Genotype Analysis of Complex Genetic Models using Machine Learning SHAP

Kevin Delao, Maya Singh

June 2020

1 Introduction

Determining the relationship between genotype variation and phenotypic traits is a challenging task to accomplish. Part of the challenge has to do with varying biological factors that make it difficult to determine which genetic variants led to a specific trait. Additionally, gathering enough data to understand how rare genetic variants and common variants affect traits is difficult to do even in genome wide association studies (GWAS).

Machine learning models have been utilized to try to predict traits from genotype data, but with increasingly complex models it becomes difficult to determine which loci had the largest contribution towards a predicted trait. In this research project we attempt to solve the problem of causal loci identification in complex machine learning models by determining the viability of SHAP at identifying causal loci. We created increasingly complex simulations of genotype and phenotype data that are passed into machine learning models for supervised learning.

By using simulated data, we had scenarios where we knew the correct loci beforehand which let us determine in what scenarios was SHAP viable to determine causal loci and in which scenarios did SHAP fail. We performed multiple simulation to find the accuracy of SHAP at determining causal loci where we varied biological factors such as environmental noise, genetic effects, and loci interactions. The results we obtained from our research indicates that regardless of the complexity of our models and varying biological factors SHAP was able to identify casual loci for Linear Regression, Random Forest, and Neural Network model with high accuracy for one loci.When there were two loci Random Forest and Neural Nets both struggled to get both causal loci correct. We also found that Shapley values alone were not able detect interactions occurring between loci, but SHAP interaction values could be used to determine if interactions took place. From our results we can conclude that SHAP offers a promising method to identify casual loci and can be used to determine if interactions occurred between loci.

2 Methods

2.1 Data Simulation

2.1.1 Genotype

The genotype data is simulated using the number of samples (n) and the length of the chromosome (m) to generate a genotype matrix n x m (G). The frequency of the allele at each loci (j) is obtained from a continuous uniform distribution. The genotype for each individual (i) at a specific loci position (j) is determined using f_j

$$f_j \sim U(0, 1) \tag{1}$$

$$G_{ij} \sim B(2, f_j) \tag{2}$$

The matrix is generated by initializing an $n \times m$ zero matrix and the allele frequencies for all the loci using (1). For each sample position at each loci in the matrix the frequency from (1) is used as the probability for the binomial distribution (2) to determine genotype at that position.

2.2 Heritability

The variance of the environmental effect and the variance of the genetic effect are used to generate the phenotype of the model, while following the constraint in (2).

$$\sigma_g^2 + \sigma_e^2 = 1 \tag{3}$$

Using (3) the heritability of the the target gene is determined.

$$h^2 = \frac{\sigma_g^2}{\sigma_g^2 + \sigma_e^2} \tag{4}$$

2.3 Phenotype

2.3.1 One Loci

$$\boldsymbol{\beta} \sim N(0, \sigma_g^2 \boldsymbol{I}) \tag{5}$$

$$\epsilon_j \sim N(0, \sigma_e^2 \boldsymbol{I}) \tag{6}$$

The phenotype expression for one loci is determined by randomly selecting a causal SNP from the genotype G. The the genetic effect for the causal SNP is a random variable from (5) and the genetic effect for all other SNPs will be 0. The environmental effect on the phenotype per sample will be a random variable from (6).

$$Y_i = X_i \boldsymbol{\beta} + \epsilon_i \tag{7}$$

$$\epsilon_j \sim N(0, \sigma_e^2 \boldsymbol{I}) \tag{8}$$

2.3.2 Two Loci

$$\boldsymbol{\beta} \sim N(0, \frac{\sigma_g^2}{2}\boldsymbol{I}) \tag{9}$$

$$\epsilon_j \sim N(0, \sigma_e^2\boldsymbol{I}) \tag{10}$$

The phenotype expression for two loci is determined by randomly selecting 2 causal SNPs from the genotype G. The the genetic effect for the causal SNP will be determined from (7) and the genetic effect for all other SNPs will be 0. The environmental effect on the phenotype per sample will be determined from (9).

$$Y_i = X_{i_1}\boldsymbol{\beta_1} + X_{i_2}\boldsymbol{\beta_2} + \epsilon_i \tag{11}$$

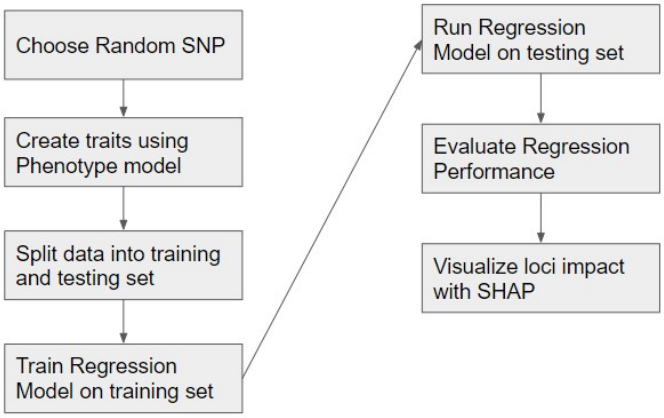
2.3.3 Two Loci with interaction

The phenotype expression for two loci with interaction is the same as for two loci, with an added term for the phenotype expression when both loci are present together.

$$Y_j = X_{i_1 j}\beta_{i_1} + X_{i_2 j}\beta_{i_2} + X_{i_1 j}X_{i_2 j}\beta_{i_{12}} + \epsilon_j \tag{12}$$

2.4 Machine Learning Models

2.4.1 Supervised Learning Steps



2.4.2 Linear Regression

The first machine learning model that we used in conjunction with SHAP was Linear Regression. Linear Regression is a linear model that tries to fit a linear equation to the data. We used Linear Regression first for our one loci simulation, because our one loci simulation was a linear problem. Another factor in utilizing Linear Regression was due to the additive nature of how features in the model

are used to make a prediction. With Linear Regression it is possible to determine which features contributed the most towards to a prediction by simply looking at the regression coefficient of a feature and the feature value itself as seen (12):

$$h(\theta) = \theta_0 X_0 + \theta_1 X_1 + ...\theta_n X_n \tag{13}$$

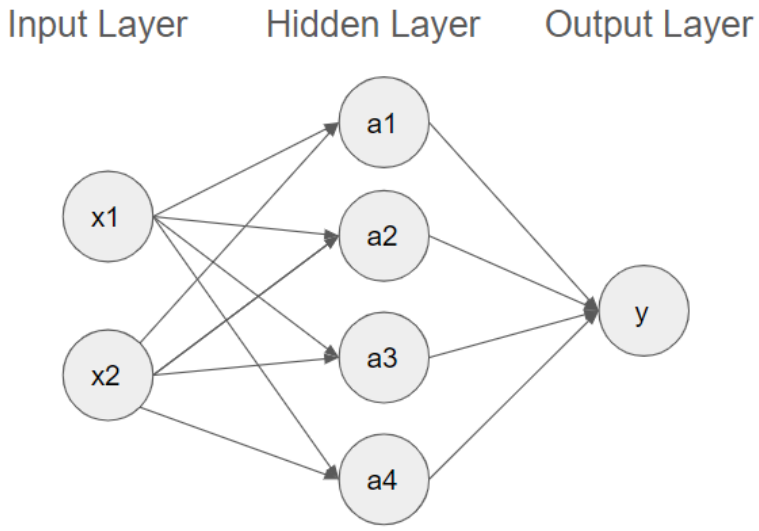
By manually extracting the loci that contributed the most towards a prediction, we were able to compare how accurate SHAP was at determining causal loci. Linear Regression served as our ground truth model due to the fact that it offers the capability to determine the feature impacting a prediction which is not possible to do with more complex models.

2.4.3 Random Forest Regressor

We utilized a Random Forest model fitted for regression to determine how well non-linear regression models can perform on our simulations. Random Forest algorithms are popular to use due to their performance and speed in making a prediction. The parameters chosen for the Random Forest Regressor included 50 decision trees and 10 splits. Due to time constraints we did not have time to determine what parameters would work best for our project so it would be interesting to determine in the future what the best parameters would be for our Random Forest model.
Another factor in deciding to use a Random Forest model has to due with Random Forest models being compatible with TreeSHAP and SHAP Interaction Values, both which will be discussed in following sections.

2.4.4 Neural Network

The last machine learning model that was used in this research project was a neural network. Normally neural networks are black boxes due to their complexity, which makes it difficult to determine which of features produced the prediction made by the neural network. Even more challenging are the scenarios where a neural network creates its own features in order to make a prediction and in such scenarios determining feature importance can be a daunting task. SHAP fortunately can extract feature importance from a neural network regardless of the complexity of the model. We designed simple network with one hidden layer in order to determine how accurate SHAP would be in the simplest network possible. Our network had 32 nodes in the hidden layer and the input layer and hidden layer were dense meaning they were fully connected to the layer that followed as seen in the figure below. The results for how accurate SHAP was when used in conjunction with our neural network for one and two loci will be shown in the results section.



For next steps it would be beneficial to determine how well SHAP would perform on a network with more layers and varying nodes in the hidden layers. It would also be beneficial to determine the accuracy of the neural network for larger sample sizes as due to time constraints and the slow computation time of SHAP we were restricted to using only moderately large sample sizes.

2.5 SHAP (SHapley Additive exPlanations)

2.5.1 The Shapley Value

The SHAP software allows us to determine which features are the most important for the prediction made by a machine learning model. Shap determines feature importance by assigning Shapley values to each feature and features with large Shapley values represent features that have the largest contribution to a prediction. In order assign Shapley values to features SHAP uses (13) to calculate a Shapley for each feature. Equation (13) calculates a Shapley for a feature i by summing through all possible coalitions S which represents a group of features from our total group of features N. All coalitions S are groupings that do no include feature i. In each summation the marginal calculation of the difference between a prediction including i and not including i will be weighted by all possible orderings in S multiplied by the feature orderings that are still possible to make. After summing over all possible coalitions, the sumation will be divided by the total possible feature orderings possible. The end result is a Shapley value for feature i which represents feature i's contribution to the prediction made by the model.

$$\phi_i(p) = \frac{1}{n!} \sum_{S \subseteq N/i} |S|!(n - |S| - 1)!(p(S \cup i) - p(S)) \tag{14}$$

2.5.2 Kernel SHAP

Kernel SHAP is a model agnostic method to determine SHAP values. It is mainly used in Linear Regression, Logistic Regression, and Neural Network models to extract SHAP values for features. Kernal SHAP is a slow method to determine SHAP values as it uses the standard method to determine Shapley value which a have time complexity that is 2^n.In order to make calculations feasible, Kernal SHAP allows data to be sampled in order to speed up calculations. Even with sampling Kernal SHAP had a slow computation which resulted in our simulations that used Linear Regression taking hours took complete. The slow computation time resulted in our sample size not being too large, because if we were to create larger genotype matrices then our computation time would run for days.

2.5.3 Tree SHAP

Tree SHAP is fast method to extract feature values from tree based machine learning models and also ensemble learners. The time complexity for Tree SHAP is much faster than Kernal SHAP, taking a time of n^2 to calculate SHAP values. The faster computation of SHAP values allowed us to create larger genotype matrices that the Random Forest Regressor could train on.

2.6 SHAP Interactions

2.6.1 SHAP Interaction Values

In our project one of the factors we were interested in discovering was whether SHAP could detect interactions between loci. Our initial research into Shapley values determined that Shapley values alone cannot detect interaction, because Shapley values assume features are independent. We looked into various methods that could identify interactions between features and we came across SHAP interaction values. SHAP interaction values are a method to detect interactions between pairs of features using (14). Using (14) SHAP calculates a SHAP interaction value for a pair of features i,j for every feature value of i and j where in each calculation SHAP will make prediction using i and j and subtract from that a prediction made with only i. The result will be a SHAP value that represents how feature j impacts feature i when they are together and apart. The resulting SHAP interaction value can be negative, positive, or zero. Where a SHAP interaction of zero represents no interaction, a positive value means that when feature i and j are together they have a higher SHAP value, a negative value means that that when feature i and j are together they have lower SHAP value.

$$\phi_{i_j}(p) = \frac{1}{2(M-1)!} \sum_{S \subseteq N/i,j} |S|!(M-|S|-2)!\delta_{i_j}(S) \tag{15}$$

$$\delta_{i_j}(S) = f_x(S \cup i, j) - f_x(S \cup i) - f_x(S \cup j) + f_x(S) \tag{16}$$

Using SHAP interaction values we were able to determine if interactions took place for our simulated data. In our simulations we controlled if interactions occurred or not, but we initially had no method to visually determine the interactions. SHAP interaction values allowed for a method to plot the interactions occurring between pairs of loci and more specifically the interactions occurring between two causal SNP's. With SHAP Interaction value we could determine how different interaction scenarios that we simulated affected the prediction made by a machine learning algorithm.
The results for the SHAP interaction values are shown in the results section.

2.7 Accuracy Testing

The accuracy of each trail of the model was determined using the SHAP values. The highest SHAP value provides the feature with the highest effect on the model. The accuracy of the model is tested by comparing the feature with the highest SHAP value to the causal loci. A match indicates accuracy, a mismatch indicates a mistake.
The accuracy is measured for two overarching cases, Value of Genetic Effect and Variance of genetic effect. For value of genetic effect the genetic effect is set at a specific β value, this represents the best case scenario with the maximum genetic effect. For variance of genetic effect the β value is obtained from (5), this represents the real world scenario where the β value is normally distributed.

2.7.1 One Causal Loci

Random Forest Regression, Linear Regression and Neural Network machine learning models were run on phenotype data affected by one causal loci. The accuracy for each test was measured by providing 1 for a correct match and 0 for an incorrect match. The accuracy test for 100 trials is run for different σ_e^2 and σ_g^2 combinations.

$$\mathbb{1}Accuracy \begin{cases} 1 \text{ Feature with highest SHAP} = \text{Causal Loci} \\ 0 \text{ Feature with highest SHAP} \neq \text{ Causal Loci} \end{cases} \tag{17}$$

2.7.2 Two Causal Loci

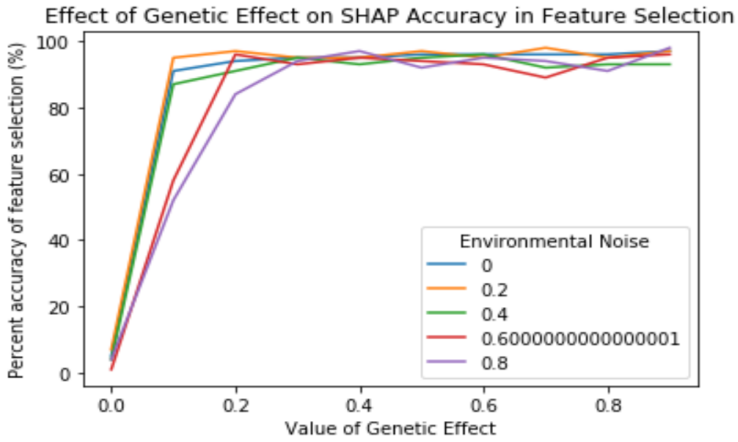
For our simulations for two causal loci with and without interactions we determined the accuracy of SHAP by measuring the times SHAP got both loci wrong, both loci right, and only one one loci right when using Random Forest Regression, Linear Regression and Neural Network models. When SHAP got both causal loci right it counted right it counted as 1, half right counted as .5, and both wrong counted as 0. We then added up the different counts across multiple simulations and then divided by the total number of simulation to give us the accuracy for a specific variance of genetic effect.
An alternative to view the accuracy of SHAP was to make a bar plot showing the percentage of the times SHAP got both causal loci wrong, one right, and

both wrong for multiple interaction scenarios. Showing the percentage of how SHAP performed gave us a better understanding on how well our models are constructed, whether we are providing enough data, and the limations of SHAP determining causal loci in more complex scenarios.

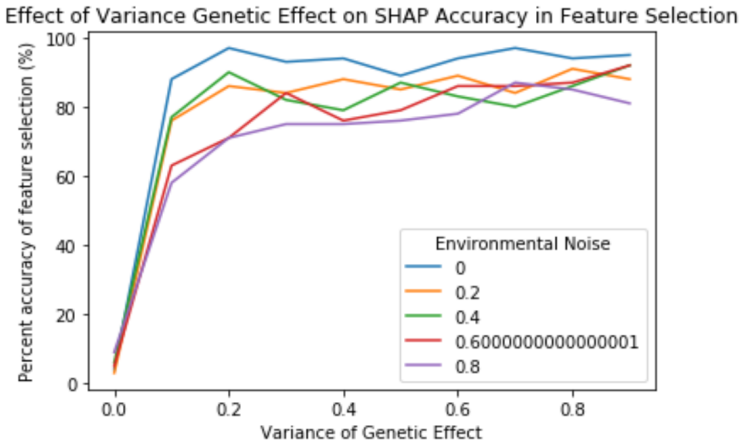
3 Results

3.1 SHAP Accuracy

Random Forest Regressor One Loci, Set Genetic Effect
Testing the effectiveness on SHAP with Random Forest Regressor with the set value of genetic effect shows that in the best case scenario RFR is above 80 percent for all environmental noise levels after a genetic effect of 0.2.

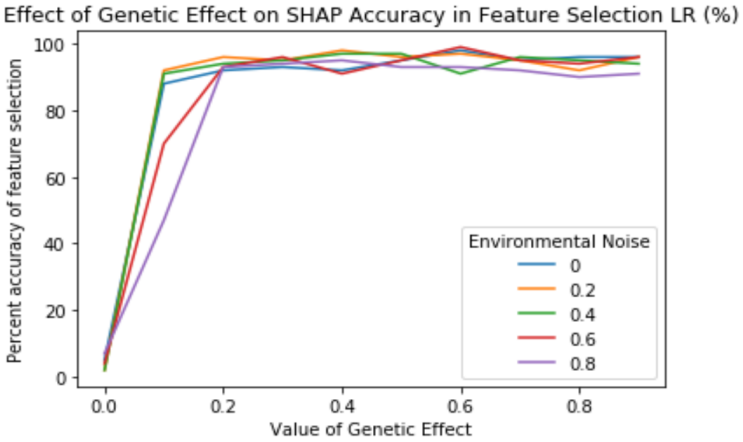


Random Forest Regressor One Loci, Set Varient Effect
Testing the effectiveness on SHAP with Random Forest Regressor with the variance of genetic effect shows that in the real world scenario RFR is above 80 percent accurate for all environmental noise levels after a genetic variance of 0.6. For environmental noise levels less than 0.4 the accuracy is above 80 percent after a genetic variance of 0.1.



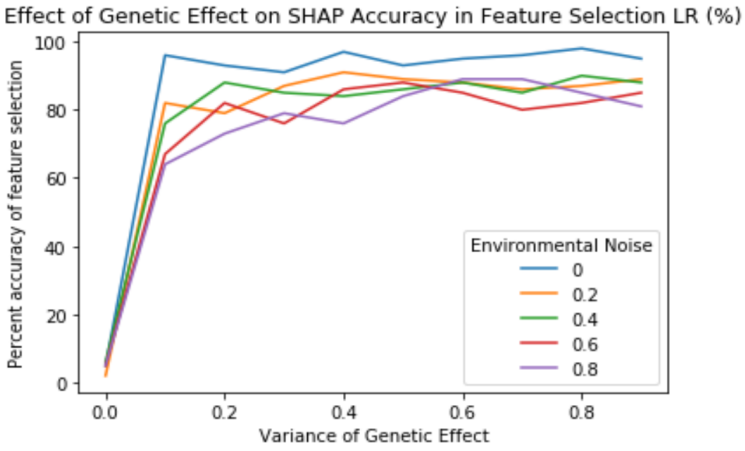
Linear Regression One Loci, Set Genetic Effect

Testing the effectiveness on SHAP with Linear Regression with the set value of genetic effect shows that in the best case scenario LR is around 90 percent for all environmental noise levels after a genetic effect of 0.2.

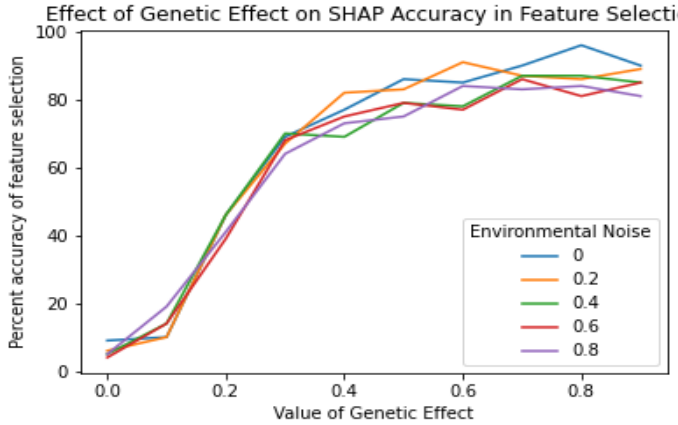


Linear Regression One Loci, Set Varient Effect

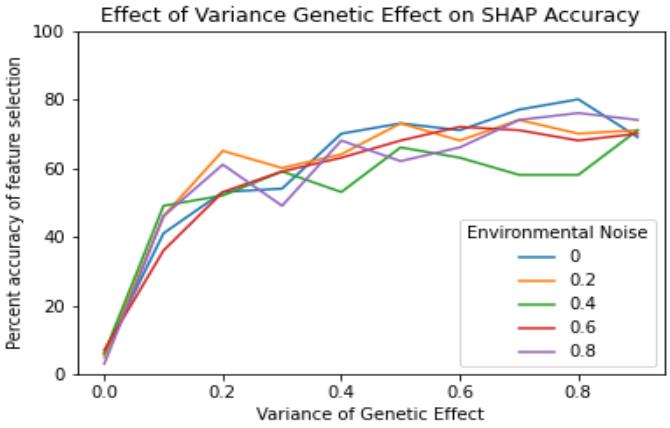
Testing the effectiveness on SHAP with Linear Regression with the variance of genetic effect shows that in the real world scenario LR is above 80 percent accurate for environmental noise levels below 0.8.



Neural Network Result One Loci Set Environmental Noise
Here we plotted how SHAP performs at detecting one causal loci when the variance for the environmental noise is set.



Neural Network Result One Loci Varied Environmental Noise
Here we plotted how SHAP performs at detecting one causal loci when the variance for the environmental noise varies for multiple values.
When the environmental noise is set for one loci

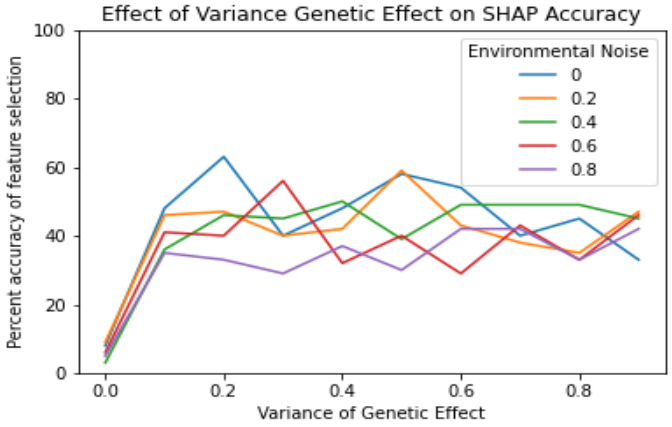

Effect of Variance Genetic Effect on SHAP Accuracy
Percent accuracy of feature selection
Variance of Genetic Effect
Environmental Noise
0
0.2
0.4
0.6
0.8
0
20
40
60
80
100
0.0
0.2
0.4
0.6
0.8

SHAP Accuracy for two loci

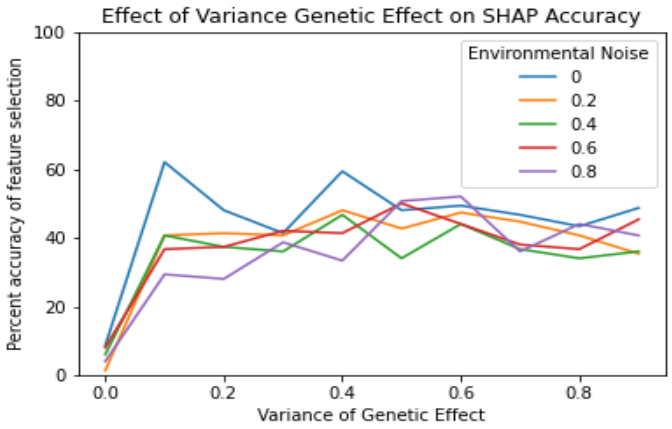
Random Forest Regressor accuracy for two loci

Here we plotted the accuracy of SHAP for two loci when there were no interactions between the the two SNPS. Both plots have 1000 samples and 20 loci.Here $\beta_{1,2}$ was equal to zero. Each variance is run 100 times each.

The first plot shows the accuracy when e_j is set



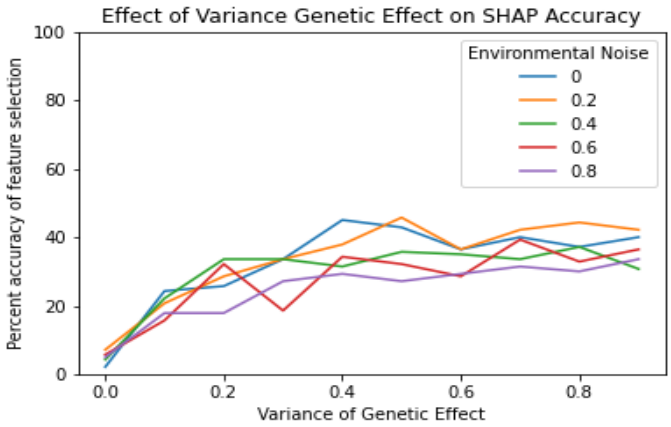
The second plot shows the accuracy when e_j varies and is a more accurate representation of the accuracy



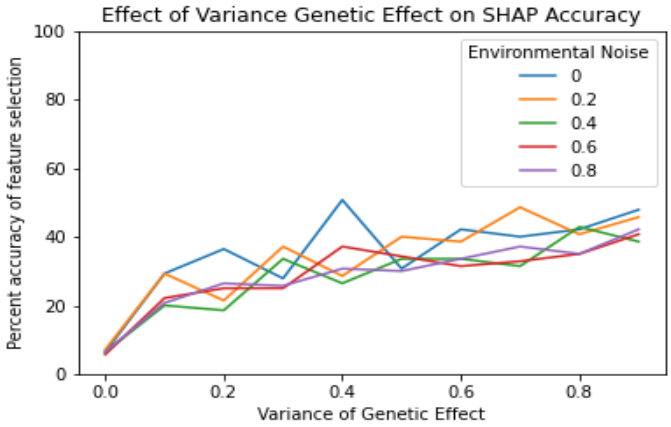
Neural Network accuracy for two loci

Here we plotted a 2000x20 genotype matrix to a neural network. In this scenario the the batch size was 16, epochs 50 and each variance parameters was run 100 times. Here the environmental noise was set so it did not vary for multiple values. We plotted for multiple variances how well SHAP when

used on neural networks is at detecting two causal loci. Here $\beta_{1,2}$ was zero because SHAP interaction values are not compatible with neural networks.



Here the same parameters as before but now the environmental noise is varied which means this plot is a more true representation of how well the neural network would perform for multiple environmental noises.

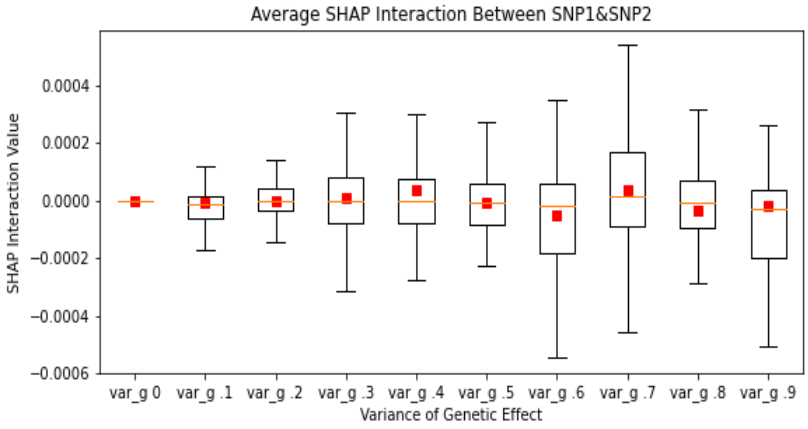


3.2 SHAP interaction value plots

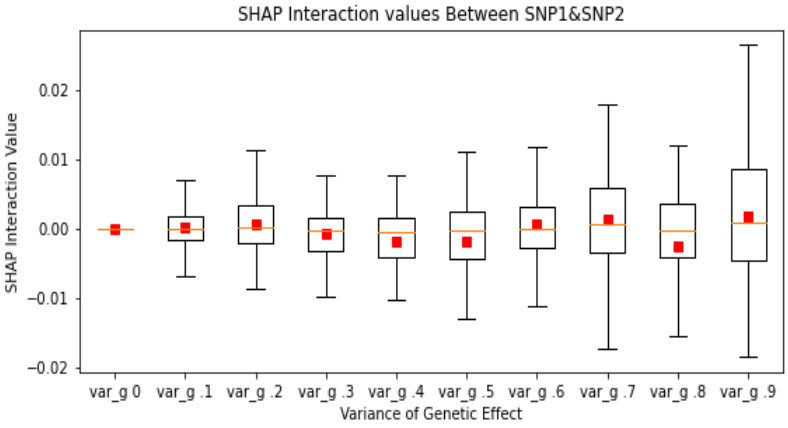
The following plots shows the SHAP interaction values for both SNPS across different variances for genetic effect and for different interaction scenarios for $\beta_{1,2}$. Each plot was made for 5000 samples and 100 loci and each variance was run for 100 trials. The red squares represent the average SHAP interaction value and the orange line represents the median of the interaction values.The whiskers extending from the box plot represent the full range of the SHAP interaction

values.

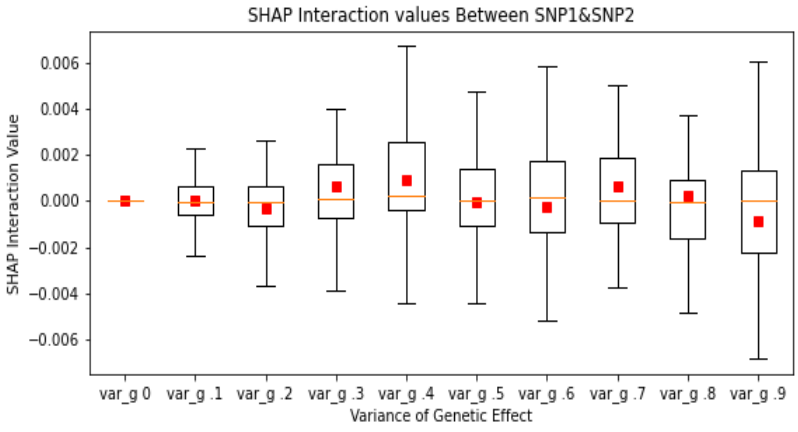
$\beta_{1,2} = 0$
Here we plotted the interactions between SNPS when there should be no interaction occurring. It can be immediately some interactions are occuring even there should be none, this will be discussed in the discussion section.



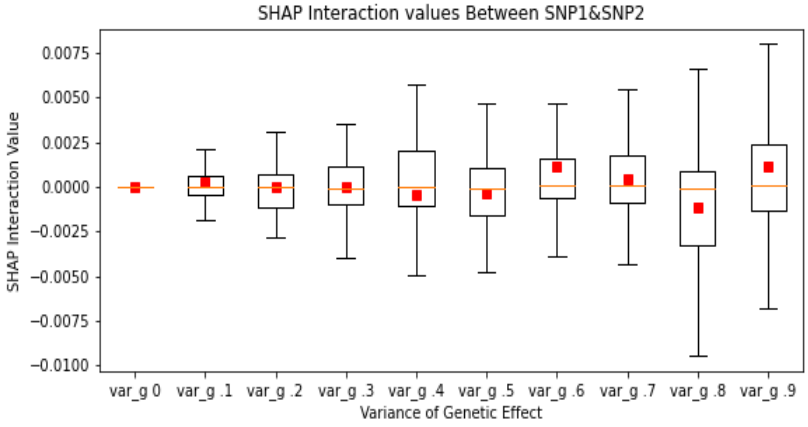
$\boldsymbol{\beta} = N(0, \sigma_g^2 \boldsymbol{I})$
Here we plotted the interaction between SNP's when $\beta_{1,2}$ comes from a a random normal distribution with some variance.



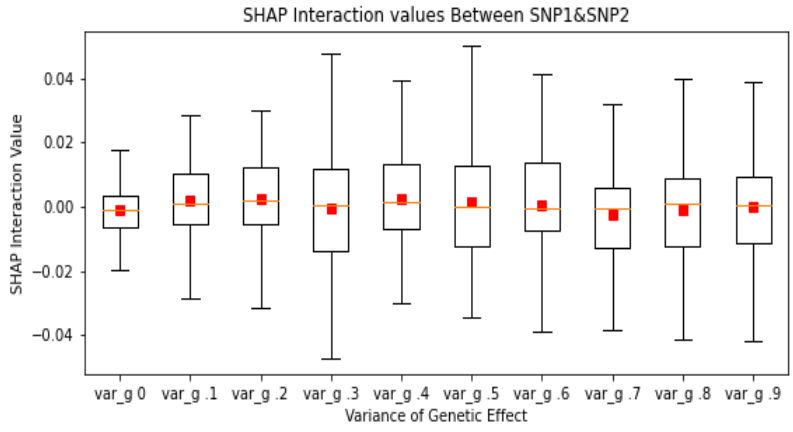
$0 < \beta_{1,2} < \beta_1$
Here we plotted the interaction between SNP's when $\beta_{1,2}$ is less than β_1



$0 < \beta_{1,2} < \beta_2$
Here we plotted the interaction between SNP's when $\beta_{1,2}$ is less than β_2

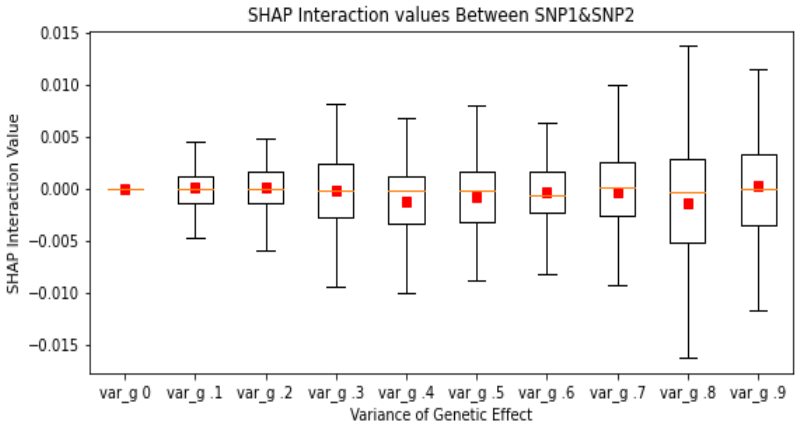


$\beta_1 + \beta_2 < \beta_{1,2}$
Here we plotted the interaction between SNP's when $\beta_{1,2}$ is greater than both β_1 and β_2



$\beta_{1,2} < 0$

Lastly we plotted the interaction between SNP's when $\beta_{1,2}$ is less than 0



4 Discussion

For single Loci SHAP shows a high level of accuracy and a high tolerance for environmental noise. However, the question still remains as to why the machine learning models do not have perfect accuracy at 0 environmental noise. This is an area to explore in the future, by both checking the models for their phenotype prediction accuracy and then testing SHAP.

For the accuracy of SHAP at identifying two causal loci we found that for Random Forest Regressor and Neural Networks, SHAP performed poorly at identifying the correct loci. This could be due to a few reasons, one being maybe not enough data is provided to the machine learning models or that the parameters of the models themselves need to tuned to make the model more optimal for the data. Another reason could be due to the nature of our problem in that we developed a regression problem, but Random Forest and Neural Networks are often used for classification. A future step could be run all our models again and only use binary traits to determine SHAP accuracy.

In our research we found an interesting finding for detecting loci interactions. Namely that even when we set $\beta_{1,2}$ equal to zero we still see some interactions taking place. This could be due to not enough samples, but we did make the samples equal to 5000 which is a fairly large sample. Another possibility is that SHAP is limited at determining interactions. SHAP interaction values are based on Shapley values which could mean that that due to Shapley values assuming features are independent there could be some noise being introduced by SHAP when calculating the interaction values between loci. It would be interesting to determine how the noise is introduced either by trying larger sample sizes or by performing statistical analysis on SHAP interaction values in order to understand their limitations.

For next steps we also need to develop more complex simulations that are more similar to real data. Introducing linkage disequilibrium into our simulation would be a good way to further determine the accuracy of SHAP in more complex simulation besides just interactions between loci. We could also try Neural Networks with more than one hidden layer in order to determine how well SHAP is at determining casual loci in a truly deep network.

Further study into SHAP values should also be done to attempt to determine the causal loci in the case where the number of causal loci is unknown.

5 Acknowledgements

We would like to thank Boyang Fu, Dr.Nandita Garud, and Dr.Sriram Sankararaman for advising us for this summer project and providing advice on how to carry out our project and how to advance it.