Detecting Recent Frequency-Dependent Balancing Selection using Random Forests and Bayesian Networks

Jackson Shen Vidhu Naik Robert J Williamson

Rose-Hulman Institute of Technology

Background

Frequency-dependent balancing selection is a form of natural selection that creates and maintains diversity within a population [3]. Balancing selection has been used to explain the certain patterns in genes, such as the variability patterns in the MHC gene, which codes for essential proteins required for vertebrate immune systems. By identifying loci under similar selection, we might be able to identify important alleles that could help patients recovering from other diseases.

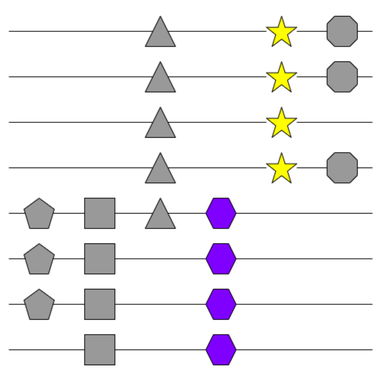
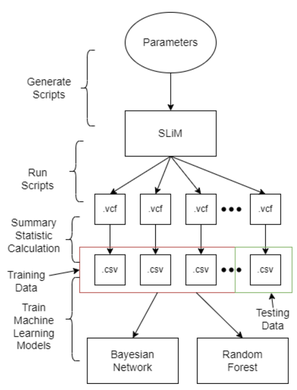


Figure 1. Each line represents the genome of an individual, and symbols signify mutations at that location within the genome. The highlighted mutations are affected by balancing selection, each of which have certain linked alleles that comprise the main haplotypes in the population.

Detection of balancing selection in populations has led to insights on the applications of domestication on crops and livestock. Loci under balancing selection are candidates for those controlling various agronomic traits [1] including total water requirement of crop, plant density, and fertilizer responsiveness.

Data Generation

Data is generated for training using SLiM [4] combined with a custom data pipeline. Eidos scripts are generated with the following variable parameters: mutation rate, recombination rate, selection coefficient, population size, and the balancing selection locus. Training sets consisting of multiple Eidos scripts will be generated with each of its parameters given a uniform random value within given limits. Running these Eidos scripts generate VCF files, which are used to calculate the following summary statistics for constant variant site windows: the π, θ_w, D, and H statistics.



Random Forests

A random forest is a classifier consisting of a collection of decision trees, which operate on a majority vote for the classification of an input [2]. This model is trained by growing multiple decision trees on random subsets of the overall training data. One of the key features of Random Forests is its resistance to overfitting. This allows for us to generate training data sets with a large number of parameters without needing to worry about limiting the data set generation process.

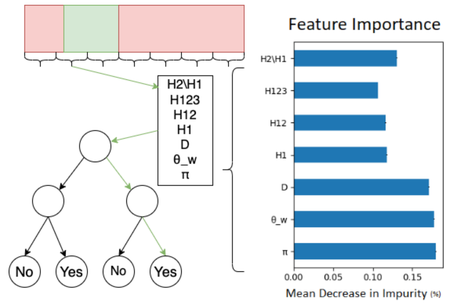


Figure 2. The relative weighting given to each summary statistic within our random forest model

We use SciKit-Learn's Random Forest implementation [5] to train the Random Forest model. The training data is the previously calculated summary statistics from sliding sections of the simulated populations. The Random Forest model will then classify regions of these genomes into whether it is under the effect of balancing selection or not.

Discussion

Metric	RF	BN
Accuracy	0.711098	0.756293
Precision	0.262108	0.235394
Recall	0.109134	0.004745
F1	0.154103	0.009302

Table 1. Metrics for a Random Forest and a Bayesian Network model trained on recent frequency-dependent balancing data simulated on SLiM3

Low recall and precision indicate that the models had low predictive power on areas with balancing selection. However, the models were strong in predicting areas without pressure from balancing selection. Overall, both of the models were very conservative in predicting balancing selection, with the Random Forest being slightly more likely to detect balancing selection than the Bayesian Network.

Bayesian Networks

Bayesian Networks are machine learning models that use the power of conditional probabilities, represented using a graph structure where each node is a probability distribution and each edge is a conditional dependency between two nodes.

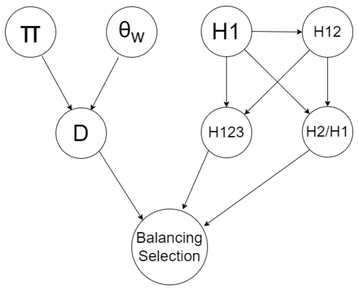


Figure 3. The structure of the Bayesian Network

Each vertex in our Bayesian Network represents a probability distribution associated with a test statistic calculated from a simulated genome. Each edge represents a conditional dependency between two test statistics. Conditional probability tables (CPTs) are calculated for each vertex using the training data, which produces the model's expectation of whether balancing selection has been detected.

Future Work

- Detect other types of balancing selection
- Create a model to model more diverse population histories
- Increase flexibility of metric generation for populations
- Adapt the data pipeline for a non-technical audience

References

[1] J. Arro, J.-W. Park, C. M. Wai, R. VanBuren, Y.-B. Pan, C. Nagai, J. da Silva, and R. Ming. Balancing selection contributed to domestication of autopolyploid sugarcane (saccharum officinarum l.). *Euphytica*, 209(2):477–493, 2016.

[2] L. Breiman. Random forests. 45:5–32, 2001.

[3] A. Fijarczyk and W. Babik. Detecting balancing selection in genomes: Limits and prospects. *Molecular Ecology*, 24, 2015.

[4] B. C. Haller and P. W. Messer. Slim 3: Forward genetic simulations beyond the wright-fisher model. *Molecular Biology and Evolution*, 36, 2019.

[5] 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. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.