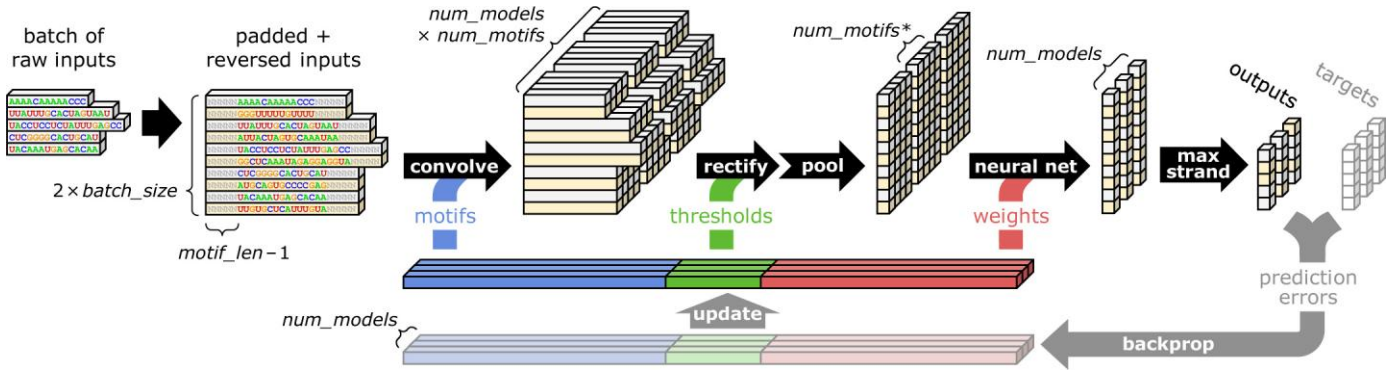


Supplementary Figure 1

An extended version of Figure 2a**, depicting multi-model training and reverse-complement mode**

To use the GPU's full computational power, we train several independent models in parallel on the same data, each with different calibration parameters. The calibration parameters with validation performance are used to train the final model. Shown is an example with batch_size=5, motif_len=6, num_motifs=4, num_models=3. Sequences are padded with 'N's so that the motif scan operation can find detections at both extremities. Yellow cells represent the reverse complement of the input located above; both strands are fed to the model, and the strand with the maximum score is used for the output prediction (the *max strand* stage). The output dimension of the *pool* stage, depicted as num_motifs (*), depends on whether "max" or "max and avg" pooling was used.

a

	Mean	Essrrb	Gata4	Tbx20	Tbx5	Zfx
DeepBind	0.726	0.731	0.741	0.633	0.720	0.807
BEEML-PBM_sec	0.714	0.703	0.736	0.661	0.675	0.793
BEEML-PBM	0.708	0.688	0.726	0.663	0.699	0.765
BEEML-PBM_dinuc	0.703	0.677	0.744	0.573	0.716	0.803
FeatureREDUCE_dinuc	0.696	0.685	0.729	0.624	0.679	0.761
FeatureREDUCE_PWM	0.695	0.684	0.726	0.631	0.679	0.753
FeatureREDUCE	0.673	0.625	0.725	0.529	0.683	0.805
Team_E	0.663	0.577	0.714	0.636	0.599	0.789
PWM_align	0.651	0.698	0.702	0.618	0.473	0.763
FeatureREDUCE_sec	0.650	0.699	0.637	0.627	0.582	0.704
MatrixREDUCE	0.587	0.347	0.659	0.568	0.572	0.791
Team_D	0.532	0.580	0.670	0.468	0.470	0.470

b

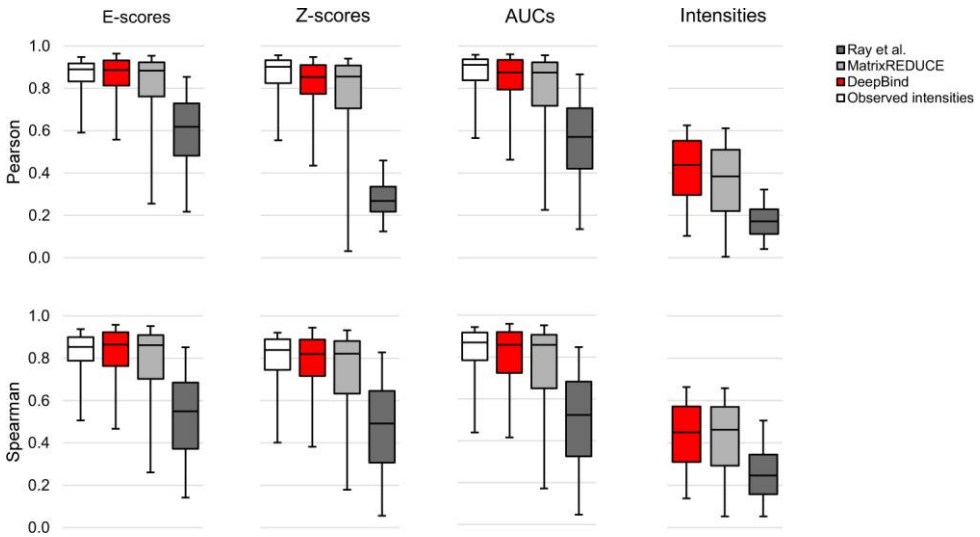
AUC (dinuc shuffle background)

	Mean	Essrrb	Gata4	Tbx20	Tbx5	Zfx
DeepBind	0.700	0.673	0.742	0.793	0.598	0.695
BEEML-PBM	0.651	0.696	0.726	0.638	0.614	0.583
FeatureREDUCE	0.651	0.675	0.737	0.626	0.614	0.603
ChIPmunk	0.642	0.682	0.658	0.761	0.560	0.553
MEME-ChIP	0.636	0.658	0.702	0.761	0.555	0.504

Supplementary Figure 2

Performance of *in vitro* trained TF models on *in vivo* data (DREAM5 ChIP-seq)

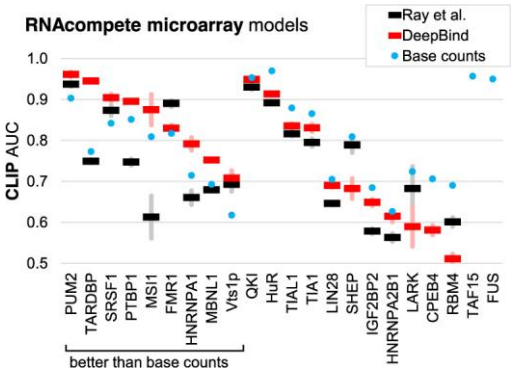
(**a**) All DREAM5 ChIP-seq AUCs used to compute mean performance shown in **Figure 3b**. The models were trained on the DREAM5 PBM training data only, and evaluated against three different backgrounds[22]. (**b**) Cross-validation performance of methods trained directly on ChIP-seq data (sequence length 100), evaluated against a dinucleotide shuffled background (**Supplementary Table 1**).



Supplementary Figure 3

Performance on *in vitro* RBP data using several evaluation metrics

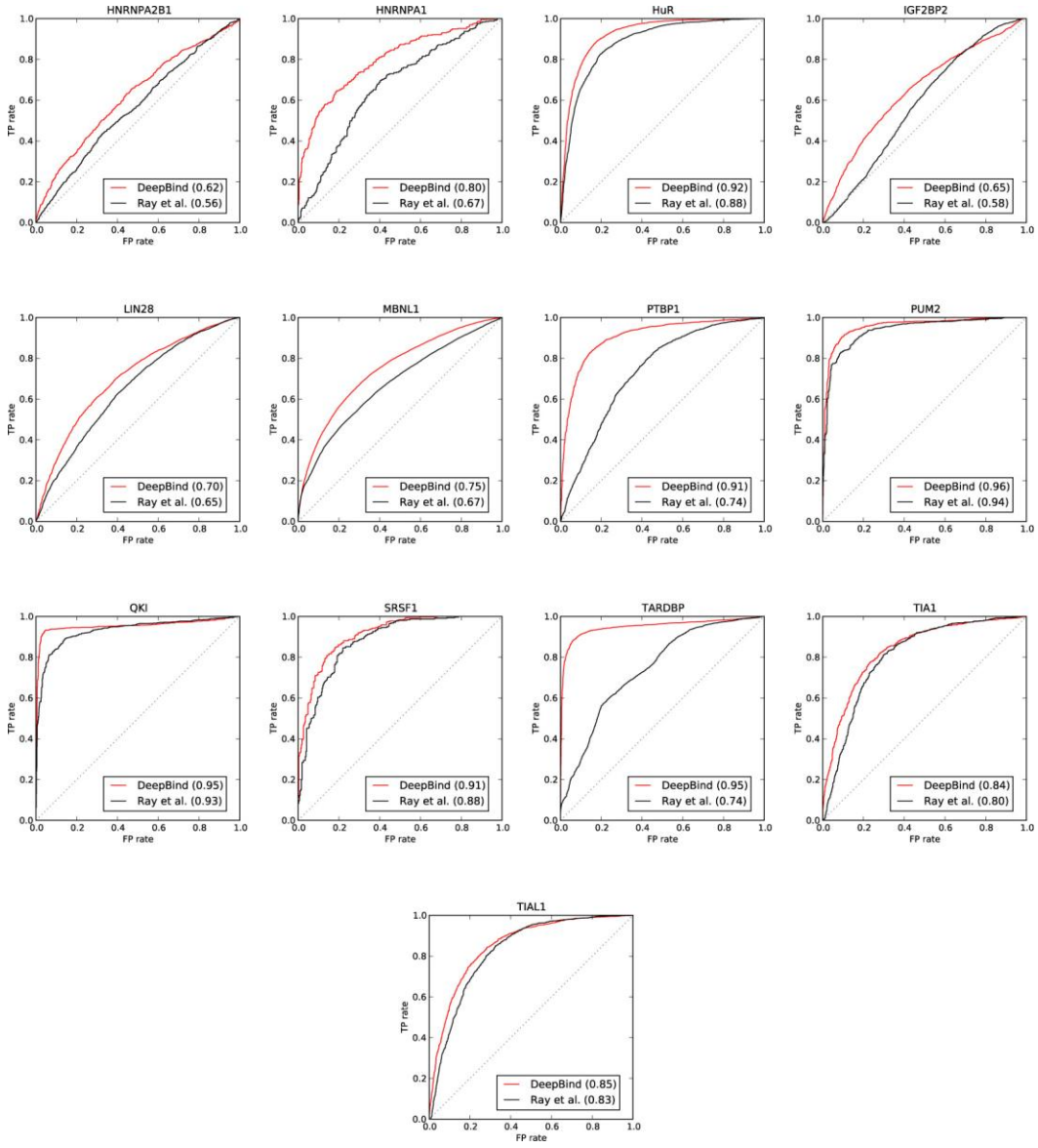
Box plots showing distribution of RNAcompete *in vitro* RBP performance over 244 different microarray experiments using 6 evaluation metrics (columns) and two types of correlation (rows). Models were trained on RNAcompete PBM probes labeled “Set A”, and tested on “Set B” probes.



Supplementary Figure 4

Performance of *in vitro* trained RBP models on *in vivo* data

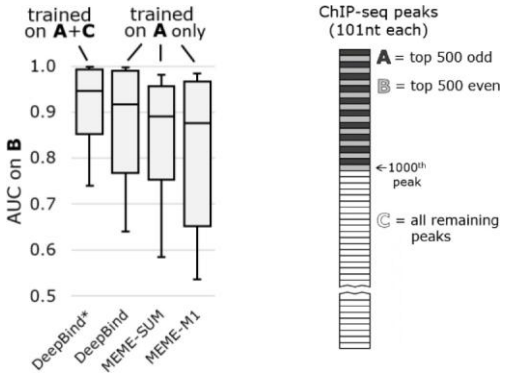
Performance of all RBP models for which RNAcompete *in vivo* data was available (c.f. Ray et al.[19], Fig. 1C). **Figure 3d** shows only the subset of RBPs for which the *in vivo* test sequences has average length <1000. All AUCs are calculated with 100 bootstrap samples, and the standard deviation is shown as vertical lines. “Base counts” show the best performance achievable from ranking test sequences by the proportion of a single nucleotide or by sequence length; for example, ranking the QKI test sequences by 1/(fraction of Gs) gives AUC of 0.95. There are 9 RBPs for which at least one method can perform better than base counts on this test data. RNAcompete PFMs beat base counts for PUM2, SRSF1, FMR1, and Vts1p. DeepBind beats base counts for 8 RBPs (no significant improvement for FMR1). See **Supplementary Table 3** (“In vivo AUCs”) for raw data for this plot.



Supplementary Figure 5

ROC curves for the AUCs shown in Figure 3d

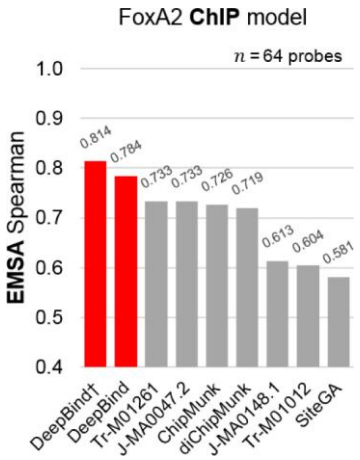
ROC curves for the AUCs shown in **Figure 3d**, where the RNAcompete-trained (*in vitro*) RBP models were applied to *in vivo* (CLIP, RIP) sequences. Importantly, several DeepBind models have higher recall at low false positive rates.



Supplementary Figure 6

Detailed explanation of how ChIP-seq peaks were divided into training and testing data for each experiment.

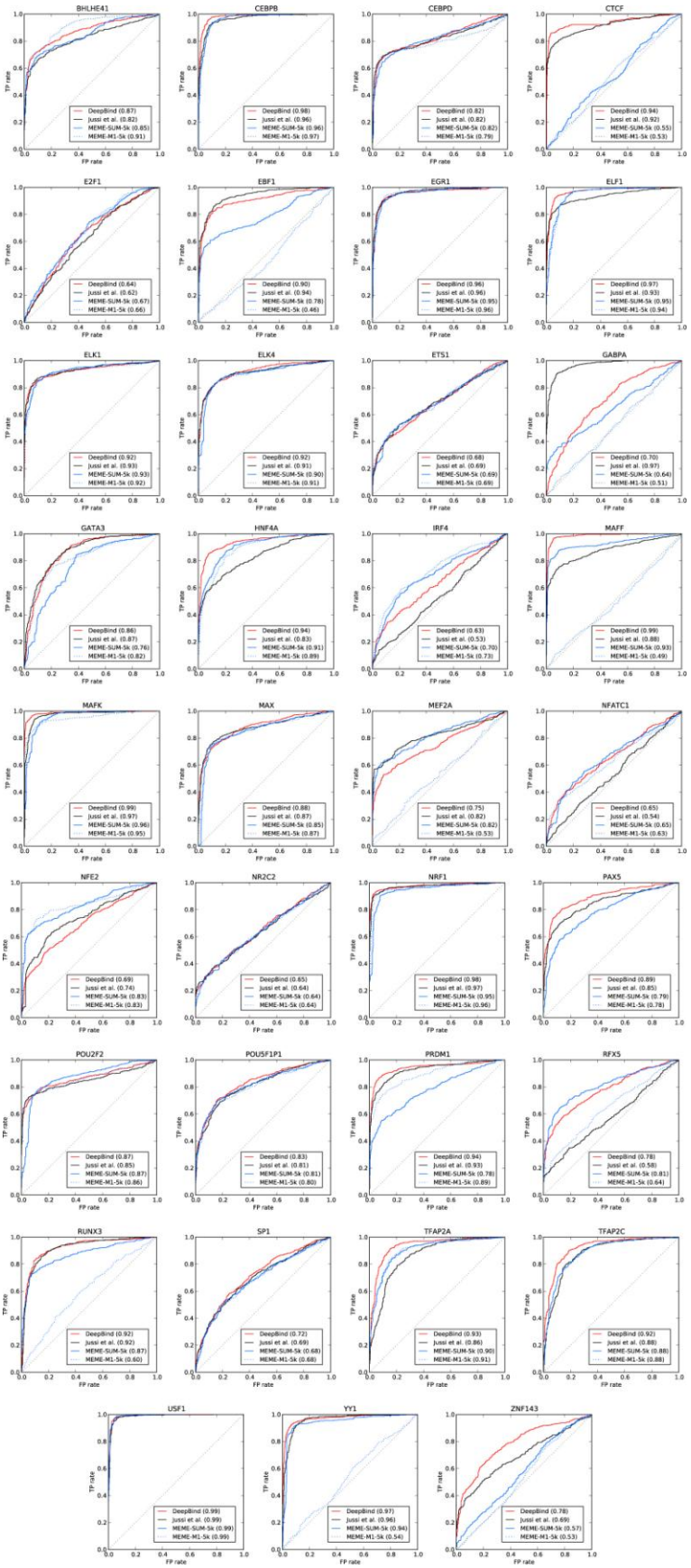
The ChIP-seq performance from **Figure 3e** are reproduced at left with extra annotations for clarity. At right is the breakdown of ChIPseq peaks used to train a model on each ChIP experiment. We train each method on peaks labeled A ("top 500 odd"), then test each method on peaks labeled B ("top 500 even"). DeepBind* is a special case where we show that including the lower -ranked peaks labeled C ("all remaining peaks") in the training set can significantly improve the accuracy when scoring the top-ranked peaks labeled B.



Supplementary Figure 7

Evaluation of FoXA2 models learned from ChIP-seq data on EMSA-measured affinities

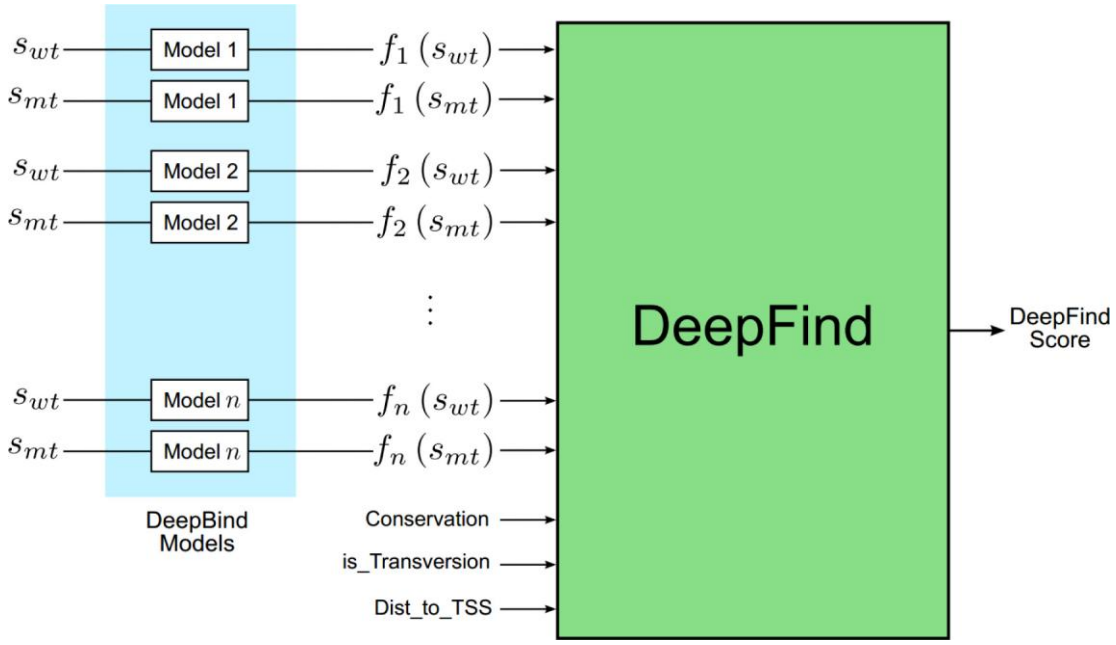
FoxA2 ChIP model predictions validated by EMSA-measured affinities of FoxA2 binding to 64 probe sequences[32]. The column marked "DeepBind†" is an extra model that we trained on the same ENCODE ChIP data as "DeepBind", but where we used motif_len=16 instead of the usual motif_len=24. The shorter motif length was tried due to the post-hoc observation that our FoxA2 model learns patterns of length 10, and we heuristically found that motif_len of ~1.5x the true motif length often works well. The fact that DeepBind† performed best suggests that there is still room for refinement in the DeepBind training procedure we use.


BHLHE41
DeepBind (0.87)
Jussi et al. (0.82)
MEME-SUM-5k (0.85)
MEME-M1-5k (0.91)
CEBPB
DeepBind (0.98)
Jussi et al. (0.96)
MEME-SUM-5k (0.96)
MEME-M1-5k (0.97)
CEBPD
DeepBind (0.82)
Jussi et al. (0.82)
MEME-SUM-5k (0.82)
MEME-M1-5k (0.79)
CTCF
DeepBind (0.94)
Jussi et al. (0.92)
MEME-SUM-5k (0.55)
MEME-M1-5k (0.53)
E2F1
DeepBind (0.64)
Jussi et al. (0.62)
MEME-SUM-5k (0.67)
MEME-M1-5k (0.66)
EBF1
DeepBind (0.90)
Jussi et al. (0.94)
MEME-SUM-5k (0.78)
MEME-M1-5k (0.46)
EGR1
DeepBind (0.96)
Jussi et al. (0.96)
MEME-SUM-5k (0.95)
MEME-M1-5k (0.96)
ELF1
DeepBind (0.97)
Jussi et al. (0.93)
MEME-SUM-5k (0.95)
MEME-M1-5k (0.94)
ELK1
DeepBind (0.92)
Jussi et al. (0.93)
MEME-SUM-5k (0.93)
MEME-M1-5k (0.92)
ELK4
DeepBind (0.92)
Jussi et al. (0.91)
MEME-SUM-5k (0.90)
MEME-M1-5k (0.91)
ETS1
DeepBind (0.68)
Jussi et al. (0.69)
MEME-SUM-5k (0.69)
MEME-M1-5k (0.69)
GABPA
DeepBind (0.70)
Jussi et al. (0.97)
MEME-SUM-5k (0.64)
MEME-M1-5k (0.51)
GATA3
DeepBind (0.86)
Jussi et al. (0.87)
MEME-SUM-5k (0.76)
MEME-M1-5k (0.82)
HNF4A
DeepBind (0.94)
Jussi et al. (0.83)
MEME-SUM-5k (0.91)
MEME-M1-5k (0.89)
IRF4
DeepBind (0.63)
Jussi et al. (0.53)
MEME-SUM-5k (0.70)
MEME-M1-5k (0.73)
MAFF
DeepBind (0.99)
Jussi et al. (0.88)
MEME-SUM-5k (0.93)
MEME-M1-5k (0.49)
MAFK
DeepBind (0.99)
Jussi et al. (0.97)
MEME-SUM-5k (0.96)
MEME-M1-5k (0.95)
MAX
DeepBind (0.88)
Jussi et al. (0.87)
MEME-SUM-5k (0.85)
MEME-M1-5k (0.87)
MEF2A
DeepBind (0.75)
Jussi et al. (0.82)
MEME-SUM-5k (0.82)
MEME-M1-5k (0.53)
NFATC1
DeepBind (0.65)
Jussi et al. (0.54)
MEME-SUM-5k (0.65)
MEME-M1-5k (0.63)
NFE2
DeepBind (0.69)
Jussi et al. (0.74)
MEME-SUM-5k (0.83)
MEME-M1-5k (0.83)
NR2C2
DeepBind (0.65)
Jussi et al. (0.64)
MEME-SUM-5k (0.64)
MEME-M1-5k (0.64)
NRF1
DeepBind (0.98)
Jussi et al. (0.97)
MEME-SUM-5k (0.95)
MEME-M1-5k (0.96)
PAX5
DeepBind (0.89)
Jussi et al. (0.85)
MEME-SUM-5k (0.79)
MEME-M1-5k (0.78)
POU2F2
DeepBind (0.87)
Jussi et al. (0.85)
MEME-SUM-5k (0.87)
MEME-M1-5k (0.86)
POU5F1P1
DeepBind (0.83)
Jussi et al. (0.81)
MEME-SUM-5k (0.81)
MEME-M1-5k (0.80)
PRDM1
DeepBind (0.94)
Jussi et al. (0.93)
MEME-SUM-5k (0.78)
MEME-M1-5k (0.89)
RFX5
DeepBind (0.78)
Jussi et al. (0.58)
MEME-SUM-5k (0.81)
MEME-M1-5k (0.64)
RUNX3
DeepBind (0.92)
Jussi et al. (0.92)
MEME-SUM-5k (0.87)
MEME-M1-5k (0.60)
SP1
DeepBind (0.72)
Jussi et al. (0.69)
MEME-SUM-5k (0.68)
MEME-M1-5k (0.68)
TFAP2A
DeepBind (0.93)
Jussi et al. (0.86)
MEME-SUM-5k (0.90)
MEME-M1-5k (0.91)
TFAP2C
DeepBind (0.92)
Jussi et al. (0.88)
MEME-SUM-5k (0.88)
MEME-M1-5k (0.88)
USF1
DeepBind (0.99)
Jussi et al. (0.99)
MEME-SUM-5k (0.99)
MEME-M1-5k (0.99)
YY1
DeepBind (0.97)
Jussi et al. (0.96)
MEME-SUM-5k (0.94)
MEME-M1-5k (0.54)
ZNF143
DeepBind (0.78)
Jussi et al. (0.69)
MEME-SUM-5k (0.57)
MEME-M1-5k (0.53)
TP rate
FP rate

Supplementary Figure 8

ROC curves for the AUCs shown in Figure 3f

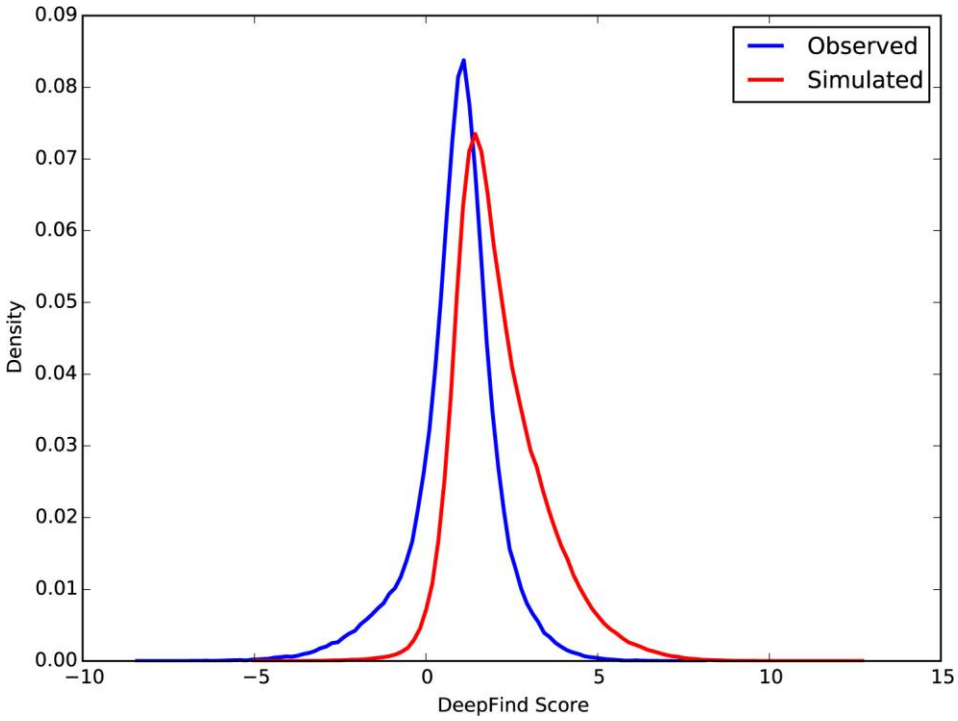
ROC curves for the AUCs shown in **Figure 3f**, where the HT-SELEX-trained (*in vitro*) TF models were applied to *in vivo* (ChIP) sequences. For the semi-automatic method of Jolma et al. we show the curve for whichever PWM performed best on the test data; summing the scores of their choices of PFMs resulted in worse performance overall, so it is not shown.



Supplementary Figure 9

Schematic diagram of the DeepFind model

Schematic diagram of the DeepFind model, using 2*n* TF scores (*n* wild type, *n* mutant) as features to a deep neural network.



Supplementary Figure 10

DeepFind score distributions for the observed and simulated SNVs.

DeepFind score distributions for the observed and simulated SNVs.