A Detailed Overview of the GENIE3 Algorithm

1 Introduction

The GENIE3 algorithm is a method for inferring gene regulatory networks from gene expression data. The algorithm utilizes Random Forests, an ensemble learning technique based on decision trees, to rank genes based on their regulatory potential.

2 Algorithm Steps

2.1 Input Data

The input data for the GENIE3 algorithm consists of a gene expression matrix X, with G genes and S samples:

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1S} \\ x_{21} & x_{22} & \cdots & x_{2S} \\ \vdots & \vdots & \ddots & \vdots \\ x_{G1} & x_{G2} & \cdots & x_{GS} \end{bmatrix} \tag{1}$$

Each element x_{ij} represents the expression level of gene i in sample j. The goal of the algorithm is to identify potential regulatory relationships between genes based on their expression patterns.

2.2 Random Forests

For each target gene g_{target}, the GENIE3 algorithm trains a Random Forest to predict the expression level of g_{target} based on the expression levels of all other genes. The Random Forest is an ensemble of decision trees, each of which is trained on a bootstrap sample of the data and a random subset of the predictor genes.

2.3 Impurity Decrease and Feature Importance in a Single Tree

The importance of a predictor gene in a single decision tree is determined by the impurity decrease it brings when used as a split. Figure 1 demonstrates an

example of such a split, where data points from the parent node are divided into two parts based on the threshold of the predictor variable (feature), and the choice of the predictor variable. Each split in the tree formation process is chosen to maximize the impurity decrease.

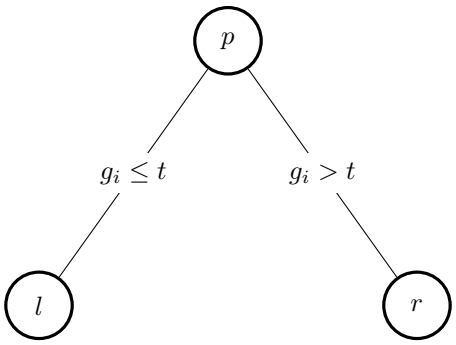

Figure 1: Parent node (P) splits into two child nodes (L and R). g_j, t are feature (particular gene) and treshold.

In GENIE3, the Impurity Decrease (**ID**) for one split is computed as:

$$\text{ID} = N_p \cdot \text{Var}(y_p) - N_l \cdot \text{Var}(y_l) - N_r \cdot \text{Var}(y_r) \tag{2}$$

Here, N_p, N_l, and N_r represent the number of samples in the parent node, left child node, and right child node, respectively, and $\text{Var}(y_p)$, $\text{Var}(y_l)$, and $\text{Var}(y_r)$ denote the variances of the target gene expression levels in the corresponding nodes.

Note that this is slightly different from common impurity decrease equation used in Random Forests.

The feature importance (**I**) for gene g_i in a single tree is **the sum of impurity decreases associated with all splits in the tree** that use gene g_i as the splitting feature:

$$\text{I}_t(g_i) = \sum_{s \in S_{g_i}} \text{ID}_s \tag{3}$$

Here, S_{g_i} represents the set of all splits in the tree k that **use gene** g_i as the splitting feature, and $\text{Impurity Decrease}_s$ denotes the impurity decrease for split s.

2.4 Feature Importance

After the Random Forest is trained, the importance scores of the predictor genes are averaged **across all trees** in the ensemble:

$$w_{i,target} = \frac{1}{T} \sum_{t=1}^{T} \text{I}_t(g_i) \tag{4}$$

Here, $w_{i,target}$ is feature importance for the feature g_i in prediction of target expression of target gene g_{target}. T represents the number of trees in the Random Forest, and $\text{Importance}_t(g_i)$ denotes the importance score of gene g_i in tree t.

2.5 Weighted Adjacency Matrix

Once the importance scores are averaged, a weighted adjacency matrix W is constructed to represent the inferred gene regulatory network:

$$W = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1G} \\ w_{21} & w_{22} & \cdots & w_{2G} \\ \vdots & \vdots & \ddots & \vdots \\ w_{G1} & w_{G2} & \cdots & w_{GG} \end{bmatrix} \tag{5}$$

Each element w_{ij} represents the regulatory potential of gene i on gene j, which is approximated by the averaged importance score of gene i in the Random Forest trained to predict the expression of gene j.

2.6 Discussion of the conservation of mean values across all adjacency matrices

In the article [1] there is the following equation (equation 4, page 4):

$$\sum_{i \neq j} w_{i,j} \approx N\text{Var}(S) \tag{6}$$

S is a learning sample from which **a tree** was built. $\text{Var}(S)$ is a variance of target gene, estimated in the corresponding learning sample. Let's understand this equation and try to derive the equation step by step. At first, here is the **citation from the article** above this equation: "Each tree-based model yields a separate ranking of the genes as potential regulators of a target gene in the form of weights $w_{i,j}$ computed as sums of total variance reductions. The sum of the importances of all variables for a tree is equal to the total variance of the output variable explained by the tree, which in the case of unpruned trees (as they are in the case of Random Forests and Extra-Trees ensembles) is usually very close to the initial total variance of the output."

Note that in the citation we clearly see that this equation is written for **one tree** in the ensemble that predicts the expression of a specific gene.

Note that in this text we denote S_{g_i} as the set of all splits in the tree k that **use gene** g_i as the splitting feature.

For a tree in the ensemble we can sum the Impurity Decreases (equation 2) across all splits (some features can be chosen for splits more than once):

$$\sum_{s \in S_{all}} \mathrm{ID}_s = \sum_{s \in S_{all}} N_p \cdot \mathrm{Var}(y_p) - N_l \cdot \mathrm{Var}(y_l) - N_r \cdot \mathrm{Var}(y_r) \tag{7}$$

Here S_{all} is the set of all splits in **one tree**. It can be seen that for the sum over the entire tree, the intermediate terms will cancel out, and we will obtain:

$$\sum_{s \in S_{all}} \mathrm{ID}_s = N_p \cdot \mathrm{Var}(y_p) - \sum_{c} N_c \cdot \mathrm{Var}(y_c) \tag{8}$$

Where c belongs to the set of terminal nodes of the tree.

Now let's move on to the sum of feature weights in the ensemble of trees. Let's denote tree feature set for a target gene g_j as G_j.

$$\sum_{g_i \in G_j} w_{i,j} = \sum_{g_i \in G_j} \frac{1}{T} \sum_{t=1}^{T} \mathrm{I}_t(g_i) = \sum_{g_i \in G_j} \frac{1}{T} \sum_{t=1}^{T} \sum_{s \in S_{g_i}} \mathrm{ID}_s \tag{9}$$

Let's swap the summation signs:

$$\sum_{g_i \in G_j} w_{i,j} = \frac{1}{T} \sum_{t=1}^{T} \sum_{s \in S_{all}} \mathrm{ID}_s \tag{10}$$

Using the equation 8, we get:

$$\sum_{g_i \in G_j} w_{i,j} = \frac{1}{T} \sum_{t=1}^{T} (N_p \cdot \mathrm{Var}(y_p) - \sum_{c} N_c \cdot \mathrm{Var}(y_c)) \tag{11}$$

$$\sum_{g_i \in G_j} w_{i,j} = N_p \cdot \mathrm{Var}(y_p) - \frac{1}{T} \sum_{t=1}^{T} (\sum_{c} N_c \cdot \mathrm{Var}(y_c)) \tag{12}$$

If we neglect the average sum over the terminal nodes, we will obtain the exact formula from the article.

We have seen that the equation in the article is written for the sum of feature weights for a particular target gene, rather than for the sum of all weights in the matrix.

Now let's write an expression for the average edge weight in the gene network. To do this, the equation 12 needs to be summed over the number of genes in the dataset:

$$\sum_{i,j} w_{i,j} \approx S \sum_{g_j} \mathrm{Var}(y_j) \tag{13}$$

Where $\mathrm{Var}(y_j)$ denotes the variance of the target gene g_j expression levels, and the sum is done across all genes in the network.

The article mentions that the variations of all genes in the expression data were equalized before the calculation of adjacency matrix. This was done to avoid positive bias for regulatory links towards more highly variable genes.

2.7 Gene Regulatory Network

Finally, the weighted adjacency matrix W is used to represent the inferred gene regulatory network, where nodes correspond to genes, and edges are weighted by the regulatory potential of one gene on another.

3 Biological Interpretation

The GENIE3 algorithm provides a data-driven approach to uncover potential regulatory relationships between genes based on their expression patterns across different samples. The resulting gene regulatory network can be used to generate hypotheses about the biological mechanisms underlying gene regulation, such as transcription factors controlling the expression of target genes. By integrating additional data sources or performing further experimental validation, researchers can use the GENIE3 algorithm as a starting point to better understand the complex regulatory processes in living organisms.

References

[1] V.A. Huynh-Thu, A. Irrthum, L. Wehenkel and P. Geurts, *Inferring regulatory networks from expression data using tree-based methods*, *PLOS ONE* **5** (2010) 1.