A Neural Network Based Algorithm for Gene Expression Prediction from Chromatin Structure

Marco Frasca and Giulio Pavesi
University of Milan
{marco.frasca, giulio.pavesi}@unimi.it

***Abstract*—Gene expression is a very complex process, which is finely regulated and modulated at different levels. The first step of gene expression, the transcription of DNA into mRNA, is in turn regulated both at the genetic and epigenetic level. In particular, the latter, which involves the structure formed by DNA wrapped around histones (chromatin), has been recently shown to be a key factor, with post–translational modifications of histones acting combinatorially to activate or block transcription. In this work we addressed the problem of predicting the level of expression of genes starting from genome–wide maps of chromatin structure, that is, of the localization of several different histone modifications, which have been recently made available through the introduction of technologies like ChIP–Seq. We formalized the problem as a multi–class bipartite ranking problem, in which for each class a gene can be under– or over–expressed with respect to a given reference expression value. In order to deal with this problem, we exploit and extend a semi–supervised method (COSNet) based on a family of Hopfield neural networks. Benchmark tests performed on genome–wide tests in six different human cell lines yielded satisfactory results, with clear improvements over the alternative approach most commonly adopted in the literature.**

I. Introduction

Gene expression refers to the process of producing a protein from sequence information encoded in DNA. It is highly regulated at different levels, including transcriptional regulation, splicing, and modification, export, and degradation of protein products. The first step, the regulation of the transcription of DNA into mRNA, can take place both at the genetic and epigenetic level. Usually, the former is defined as a direct or indirect interaction between DNA and dedicated molecules called transcription factors, while epigenetic regulation makes DNA accessible to transcription factors through chemical modifications of chromatin. The basic unit of chromatin is structured like beads on a string, where the string is DNA and each bead is a DNA–protein complex called a *nucleosome*. Nucleosomes are in turn made of proteins called *histones*, and, more in detail, by two copies of four core histones (H2A, H2B, H3 and H4) with roughly 147 base pairs (bp) of DNA wrapped around it (Figure 1). Several post-translational modifications, such as methylation, acetylation, and phosphorylation, may occur on some of the aminoacids forming the histones. These modifications can alter the structure and function of chromatin, in turn making DNA accessible (or blocking it) to transcription factors and the transcriptional machinery, and allowing gene expression to start. It has been proposed that these histone modifications can occur combinatorially to form a 'histone code' that is able to switch on or off the transcription of genes by co–operatively altering chromatin structure in a suitable way [1], [2], [3]. For example, methylation of lysine (K) in position four of histone H3 (H3K4me1) or its tri-methylation (H3K4me3) are usually associated with transcriptional activation and are localized around the transcription start site (TSS) of genes; H3K9me3 and H3K27me3 are considered to be responsible for transcriptional repression; H3K36me3 is usually found within the transcribed regions of active genes [4]. Hence, the histone code seems able to mark in a different way the parts of a transcribed gene, that is, some modifications are associated with the TSS, others with the region that has to be transcribed and where transcription has to stop. Transcription starts and continues producing a RNA only if the right combinations of histone modifications are found at the right positions on the genome.

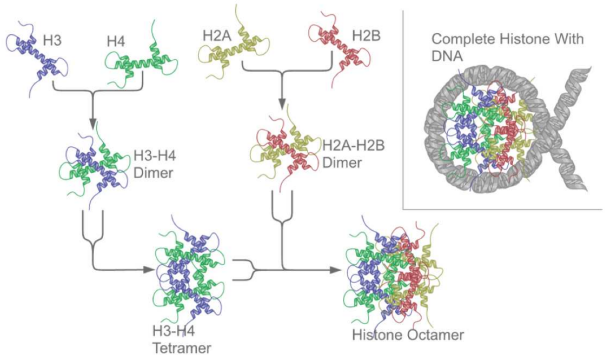

Fig. 1. The structure of a nucleosome, with the eight core histones building it.

On the other hand, while information encoded by DNA can be considered as static, epigenetic factors are highly dynamic. Thus, as gene expression changes according to 'cell type' and 'status' (i.e. tissue, developmental stage, external stimuli, and so on), chromatin structure and histone modifications change likewise. ChIP–Seq experiments, that is, chromatin immunoprecipitation (ChIP) followed by sequencing using next–generation technologies, are very powerful techniques that permit to build genome–wide maps of protein–DNA interactions. That is, given a protein of interest which can interact with DNA, it can be extracted from cell nuclei together

with the DNA bound to it, which in turn can be sequenced and analyzed. Applied to histones, ChIP–Seq permits to extract those nucleosomes which carry a specific histone modification: in other words, it can be used to build genome–wide maps of, for example, the position of nucleosomes where histone H3 has lysine four mono– or di– or trimethylated, and separate maps for each of these three possibilities.

Indeed, large–scale application of ChIP-Seq in the last few years has culminated in "whole epigenome projects" [5], which aim to build genome wide maps of the most relevant histone modifications in several different cell lines and tissues for the most widely studied organisms, first of all human. But, rather than yes/no labels marking the presence/absence of a given modification in a given position of the genome, ChIP–Seq experiments can produce maps of *enrichment* across the whole genome. In other words, enrichment provides an estimate of the abundance of the corresponding DNA segment in the sample extracted from the cells studied, and thus of the frequency with which a given histone modification can be found associated with a given DNA region (Figure 2). Usually, regions of interest are immediately upstream and downstream of the transcription start site, and/or the whole transcribed region, that is, those regions which are characterized by different modifications.

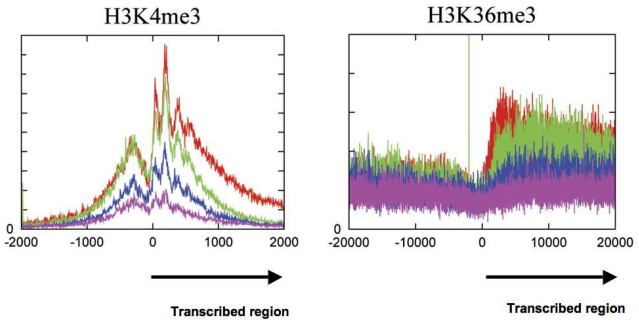

Fig. 2. Enrichment distribution obtained through ChIP–Seq experiments for two different histone modifications, plotted around the TSS (coordinate 0 on the x axis) of genes. H3K4me3 (left) is enriched in the regions flanking the TSS for genes with high (red), and medium (green) expression levels, much less so for those expressed at low (blue) or not expressed (purple). H3K36me3 (right) is again more enriched for expressed genes, but only in the transcribed region (positive values on the x axis). Distances on the x axis are measured in DNA base pairs. Transcription of the gene starts at position 0, proceeding towards the right.

II. Predicting Gene Expression

The transcript (expression) level of genes, that is, how much RNA is produced by each gene on a given genome, can be nowadays measured with different technologies, like microarrays or RNA-Seq, and it is usually denoted by a real number ≥ 0. Clearly, 0 indicates that no transcription (expression) is taking place. The availability of these data, crossed with those derived from ChIP–Seq experiments, has, in turn, led quite naturally to the following hypothesis: if chromatin structure and histone modifications are responsible for the regulation of transcription, and if now genome–wide maps of several histone modifications are available, then it should be possible, to some extent, to infer and predict the level of transcription of genes starting from these maps, that is, predict gene expression from chromatin structure and the relative position of histone modifications. Indeed, even by using quite naïve approaches, it can be seen how some histone modifications, and in particular their enrichment determined through ChIP–Seq, yield clear correlations (or anti–correlations) with the level of gene expression.

Starting from these observations, suitable models can be built, that, starting from enrichment maps of different histone modifications, can infer with good accuracy the activation of the gene transcription and the respective level. The problem itself can be recast in different ways, at different levels of complexity: we could predict which genes have an expression level > 0 versus those whose expression level is 0 (predicting transcribed versus silent genes), or which genes are over- or under–expressed with respect to a mean (or median, or modal) expression value, or in detail estimate the expression level of each gene.

In this work, we define the *Gene Expression Prediction Problem* (GEP) as follows. Given a class (cell line under fixed conditions) and the chromatin features for each gene, the problem is inferring a ranking on genes such that high (resp. low) ranks correspond to high (resp. low) levels of gene transcripts. Genes with high (low) rank are then predicted being over (under)–expressed. Classes usually are not balanced, that is, there is no guarantee to have approximately 50% of the genes over– or under–expressed.

Since the quite recent introduction of ChIP–Seq experiments, a few approaches have been proposed for the GEP problem in the last couple of years. Most of these approaches adopt inductive methods such as linear regression techniques [6], [7], [8]. Unfortunately, they do not take into account the label imbalance that may characterize the class labels, and many learning systems may suffer a decay in performance when predicting on unbalanced data [9]. Moreover, due to their time complexity, techniques like support vector machines [10] do not properly scale on large size data, and for most of the widely studied organisms (e.g. human), the number of genes is considerably large. Finally, adopting supervised machine learning methods does not take into account, during the learning phase, the relationships existing among labeled and unlabeled genes (that is, those genes for which we want to make a prediction).

In order to deal with these learning issues, we propose a semi-supervised method which adapts to GEP a recently proposed algorithm to predict gene functions [12]. The method, which is based on a family of parametrized Hopfield networks, can be summarized as follows:

1. The network parameters are automatically learnt from the labeled genes to deal with the data imbalance.

2. The connections among labeled and unlabeled genes are

embedded in the network.

3. The method is able to infer both the genes that are over–expressed with respect to a given expression boundary and the detailed gene expression levels.

4. The network dynamics is restricted solely to the genes to be predicted, considerably reducing the time complexity.

The method has been validated in predicting the expression level of the human genes in six different cell lines and with a genome–wide approach, considering, in average, more than thirty chromatin features. The results revealed significant improvements of the method we propose w. r. t. the approach most commonly used in the literature to predict gene expression levels.

III. Gene Expression Prediction (GEP) as semi-supervised bipartite ranking problem

All in all, the GEP problem can be formalized as semi-supervised bipartite ranking problem [13] on undirected graphs, in which genes $V = \{1, 2, \ldots, n\}$ are only partially labeled and $\boldsymbol{H}$ is the matrix containing the histone modifications levels, where H_{ij} is the level of histone modification j at gene i, for each $i \in \{1, 2, \ldots, n\}$ and $j \in \{1, 2, \ldots, m\}$. The matrix $\boldsymbol{H}$ is processed in a squared similarity matrix $\boldsymbol{W}$ by defining the component w_{ij} as the Pearson's correlation coefficient of the vectors $\boldsymbol{H}_i$ and $\boldsymbol{H}_j$, and setting to zero the diagonal and the negative components. The weight $w_{ij} \in [0, 1]$ represents a similarity index between genes i and j, with $w_{ij} = w_{ji}$, that is, how similar genes i and j are with respect to chromatin structure and histone modifications: genes with similar chromatin structure should yield similar expression values. Moreover, genes V are bipartitioned in the sets (up to a permutation) $U = \{1, 2, \ldots, h\}$ and $L = \{h+1, h+2, \ldots, n\}$ of the unlabeled and labeled genes respectively. The label $y_i \in \{-1,\ 1\}$ of gene $i \in L$ describes the known condition of the gene i w. r. t. a previously fixed cell line condition: 1 stands for *over–expressed*, -1 for *under–expressed*. Moreover, let $L^+ = \{i \in L | y_i = 1\}$ and $L^- = \{i \in L | y_i = -1\}$ be the sets of positive and negative instances respectively. We can refer to $L^+,\ L^-$ and $\boldsymbol{W}$ as the prior information.

The *gene expression prediction problem* consists of inferring, on the basis of prior information, a ranking function $\phi : U \longrightarrow \mathbb{R}$ which associates each gene $i \in U$ with an expression value $\phi(i)$ such that, future positive instances have higher rank (expression value) than negative ones. From this point of view, GEP is cast as a semi-supervised learning problem on graphs, since gene expression profiles can be predicted by exploiting both labeled and unlabeled nodes (genes) and the weighted connections among them.

Finally, in order to simplify the problem, we assume that prior information is not affected by noise. Nevertheless, it is worth noting that usually known technologies for measuring gene transcript levels still are affected by both experimental and biological noise [14], [15].

IV. Hopfield Network model for GEP

Hopfield networks are artificial neural networks whose dynamics admit a Lyapunov function [16]. This model has been widely used to address different issues, including content-addressable memory [17], [18], [19], discrete nonlinear optimization [20], binary classification for protein function prediction [21]. In particular, a Hopfield network–based algorithm, *COSNet*, has been recently proposed to predict the biological functions of genes, resulting among the best algorithms for this task [11], [12]. This algorithm, originally designed to predict gene functions, can be extended in order to predict also the gene expression levels.

A. COSNet

Informally, COSNet is a Hopfield network model which, unlike classical Hopfield networks, conceptually separates labels and neuron states: neuron states are set to $\sin\alpha$ for "positive" neurons and to $-\cos\alpha$ for "negative" neurons, where α is a real number in the interval $[0, \frac{\pi}{2}[$. By automatically learning from training data the parameters α and γ (which represents the activation threshold for each neuron), this algorithm reaches accurate predictive capabilities also in presence of highly unbalanced data [11]. More in detail, for each functional class, a Hopfield network with neuron V and connection strengths encoded in the similarity matrix $\boldsymbol{W}$ is considered. The initial state is set to $u_i = 0$ for neurons $i \in U$, to $l_i = \sin\alpha$ for neurons $i \in L^+$ and to $l_i = -\cos\alpha$ for neurons in $i \in L^-$. The parameters $\alpha,\ \gamma$ are estimated

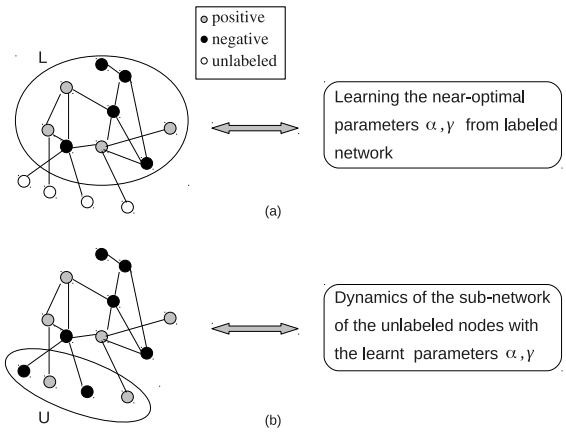

Fig. 3. Main steps of COSNet: (a) the labeled part of the network is used to learn the near–optimal parameters; (b) the learnt parameters are extended to the unlabeled part of the network which is simulated until convergence to infer labels for unlabeled neurons.

on the sub–network restricted to neurons in L by an efficient approximated algorithm (Figure 3 (a)) which preserves the minimization of the overall energy, and then extended to the sub–network of neurons U, which is simulated to infer a binary

prediction for unlabeled neurons (Figure 3 (b)). The network evolves according to the following asynchronous dynamics :

$$u_i(t) = \begin{cases} \sin\alpha & \text{if } \sum_{j=1}^{i-1} w_{ij}u_j(t) + \sum_{k=i+1}^{h} w_{ik}u_k(t-1) - \theta_i > 0 \\ -\cos\alpha & \text{if } \sum_{j=1}^{i-1} w_{ij}u_j(t) + \sum_{k=i+1}^{h} w_{ik}u_k(t-1) - \theta_i \leq 0 \end{cases} \quad (1)$$

where $u_i(t)$ is the value of neuron $i \in U$ at time t and $\theta_i = \gamma - \sum_{j=h+1}^{n} w_{ij}l_j$ is the corresponding activation threshold, which also includes the influence on this node of the labeled neurons L (whose values are clamped during the network dynamics). The state of the network at time t is $\boldsymbol{u}(t) = (u_1(t), u_2(t), \ldots, u_h(t))$. The system admits a Lyapunov state function named *energy function*:

$$E(\boldsymbol{u}) = -\frac{1}{2}\sum_{i\neq j} w_{ij}u_iu_j + \sum_{i=1}^{h} u_i\theta_i \quad (2)$$

It is easy to see that the dynamics (1) converges to an equilibrium state $\overline{\boldsymbol{u}}$ which corresponds to a local minimum of the energy function E. Finally, each neuron i in U is classified as positive if $\overline{u}_i = \sin\alpha$, as negative otherwise.

B. COSNet for GEP

In our context, the approach of COSNet is motivated by the fact that minimizing the energy defined in Equation (2) means maximizing the weighted sum of the edges connecting neurons with the same activation values. Even though this model cannot rank the instances and it can just assign a binary label $\{+, -\}$ to each neuron, we can observe that neurons "strongly connected" with other positive neurons tend in turn to be positive (over–expressed), while the opposite is true when strong connections with negative (under–expressed) neighbors prevail. Furthermore, such an approach lets the node labels propagate through the network, so that neurons can get information also from non neighboring neurons. In this way the final labeling corresponds to local and global consistency. We can thereby effectively deal with the GEP problem by considering the equilibrium state $\overline{\boldsymbol{u}}$ reached by the network dynamics. In order to define a ranking for genes in U, we can associate the ranking scores with the "strength" of positive and negative predictions. That is, some positive predicted neurons are connected to positive neurons "more strongly" than the others; accordingly, since in our context a positive prediction means over–expression, these neurons should have a higher rank. On the other hand, negative predicted neurons with stronger negative neighbors should be in a lower rank. Furthermore, our aim is also to define a score for each gene which corresponds to both local and global network stability. In this regard, we consider the energy contribution $E(\overline{u}_i)$ of a single node $i \in U$ to the overall energy (2) at equilibrium:

$$E(\overline{u}_i) = -\overline{u}_i \sum_{j\neq i}(w_{ij}\overline{u}_j - \theta_i) \quad (3)$$

Indeed, since the energy E is minimized through the dynamics, low values of $E(\overline{u}_i)$ correspond to stable states $\overline{u}_i$ for the node i, and it can be interpreted as more reliable predictions. From (3) we can derive a score $\phi(i)$ associated to each node i:

$$\phi(i) = \sum_{j\neq i}(w_{ij}\overline{u}_j - \theta_i) \quad (4)$$

It is easy to see that this choice deals with the GEP ranking requirements described in Section III. For positive predictions (corresponding to $\overline{u}_i = \sin\alpha$), the score $\phi(i)$ is positive and large values of $\phi(i)$ correspond to low values of the energy $E(\overline{u}_i)$. Note that this is true when we have a large number of strongly connected positive nodes in the neighborhood of node i. The opposite is true for negative predictions: the score $\phi(i)$ is negative and low values of $\phi(i)$ correspond to low energy states for the node i.

Finally, we point out that, since the score $\phi(i)$ depends on the number of neighbors j ($w_{ij} \neq 0$) of node i (node degree), the algorithm may suffer a decay in performance when the distribution of node degrees has a high variance. In other words, high (resp. low) values $\phi(i)$ may be due not solely to a prevalence of positive (resp. negative) neighbors, but also to a high node degree. To prevent it, we divide the score $\phi(i)$ by the node degree $d_i = \sum_{j=1}^{n} w_{ij}$, obtaining a final score

$$\phi'(i) = \frac{\sum_{j\neq i}(w_{ij}\overline{u}_j - \theta_i)}{d_i} \quad (5)$$

In summary, the method we propose predict the expression levels of genes by assigning to each gene $i \in U$ the score $\phi'(i)$; on the other hand, we can also predict the under– and over–expressed genes by means of the COSNet binary predictions.

V. Algorithm validation

A. Experimental setting

Since, as briefly discussed in the introduction, gene expression as well as chromatin structure significantly changes according to the type of cell/tissue and/or external stimuli, we validated the algorithm by predicting expression of 12018 human RefSeq genes in six different cell lines, taking advantage of the data produced in the ENCODE project[22] and made available through the UCSC Genome Browser database[23]. We considered the cell lines with the highest number of genome–wide histone modifications maps available (at least 10) and with expression data available as well. Each histone modification was then associated with each gene by considering the enrichment averaged 1) in the region 500 bp upstream of the TSS; 2) in the region 500 bp downstream of the TSS, and 3) in the whole transcribed region. Each modification, thus, yielded three separate features for each gene. Since genes annotated in complex organisms like human might overlap one another, we considered only genes that did not overlap others in the transcribed region. It is worth mentioning that, although as in the examples shown in Figure 2 preferred enrichment regions are known for several histone modifications (e.g. H3K4me3 for the regions flanking the TSS; H3K36me3 for the transcribed region), we did not introduce in our model any prior knowledge of this kind. The cell lines and the corresponding number of features are listed in Table I,

together with the number of histone modifications available for each (Table II).

TABLE I
CELL LINES ADOPTED IN THE EXPERIMENTAL VALIDATION OF THE ALGORITHM, AND THE CORRESPONDING NUMBER OF CHROMATIN FEATURES AVAILABLE.

Cell line	Description	Features
H1 hESC	Human embryonic stem cells	33
HepG2	Cell line derived from patient with liver carcinoma	33
HeLa-S3	Immortalized cell line derived from a cervical cancer patient	33
GM12878	Lymphoblastoid cell line produced by EBV transformation	33
HUVEC	Human umbilical vein endothelial cells	33
K562	Immortalized cell line produced from a patient with chronic myelogenous leukemia	54

B. Preprocessing

The matrix similarity $\boldsymbol{W}$ has been thresholded by ensuring each row has at least a non zero component. Then, the thresholded matrix has been normalized as follows:

$$\hat{\boldsymbol{W}} = \boldsymbol{T}^{-\frac{1}{2}} \boldsymbol{W} \boldsymbol{T}^{-\frac{1}{2}}$$

where $\boldsymbol{T}$ is a diagonal matrix in which $T_{ii} = \sum_j W_{ij}$ for each $i \in \{1, 2, ..., n\}$. Observe that $\hat{\boldsymbol{W}}$ is still symmetric.

In order to define gene labels, we once again took advantage of the ENCODE data at the UCSC Genome Browser database[23]. We retrieved the RNA–Seq signal map for the six cell lines considered, and estimated the expression (transcription) level of each gene by computing the mean RNA–Seq signal across the exons of the gene. This yielded the expression vector $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)$. Then, we transformed each component of the expression vector to a logarithmic scale, with $\hat{y}_i = log(\frac{(y_i+1)}{\mu_y})$, where μ_y is the mean value of vector $\mathbf{y}$. We added 1 to each expression value in order to assure the logarithm would always be defined. Finally, each component $\hat{y}_i$ has been scaled in the $[-1, 1]$ interval by setting $y'_i = \frac{(\hat{y}_i - \min \hat{\boldsymbol{y}}) * 2}{\max \hat{\boldsymbol{y}} - \min \hat{\boldsymbol{y}}} - 1$. The resulting expression vector $\boldsymbol{y'}$ is transformed in a binary vector by setting a threshold t_y such that the components $y'_i > t_y$ are scaled in $]0, 1]$ and the components $y'_i \leq t_y$ are scaled in the interval $[-1, 0]$, obtaining in this way a new vector $\overline{\boldsymbol{y}}$. In order to define t_y, we adopted two different criteria: in the first (C1) we simply set $t_y = 0$. The second one (C2) derives from a further analysis of the distribution of gene expression values $\boldsymbol{y'}$, shown in Figure 4 for the H1 hESC cell line. We can observe a bimodal distribution, and, accordingly, two categories of genes can be defined: genes whose expression is lower than the value corresponding to the local minimum (marked by the vertical line) between the modal peaks, and genes whose expression is greater than this value. In C2 we set t_y equal to this value, thus defining two classes of over– and under–expressed genes. While the bimodal distribution appears in every cell line, the local minimum value is clearly cell line–specific.

TABLE II
DETAILS OF THE FEATURES EMPLOYED IN EACH CELL LINE ANALYZED. FOR FIVE CELL LINES WE EMPLOYED HISTONE MODIFICATIONS ONLY, WHILE K562 ALSO INCLUDES GENOME–WIDE BINDING MAPS FOR HISTONE–MODIFYING AND CHROMATIN–REMODELLING PROTEINS (HDAC1SC6298, HDAC2A300705A, P300 M PHF8A301772A, PLU1, SAP3039731)

Cell line	Histone Modification
H1 hESC HepG2 HeLa-S3 GM12878 HUVEC	H2az, H3k9me3, H3k27ac H3k27me3, H3k36me3, H3k4me1 H3k4me2, H3k4me3, H3k79me2 H3k9ac, H4k20me1
K562	H2az, H3k27ac, H3k27me3 H3k36me3, H3k4me1, H3k4me2 H3k4me3, H3k79me2, H3k9ac H3k9me1, H3k9me3, H4k20me1 Hdac1sc6298, Hdac2a300705a, P300 Phf8a301772a, Plu1, Sap3039731

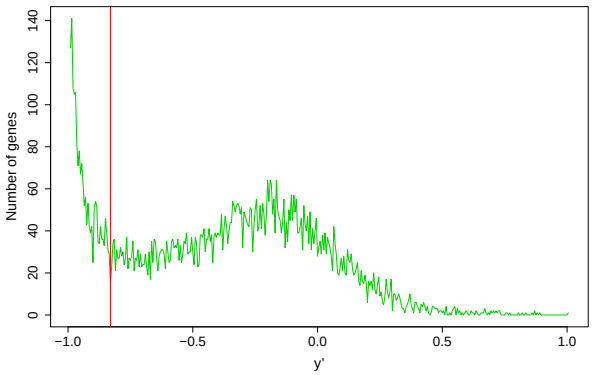

Fig. 4. Expression level distribution of genes in the H1 hESC cell line. The vertical line marks the boundary (expression value) for over–expressed genes.

Finally, we defined the label s_i for each gene i such that $s_i = 1$ iff gene $\overline{y}_i > 0$, $s_i = -1$ otherwise.

C. Learning task

We applied our algorithm with a 10-fold (randomly chosen) cross validation procedure, in which at each step the binary labels of a fold are hidden and the corresponding expression levels and binary labels predicted by using the other nine fold as training set. We set the regularization parameter of the algorithm as $\beta = 0.0001$, as suggested in [11]. We employed as performance measures the Pearson's correlation between the measured $\overline{\boldsymbol{y}}$ and predicted expression values, and accuracy in the prediction of the two classes of genes.

D. Results

We compared our method with the linear regression algorithm (LinReg), a widely used algorithm in the context of

gene expression prediction [6], [7], [8]. We used the linear regression algorithm implementation provided by the *lm* function of the R package *stats*. Figure 5 shows a summary of the results. Most of the experiments reveal a strong correlation for

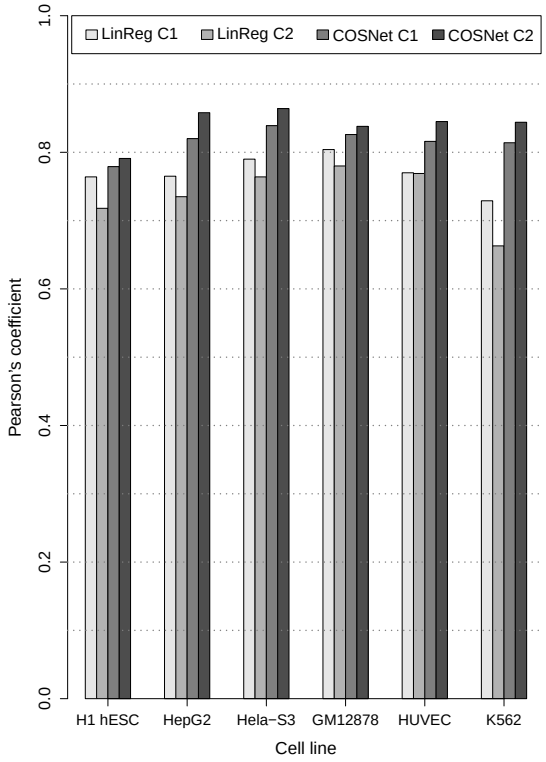

Fig. 5. Performance comparison between linear regression and our approach in terms of Pearson's correlation coefficient between the measured and predicted expression values in all the six considered cell lines. The results relative to both the adopted criterions C1 and C2 to define the threshold t_y are shown.

COSNet between the predicted and measured expression levels for both the two thresholding criteria employed (median 0.818 and 0.844 for C1 and C2, respectively), with C2 performing slightly better than C1 in all the data sets. Moreover, the LinReg algorithm achieves significantly worse results in all the cell lines and with both criteria. Since COSNet is also a binary classifier, in Table III we also report the accuracy and the number of positive instances (those considered over–expressed) for each experiment, which obviously changes according to t_y. COSNet significantly outperforms the LinReg algorithm also in terms of accuracy, that for the LinReg has been computed by scaling the predicted expression values in the $[-1, 1]$ interval, and then considering over–expressed those genes with a positive predicted expression (we remark that this choice may be suboptimal, and more refined strategies may lead to better accuracies). The choice of t_y according to criterion C1 leads to unbalanced data, and we can observe a decay of performance in terms of accuracy for both algorithms (median 0.603 and 0.874) w. r. t. the results obtained by adopting criterion C2 (median 0.858 and 0.890), where the labels are more balanced. This is not surprising, since it is well known that the class imbalance affects the performances of several classification systems [9]. Nevertheless, we can observe that, since we adopted a cost-sensitive algorithm, the accuracy obtained by our method on balanced and unbalanced data are comparable, whereas LinReg achieves a significantly worse accuracy on unbalanced data.

TABLE III
PREDICTION PERFORMANCE IN TERMS OF ACCURACY (A) AND PEARSON'S CORRELATION COEFFICIENT (PC). FOR EACH EXPERIMENT ARE ALSO SHOWN THE NUMBER OF POSITIVE INSTANCES W. R. T. THE DIFFERENT VALUES OF t_y.

Data set	LinReg		COSNet		t_y	Pos
	A	PC	A	PC	C1	
H1 hESC	0.550	0.764	0.848	0.779	0	1479
HepG2	0.582	0.765	0.897	0.820	0	886
HeLa-S3	0.682	0.790	0.817	0.839	0	1685
GM12878	0.635	0.804	0.871	0.826	0	1227
HUVEC	0.592	0.770	0.878	0.816	0	1215
K562	0.614	0.729	0.925	0.814	0	628
	A	PC	A	PC	C2	
H1 hESC	0.859	0.718	0.866	0.791	-0.83	7668
HepG2	0.852	0.735	0.895	0.858	-0.615	5448
Hela-S3	0.895	0.764	0.902	0.864	-0.74	5625
GM12878	0.857	0.780	0.889	0.838	-0.545	4940
HUVEC	0.847	0.769	0.872	0.845	-0.64	5507
K562	0.881	0.663	0.890	0.844	-0.65	5832

Finally we point out that the COSNet algorithm is efficient and scales nicely on large size data, since the time complexity is quasi–linear in the number of considered instances when the input matrix $\boldsymbol{W}$ is sparse [11]. This is fundamental in problems like GEP, where the number of genes is large. In Table IV we also report the time in seconds needed by the algorithm to perform 10–fold cross validations for each separate data set on a Linux system with 64 Gb RAM and Intel Xeon CPU 2.00GHz. As we expected, the method is fast, taking for each dataset around one minute to complete the computation.

TABLE IV
TIME IN TERMS OF SECONDS NEEDED BY COSNET TO PERFORM 10-FOLDS CROSS VALIDATION IN EACH SEPARATE EXPERIMENT.

Data set	Time (sec)
H1 hESC	77.335
HepG2	71.82
Hela-S3	67.563
GM12878	73.267
HUVEC	68.516
K562	64.24

VI. CONCLUSION

In this paper we introduced a method to predict gene expression levels starting from chromatin structure and histone

modifications maps, by considering their correlation with gene expression. We formalized the problem as bipartite ranking on undirected graphs, and employed parametrized Hopfield networks to infer a ranking for the studied genes. We validated the algorithm by predicting the genome–wide gene expression levels in six different human cell lines. We compared our method with a widely used technique to predict the expression level of genes, obtaining considerably higher accuracy and correlation in all the cell lines we considered. Moreover, the algorithm is also efficient and nicely scales on large size data, permitting its straightforward application at the whole–genome level for virtually all the most studied organisms, beside human. Finally, in the benchmark tests we performed each cell line has been processed independently from the others, that is, without considering the existing relationships among the expression profiles of different cell lines. The next logical step is embedding these relationships in the model, and to exploit them for the prediction tasks, with possible significant improvements on the performance of the algorithm.

ACKNOWLEDGMENT

This work has been supported by the Italian National Research Council flagship project "Epigen".

REFERENCES

[1] T. Kouzarides *et al.*, "Chromatin modifications and their function," *Cell*, vol. 128, no. 4, p. 693, 2007.

[2] B. D. Strahl, C. D. Allis *et al.*, "The language of covalent histone modifications," *Nature*, vol. 403, no. 6765, p. 41, 2000.

[3] T. Jenuwein and C. D. Allis, "Translating the histone code," *Science Signalling*, vol. 293, no. 5532, p. 1074, 2001.

[4] A. Barski, S. Cuddapah, K. Cui, T.-Y. Roh, D. E. Schones, Z. Wang, G. Wei, I. Chepelev, K. Zhao *et al.*, "High-resolution profiling of histone methylations in the human genome," *Cell*, vol. 129, no. 4, pp. 823–837, 2007.

[5] L. H. Chadwick, "The nih roadmap epigenomics program data resource," *Epigenomics*, vol. 4, no. 3, pp. 317–324, 2012.

[6] Z. Ouyang, Q. Zhou, and W. H. Wong, "ChIP-Seq of transcription factors predicts absolute and differential gene expression in embryonic stem cells," *Proceedings of the National Academy of Sciences*, vol. 106, no. 51, pp. 21 521–21 526, Dec. 2009. [Online]. Available: http://dx.doi.org/10.1073/pnas.0904863106

[7] R. Karlić, H.-R. Chung, J. Lasserre, K. Vlahoviček, and M. Vingron, "Histone modification levels are predictive for gene expression," *Proceedings of the National Academy of Sciences*, vol. 107, no. 7, pp. 2926–2931, Feb. 2010. [Online]. Available: http://dx.doi.org/10.1073/pnas.0909344107

[8] X. Dong, M. Greven, A. Kundaje, S. Djebali, J. Brown, C. Cheng, T. Gingeras, M. Gerstein, R. Guigo, E. Birney, and Z. Weng, "Modeling gene expression using chromatin features in various cellular contexts," *Genome Biology*, vol. 13, no. 9, pp. R53+, Sep. 2012. [Online]. Available: http://dx.doi.org/10.1186/gb-2012-13-9-r53

[9] N. Japkowicz and S. Stephen, "The Class Imbalance Problem: A Systematic Study," *Intelligent Data Analysis*, vol. 6, no. 5, pp. 429–449, Oct. 2002. [Online]. Available: http://portal.acm.org/citation.cfm?id=1293951.1293954

[10] N. Cristianini and J. Shawe-Taylor, *An introduction to support Vector Machines: and other kernel-based learning methods*. New York, NY, USA: Cambridge University Press, 2000.

[11] M. Frasca, A. Bertoni, M. Re and G. Valentini, "A neural network algorithm for semi-supervised node label learning from unbalanced data," *Neural Networks*, in press. http://dx.doi.org/10.1016/j.neunet.2013.01.021

[12] A. Bertoni, M. Frasca, and G. Valentini, "Cosnet: A cost sensitive neural network for semi-supervised learning in graphs." in *ECML/PKDD (1)*, D. Gunopulos, T. Hofmann, D. Malerba, and M. Vazirgiannis, Eds., vol. 6911, 2011, pp. 219–234.

[13] S. Agarwal, *A Study of the Bipartite Ranking Problem in Machine Learning*. University of Illinois at Urbana-Champaign, 2005. [Online]. Available: http://books.google.it/books?id=1dP6GwAACAAJ

[14] V. M. Aris, M. J. Cody, J. Cheng, J. J. Dermody, P. Soteropoulos, M. Recce, and P. P. Tolias, "Noise filtering and nonparametric analysis of microarray data underscores discriminating markers of oral, prostate, lung, ovarian and breast cancer." *BMC Bioinformatics*, vol. 5, p. 185, 2004. [Online]. Available: http://dblp.uni-trier.de/db/journals/bmcbi/bmcbi5.html#ArisCCDSRT04

[15] E. Marshall, "Getting the Noise Out of Gene Arrays," *Science*, vol. 306, no. 5696, pp. 630–631, Oct. 2004. [Online]. Available: http://dx.doi.org/10.1126/science.306.5696.630

[16] J. Hopfield, "Neural networks and physical systems with emergent collective compatational abilities," *Proc. Natl Acad. Sci. USA*, vol. 79, pp. 2554–2558, 1982.

[17] D. Wang, "Temporal pattern processing," in *The Handbook of Brain Theory and Neural Networks*, 2003, pp. 1163–1167.

[18] H. Liu and Y. Hu, "An application of hopfield neural network in target selection of mergers and acquisitions," *Business Intelligence and Financial Engineering, International Conference on*, vol. 0, pp. 34–37, 2009.

[19] F. Zhang and H. Zhang, "Applications of a neural network to watermarking capacity of digital image," *Neurocomputing*, vol. 67, pp. 345–349, 2005.

[20] A. G. Tsirukis, G. V. Reklaitis, and M. F. Tenorio, "Nonlinear optimization using generalized hopfield networks," *Neural Comput.*, vol. 1, pp. 511–521, 1989.

[21] U. Karaoz *et al.*, "Whole-genome annotation by using evidence integration in functional-linkage networks," *Proc. Natl Acad. Sci. USA*, vol. 101, pp. 2888–2893, 2004.

[22] E. P. Consortium *et al.*, "A user's guide to the encyclopedia of dna elements (encode)," *PLoS Biol*, vol. 9, no. 4, p. e1001046, 2011.

[23] K. R. Rosenbloom, T. R. Dreszer, J. C. Long, V. S. Malladi, C. A. Sloan, B. J. Raney, M. S. Cline, D. Karolchik, G. P. Barber, H. Clawson *et al.*, "Encode whole-genome data in the ucsc genome browser: update 2012," *Nucleic acids research*, vol. 40, no. D1, pp. D912–D917, 2012.