Prediction of Histone Post-translational Modifications using Deep Learning

Dipankar Ranjan Baisya*
dbais001@ucr.edu
Department of Computer Science and Engineering,
University of California
Riverside, CA, US

Stefano Lonardi
stelo@cs.ucr.edu
Department of Computer Science and Engineering,
University of California
Riverside, CA, US

ABSTRACT

Motivation: Histone post-translational modifications (PTMs) are epigenetic regulators involved in a variety of essential cellular processes. Recent studies have demonstrated that histone PTMs can be accurately predicted from the knowledge of transcription factor binding either at the promoter or distal regulatory regions, or from the underlying DNA primary sequence.
Results: We propose a deep learning architecture called DeepHistone for predicting histone PTMs from transcription factor binding data and the primary DNA sequence. Extensive experimental results show that our deep learning model outperforms the prediction accuracy of the model proposed in Benveniste *et al.* (PNAS 2014) by 8.5%–36.3% improvement in AUPR. The competitive advantage of our framework lies in synergistic use of deep learning combined with an effective pre-processing step. Our framework has also enabled the discovery that the knowledge of a small subset of transcription factors (which are histone-PTM- and cell-type-specific) can provide almost the same prediction accuracy that can be obtained using all the transcription factors.
Availability: https://github.com/ucrbioinfo/DeepHistone
Contact: dbais001@ucr.edu

CCS CONCEPTS

• **Applied computing → Computational biology**.

KEYWORDS

Epigenetic, Histone Modification, Transcription Factor, DNA Sequence, Neural Network, Gradient Boosted Classifier

ACM Reference Format:
Dipankar Ranjan Baisya and Stefano Lonardi. 2018. Prediction of Histone Post-translational Modifications using Deep Learning. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Histones are a class of proteins that bind to DNA and help to condense the DNA into chromatin. Histones contain a large proportion of positively charged amino acids, while DNA is negatively charged. These opposite charges create a high-binding affinity structure between histones and DNA, called the *nucleosome*.

Histone proteins can be classified into *core* histones (H2A, H2B, H3, H4) and *linker* histone (H1). The eight histones in the core are arranged into a (H3)2(H4)2 tetramer and a pair of H2A-H2B dimers, called the *histone octamer*. Each core histone has an amino-terminal extension called *histone tail* which is the target for *post-translational modifications* (PTMs), also known as histone *marks*.

Research in epigenetics has shown that post-transcriptional modifications of core histone are involved in a variety of essential regulatory processes in the cell, including transcription control (see, e.g., [3, 20, 22]). Epigenetics factors (including PTMs and DNA methylation) and the complex of interaction between nucleosomes and transcription factors are arguably the most critical factors influencing gene expression. In fact, some studies (e.g., [9, 11]) have shown that gene expression can be accurately predicted from the knowledge of histone tail PTMs.

In [4], the authors investigated the opposite problem, that is, the prediction of histone PTMs from the knowledge of transcription factors (TF) binding and the underlying DNA sequence data. They reported that histone PTMs can be predicted accurately from the knowledge of TF binding either at the promoter or distal regulatory regions for three different human cell lines. They also showed that the prediction from TFs is more accurate than the prediction from the DNA sequence alone, and that the predictive power of TF binding data can be extended to predict histone modifications across cell lines. They suggested that interactions between TFs and histone-modifying enzymes might be important in driving the deposition of histone modifications.

In this work, we propose for the first time a deep learning architecture called DeepHistone for predicting histone PTMs from TF binding and DNA sequence data. Extensive experimental results show that our neural network achieves a prediction accuracy substantially higher than the logistic regression model proposed in [4]. The competitive advantage of our framework lies in synergistic use of deep learning combined with an effective data cleaning preprocessing step. Our framework has also enabled the discovery that the knowledge of a small subset of transcription factors (which are histone-PTM- and cell-type-specific) can provide almost the same prediction accuracy that can be obtained using all the transcription factors.

Woodstock '18, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-9999-9/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

2 METHODS

2.1 Data Collection

Human epigenetic and genetic data was obtained from ENCODE [8], as follows. Exactly 29,828 unique protein-coding transcription start sites (TSS) for the human reference genome were obtained from the ENSEMBL database. Histone tail modifications in the proximity of these TSS were assigned based on ChIP-Seq analysis for three ENCODE Tier 1 cell lines, namely H1 ES cells, K562 erythroleukemia cells, and GM12878 lymphoblastoid cells. We considered three-four histone PTMs that are known to be relevant for transcription, namely H3K4me3, H3K9ac, H3K27ac for all three cell lines. For cell line H1, we also considered H3K27me3.

As it was done in [4], for each histone PTM, a transcription start site (TSS) was assigned a positive label if a ChIP-Seq peak was detected within a 100 bp window center at the TSS (negative otherwise). In ChIP-Seq, (i) DNA-bound proteins are cross-linked to their DNA, which is then fragmented, (ii) DNA fragments those that are not cross linked with proteins are removed by immuno-precipitation, and (iii) short reads from either ends of cross-linked DNA fragments are produced. We aligned ChIP-Seq reads to the human genome using BWA, then used MACS2 [21] using a false-discovery rate of 0.01 to detect peaks. As a result of this process, the proportion of positive/negative examples in the training set for cell line H1 ranges from 8.3% to 39.8% (see Supplementary Table 1). A pre-processing step described below is aimed at correcting the imbalance in the training set.

The DNA sequence data was obtained following the protocol in [4]. We extracted the sequence ±2,000 bp upstream and downstream of each TSS. Then, we computed 6-mer counts for these 4,000 bp-long sequences. We combined counts of 6-mers which are the reverse-complement of each other, ending up with 2,080 6-mer counts. As result of this process, the entire DNA sequence dataset was represented by 29,828 vectors of dimension 2,080. Each vector had a positive or negative label for a particular pair of (histone PTM, cell line) according to the presence of a corresponding ChIP-Seq peak around the TSS (as described above).

The transcription factor (TF) dataset was obtained following the protocol in [4]. First, ChIP-Seq data from the ENCODE project was filtered to remove proteins that lacked sequence-specific DNA-binding transcription factor (TF) activity. Overall, 30 TFs were assayed in H1 cells, 45 in K562, and 51 in GM12878. Among these, 17 TFs were assayed in all three cell lines. ChIP-Seq reads were aligned to the human genome, then the number of reads mapped within 2,000 bp of the TSS were counted. Raw read counts were normalized by dividing the counts by the total number of reads from the control dataset. For the H1 cell line, we considered 29,828 TSS and 30 normalized read counts. For the K562 cell line, we had 45 normalized read counts. For the GM12878 cell line, we considered 51 normalized read counts. Again, each vector had a positive or negative label for a particular pair of (histone PTM, cell line) according to the presence of a corresponding ChIP-Seq peak around the TSS (as described above).

2.2 Neighborhood Cleaning Rule

To gain insights into the structural properties of the training examples, we generated two-dimensional projections for the feature vectors in the training set. For the transcription factor binding sites obtained from ChIP-Seq data, k-dimensional vectors and their respective labels for all histone PTMs ($k = 30$ for the H1 cell line, $k = 45$ for the K562 cell line, and $k = 51$ for the GM12878 cell line) were processed using t-SNE [15] with default settings such as $n\_components = 2$, $perplexity = 30.0$, $learning\_rate = 200.0$, $n\_iter = 1000$ etc. Figure 1 shows the results for cell line H1. Observe that positive examples (green) and negative examples (red) are not well separated. The same lack of separation can be observed on t-SNE plots for H3K9ac in cell line K562, H3K4me3 and H3K9ac in cell line GM12878 (Supplementary Figure 6). For DNA sequence based features for histone PTM H3K4me3 in H1 cells, H3K9ac in K562 cells and H3K27ac in GM12878 cells the t-SNE plots positive examples and negative examples are equally not-well separated (Supplementary Figure 5). Based on these observations, we decided to carefully discard training examples that could hamper the learning inference.

We employed the *neighborhood cleaning rule* (NCL) introduced in [13]. NCL is a down-sampling method that removes only samples from the majority class. For a two-class problem the algorithm works as follows. For each example E_i in the training set, identify its three nearest neighbors. If E_i belongs to the majority class and its three nearest neighbors contradicts the label of E_i, then E_i is removed. If E_i belongs to the minority class and its three nearest neighbors contradict the label of E_i, then the nearest neighbors that belong to the majority class are removed. Figure 2 shows a portion of the t-SNE projection of the data before and after applying NCL. Observe how the data is much better separated after NCL. The positive/negative ratio in the training set after NCL are reported in Supplementary Table 2 (TF-data) and Supplementary Table 3 (DNA sequence data).

2.3 Choice of the Classifier and Training

A logistic regression (LR) classifier was used in [4] because this type of classifiers produce interpretable weights that were used to verify afterward that their method recapitulates known TF-histone PTM interactions. In this work, we have used a feed-forward fully-connected neural network because it vastly improves the prediction accuracy compared to LR. However, to determine the *minimal sets* of TFs (described below) we have used a Gradient Boosted Classifier (GBC) because it provides interpretable weights (i.e., Gini importance).

While the architecture of the neural network for TF ChIP-Seq data is different from the architecture for DNA sequences (see below), the training procedure is same. The steps of our training/testing procedure are as follows.

(1) Let A be the input data set (unbalanced)
(2) Split A uniformly at random into a training set A_{prime} (80% of A), validation set B (10% of A) and a test set C (10% of A)
(3) Let D the dataset after the Neighborhood Cleaning Rule on $A_\prime$
(4) Train the model on D and validate on B
(5) Evaluate on C

It is important to note that we have applied NCl only on training data set not on the test data set. However, to make a performance comparison of our model with noisy and clean test set, we have

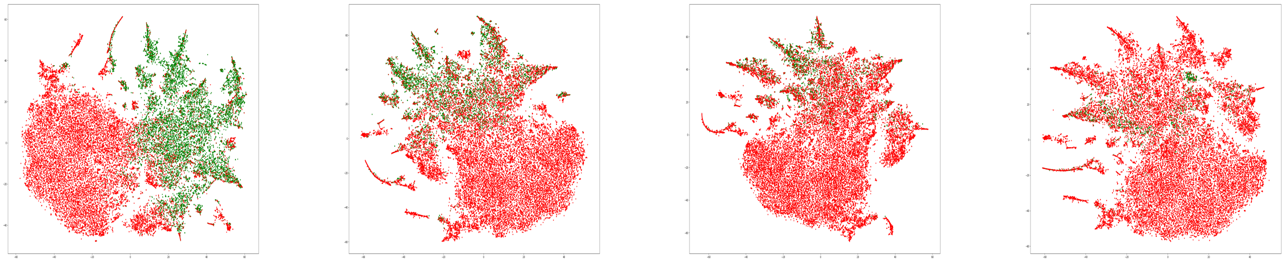

Figure 1: Two-dimensional projection using t-SNE for H3K4me3, H1-H3K9ac, H1-H3K27ac and H1-H3K27me3 (left to right) on TF-ChIP-Seq data for cell line H1; red points are negative examples; greens points are positive examples.

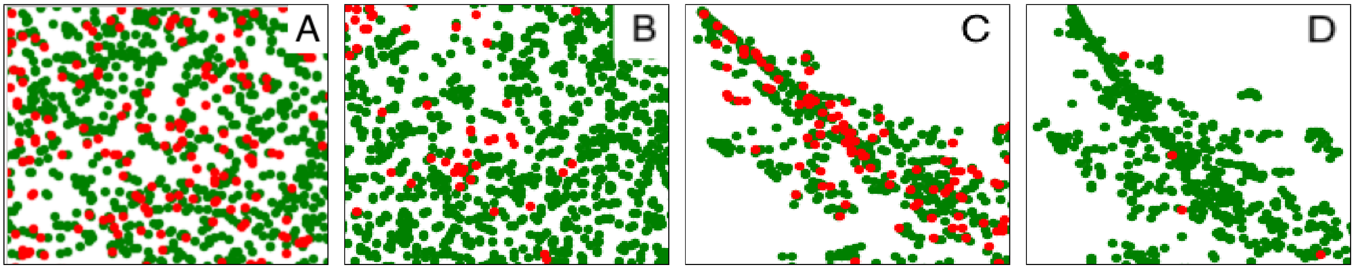

Figure 2: Two-dimensional projection using t-SNE for H3K4me3 data before (A) and after NCL (B) and for H3K9ac data before (C) and after NCL (D) for cell line H1; red points are negative examples; greens points are positive examples.

applied NCL to test set and reported the the performance of our model in test set without applying NCL and test set after applying NCL separately.

2.4 Model Architecture for DNA sequences

To determine the optimal architecture for DeepHistone to learn from the DNA sequence dataset we considered 1–5 hidden layers, each with a number of hidden units between 50 and 550. For dropout [19], we considered three values, namely, 0.3 corresponding to a strong dropout, 0.5 (medium dropout) and 0.7 (weak dropout). We evaluated batch sizes of $4, 8, 16, \ldots, 128$. The learning rate was chosen from the interval $[1e^{-5}, 1e^{-1}]$ evenly spaced in log scale. The optimal architecture for DNA sequences used three hidden layers (with 512, 180 and 70 hidden units, respectively), with a dropout of 0.5, a batch size of 128, and an initial learning rate of $1e^{-4}$.

2.5 Model Architecture for TF data

We searched for the optimal architecture to learn from the TF ChIP-Seq dataset, as described in the Section 2.4. For the TF dataset, the optimal architecture of DeepHistone has three hidden layers (with 256, 180 and 60 hidden units, respectively), with a dropout of 0.3, a batch size of 16, and an initial learning rate of $1e^{-4}$. The architecture DeepHistone for the TF dataset is illustrated in Figure 3.

In both architectures (DNA and TF), we used the procedure in [10] to initialize the weights, ReLU as the activation function for the hidden layers [14], a sigmoid activation function in the output layer, binary cross entropy (Equation 1 below) as the loss function, and the Adam optimizer [12]. We allowed up to 90 epochs for training, with the possibility of early stopping when the value of loss function does not improve over ten iterations. For model architecture selection and hyper-parameter tuning, we followed [1, 2, 5, 17, 18]. As said, the loss function used in DeepHistone is

$$L(y, y') = -\frac{1}{N}\sum_{i=1}^{N} y_i \log(y'_i) + (1 - y_i)\log(1 - y'_i) \qquad (1)$$

where y is 1 for a positive example and 0 for a negative example, y' is classifier's predicted label/probability, and N is the total number of examples in the training set.

2.6 Feature Selection

On the TF binding data have computed the *Gini importance* on the feature set. The *Gini importance* depends on the number of times a feature is used to split a node in a decision tree, weighted by the number of samples it splits.

In order to define formally the *Gini importance* we need to introduce the notion of *impurity*. Impurity is a quantity defined on each node t in a decision tree: the smaller the impurity of t (i.e., the purer is t), the better is the accuracy in prediction provided by t. The *Gini impurity* is one type of impurity. Given a set of objects with J classes, where t_i be the fraction of items labeled with class $i \in \{1, 2, \ldots, J\}$ in the set, the Gini impurity $i(t)$ is defined as

$$i(t) = \sum_{i=1}^{j} t_i(1 - t_i).$$

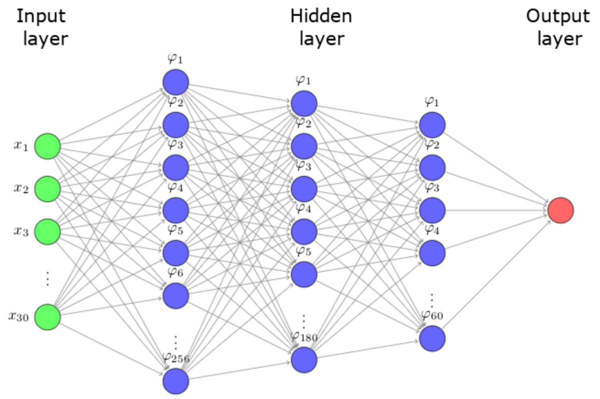

Figure 3: The proposed architecture for DeepHistone for the TF ChIP-Seq dataset: one input layers, three hidden layers, one output layer

The *impurity decrease* Δi for a binary split s_t dividing node t into a left node t_L and a right node t_R is defined as

$$\Delta i(s_t, t) = i(t) - p_L i(t_L) - p_R i(t_R)$$

where p_L (respectively, p_R) is the fraction N_{t_L}/N_t (respectively, N_{t_R}/N_t) of learning samples assigned to the left node t_L (respectively, assigned to the right node t_R), N_t is the total size of the subset, and $i(t)$ is the Gini impurity of node t.

The *Gini importance* $Imp(X_j)$ for a variable X_j used to predict Y is defined by adding the weighted impurity decreases $p(t)\Delta i(s_t, t)$ for all nodes t where X_j is used, and averaging over all trees ϕ_m (for $m = 1, \ldots, M$) in the forest [6]. Formally,

$$Imp(X_j) = \frac{1}{M} \sum_{m=1}^{M} \sum_{t \in \phi_m} 1(j_t = j)\,[p(t)\Delta i(s_t, t)]$$

where $p(t)$ is the fraction N_t/N of samples reaching t, j_t denotes the identifier of variable used for splitting node t, and $\Delta i(s_t, t)$ is the impurity decrease defined above.

Based on the *Gini importance*, we computed the *minimal feature set* of features as follows.

(1) Sort features by Gini importance in descending order
(2) Use the top k features to feed the DeepHistone
(3) Add k features in to S
(4) Train/Test on the the set S
(5) Find the best k by incrementing 1 ($k = k + 1$) until the prediction accuracy of the classifier using S is within 1% of the prediction accuracy based on all features

Supplementary Figure 1 illustrates the process of producing minimal sets. The Gini importance for each TF over all pairs of (PTM, cell line) are shown in Supplementary Tables 15–24.

3 RESULTS AND DISCUSSION

All training/testing experiments were carried out on a Titan GTX 1080 Ti GPU, running Keras 2.1.3 [7]. NCL and Gradient Boosted Classifier were implemented using Sklearn [16]. Unless otherwise noted, all experimental results reported for DeepHistone are after data cleaning (NCL).

3.1 Prediction of Histone PTMs from DNA Sequence

To evaluate the prediction quality of our classifier, we computed the Area Under Precision-Recall curve (AUPR), the Area under ROC curve (AUROC), and accuracy. To quantify the stability of these predictions, we repeated each training/testing experiment five times and recorded the mean and standard deviation of each statistical measure. We compared the performance of DeepHistone to the logistic regression classifier (LR) described in [4].

First, we investigated the impact of the Neighborhood Cleaning Rule (NCL) on the performance of LR and DeepHistone. Supplementary Table 4 shows that LR performs better after NCL in predicting all histone PTMs. DeepHistone also performs better after NCL for all histone PTM predictions (see Supplementary Table 5). In general, we found that DeepHistone performs better than LR for all histone PTMs when cleaned data is used (Supplementary Table 6). In the rest of the experiments below we always used NCL as a pre-processing step for DeepHistone.

Table 1 shows AUPR, AUROC and accuracy for DeepHistone and LR when trained and tested on DNA sequence data (represented by the 2,080 6-mer counts). Here, NCL is used as a pre-processing steps of DeepHistonenot for LR. For further investigation, we also compared the performance of DeepHistone on raw test set (without applying NCL) and test set that we get after applying NCL on raw test set. Figure 4 shows the corresponding precision/recall curves. Observe that the AUPR for DeepHistone on raw test set ranges from 0.607 (for H3K27me3 on cell line H1) to 0.956 (for H3K4me3 on cell line H1). DeepHistone's predictions outperform LR on all histone PTMs. DeepHistone's AUPR improvement over LR ranges from 10.8% (for H3K4me3 on cell line H1) to 27.5% (for H3K4me3 on cell line H1). Table 1, we can also observe that for all PTMs performance of DeepHistone improves on the test set that that we get after applying NCL. We can conclude that less noisy the test set better the performance of DeepHistone .

To make an apples-to-apples comparison we have applied NCL as a pre-processing step of LR and then compare LR and DeepHistone. To compare the performance of LR and DeepHistone, we have used the raw test set (without applying NCL). Table 2 shows AUPR, AUROC and accuracy for DeepHistone and LR when trained and tested on DNA sequence data. Observe that AUPR for LR ranges from 0.512 (for H3K27me3 on cell line H1) to 0.85 (for H3K4me3 on cell line H1). Predictions of DeepHistone outperform LR on all histone PTMs. DeepHistone's AUPR improvement over LR ranges from 9.5 % (for H3K27ac on cell line H1) to 16.4 % (for H3K9ac on cell line H1).

3.2 Prediction of Histone PTMs from TF bindings

As we did in the previous section, we recorded mean and standard deviation for AUPR, AUROC, and accuracy over five iterations of each experiment. First, we investigated the impact of NCL on TF ChIP-Seq data. Both DeepHistone and LR performed better on cleaned data than on raw data for all histone PTM predictions (Supplementary Table 8 for LR, and Supplementary Table 9 for DeepHistone). In addition, DeepHistone performed better than LR for all histone PTMs when cleaned data was used (Supplementary

		AUPR			AUROC			Accuracy		
		DeepHistone		LR	DeepHistone		LR	DeepHistone		LR
Cell Line	PTM	Test Set with NCL	Test Set w/o NCL	w/o NCL	Test Set with NCL	Test Set w/o NCL	w/0 NCL	Test Set with NCL	Test Set w/o NCL	w/o NCL
H1	H3K4me3	**0.982**$_{\pm 0.000}$	**0.916**$_{\pm 0.000}$	0.848	**0.985**$_{\pm 0.001}$	**0.955**$_{\pm 0.001}$	0.918	**0.951**$_{\pm 0.002}$	**0.923**$_{\pm 0.002}$	0.856
H1	H3K9ac	**0.922**$_{\pm 0.001}$	**0.801**$_{\pm 0.001}$	0.649	**0.966**$_{\pm 0.000}$	**0.92**$_{\pm 0.000}$	0.867	**0.908**$_{\pm 0.003}$	**0.827**$_{\pm 0.003}$	0.825
H1	H3K27ac	**0.730**$_{\pm 0.003}$	**0.701**$_{\pm 0.003}$	0.447	**0.927**$_{\pm 0.001}$	**0.883**$_{\pm 0.001}$	0.828	**0.896**$_{\pm 0.001}$	**0.823**$_{\pm 0.001}$	0.861
H1	H3K27me3	**0.614**$_{\pm 0.014}$	**0.607**$_{\pm 0.014}$	0.332	**0.912**$_{\pm 0.005}$	**0.903**$_{\pm 0.005}$	0.806	**0.927**$_{\pm 0.000}$	**0.906**$_{\pm 0.000}$	0.902
K562	H3K4me3	**0.952**$_{\pm 0.001}$	**0.858**$_{\pm 0.001}$	0.751	**0.960**$_{\pm 0.000}$	**0.918**$_{\pm 0.001}$	0.865	**0.907**$_{\pm 0.001}$	**0.891**$_{\pm 0.001}$	0.814
K562	H3K9ac	**0.927**$_{\pm 0.006}$	**0.836**$_{\pm 0.006}$	0.735	**0.946**$_{\pm 0.005}$	**0.923**$_{\pm 0.005}$	0.865	**0.894**$_{\pm 0.003}$	**0.831**$_{\pm 0.003}$	0.822
K562	H3K27ac	**0.914**$_{\pm 0.000}$	**0.823**$_{\pm 0.000}$	0.705	**0.939**$_{\pm 0.000}$	**0.907**$_{\pm 0.000}$	0.853	**0.888**$_{\pm 0.001}$	**0.852**$_{\pm 0.001}$	0.821
GM12878	H3K4me3	**0.931**$_{\pm 0.001}$	**0.831**$_{\pm 0.001}$	0.693	**0.950**$_{\pm 0.000}$	**0.911**$_{\pm 0.000}$	0.845	**0.895**$_{\pm 0.001}$	**0.825**$_{\pm 0.001}$	0.800
GM12878	H3K9ac	**0.923**$_{\pm 0.001}$	**0.823**$_{\pm 0.001}$	0.700	**0.965**$_{\pm 0.000}$	**0.925**$_{\pm 0.000}$	0.855	**0.906**$_{\pm 0.003}$	**0.860**$_{\pm 0.003}$	0.814
GM12878	H3K27ac	**0.879**$_{\pm 0.001}$	**0.780**$_{\pm 0.001}$	0.647	**0.921**$_{\pm 0.001}$	**0.901**$_{\pm 0.001}$	0.827	**0.869**$_{\pm 0.001}$	**0.83**$_{\pm 0.001}$	0.806

Table 1: Histone PTM prediction performance (mean ± standard deviation, over five iterations) of DeepHistone (after data cleaning) vs. the logistic regression classifier (LR) from DNA sequence data; numbers in boldface indicate the best performance

		AUPR		AUROC		Accuracy	
Cell Line	PTM	Test Set with NCL DeepHistone	Test Set w/o NCL LR	Test Set with NCL DeepHistone	Test Set w/0 NCL LR	Test Set with NCL DeepHistone	Test Set w/o NCL LR
H1	H3K4me3	0.956	0.85	0.955	0.936	0.923	0.806
H1	H3K9ac	0.801	0.637	0.92	0.872	0.827	0.798
H1	H3K27ac	0.701	0.606	0.883	0.847	0.823	0.799
H1	H3K27me3	0.607	0.512	0.903	0.822	0.906	0.912
K562	H3K4me3	0.858	0.747	0.918	0.87	0.891	0.82
K562	H3K9ac	0.836	0.723	0.923	0.869	0.831	0.806
K562	H3K27ac	0.823	0.706	0.907	0.864	0.852	0.823
GM12878	H3K4me3	0.831	0.698	0.911	0.863	0.825	0.802
GM12878	H3K9ac	0.823	0.705	0.925	0.863	0.860	0.802
GM12878	H3K27ac	0.780	0.666	0.901	0.847	0.83	0.798

Table 2: Histone PTM prediction performance (mean ± standard deviation, over five iterations) of logistic regression classifier (LR) after applying NCL on training data of DNA sequence ; numbers in boldface indicate the best performance

Table 10). In the rest of the experiments below we always used NCL as a pre-processing step for DeepHistone (but not for LR).

Table 3 reports these performance metrics for DeepHistone and logistic regression (LR) when trained and tested on TF binding data, which is represented as normalized ChIP-Seq read counts. As we did in the previous section, we compared the performance of DeepHistone on raw test set (without applying NCL) and test set that we get after applying NCL on raw test set. Figure 5 shows the corresponding precision/recall curves.

Observe that the AUPR for DeepHistone on raw test set ranges from 0.678 (for H3K27me3 on cell line H1) to 0.914 (for H3K4me3 on cell line H1). DeepHistone's predictions outperform LR on all histone PTMs. DeepHistone's AUPR improvement over LR ranges from 1.5% (for H3K9ac on cell line K562) to 23.1% (for H3K27me3 on cell line H1). we can also observe that for all PTMs performance of DeepHistone improves on the test set that that we get after applying NCL. That implies less noisy the test set better the performance of DeepHistone .

As we did in the previous section, to make an apples-to-apples comparison we have applied NCL as a pre-processing step of LR and then compare LR and DeepHistone. To compare the performance of LR and DeepHistone, we have used the raw test set (without applying NCL). Table 4 shows AUPR, AUROC and accuracy for DeepHistone and LR when trained and tested on TF-ChIP-Seq data. Observe that AUPR for LR ranges from 0.52 (for H3K27me3 on cell line H1) to 0.891 (for H3K9ac on cell line K562). Predictions of DeepHistone outperform LR on all histone PTMs. DeepHistone's AUPR improvement over LR ranges from 3 % (for H3K9ac on cell line K562) to 15.8 % (for H3K9ac on cell line H1).

We tested the impact of the choice of the window size used to assign PTMs to a particular TSS (the default window size was 100bp). Supplementary Table 14 shows very minor difference in

		AUPR			AUROC			Accuracy		
		DeepHistone		LR	DeepHistone		LR	DeepHistone		LR
Cell Line	PTM	Test Set with NCL	Test Set w/o NCL	w/o NCL	Test Set with NCL	Test Set w/o NCL	w/o NCL	Test Set with NCL	Test Set w/o NCL	w/o NCL
H1	H3K4me3	**0.996**$_{\pm0.000}$	**0.924**$_{\pm0.000}$	0.885	**0.996**$_{\pm0.000}$	**0.962**$_{\pm0.000}$	0.950	**0.977**$_{\pm0.002}$	**0.909**$_{\pm0.002}$	0.888
H1	H3K9ac	**0.970**$_{\pm0.002}$	**0.853**$_{\pm0.002}$	0.729	**0.988**$_{\pm0.001}$	**0.952**$_{\pm0.001}$	0.921	**0.953**$_{\pm0.002}$	**0.867**$_{\pm0.002}$	0.844
H1	H3K27ac	**0.901**$_{\pm0.006}$	**0.787**$_{\pm0.006}$	0.545	**0.978**$_{\pm0.001}$	**0.944**$_{\pm0.001}$	0.909	**0.940**$_{\pm0.002}$	**0.898**$_{\pm0.002}$	0.878
H1	H3K27me3	**0.810**$_{\pm0.014}$	**0.678**$_{\pm0.014}$	0.447	**0.957**$_{\pm0.006}$	**0.902**$_{\pm0.006}$	0.877	**0.954**$_{\pm0.002}$	**0.914**$_{\pm0.002}$	0.922
K562	H3K4me3	**0.992**$_{\pm0.000}$	**0.922**$_{\pm0.000}$	0.901	**0.995**$_{\pm0.000}$	**0.969**$_{\pm0.000}$	0.961	**0.974**$_{\pm0.000}$	**0.915**$_{\pm0.000}$	0.913
K562	H3K9ac	**0.991**$_{\pm0.001}$	**0.921**$_{\pm0.001}$	0.906	**0.995**$_{\pm0.000}$	**0.972**$_{\pm0.000}$	0.969	**0.978**$_{\pm0.002}$	**0.919**$_{\pm0.002}$	0.919
K562	H3K27ac	**0.988**$_{\pm0.008}$	**0.905**$_{\pm0.008}$	0.887	**0.994**$_{\pm0.000}$	**0.967**$_{\pm0.000}$	0.965	**0.974**$_{\pm0.001}$	**0.904**$_{\pm0.001}$	0.912
GM12878	H3K4me3	**0.986**$_{\pm0.001}$	**0.898**$_{\pm0.001}$	0.829	**0.991**$_{\pm0.001}$	**0.958**$_{\pm0.001}$	0.942	**0.966**$_{\pm0.001}$	**0.902**$_{\pm0.001}$	0.883
GM12878	H3K9ac	**0.988**$_{\pm0.002}$	**0.915**$_{\pm0.002}$	0.873	**0.994**$_{\pm0.001}$	**0.988**$_{\pm0.001}$	0.962	**0.972**$_{\pm0.002}$	**0.93**$_{\pm0.002}$	0.906
GM12878	H3K27ac	**0.986**$_{\pm0.001}$	**0.889**$_{\pm0.001}$	0.855	**0.991**$_{\pm0.000}$	**0.971**$_{\pm0.000}$	0.956	**0.969**$_{\pm0.001}$	**0.906**$_{\pm0.001}$	0.901

Table 3: Histone PTM prediction performance (mean ± standard deviation, over five iterations) of DeepHistone (after data cleaning) vs. the logistic regression classifier (LR) using TF ChIP-Seq data; numbers in boldface indicate the best performance

		AUPR		AUROC		Accuracy	
Cell Line	PTM	Test Set with NCL DeepHistone	Test Set w/o NCL LR	Test Set with NCL DeepHistone	Test Set w/0 NCL LR	Test Set with NCL DeepHistone	Test Set w/o NCL LR
H1	H3K4me3	0.924	0.856	0.962	0.946	0.909	0.858
H1	H3K9ac	0.853	0.798	0.952	0.908	0.867	0.841
H1	H3K27ac	0.787	0.729	0.944	0.862	0.898	0.881
H1	H3K27me3	0.678	0.52	0.902	0.901	0.914	0.920
K562	H3K4me3	0.922	0.887	0.969	0.961	0.915	0.919
K562	H3K9ac	0.921	0.891	0.972	0.967	0.919	0.921
K562	H3K27ac	0.905	0.852	0.967	0.921	0.904	0.90
GM12878	H3K4me3	0.898	0.843	0.958	0.944	0.902	0.896
GM12878	H3K9ac	0.915	0.860	0.988	0.961	0.93	0.911
GM12878	H3K27ac	0.889	0.840	0.971	0.95	0.906	0.83

Table 4: Histone PTM prediction performance (mean ± standard deviation, over five iterations) of logistic regression classifier (LR) after applying NCL on training data of TF ChIP-Seq ; numbers in boldface indicate the best performance

DeepHistone's AUPR for other choices of the window size (500bp, 1 kbp and 2 kbp).

We finally compared the predictive performance of DeepHistone trained on TF ChIP-Seq data against its performance when trained on DNA sequence data. Table 5 shows that DeepHistone performed better when trained on TF ChIP-Seq data on all cell lines.

3.3 Prediction of histone PTMs from the DNA sequence & TF bindings (combined)

We combined the 6-mer count from the DNA sequence data and the normalized read count from TF bindings, then we used NCL for data cleaning. We tested the two architectures that were optimized for TF data and DNA sequence data, respectively, and based on their performance we chose the architecture optimized for DNA sequence (with a larger input layer). DeepHistone's prediction performance using both DNA sequence and TF ChIP-Seq is shown in Table 6 for cell line H1.

By comparing these experimental results to the results in Table 5 we observed that the performance of DeepHistone using the combined features is similar to its performance using only DNA sequence features, and lower than its performance using only TF-binding features.

To investigate the reason, we considered H3K4me3 ptm for H1 cell line and ranked the features according to gini importance. We found that top seven features are from TF bindings such as SIN3A, E2F6, TCF12, NRSF, ATF2, SP4 and GABP. Also among top ten features eight of them are TF binding features and two of them k-mer count features. So it seems TF - binding features have played more important role then k-mer count features and adding more k-mer count features does not help.

		AUPR		AUROC		Accuracy	
Cell Line	PTM	TF ChIP-Seq	DNA sequence	TF ChIP-Seq	DNA sequence	TF ChIP-Seq	DNA sequence
H1	H3K4me3	0.996	0.982	0.996	0.985	0.977	0.951
H1	H3K9ac	0.970	0.922	0.988	0.966	0.953	0.908
H1	H3K27ac	0.901	0.730	0.978	0.927	0.940	0.896
H1	H3K27me3	0.810	0.614	0.957	0.912	0.954	0.927
K562	H3K4me3	0.992	0.952	0.995	0.960	0.974	0.907
K562	H3K9ac	0.991	0.927	0.995	0.946	0.978	0.894
K562	H3K27ac	0.988	0.914	0.994	0.939	0.974	0.888
GM12878	H3K4me3	0.986	0.931	0.991	0.950	0.966	0.895
GM12878	H3K9ac	0.988	0.923	0.994	0.965	0.972	0.906
GM12878	H3K27ac	0.986	0.879	0.991	0.827	0.969	0.869

Table 5: Comparing the prediction performance DeepHistone (mean AUPR, AUROC and accuracy) when training on TF ChIP-Seq or DNA sequence data (after data cleaning)

Cell Line	PTM	AUPR	AUROC	Accuracy
H1	H3K4me3	$0.955_{\pm 0.001}$	$0.954_{\pm 0.000}$	$0.922_{\pm 0.003}$
H1	H3K9ac	$0.782_{\pm 0.004}$	$0.913_{\pm 0.002}$	$0.813_{\pm 0.003}$
H1	H3K27ac	$0.70_{\pm 0.003}$	$0.882_{\pm 0.004}$	$0.823_{\pm 0.001}$
H1	H3K27me3	$0.643_{\pm 0.030}$	$0.915_{\pm 0.010}$	$0.925_{\pm 0.004}$

Table 6: DeepHistone's prediction performance (mean AUPR ± standard deviation) using both DNA sequence and TF ChIP-Seq data (after data cleaning)

3.4 Prediction Across Different Cell Lines using TF binding data

Given the excellent performance of DeepHistone on TF binding data we tested its performance across cell lines, i.e., we trained DeepHistone on one cell line and tested on another. For this purpose, we selected as features the 17 TFs which are common to all cell lines, then applied NCL on all datasets. Like before, NCL is used to pre-process the train set not the test set primarily. However, for further exploration, we also compared the performance of DeepHistone on raw test set(without applying NCL) and test set that we get after applying NCL on raw test set. Observe in Table 7 that the prediction performance (AUPR) of DeepHistone for H3K4me3 across cell lines is lower than the prediction performance on the same cell line. The same is true for H3K9ac (Supplementary Table 12) and for H3K27ac (Supplementary Table 13). This suggests that cell type specific TFs are responsible for determining histone PTM profiles, which reconfirms the finding in [4]. Observe however, that the prediction performance of DeepHistone using the common 17 TFs is almost the same as using all the TFs. This is not the case for the logistic regression classifier. Finally, observe that DeepHistone significantly outperforms LR on all cross cell line predictions. Precision/recall curves for cross cell line prediction for H3K4me3 are shown in Supplementary Figure 4. Experimental results for H3K9ac and H3K27ac are shown in Supplementary Figure 2 and Supplementary Figure 3, respectively. Like before, observe that for all PTMs performance of DeepHistone improves on the test set that that we get after applying NCL. That implies less noisy the test set better the performance of DeepHistone.

3.5 Minimal Sets of Transcription Factors

We followed the procedure described in Section 2.6 to compute the minimal sets of TFs. Table 8 shows these sets for each pair of cell type and histone PTM. Recall that the TFs in each minimal set provides almost the same prediction performance of the entire available set of TFs. For example, choosing only SIN3A and MAX for H3K4me3 on H1 cell line provides a prediction accuracy within 1% of the accuracy obtained using all thirty TFs.

Figure 6 shows Venn diagrams for the minimal sets of each histone PTM for a particular cell line. Observe that H3K4me3 can be predicted by only two TF in H1; the other PTMs require significantly more TFs in H1. In contrast, the minimal set for H3K4me3 is the largest for cell line GM12878. Observe that (1) SIN3A is shared by all the histone PTMs for cell line H1, (2) TEAD4, E2F6, MAX are shared by all the histone PTMs for cell line K562 and (3) ELF1 is shared by all the histone PTMs for cell line GM12878.

Figure 7 shows Venn diagrams for the minimal sets of each cell lines for a particular histone PTM. Observe that there are no TFs for H3K4me3 and H3K9ac that are shared by all three cell lines, i.e., the minimal set for H3K4me3 and H3K9ac are highly specific to that cell-line. Also observe that there is only one TF for H3K27ac shared by all three cell lines, namely CTCF.

4 CONCLUSIONS

We proposed for the first time a deep learning architecture to predict histone PTMs from TF binding and DNA sequence data. We determined that histone PTM can be predicted more accurately from TF ChIP-Seq data than from DNA sequence, which reconfirms the finding in [4]. We also determined that combining both feature sets does not seem to improve the prediction. Our experimental results show that DeepHistone achieves a prediction accuracy substantially higher than the logistic regression model proposed in [4]. The competitive advantage of DeepHistone lies in synergistic use of deep learning combined with an effective data cleaning preprocessing step. Our framework has also enabled the discovery that

	Test on H1			Test on K562			Test on GM12878		
	DeepHistone		LR	DeepHistone		LR	DeepHistone		LR
	Test Set with NCL	Test Set w/o NCL	w/o NCL	Test Set with NCL	Test Set w/o NCL	w/o NCL	Test Set with NCL	Test Set w/o NCL	w/o NCL
Train on H1	0.993	0.904	0.878	0.951	0.855	0.777	0.848	0.746	0.651
Train on K562	0.970	0.871	0.793	0.990	0.901	0.882	0.733	0.670	0.479
Train on GM12878	0.972	0.882	0.848	0.959	0.871	0.825	0.981	0.878	0.790

Table 7: DeepHistone's predictive performance (AUPR) across cell lines for H3K4me3 (after data cleaning)

Cell line	PTM	Minimal Set
H1	H3K4me3	{SIN3A, MAX}
H1	H3K9ac	{SIN3A, TCF12, NRSF, CREB1, YY1, SP4, SIX5}
H1	H3K27ac	{YY1, SIN3A, ATF2, CREB1, TCF12, SP4, NRSF, SP1, CTCF}
H1	H3K27me3	{TCF12, SIN3A, E2F6, ATF2, GABP, NRSF, SIX5, TEAD4, SP4, MAX, BACH1, SP1, CMYC, CREB1, YY1}
K562	H3K4me3	{E2F6, MAX, TEAD4, ATF3, GATA2, ZNF263, CREB1, E2F4}
K562	H3K9ac	{E2F6, MAX, CMYC, TEAD4, CTCF, ATF3}
K562	H3K27ac	{MAX, CMYC, TEAD4, E2F6, CTCF}
GM12878	H3K4me3	{ATF2, BATF, BCL3, BHLHE40, CFOS, E2F4, EBF1, ELF1, ELK1, ERRA, ETS1, FOXM1, IKZF1, IRF3, IRF4}
GM12878	H3K9ac	{SP1, NFATC1, CREB1, TCF3, STAT3, BATF, ELF1, RUNX3, POU2F2}
GM12878	H3K27ac	{NFATC1, SP1, CREB1, ELF1, CTCF, TCF3, BCL3}

Table 8: Minimal sets of TFs for cell specific histone PTM

the knowledge of a small subset of transcription factors (which are histone-PTM- and cell-type-specific) can provide almost the same prediction accuracy that can be obtained using all the transcription factors.

ACKNOWLEDGEMENTS

The authors thank Prof. Duncan Sproul (University of Edinburgh) for providing help and for useful discussions.

FUNDING

This work was supported in part by the US National Science Foundation [IOS-1543963, IIS-1526742, IIS-1814359].

REFERENCES

[1] Babak Alipanahi, Andrew Delong, Matthew T Weirauch, and Brendan J Frey. 2015. Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning. *Nature biotechnology* 33, 8 (2015), 831.

[2] Christof Angermueller, Heather J Lee, Wolf Reik, and Oliver Stegle. 2017. DeepCpG: accurate prediction of single-cell DNA methylation states using deep learning. *Genome biology* 18, 1 (2017), 67.

[3] Artem Barski, Suresh Cuddapah, Kairong Cui, Tae-Young Roh, Dustin E Schones, Zhibin Wang, Gang Wei, Iouri Chepelev, and Keji Zhao. 2007. High-resolution profiling of histone methylations in the human genome. *Cell* 129, 4 (2007), 823–837.

[4] Dan Benveniste, Hans-Joachim Sonntag, Guido Sanguinetti, and Duncan Sproul. 2014. Transcription factor binding predicts histone modifications in human cell lines. *Proceedings of the National Academy of Sciences* 111, 37 (2014), 13367–13372. https://doi.org/10.1073/pnas.1412081111 arXiv:http://www.pnas.org/content/111/37/13367.full.pdf

[5] James Bergstra and Yoshua Bengio. 2012. Random search for hyper-parameter optimization. *Journal of Machine Learning Research* 13, Feb (2012), 281–305.

[6] Leo Breiman. 2001. Random forests. *Machine learning* 45, 1 (2001), 5–32.

[7] François Chollet et al. 2015. Keras. https://github.com/keras-team/keras.

[8] ENCODE Project Consortium et al. 2004. The ENCODE (ENCyclopedia of DNA elements) project. *Science* 306, 5696 (2004), 636–640.

[9] Xianjun Dong, Melissa C Greven, Anshul Kundaje, Sarah Djebali, James B Brown, Chao Cheng, Thomas R Gingeras, Mark Gerstein, Roderic Guigó, Ewan Birney, et al. 2012. Modeling gene expression using chromatin features in various cellular contexts. *Genome biology* 13, 9 (2012), R53.

[10] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. 249–256.

[11] Rosa Karlić, Ho-Ryun Chung, Julia Lasserre, Kristian Vlahoviček, and Martin Vingron. 2010. Histone modification levels are predictive for gene expression. *Proceedings of the National Academy of Sciences* 107, 7 (2010), 2926–2931.

[12] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[13] Jorma Laurikkala. 2001. Improving identification of difficult small classes by balancing class distribution. In *Conference on Artificial Intelligence in Medicine in Europe*. Springer, 63–66.

[14] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature* 521, 7553 (2015), 436.

[15] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, Nov (2008), 2579–2605.

[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python . *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[17] Daniel Quang and Xiaohui Xie. 2016. DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. *Nucleic acids research* 44, 11 (2016), e107–e107.

[18] Shashank Singh, Yang Yang, Barnabas Poczos, and Jian Ma. 2016. Predicting enhancer-promoter interaction from genomic sequence with deep neural networks. *bioRxiv* (2016), 085241.

[19] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research* 15, 1 (2014), 1929–1958.

[20] Matthew D VerMilyea, Laura P O'Neill, and Bryan M Turner. 2009. Transcription-independent heritability of induced histone modifications in the mouse preimplantation embryo. *PLoS One* 4, 6 (2009), e6086.

[21] Yong Zhang, Tao Liu, Clifford A Meyer, Jérôme Eeckhoute, David S Johnson, Bradley E Bernstein, Chad Nusbaum, Richard M Myers, Myles Brown, Wei Li, et al. 2008. Model-based analysis of ChIP-Seq (MACS). *Genome biology* 9, 9 (2008), R137.

[22] Yi Zhang and Danny Reinberg. 2001. Transcription regulation by histone methylation: interplay between different covalent modifications of the core histone

tails. *Genes & development* 15, 18 (2001), 2343–2360.

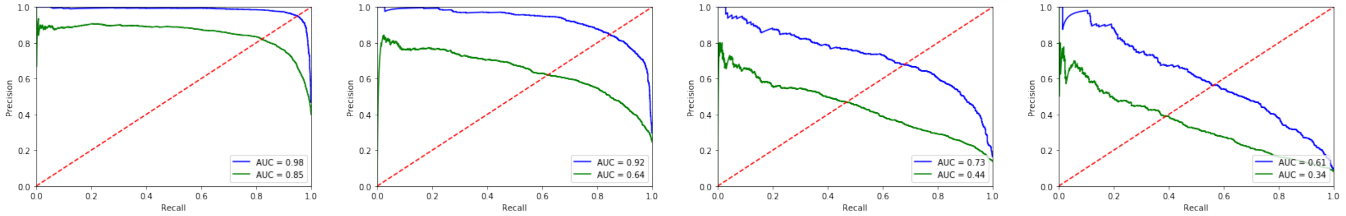

Figure 4: Precision/recall curves for the logistic regression (green line) and DeepHistone (blue line) based on DNA sequence data (after data cleaning); left to right: H3K4me3, H3K9ac, H3K27ac, H3K27me3

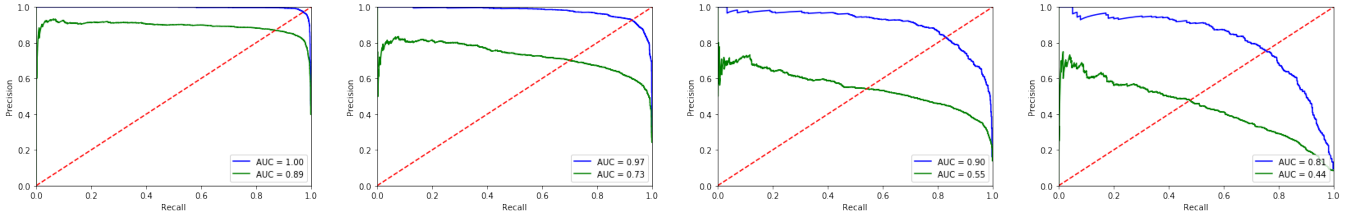

Figure 5: Precision/recall curves for logistic regression (green line) and DeepHistone (blue line) on TF ChIP-Seq data (after data cleaning); left to right: H3K4me3, H3K9ac, H3K27ac, H3K27me3

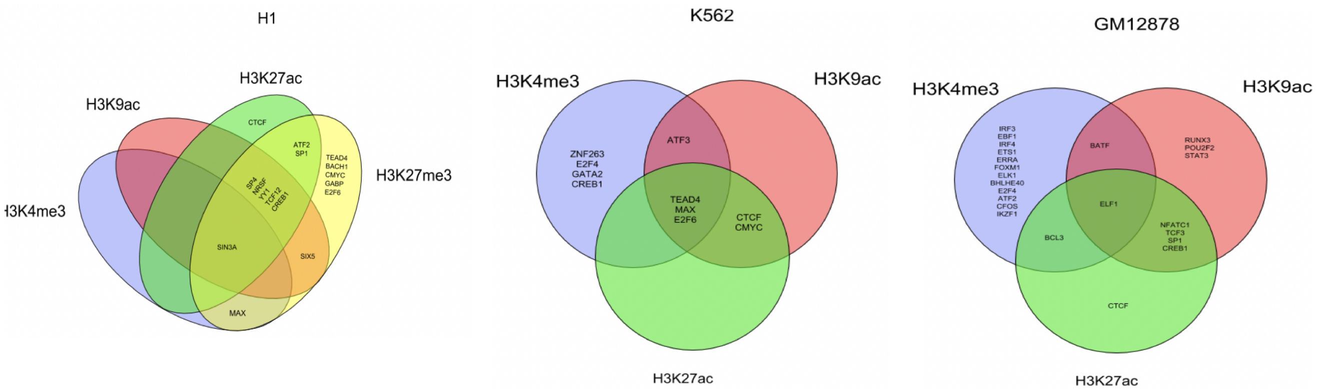

Figure 6: Minimal sets of TFs of each histone PTM for three cell lines

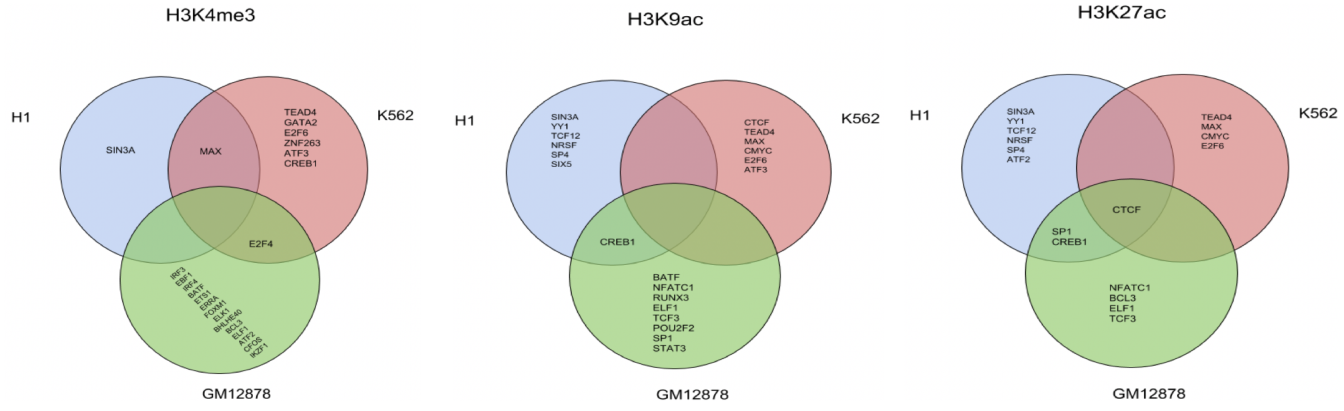

Figure 7: Minimal sets of TFs of each cell line for the three PTMs