Selection of Negative Examples in Learning Gene Regulatory Networks

Michele Ceccarelli, Luigi Cerulo
Department of Biological and Environmental Studies, University of Sannio
Via Port’Arsa, 11 – 82100 Benevento, Italy
Bioinformatics CORE-Lab – Biogem s.c.ar.l.
Via Campo Reale – 83031 Ariano Irpino (AV), Italy
ceccarelli@unisannio.it, lcerulo@unisannio.it

Abstract

Supervised learning methods have been recently exploited to learn gene regulatory networks from gene expression data. They consist of building a binary classifier from feature vectors composed of expression levels of a set of known regulatory connections, available in public databases (eg. RegulonDB, TRRD, Transfac, IPA), and using such a classifier to predict new unknown connections.

The input to a binary supervised classifier consists normally of positive and negative examples, but usually the only available information are a partial set of gene regulations, i.e. positive examples, and unlabeled data which could include both positive and negative examples. A fundamental challenge is the choice of negative examples from such unlabeled data to make the classifier able to learn from data.

We exploit the known topology of a gene network to select such negative examples and show whether such an assumption benefits the performance of a classifier.

Keywords: gene regulatory networks, machine learning.

1 Introduction

Inferring Biological networks is fundamental to understand the complexity of interdependencies among Biological elements, such as genes, proteins, and metabolites. In silico methods represent a promising direction aiming at extract Biological networks from prior biological knowledge and available genomic and post-genomic data. Such networks are modeled as directed graphs where nodes represent elements of interactions, eg. genes, proteins, metabolites, and edges represent interaction activities between such network components. Different approaches have been introduced in literature [6]: i) information theory models, ii) boolean network models, iii) differential and difference equation models, iv) Bayesian models.

Information theory models correlate two genes by means of a correlation coefficient and a threshold. Two genes are predicted to interact if the correlation coefficient of their expression levels is above a threshold. ARACNE [10] and CLR [4] infer the network structure with a statistical score derived from the mutual information and a set of pruning heuristics.

Boolean network models use a binary variable to represent the state of a gene activity and a directed graph, where edges are represented by boolean functions, to represent the interaction between genes. REVEAL [7] is an algorithm that infer a boolean network model from gene expression data.

Differential and difference equations describe gene expression changes as a function of the expression level of other genes. They are particular suitable to model the dynamic behavior of gene networks. The basic mathematical model of such approaches are a set of ordinary differential equations (ODE) [12].

Bayesian models, or more generally graphical models, make use of Bayes rules and consider gene expressions as random variables. The major advantage is that the Bayesian framework allows for combining different types of data and prior knowledge in the process of gene networks inference [14].

Recently, supervised learning methods have been exploited to learn gene regulatory networks from gene expression data. They consist of building a binary classifier from feature vectors composed of expression levels of a set of known regulatory connections, available in public databases (eg. RegulonDB1, TRRD2, Transfac3, IPA4), and using such a classifier to predict new unknown connections.

---

1http://regulondb.ccg.unam.mx
2http://wwwmgs.bionet.nsc.ru/mgs/gnw
3http://www.gene-regulation.com
4http://www.ingenuity.com
They differ from the above mentioned approaches in that they require as inputs not only gene expression data, but also a list of known regulation relationships, that act as a training set. The necessity to know some regulations is not a serious restriction in many practical applications, as many regulations have already been characterized in model organisms (eg. E.coli). The basic principle is to use the natural inductive reasoning to predict new regulations: if a gene A having expression profile $e(A)$ is known to regulate a gene B with expression profile $e(B)$, then all other couples of genes X and Y, having expression profiles similar to $e(A)$ and $e(B)$ are likely to interact. Expression profiles play the role of feature vectors in the machine learning algorithm, while the output is a binary variable representing whether two genes interact or not.

Similarly, the prediction of protein–protein interaction [1, 2] and metabolic networks [16] make use of a feature vector built upon the sequence representation of protein and metabolites. A large variety of machine learning algorithms have been proposed in literature and are available as working tools [15]. In the context of gene regulatory networks a first attempt has been made with Bayesian Networks, Linear Regression, Decision Trees, and Support Vector Machines (SVM) [5]. Among all the Support Vector Machine algorithm have attracted the attention of the bioinformatics community. SIRENE [11] is the state-of-the-art method for the reconstruction of gene regulatory networks with a Support Vector Machine algorithm. The authors tested SIRENE on a benchmark experiment of Escherichia coli genes composed by a compendium of gene expression data and a set of known regulations.

A binary supervised classifier algorithm learns from inputs which consist normally of positive and negative examples. In gene regulatory networks, although prior known regulatory connections can safely be taken as a partial set of positive training examples, the choice of negative examples is a critical point. In fact, the only available information are a partial set of gene regulations, i.e. positive examples, and unlabeled data which could include both positive and negative examples. A common adopted solution is to consider all, or a random fraction of, unknown examples as negatives and learn a classifier from such a selection of examples [11]. This assumption limits the performance of the classification algorithm as it could learn wrongly potentially positive examples as negatives.

In this paper we introduce an heuristics to select negative examples from unlabeled data in learning gene regulatory networks. We make use of the known topology of the network model to improve the quality of the selected set of negatives examples and consequently improve the performance of a classifier trained with such examples. The paper is organized as follows. Section 2 introduces the heuristics to select negative examples from unlabeled gene interaction networks. Section 3 introduces the research questions and outlines the process followed to answer such questions. Section 4 describes the context where the process is applied. Section 5 reports and discusses the results obtained. Finally, Section 6 concludes the paper and outlines directions for future work.

## 2 Selection of Negative Examples

A gene interaction network can be modeled as a directed graph $G = \langle \text{node}(G), \text{edge}(G) \rangle$, where, $\text{node}(G)$, represents the set of genes, and, $\text{edge}(G)$, represents the set of interactions between genes. Let $P \subseteq G$, a subgraph of $G$, be the known gene–gene interactions. Let $Q = G – P$ be the unknown regulatory links, and $N = \text{Complement}(G)$, the graph containing the edges not contained in $G$. A machine learning scheme can be used to infer the unknown gene regulatory connections, $Q$, by using a training set of known regulatory connections. In machine learning terminology, $\text{edge}(P)$ is the set of known positive examples, $\text{edge}(N)$ is the set of all unknown negative examples, $\text{edge}(Q)$ is the set of unknown positive examples. The set $\text{edge}(N) \cup \text{edge}(Q)$ is known also as the unlabeled set.

A critical issue is that a binary classifier should be trained with both positive and negative examples, to work properly. In the context of gene regulatory networks it is more likely to know positive examples, i.e. two gene interact, rather than the inverse, i.e. two gene do not interact. Public databases of known regulatory interaction usually report only the first information. The problem is then to select from the unlabeled set $\text{edge}(N) \cup \text{edge}(Q)$ of unknown connections, a sub set of negative examples $\text{edge}(S)$ which could be as much as possible composed of negative examples, i.e. $\text{edge}(S) \cap \text{edge}(Q) \simeq \emptyset$.

A common approach in the state of the art methods for the reconstruction of gene regulatory networks is to adopt a random selection heuristic. For example in [11] the authors partition the set $\text{edge}(N) \cup \text{edge}(Q)$ into three random subsets and then use alternatively one of such a sub set as a test set and force the remaining two as the training set of negative examples. The problem is that the classification algorithm can learn wrongly potentially positive examples as negative.

The heuristic proposed in this paper, tries to overcome such a limitation, by exploiting the known topology of the network model in order to limit the presence of positive examples in the selected set of negatives. The heuristic is based on two assumptions:

- The network has no or few cycles. In a regulation path $g_1, g_2, \ldots, g_k$, which means that $g_i$ regulates $g_{i+1}$ for each $i = 1, \ldots, k – 1$, it is likely that the gene $g_k$ does not regulate $g_1$. 


• The network has a tree like structure. In a regulation path $g_1, g_2, \ldots, g_k$, it is likely that the gene $g_1$ does not regulate $g_k$.

Both assumption are not always true but we feel they hold at least in the simulated networks we analyze in the following sections. As a future work we aim to verify such assumptions also in experimental data.

The heuristic proposed is summarized in Figure 1. The selected set of potentially negative examples $\text{edge}(S)$ is defined as follows:

$$S = \text{TC}(P) + \text{Transpose}(\text{TC}(P)) + \text{Transpose}(P)$$

where $\text{TC}(P)$ is the transitive closure of $P$, i.e. the graph composed by the same nodes of $P$ and the set of edges $(g_i, g_j)$ such that there is a non-null path from $g_i$ to $g_j$ in $P$; while, $\text{Transpose}(X)$ is the graph containing the edges of $X$ reversed.

3 Research questions

In this section we introduce the research questions we aim at answering. In order to simplify the notations we use the symbol $X$ to indicate also the set $\text{edge}(X)$.

• **RQ1:** How does the precision/recall of positives, and the precision/recall of negatives of the selected set $S$ vary with the percentage of known positives? In particular, this research question aims to measure the quality of the set $S$ when the percentage of known positives varies. The quality of $S$ is intuitively correlated with the precision/recall of negatives and inversely correlated with the precision/recall of positives. Such quantities are then compared with a random selection heuristic.

• **RQ2:** What are the performances of a classifier trained with the selected set $S$ and one trained with a random set $R$? Specifically, it investigates whether the selection heuristic, introduced in the previous section, could improve the training set of a classifier in terms of accuracy of prediction. For such a purpose an SVM (Support Vector Machine) [8] classifier is used.

To answer RQ1 the following process is executed. From a simulated dataset where all interactions are known, let $P_F$ be a percentage fraction $F$ of randomly selected positive examples assumed to be known, $Q_F$ the set of positive examples assumed to be unknown, and $N$ the set of all negative examples. The percentage fraction $F = \frac{|P_F|}{|P_F \cup Q_F|}$ is assumed to vary in $F \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. Let $S_F$ be the set of potentially negative examples selected from the unlabeled set $Q_F \cup N$, with the heuristic proposed in the previous section. The following quantities, computed within each $S_F$, are used to answer RQ1:

$\text{Precision of Positives}(S_F) = \frac{|S_F \cap N|}{|S_F|}$

$\text{Precision of Negatives}(S_F) = \frac{|S_F \cap Q_F|}{|S_F|}$

$\text{Recall of Positives}(S_F) = \frac{|S_F \cap N|}{|N|}$

$\text{Recall of Negatives}(S_F) = \frac{|S_F \cap Q_F|}{|Q_F|}$

Instead, a random selection of $R_F$ potentially negative examples from the unlabeled set $Q_F \cup N$ produces the following quantities:

$\text{Precision of Positives}(R_F) = \frac{|N|}{|N| + |Q_F|} = \frac{1 - \rho}{1 - \rho F}$

$\text{Precision of Negatives}(R_F) = \frac{|Q_F|}{|Q_F| + |N|} = \frac{\rho(1 - F)}{1 - \rho F}$

$\text{Recall of Positives}(R_F) = \frac{|R_F \cap N|}{|N|} = \frac{|R_F|}{|Q| + |N|}$

$\text{Recall of Negatives}(R_F) = \frac{|R_F \cap Q_F|}{|Q_F|} = \frac{|R_F|}{|Q| + |N|}$

![Figure 1. Selection of negatives examples](image-url)
where \( \rho = \frac{|P_F + Q_F|}{|P_F + Q_F + N_F|} \) is the percentage fraction of positive examples in the network. It can be noticed that such precision/recall quantities depend only from \( \rho \), \( F \) and the size of \( R_F \), therefore we consider a random selection heuristic as reference limit. It is important to specify that a random selection from a set assumes that each element of the set have the same probability to be chosen. A new selection heuristic should have a precision/recall of negatives higher and a precision/recall of positives lower that those exhibited by a random selection heuristic.

To answer RQ2 the following process is executed. A fraction \( F \) of positive examples, \( P_F \), is selected randomly from the set of all positive examples of a simulated dataset, where all interactions are known. In such a way the set of known positive examples can be simulated. Then, an SVM classifier is trained with the known set of positive examples \( P_F \) and with a set of negative examples, alternatively, selected with the proposed heuristic, \( S_F \), and with a random heuristic, \( R_F \). In both cases the classifier is tested with the remaining set of examples, which is \( Q_F \cup N \setminus S_F \) in the first case, and \( Q_F \cup N \setminus R_F \) in the latter case. Such a process is repeated for each \( F \) and, within an \( F \), it is iterated about ten times in order to distribute statistically the random selection of \( P_F \). For each trial the indexes of precision and recall of positives, PR\(_F\) and RC\(_F\) (i.e. Positive Predictive Value and Sensitivity) are computed to evaluate the performance of both classifiers.

## 4 Context

This section reports the context where the proposed heuristics have been evaluated. The context consists of simulated data published in the DREAM3 challenge initiative [9, 13]. It is composed by five datasets, namely \( Ecoli_1 \), \( Ecoli_2 \), \( Yeast_1 \), \( Yeast_2 \), and \( Yeast_3 \), each of which is composed by 10, 50, and 100 number of genes. The datasets contain the steady state levels for the wild-type and the heterozygous knock-down strains for each gene. This means that for a network of \( N \) genes there are \( N + 1 \) experiments (wild-type and knock-down of every gene) leading to a feature vector composed of \( 2(N + 1) \) attributes. Table 1 reports the number of positives and negatives contained in the goal standard of every DREAM3 dataset.

The data corresponds to noisy measurements mRNA levels which have been normalized such that the maximum value in a given dataset is one. Network topologies were obtained by extracting sub-networks from the gene-to-gene interaction network of \( E.coli \) and \( Saccharomyces cerevisiae \) (Yeast). Auto-regulatory interactions were removed, i.e. no self-interactions are considered in the networks. As reported in the DREAM3 documentation, great care was taken to generate both network structure and dynamics that are biologically plausible.

### Table 1. DREAM3 in-silico networks: number of Positives (Negatives)

<table>
<thead>
<tr>
<th>Dataset</th>
<th>10</th>
<th>50</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Ecoli_1)</td>
<td>11 (79)</td>
<td>62 (2388)</td>
<td>125 (9775)</td>
</tr>
<tr>
<td>(Ecoli_2)</td>
<td>15 (75)</td>
<td>82 (2368)</td>
<td>119 (9781)</td>
</tr>
<tr>
<td>(Yeast_1)</td>
<td>10 (80)</td>
<td>77 (2373)</td>
<td>166 (9734)</td>
</tr>
<tr>
<td>(Yeast_2)</td>
<td>25 (65)</td>
<td>160 (2290)</td>
<td>389 (9511)</td>
</tr>
<tr>
<td>(Yeast_3)</td>
<td>22 (68)</td>
<td>173 (2277)</td>
<td>551 (9349)</td>
</tr>
</tbody>
</table>

## 5 Results and Discussion

This section reports results of the research questions introduced in Section 3. Due to space limitations, we only report the results of a single dataset (\( Yeast_2 \)); results obtained with other datasets are very similar and are included in a detailed technical report\(^5\). To answer both research questions, RQ1 and RQ2, we applied the process described in Section 3 to all the DREAM3 datasets detailed in Section 4.

Figures 2 and 3 show the results answering RQ1 in DREAM3 \( Yeast_2 \) dataset. In particular they show the precision/recall of positives and the precision/recall of negatives at different percentage, \( F \times 100 \in \{10, 30, 50, 70, 90\} \), of known positive connections. Figure 2 shows such quantities within the set built with the proposed heuristic, while Figure 3 shows such quantities within the set built by selecting randomly 2/3 of examples from unlabeled data, an approach adopted also in [11].

In the selection made with the proposed heuristic, the precision of positives has a trend that increases initially, reaching a maximum at about \( F = 30\% \), and then decreases. Consequently, the precision of negatives decreases firstly, reaching a minimum at about \( F = 30\% \) and then increases. Recall of negatives increases reaching the maximum at \( F = 90\% \), while recall of positives increases firstly, reaching a maximum at \( F = 50\% \) and the decreases.

In the random selection of 2/3 of unlabeled data, the precision of positives decreases with \( F \), and consequently the precision of negatives increases with \( F \). Recall of both negatives and positives is constant at 2/3, i.e. the fraction of unlabeled examples randomly selected.

Intuitively, a good selection of negatives should have a low precision/recall of positives, and a high precision/recall of negatives. In terms of precision of negatives/positives the set selected with the proposed heuristic exhibits a better tradeoff when the percentage of known positives \( F \) and the number of genes are low. For example, in the dataset composed of 10 genes and at \( F = 10\% \), the proposed heuristic exhibits a precision of negatives equal to 1 and a precision\(^5\) https://www.scoda.unisannio.it/tr/TR-BIOCORE-09-01.pdf
of positives equal to 0 (Figure 2–a). This means that the set
selected with the proposed heuristic is entirely composed
of true negative examples. On the other hand, in the same
dataset, the random selection heuristic exhibits a precision
of negatives equal to 0.75 and a precision of positives equal
to 0.25 (Figure 3–a). The difference between the random
selection and the proposed heuristic becomes less evident
when both the number of genes and \( F \) increase.

In terms of recall of negatives/positives the set selected
with the proposed heuristic exhibits a better tradeoff when
the percentage of known positives \( F \) is high. Instead, the
random selection heuristic exhibits a constant values of re-
call of negatives/positives.

Figure 4 shows the results answering \( RQ2 \) in the
DREAM3 Yeast2 dataset. To show the performance of an
SVM classifier we computed the F-Measure, which mea-
sures the tradeoff between the precision and recall obtained
in a trial. A trial is performed on a random selection of a
percentage fraction \( F \) of positive examples, which are those
assumed to be known. Such trial is repeated ten times and
Boxplots in Figure show the lower quartile, the median,
and the upper quartile of the F-Measures obtained with such
trials.

It is important to specify that the focus of \( RQ2 \) is to show
the relative difference of two classifiers, one trained with
a set populated with the proposed heuristic, and the other
trained with a random set. Evaluating the absolute perfor-
mance of each classifier is out of the scope as it depends
strongly on the parameters settings of the SVM classifier.
To this aim we set the parameters of the LIBSVM tool [3],
we used to build an SVM classifier, to default values.

A paired t-test was performed to show if there is a sta-
tistically significant difference in the mean performance of
two classifiers expressed in term of F-Measures. It is in-
teresting to note that a paired t-test performed on \( F = 10\% \)
and \( F = 30\% \) shows a, statistically significant (p-
value < 0.01), difference between the two F-measures
in each DREAM3 dataset. Boxplots show a clear differ-
ence in favor to the classifier trained with the set popu-
lated with the proposed heuristic. In all other percentages
\( F = 50\%, 70\%, 90\% \) no statistically significant conclusions
can be made in all datasets, although, in almost all cases, a
better performance is exhibited by the classifier trained with
a random set when \( F = 90\% \). These results confirm what
has been reported previously for \( RQ1 \) where the set obtained
with the presented heuristic exhibit a better tradeoff at low
levels of \( F \).
6 Conclusions

This paper introduced an heuristic to select negative examples from unlabeled data to learn gene regulatory networks from expression data. We found that the proposed selection heuristic is able to improve the quality of a training set, with respect to a random selection, when the percentage of known regulatory connection is low (up to 30% of the total connections). Such an increment of quality can be noticed in terms of precision of positives (false negatives) in the training set and in terms of precision of negatives. Furthermore, an SVM classifier has been trained with a set populated with the presented heuristic and with a random selection from unlabeled data. Also in this case, in each considered dataset, the first classifier outperforms, in terms of precision and recall, the second when when the percentage of known regulatory connection is low.

We are aware that results presented in this paper are partial and no general conclusions can be drawn. Results need to be empirical validated with experimental data and in different contexts. We believe that the issues presented in this paper could have an important role in the application of machine learning algorithms in gene regulatory networks discovery.

References


Figure 4. Boxplot plots of the F-Measure ($F\text{-Measure} = \frac{2PR_{+}RC_{+}}{PR_{+} + RC_{+}}$) of a SVM classifier trained with the proposed heuristic set and with a random set.