human and mouse via large-scale ChIP-Seq analysis. *Nucleic Acids Res* **46**, D252–D259 (2018).

20. Castro-Mondragon, J. A., Jaeger, S., Thieffry, D., Thomas-Chollier, M. & van Helden, J. RSAT matrix-clustering: dynamic exploration and redundancy reduction of transcription factor binding motif collections. *Nucleic Acids Research* **45**, e119–e119 (2017).
21. Quinlan, A. R. & Hall, I. M. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics* **26**, 841–842 (2010).
22. Rubin, A. J. *et al.* Coupled Single-Cell CRISPR Screening and Epigenomic Profiling Reveals Causal Gene Regulatory Networks. *Cell* **176**, 361-376.e17 (2019).
23. McInnes, L., Healy, J. & Melville, J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. (2018).
24. McDowell, I. C. *et al.* Clustering gene expression time series data using an infinite Gaussian process mixture model. *PLOS Computational Biology* **14**, e1005896 (2018).
25. Gasperini, M. *et al.* A Genome-wide Framework for Mapping Gene Regulation via Cellular Genetic Screens. *Cell* **176**, 377-390.e19 (2019).
26. Kelley, D. R., Snoek, J. & Rinn, J. Basset: Learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome Res.* gr.200535.115 (2016) doi:10.1101/gr.200535.115.
27. ENCODE Project Consortium. An integrated encyclopedia of DNA elements in the human genome. *Nature* **489**, 57–74 (2012).
28. Kundaje, A. *et al.* Integrative analysis of 111 reference human epigenomes. *Nature* **518**, 317–330 (2015).

29. Kim, D. S. & Kundaje, A. Classification dataset for ENCODE-Roadmap DNase-seq peaks and Transcription Factor ChIP-seq peaks. (2020) doi:10.5281/zenodo.4059038.
30. Kim, D. S. & Kundaje, A. Convolutional Neural Net (CNN) models for ENCODE-Roadmap DNase-seq peaks and Transcription Factor ChIP-seq peaks - Basset architecture. (2020) doi:10.5281/zenodo.4059060.
31. Kim, D. S. & Kundaje, A. Machine learning datasets for epigenomic landscapes in epidermal differentiation. (2020) doi:10.5281/zenodo.4062509.
32. Kim, D. S. & Kundaje, A. Convolutional Neural Net (CNN) models for epigenomic landscapes in epidermal differentiation - Basset architecture, classification and regression. (2020) doi:10.5281/zenodo.4062726.
33. Simonyan, K., Vedaldi, A. & Zisserman, A. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *arXiv:1312.6034 [cs]* (2014).
34. Shrikumar, A., Greenside, P. & Kundaje, A. Learning Important Features Through Propagating Activation Differences. *arXiv:1704.02685 [cs]* (2017).
35. Harvey, C. T. *et al.* QuASAR: quantitative allele-specific analysis of reads. *Bioinformatics* **31**, 1235–1242 (2015).
36. Heinz, S. *et al.* Simple combinations of lineage-determining transcription factors prime cis-regulatory elements required for macrophage and B cell identities. *Mol Cell* **38**, 576–589 (2010).

37. Liu, T. *et al.* Cistrome: an integrative platform for transcriptional regulation studies. *Genome Biol* **12**, R83 (2011).

38. Ramírez, F. *et al.* deepTools2: a next generation web server for deep-sequencing data analysis. *Nucleic Acids Res* **44**, W160–W165 (2016).

39. Li, Z. *et al.* Identification of transcription factor binding sites using ATAC-seq. *Genome Biology* **20**, 45 (2019).

40. Reimand, J. *et al.* g:Profiler—a web server for functional interpretation of gene lists (2016 update). *Nucleic Acids Res* **44**, W83–W89 (2016).

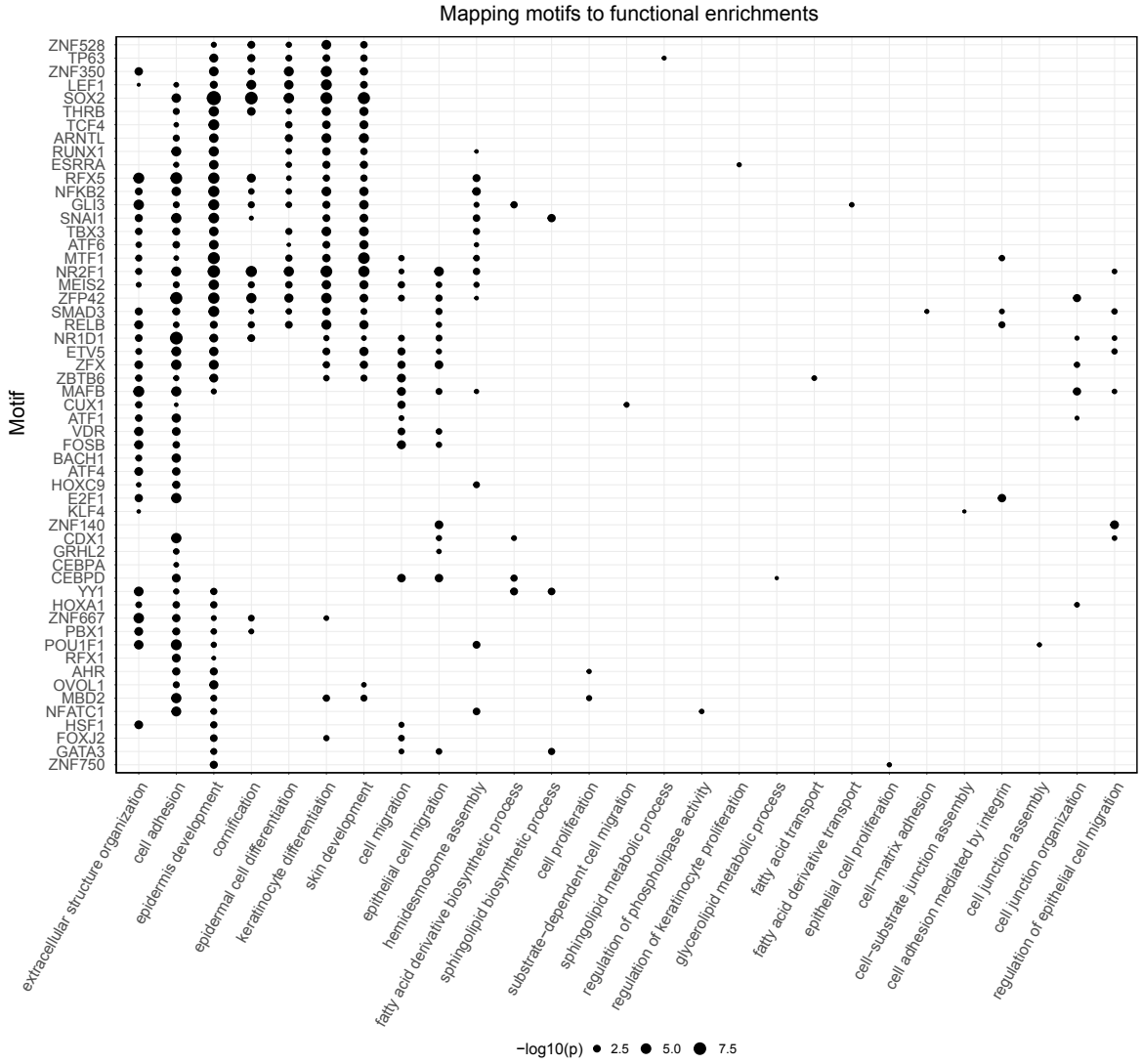
41. Greenside, P., Shimko, T., Fordyce, P. & Kundaje, A. Discovering epistatic feature interactions from neural network models of regulatory DNA sequences. *Bioinformatics* **34**, i629–i637 (2018).

42. Truong, A. B., Kretz, M., Ridky, T. W., Kimmel, R. & Khavari, P. A. p63 regulates proliferation and differentiation of developmentally mature keratinocytes. *Genes Dev.* **20**, 3185–3197 (2006).

43. Sanford, E. M., Emert, B. L., Coté, A. & Raj, A. Gene regulation gravitates towards either addition or multiplication when combining the effects of two signals. *bioRxiv* 2020.05.26.116962 (2020) doi:10.1101/2020.05.26.116962.

44. Nair, S., Kim, D. S., Perricone, J. & Kundaje, A. Integrating regulatory DNA sequence and gene expression to predict genome-wide chromatin accessibility across cellular contexts. *Bioinformatics* **35**, i108–i116 (2019).

Supplementary Figure 1



Supplementary Figure 1. Functional enrichments of genes associated with CREs containing predictive motifs. CREs containing predictive instances of each motif (y-axis) were mapped to putative target genes based on proximity. The x-axis shows the key functional ontology terms enriched for each motif's target gene set.