Modelling Gene Expression by Integrating GRNs and HMs using GCNs

Shalin Patel[1,2], TBD[1,2,3], and Ritambhara Singh[2,3]

[1]*Division of Applied Mathematics, Brown University*
[2]*Center for Computational Molecular Biology, Brown University*
[3]*Department of Computer Science, Brown University*

Abstract

1 Introduction

2 Related Work

3 Method

3.1 Formulation for Task

In this paper, we use the same inputs and outputs as Attentive and DeepChrome while also adding a gene expression matrix for each cell line. Using the same formulation as Cheng *et al.* the task is formulated as measuring the gene expression as either up (1) or down (0) regulated. First, per cell line, a GRN is precomputed which utilizes a matrix E of size $S \times G$ where S denotes the number of samples in the expression matrix while G represents the number of genes that were recorded.

Hence, for a sample gene, two pieces of information are fed. The first is H which is a graph describing gene-gene interactions for a particular cell line. In the case of this paper H was an adjacency list representation of the graph. Second, per gene, a matrix X of size $M \times T$ was utilized where M denotes the number of histone marks utilized while T is the total number of bin positions taken into account around the TSS site of a gene.

Overall, for the training data of the GCN, we utilized H and a $N \times M \times T$ sized matrix where N is the number of gene samples. The output, accordingly, is a N-sized vector with either 0 or 1.

3.2 Construction of GRNs

The gene regulatory networks for this paper are built using the standard method of random forests. Utilizing the grnboost2 algorithm from the arboreto package on `pypi.org`, a regression task was defined for each gene in a cell line.

Let E_i denote the row vector containing all samples for gene i in the expression matrix. Then, specifically, for gene i, a random forest model R_i was defined where, $L(R_i(E_{1:G\setminus i}), E_i)$ is minimized with L denoting the mean square error. Once this task is completed, denote the set $\mathfrak{N}_i = \{imp(R_i, j) \mid j \in \{1 : G\} \setminus i\}$. Here, $imp(R_i, j)$ refers to the feature importance of j in random forest model R_i. Then the final graph H is constructed with the neighbor list of a node i being $\mathfrak{I}_i = \{j \mid imp(R_i, j) > \bar{\mathfrak{N}}_i + s(\mathfrak{N}_i)\}$ with $s(\mathfrak{N}_i)$ denoting the sample standard deviation. Hence, $H = \{\mathfrak{I}_i\}, \forall i$.

4 Experimental Setup

5 Results

6 Discussion

Acknowledgments .

Funding

References