Better Understanding Genomic Architecture with the use of Applied Statistics and Explainable Artificial Intelligence

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Jonathon Romero

August 2022

Dedicated to my grandfather who instilled in me my curiosity and strive to learn.

Acknowledgments

I would to thank my friends and coworkers for being supportive throughout my journey through graduate school. I would like to thank the many teachers I have met throughout my life that have taught me lessons from the classroom and beyond. And I would like to thank Ashley Cliff, without whom none of this would have been possible, for keeping me sane and leading the way.

Abstract

With the continuous improvements in biological data collection, new techniques are needed to better understand the complex relationships in genomic and other biological data sets. Explainable Artificial Intelligence (X-AI) techniques like Iterative Random Forest (iRF) excel at finding interactions within data, such as genomic epistasis. Here, the introduction of new methods to mine for these complex interactions is shown in a variety of scenarios. The application of iRF as a method for Genomic Wide Epistasis Studies shows that the method is robust in finding interacting sets of features in synthetic data, without requiring the exponentially increasing computation time of many classic association study methods. Leveraging the non-parametric prediction capabilities of iRF, new genomic insights are used to improve Genomic Selection and Progeny Prediction. This capability enables breeders to make informed selections for crosses without first requiring a full progeny trial. A new algorithm, Tensor Iterative Random Forest (TiRF), expands upon the foundation of iRF, to provide information on relationships not only between the features and targets of the model, but also the between the targets themselves. This algorithm is validated with the capture of information from gene regulatory networks from the DREAM competition. The impact of the SARS-CoV-2 virus has necessitated a method that can capture the changing nature of the genetic architecture of the virus and incorporate potential recombination events, paving the way for a better understanding of how the virus has changed and will change. A new method is introduced that identifies likely parents of haplotypes designated to be the result of recombination. Together, these new methods aim to provide a stronger insight into genetic architecture complexities.

Table of Contents

1 Introduction **1**

1.1 Background . . . 2

1.2 Machine Learning/Artificial Intelligence . . . 4

1.2.1 Explainable Artificial Intelligence . . . 5

1.2.2 Random Forests . . . 6

1.3 Genomics and Phenotypes . . . 11

1.3.1 Genomic Data . . . 12

1.3.2 Epistasis . . . 13

1.3.3 *Populus trichocarpa* data . . . 17

1.4 Genomic Selection . . . 18

1.4.1 Progeny Trials . . . 19

1.4.2 Traditional Genomic Selection . . . 20

1.4.3 Machine Learning for Genomic Selection . . . 21

1.5 SARS-CoV-2 Background . . . 22

1.5.1 Structure and Variants . . . 22

1.5.2 Template Switching and Recombination . . . 23

1.5.3 Evolutionary Incompatibility . . . 24

1.6 Conclusion . . . 25

2 Genome Wide Epistasis Study with Iterative Random Forest **26**

2.1 Abstract . . . 27

2.2 Introduction . . . 27

2.3 Methods . . . 29

2.3.1 Iterative Random Forest . . . 29
2.3.2 iRF Metrics . . . 30
2.3.3 Generating Synthetic Data with Epistatic-Like Properties . . . 36
2.3.4 Synthetic Trials . . . 37
2.3.5 Model Evaluation Methods . . . 39
2.3.6 *Populus trichocarpa* data . . . 41
2.4 Results and Discussion . . . 43
2.4.1 Synthetic Data . . . 43
2.4.2 Poplar Data . . . 53
2.5 Conclusion . . . 55

3 Progeny Prediction with iRF **57**
3.1 Abstract . . . 58
3.2 Introduction . . . 58
3.3 Methods . . . 60
3.3.1 Heritability . . . 60
3.3.2 Progeny Trials . . . 62
3.3.3 Progeny Prediction Pipeline . . . 63
3.3.4 Iterative Random Forest . . . 65
3.3.5 *Populus trichocarpa* data . . . 65
3.3.6 Data Processing and Environmental Effect Considerations . . . 67
3.3.7 Calculating GCA and SCA . . . 68
3.4 Results and Discussion . . . 69
3.4.1 Pipeline . . . 69
3.4.2 Predictions . . . 69
3.5 Conclusion . . . 76

4 Tensor iterative Random Forest **79**
4.1 Abstract . . . 80
4.2 Introduction . . . 80
4.3 Methods . . . 83

4.3.1 Random Forests . . . 83
4.3.2 Tensor Iterative Random Forest . . . 85
4.3.3 DREAM Competition data . . . 89
4.4 Results and Discussion . . . 92
4.4.1 TiRF Model . . . 92
4.4.2 DREAM Network Construction . . . 92
4.4.3 Dream Network Accuracy . . . 94
4.5 Conclusion . . . 100

5 Building a Genetic Lineage of SARS-CoV-2 **102**
5.1 Abstract . . . 103
5.2 Introduction . . . 103
5.3 Methods . . . 106
5.3.1 SARs-CoV-2 Data . . . 106
5.3.2 Incompatibility Between SNPs . . . 108
5.3.3 Haplotype Relationship Network . . . 109
5.3.4 Potential Parental Search Method . . . 112
5.4 Results and Discussion . . . 114
5.4.1 Incompatibility of SNPs . . . 114
5.4.2 Haplotype Relationship Network . . . 118
5.4.3 Selection of Potential Parents . . . 121
5.5 Conclusion . . . 125

6 Conclusion **127**

Bibliography **130**

Appendices **160**
A Chapter 5 Supplementary Figures . . . 161

Vita **164**

List of Tables

2.1 An exhaustive combination of each parameter listed was used to to generate synthetic data trials. 38

3.1 This table shows the correlation between the predicted genetic components of the progeny trial from iRF and a simple parental trial to the True Full Cross Progeny Trial. The first column indicates which aspect of the genetic component is being compared, either the GCA from one of the sets of parents, the SCA from the crosses, or the combined prediction of GCA and SCA for the full genetic component prediction. The iRF Progeny Trial is the result of dividing the predicted iRF progeny into its genetic components and comparing it to the actual progeny trial. The Traditional Estimation is taken by defining the genetic component of the phenotype of each individual parent grown in clonal replicates at the progeny trial site. The Pearson Correlation is used to compare the effective magnitude similarity between the the Genetic Components and the Spearman Correlation is used to compare the rank order of the Genetic Components. 71

List of Figures

1.1 This figure shows the general layout of data for machine learning algorithms. Separate samples have measurements for multiple features that can be used to predict a target. 7

1.2 In a decision tree the most basic unit of model creation is a single decision that splits a single node. A) The parent node of the split contains all of the samples (and each of there features and targets) available to the split. B) The feature for the split is chosen to be the first column, and the split value is chosen to be 3. C) The Samples that have a value less than the split value for the feature move to the left child. Since the two middle samples have values for the feature greater than the split value they are not present on the left. D) The samples not present in the left child node are present in the right node. 7

1.3 An example decision tree that starts with 6 samples at the root node and splits them into 4 leaf nodes with 3 total splits. 9

1.4 An example of grouped decision trees that, when taken in aggregate, form a forest. To be a Random Forest, each tree needs a random subset of the samples, and each split needs to look at a random subset of features to split upon. 9

2.1 By training an iRF model with synthetic data that includes causative and non-causative features with included noise we can see how the split's across the entire forest for a feature behave. The left graph shows splits from a non-causative feature, with no coherent or consistent effect measured. The right graph shows splits from a causative feature with a consistent effect that shows a positive relationship between the feature and the dependent variable. . . . 44

2.2 This heat map shows the average AUROC of subsets of the iRF models trained on the synthetic data sets in determining the effect direction of causative features. The the columns represent models trained on data sets with an amount of noise, either 25%, 50%, or 75% noise. The first three columns represent the models trained on data sets with no correlated features and last three columns represent the presence of correlated features. The heat map rows represent the size and structure of the data sets with the top two rows containing 500 samples each and the bottom 2 having 1000 samples each. The rows alternate from representing 100 features to 10000. The scale of the heat map begins at .5, which is the AUROC of a random model and tops out at 1 which would be a perfect model. This shows how every axis of change of the data set has some effect on the models. More samples generally mean better AUROC, more total features generally correlate to lower AUROC, more noise in the model leads to lower AUROC, and correlated features generally leads to lower AUROC. 46

2.3 This heat map shows the average balanced accuracy of subsets of the iRF models trained on the synthetic data sets in prediction of causative features with the importance threshold of $\frac{1}{N}$ where N is the number of samples used in the model. The the columns represent models trained on data sets with an amount of noise, either 25%, 50%, or 75% noise. The first three columns represent the models trained on data sets with no correlated features and the last three columns represent the presence of correlated features. The heat map rows represent the size and structure of the data sets with the top two rows having 500 samples and the bottom 2 having 1000 samples, and the rows alternate from representing 100 features to 10,000. The scale of the heat map begins at .5, which is the balanced accuracy of a random model and tops out at 1 which would be a perfect model. 48

2.4 This heat map shows the average post test probability of subsets of the iRF models trained on the synthetic data sets in prediction of causative features with the importance threshold of $\frac{1}{N}$, where N is the number of samples used in the model. These values directly correspond to the chance that a feature that meets the importance threshold from that data set is a True causative feature for the phenotype of that model. The the columns have represent models trained on data sets with an amount of noise, either 25%, 50%, or 75% noise. The first three columns represent the models trained on data sets with no correlated features and last three columns represent the presence of correlated features. The heat map rows represent the size and structure of the data sets with the top two rows having 500 samples and the bottom 2 having 1000 samples, and the rows alternate from representing 100 features to 10000. The scale of the heat map begins at 0 which is the probability of never correctly predicting a causative feature and tops out at 1 which would be a every predicted feature being a true causative feature. 49

2.5 This heat map shows the average balanced accuracy of subsets of the iRF models trained on the synthetic data sets in prediction of causative sets of features with the four separate detection methods: RIT, *RIT_adjusted*, *set importance* (setimp), and *conditional importance* (conimp). The columns represent models trained on data sets with an amount of noise, either 25%, 50%, or 75% noise. The first three columns represent the models trained on data sets with no correlated features and last three columns represent the presence of correlated features. The heat map's rows indicate the four metrics being evaluated - RIT, *RIT_adjusted*, *set importance*, and *conditional importance*. The scale of the heat map begins at .5 which is the balanced accuracy of a random model and tops out at 1 which would be a perfect model. Because the sets examined by the other methods are determined by RIT, it makes some sense that it has the highest balanced accuracy in prediction of causative sets. While RIT and *conditional importance* look identical when comparing out to 3 decimal places, they do diverge slightly after considering more places. 51

2.6 This heat map shows the average post test probability of subsets of the iRF models trained on the synthetic data sets in prediction of causative sets of features with the four separate detection methods: RIT, *RIT_adjusted*, *set importance* (setimp), and *conditional importance* (conimp). The color scale has been set to show the range between the best performing metrics and the worst. The columns have represent models trained on data sets with an amount of noise, either 25%, 50%, or 75% noise, also first three columns represent the models trained on data sets with no correlated features and last three columns represent the presence of correlated features. The heat map's rows indicate the four metrics being evaluated - RIT, *RIT_adjusted*, *set importance*, and *conditional importance*. The scale of the heat map begins at 0 which is the probability of never correctly predicting a causative sets and tops out at 1 which would be a every predicted feature being a true causative sets. We can see here that the likelihood of a predicted set being a causative set, is significantly lower than identifying the individual features. However part of this is likely because of the imbalanced nature of true to false sets playing a more significant roll. Despite this, *set importance* does perform the best by having a predicted set being truly causative nearly 20% of the time for the lowest noise subsets. 52

3.1 This image shows the simple pipeline for Progeny prediction. Step 1 – gather genetically distinct individuals of a species from the general population. Step 2 - grow the collected individuals in a common environment and measure a selection of desired phenotypes. Step 3 - Genotype the collected population. Step 4 - Train a machine learning model to predict the collected phenotype(s) based on their unique genotypes. Step 5 - Computationally produce *in silico* progeny by crossing the genotype of two potential parents in order to create the genotype of a potential offspring. Step 6 -Use the model trained in step 4 to predict the desired phenotype for the *in silico* genotype from step 5. . . . 64

3.2 A comparison of the normalized values of the coefficients of variance attributable to the mothers from the *in silico* progeny height phenotype (green) to the coefficients of the real progeny trial(blue), and to the traditional calculation requiring the clonal replication of the parents(red). By sorting the Mothers by the Real values we can compare the trend line of the normalized values to the order from the order of the Real values. This graph illustrates how close the traditional method gets to capturing the value of GCA of the mothers but that there is still a decently strong correlation between the trend line of the Real GCA values and the iRF progeny derived Values. 73

3.3 A comparison of the normalized values of the coefficients of variance attributable to the fathers from the *in silico* progeny height phenotype (green) to the coefficients of the real progeny trial(blue), and to the traditional calculation requiring the clonal replication of the parents(red). By sorting the Fathers by the Real values we can compare the trend line of the normalized values to the order from the order of the Real values. This graph illustrates how close the traditional method gets to capturing the value of GCA of the mothers but that there is still a reasonably strong correlation between the trend lines of the Real GCA values and the iRF progeny derived Values. By comparing the closeness of the trend lines to figure 3.2 we can see that the fathers from the iRF-based method don't match the real values as well as with the mothers, especially with the value of father 2572. 74

3.4 A comparison of the normalized values of the coefficients of variance attributable to the Combinations of the parents, or the SCA, from the *in silico* progeny height phenotype (green) to the coefficients of the real progeny trial(blue). Because it is impossible to caluclate the SCA of crosses with a traditional model without including progeny from the population, those values do not exist and are not shown in this graph. By sorting the Pairs by the Real values we can compare the trend line of the normalized values to the order from the order of the Real values. This graph there is a correlation between the trend line of the Real SCA values and the iRF progeny derived Values. . 75

3.5 By ordering the families by their value in from the real progeny trial we can build trend lines for each of the predictions to illustrate the difference in each prediction. This plot illustrates the comparison of the normalized full genetic components from the Real Progeny trial shown as the blue dots and its trend line as the blue line, the iRF progeny trial shown as the green x's and green line for its trend line, and the Traditional Estimate in red circles and red line for its trend line. With the trend line we can see that the iRF Progeny trial is capturing more information than the traditional estimate when trying to predict the full Genetic component of the phenotype. 77

4.1 The top left corner shows the four columns of features and four columns of targets for an example data set with five samples. The right side of the figure shows a TiRF decision tree, where one feature is chosen at each split based on its ability to best split one or more chosen targets. The bottom left shows the different types of associations that are gathered from the tree, namely feature to feature, target to target, and feature to target. The leaf nodes at the base of each tree path denote the samples that reached that node, defining the sample sets with similar target values. 82

4.2 The process of creating non-transcription factor to non-transcription factor gold standard networks from corresponding transcription factor to non-transcription factor associations. For each non-transcription factor gene, if it is regulated by the same transcription factor as another non-TF gene, then the two no-TF genes contain an edge associating them in the new network. . 91

4.3 The gold standard and TiRF overlap from the from the Dream 5 Network 1. The blue nodes represent the transcription factors, and the orange nodes are the non-transcription factor genes. The pink edges indicate the overlap of TF to non-TF from TiRF to the gold standard, while the green indicate the TF to TF overlap. The gold standard edges not captured by TiRF are indicated in light grey. It can be seen in this image that there are regions of the network that are being better identified by TiRF, specifically some of the transcription factors are much better captured by the model than others. 93

4.4 The receiver operating curve (ROC) for the first five iterations of TiRF on the Dream 5 Networks 1, 3, and 4, measuring the accuracy of the model at capturing transcription factor genes to non-transcription factor genes as set by the "gold" standard from the Dream competition. It should be noted that the according to the original competition and construction of the "gold" standard the True Positive rate is much more important than the False Positive rate. a) The ROC for DREAM 5 Network 1 shows TiRF is able to capture the Transcription Factor to gene relationship from this synthetic data. However, while the iterative process reduces the total number of edges through refinement the overall AUROC of the model is slightly effected negatively. b) The ROC for Dream 5 Network 3 shows a slightly worse performance than the synthetic data, with the iterative process having a slightly chaotic effect on the over all AUROC, potentially indicating more noise in the data. c)The ROC for Dream 5 Network 4 has a slightly worse result than either of the other two networks, but is still above the baseline shown by the diagonal dashed line. For this network a general improvement can be seen in the AUROC with the iteration process. 95

4.5 The receiver operating curve (ROC) for the first five iterations of TiRF on the Dream 5 Network 1, 3, and 4 measuring the accuracy of the model at capturing Transcription Factor genes to other Transcription Factor genes as set by the "gold" standard from the Dream competition. These results are limited by the co-occurrence of features in the models and by the presence in the "gold" standard of Transcription Factors predicting each other. a) The ROC for Dream 5 Network 1 shows that the First iteration (Iteration 0) captures the feature to feature interaction in this data the best by far. As the iterations continue and the selection of the features becomes more heavily determined by feature weight we can see the AUROC drop off to the point where the final iteration is below the baseline. b) The ROC for Dream 5 Net 3 shows a slightly worse picture for the first iteration than Network 1's first iteration but the subsequent iterations all have much higher AUROCs. c) The ROC for Dream 5 Net 4 has by far the worst overall accuracy, with every iteration being below the baseline. 97

4.6 The receiver operating curve (ROC) for the first five iterations of TiRF on the Dream 5 Network 1, 3, and 4 measuring the accuracy of the model at capturing non-Transcription Factor genes to other non-Transcription Factor genes as set by the new "gold" standard, derived from the Dream competition. The derivation is performed by connecting all genes that are predicted by at least one common transcription factor. Because of the number of overall connections in the network reaching into the millions, these ROC graphs represent only the top 25,000 most commonly co-occurring target sets from each model. a) The ROC for Dream 5 Network 1 shows that TiRF is able to capture the target to target relationships within the synthetic data set with a small amount of variation over the course of the iterations b) The ROC for Dream 5 Network 3 shows a slightly better capture of the "gold" standard edges than the synthetic data in Network 1, but does show a slight decrease in AUROC with each iteration. c) The ROC for Dream 5 Network 4 has the overall worst results in capturing the target to target relationships, however each iteration is better than the baseline. This network also has the largest fluctuation in AUROC from iteration to iteration, with the first iteration capturing the most information. 99

5.1 The visual difference of Convergent Evolution vs Recombination. For the set of sequences: GTC, GTT, ATT, and ATC, in order to create a phylogenetic tree with purely divergent branches you create a scenario of Convergent Evolution were the first mutation of A to G (black circle) occurs and then the C to T mutation (white circle) occurs twice in the tree in separate locations. However, if recombination is allowed to occur, the same tree can be constructed with each mutation only being required to occur once. It can be seen that, by connecting GTC and ATT with recombination, it is possible to attain the sequence GTT. 105

5.2 A potential process for recombination within SARs-CoV-2. 1) In order for unique haplotype A and B to recombine they would need to be present in the same cell. 2) The replication process starts for haplotype A, and after some section of the sequence is duplicated the process is paused. In this case haplotype A's replication is paused after the AUGC sequence. 3) Replication then restarts using haplotype B as the template. This completes the overall construction of the new haplotype, successfully recombining haplotypes A and B. 107

5.3 **Haplotype Network.** Haplotypes with the sequences ATC, ATT, GTC and GTT, are all connected with number of SNP differences shown by the edge weight. The distance from ATC to GTT and from ATT to GTC is 2, the distance from ATC to ATT, ATT to GTT, GTT to GTC, and GTC to ATC is 1. Because the edges around the outside are all one and the internal edges are two we can represent the same information in the network by removing the internal edges denoted by the red dotted lines. The remaining edges form a homoplasic loop indicating that there are at least two separate feasible stepwise paths for evolution to take. 111

5.4 The process used to find a potential set of "parents" or ancestors of a SARs-CoV-2 haplotype. A) On selection of a haplotype of interest, all haplotypes that were sampled before it are added to a list of potential parents. B) The first parent is chosen as the haplotype that best covers the region in our haplotype of interest with the least coverage by the potential parents. C) The SNPs that overlap between the selected haplotype from B with the haplotype of interest are then used to create a new partial haplotype. D) The same list of haplotypes used in the search for B) is used to again search for the closest match with the haplotype of interest, excluding the SNPs from partial haplotype from C. E) The overlap from the second parent is added to the partial haplotype. This process of step D and E continues to repeat until the original haplotype is fully reconstructed, the addition of new parents no longer increases the size of the partial haplotype, or a preset number of parents is chosen. 113

5.5 Incompatibility Matrix of SARS-CoV-2 Haplotypes shown as a heat map. Each axis is the SNP locations and each red pixel represents incompatibility between the two locations. This represents a summarized view of the genome by removing SNPs from both axes that have very little information (as defined by a total incompatibility with the rest of the genome of less than 500). The darker appearing regions indicate more incompatibility between the two sites and the denser the local area of incompatibility. The lines of darker appearing color highlight sites that have likely had mutations transferred to other parts of the population through recombination resulting in incompatibility with most of the genome. 115

5.6 Incompatibility Matrix of SARS-CoV-2 Haplotypes shown as a heat map zoomed in to specific regions of the genome. Each axis is the parsimonious SNPs in the order they appear in the sequence. A) From SNP 18694 to 22000 on both axes shows a region along the diagonal that has two separate regions of incompatibility mirrored across the diagonal. Both regions appeared to be fully boxed in by at least one SNP on either side that is compatible with the whole region, potentially representing a SNP at the border of a region of recombination. This region lines up with the intersection between Beta-CoV-Nsp14 and an Nsp15-like endoribonuclease. B) From SNP 27382 to 28882 on both axes shows a region along the diagonal that has two separate regions of incompatibility mirrored across the diagonal. This shows multiple different shapes of incompatibility and aligns with the intersecting points of multiple open reading frames, including ORF6, ORF7a, ORF7b, and ORF8. C) The zoom in of the incompatibility matrix From SNP 19824 to 22800 on the Y axis and SNP 27925 to 29,904 on the X axis. This region shows both a large block of incompatibility from region of the nsp15 and 2'-O-ribose methyltransferase genes to the region of the Corona Nucleoca and ORF10. 117

5.7 A Circos plot showing the location of the largest incompatibility blocks from the heat map in the context of the SARs-CoV-2 genome and annotations. The outer most band represents the 29,904 bases in the full sequence of SARs-CoV-2, with each tick representing a distance of 1,000 bases. The next three tracks are annotated sections of the SARs-CoV-2 reference sequence from NCBI, broken down into three categories with each change in color on the track representing a different structure within the sequence. The first of these tracks is the "genes" , made up mostly of several Open Reading Frames (ORFs), the first ORF making up most of the sequence. The second track shows the peptides in each ORF and the last track showing different conserved protein domains. The connections shown in the center of the circle illustrate the largest 100 sections of the heat map of connected incompatibilities. Most of the largest incompatibility regions are incompatible with multiple other regions, and not just one singular other region. Hot spots for incompatibility favor regions where proteins or genes intersect, but also occasionally appear in the middle of large sequences from a single protein. The teal blue on the inner most track represents the region coding for the SPIKE protein and includes incompatibility hotspots both within the gene and along its intersection with the proceeding gene. 119

5.8 The network is constructed by taking unique SARS-CoV-2 haplotypes (the nodes) and connecting them if they are within 5 mutational steps away from each other. The size of each node represents the number of times a sequence was sampled that belonged to that haplotype and the color represents when the haplotype was first sampled, with lighter, yellower colors being earlier and darker, more purple colors being later in the pandemic. 120

5.9 The network is constructed by taking unique SARS-CoV-2 haplotypes (the nodes) and connecting them if they are 1 point of mutation away from each other. The size of each node represents the number of samples within that haplotype and the color represents when the haplotype was first sampled, with lighter, yellower colors being earlier and darker, more purple colors being later in the pandemic. 122

5.10 The network shows how recombination can be missed by looking at haplotypes connected by mutational distance. By searching through all haplotypes in our data set that had a sample collected before the collection of the haplotype labeled C, B.1.1.7 from the pango lineage, it was identified that the haplotype A and the haplotype B in combination account for all but 5 SNPs that were new mutations at the time, indicating that while their is significant overlap between C and its potential parents it does still introduce new mutations to the population at the same time. This network is constructed by taking unique SARS-CoV-2 haplotypes (the nodes) and connecting them if they are 5 or fewer points of mutation away from each other. The size of each node represents the number of samples within that haplotype and the color represents when the haplotype was first sampled, with lighter, yellower colors being earlier and darker, more purple colors being later in the pandemic. . . 123

5.11 The network shows the potential parents of the P.1 strain first detected in Brazil. The P.1 label, as well as the others shown in the graph, are the pango network designations, and allow for small variations in haplotype within a single designation. Using the parental search method, it was identified that haplotype B.1 and the haplotype B.1.1.7, in combination, account for 99.7% of the P.1 sequence, with B.1 providing at least 30 unique mutations not found in B.1.1.7. The other three parents shown in the image, B.1.1, B.1*, and B.1.279 each add only a single additional mutation. Both the color of the node and the relative position left to right indicate date of first sampling of the specific haplotype. Even though B.1.1.7 is a descendant of B.1 it appears before it because the specific haplotype represented by B.1 in the graph was first measured after the B.1.1.7 haplotype. This also accounts for the presence of B.1 and B.1*, with the later being only a few mutations away from the original B.1. 124

1 Incompatibility Matrix of SARS-CoV-2 Haplotypes shown as a heat map. Each axis is the parsimonious SNPs in the order they appear in the sequence with the SNPs labeled to roughly what they would be in the full Genome. The darker color means more incompatibility between the two sites. The lines of darker color highlight sites that have likely had mutations transferred to other parts of the population through recombination resulting in incompatibility with most of the genome. 161

2 This network is constructed by taking unique SARS-CoV-2 haplotypes (the nodes) and connecting them with an edge whose value is the number of mutational differences between the haplotypes. The size of each node represents the number of samples collected with that haplotype and the color represents when the haplotype was first sampled, with lighter colors being earlier and darker colors being later in the pandemic. Each edge is then pruned if there is an intermediary haplotype whose edge values to the original two haplotypes sum to the same or less value than the distance of the original edge. This is equivalent to saying that haplotype of the intermediary node is also the intermediary evolution point between the two originally connected haplotypes. This network contains 8742 unique haplotypes representing up to 2941 individual samples and 390307 edges connecting them which represent mutational differences from 1 to 722 of 14793 SNPs. 162

3 This network is constructed by taking unique SARS-CoV-2 haplotypes (the nodes) and connecting them if they are 15 points of mutation (the value of the edge) away from each other or less. The size of each node represents the number of samples within that haplotype and the color represents when the haplotype was first sampled, with lighter colors being earlier and darker, bluer colors being later in the pandemic. Each edge is then pruned if there is an intermediary haplotype whose edge values to the original two haplotypes sum to the same or less value than the distance of the original edge. This is equivalent to saying that haplotype of the intermediary node is also the intermediary evolution point between the two originally connected haplotypes. 163

Chapter 1

Introduction

1.1 Background

The amount of data collected in the field of biology is expanding exponentially[1] due to advancements in technology and methodology. With this increase in analyzable data comes a complication; due to the complex nature of biological systems, the various associations across these data are almost guaranteed to be complex themselves. However, many traditional methods of analysis for these data cannot handle the ever-increasing amounts of interaction and complexity. To better understand these intricacies requires new insights and methods.

Some of the most common analysis methods in modern biology are correlation and linear analysis. The results of these types of analyses are generally easy to understand and can provide more evidence for relationships that should be further analyzed. The linear relationship defined by these algorithms can be represented by $y = Ax + B$, where the value A has an effect on x with an offset of B to produce some property y. For example, this could represent the heart rate of a runner (y) being directly related to the speed of the runner(x) or the weight of a amphipod crustacean female (y) being due to the number of eggs(x) she is carrying[2]. While this type of relationship definition can be extremely useful, in biology and other fields, there exist more complicated, non-linear and conditional relationships that cannot be well defined with these simple methods.

To better help define these complex relationships are a variety of machine learning algorithms. These algorithms come in many forms, with the non-linear methods generally being better suited for determining non-linear relationships[3, 4]. Using machine learning techniques to account for non-linear genomic effects not only has promise, but has already seen application in several areas[5, 6, 7]. Iterative Random Forest[4] is an example of an Explainable Artificial Intelligence (X-AI) method that provides both a predictive model as well as a human understandable reason for those predictions. While many algorithms fit the definition of X-AI, the intrinsic tree-based and non-parametric nature of iRF lends itself well to understanding both non-linear and conditional effects.

The ability to analyze and identify non-linear and conditional effects is especially important for aspects of biological data such as genetic data. Pleiotropy, a single change in DNA causing a change in more than one measurable feature, and epistasis, multiple changes

in DNA working together to have a non-additive effect on a measurable feature[8], are both known types of genetic interaction and both fall outside the traditional, linear methodology used for many biological analyses. In the field of genomic selection, a deep understanding of the potential parents for the breeding process is necessary to avoid wasting time and resources on non-viable crosses. While there have been advances to domestication with the use of traditional statistics, machine learning algorithms such as Iterative Random Forest(iRF)[4, 9] could provide the ability to accurately prediction progeny traits, by including the non-linear and conditional interactions often ignored by other methods.

While traditionally considered a separate area of research, viral recombination has some similarities to traditional sexual reproduction in living organisms. Both are inherently a conditional process, and thus non-linear, because they result in offspring that have some amount of unique genetic material from each parent. The resulting traits of the offspring are not only determined by which parents they have, but how much of the unique genetic information (and its location on the genome) was inherited from each parent. This understanding can be used when trying to determine the lineage of a specific viral sequence to better understand its origin and potential strains. For SARS-Cov-2, the virus responsible for the Covid-19 pandemic, a better understanding of how the virus has changed over time may help understand severity of new strains and may prepare for future pandemics.

Many areas of biological research could benefit from new methods and tools that are able to handle and interpret the non-linear and conditional complexities of biological data. Methods such as Iterative Random Forest, that are inherently capable of working with these types of relationships, provide a solid foundation to build upon. From here, a deeper understanding of epistatic genetic relationships can be gleaned. Also possible are more advanced progeny prediction methods that can better work with the complex parental crosses while appropriately utilizing resources. And finally, these understandings can lend themselves to better understanding the viral evolution of SARS-Cov-2.

1.2 Machine Learning/Artificial Intelligence

Machine Learning (ML) is the broad classification of algorithms and techniques that allow a computer to learn information by training on a particular set of data. Artificial Intelligence(AI) is an even broader category of algorithms that allow for a machine to make decisions or display understanding in a way similar to a human. Both of these categories have the ability to greatly influence and improve many areas of science, such as analysing and imputing data, and life, with pattern recognition and self-driving cars[10]. While ML and AI are technically different, the terms themselves are often used interchangeably or together regardless of the context[11].

The field of machine learning contains many types of algorithms, ranging from neural networks to decision trees[12]. One of the ways to categorize machine learning algorithms is into supervised ML algorithms, where there is a specific value, such as height of a tree, or classification, such as if a gene is causative for a specific phenotype or not, to predict[13], or into unsupervised ML algorithms, where there is no true correct value to predict but the model can find patterns in the data, such as genes having interacting expression profiles[14, 15]. Supervised algorithms focus on understanding how features, or independent variables, change or effect a specific target, or dependent variable. If the target is a continuous variable, the method is generally called regression, and when the target is differentiated into discrete categories the method is called a classifier. Unsupervised algorithms focus on the structure of the data through understanding the relationship between features or clustering samples together based on the value of their various features.

Another useful paradigm in ML is transfer learning. A model is trained on a general problem that has a large pool of available data, and then that trained model is used for a more specific model that may not have had enough training data on its own. An example of this is within Natural Language Processing or Image Analysis, where a large general data set of millions of samples can train a model to understand the general concept of understanding text or images. Then for individual problems such as identifying a specific type of emotion in text or identifying a specific species of tree, a smaller data set can be used to refine the model to the specific problem. By training a model on a general problem in the same context as

the specific problem, it can be possible to find significantly more initial training data. Once the general model has been trained, the small amount of data for the specific problem can fine tune for model with specific problem. [16]

These different categories and styles of algorithms allow machine learning to be applicable to a broad range of fields. Applications for ML that have aided both scientists, and humans in general, include but are not limited to: aid in autopsies[17], neural networks to classify skin cancer[18], plant breeding for agricultural crops[19], mapping of soil properties[20], epidemiology[21], as well as many others. The machine learning methods used in these applications are not necessarily just for the predictions themselves but also in understanding what the model itself has learned during the training process.

1.2.1 Explainable Artificial Intelligence

Explainable Artificial Intelligence (X-AI), or ML/AI algorithms that make predictions in a way that can be understood by humans, has been around since the 1980s and have applications in many fields. X-AI is particularly suited to scientific research and medicine, where the logic and justification may be as important as the end prediction. Alternatively, 'Black Box' models provide results in such a way that the process used to arrive at the predictions are not easily understood. This becomes a more significant problem when we start asking why a specific scenario produced a false-positive or false-negative result, impacting the users ability to understand the failings of the trained model and develop best practices to prevent future failures[22].

In general, AI and ML algorithms can be put into three broad categories of explainability: "black-box", where prediction reasoning or justification is heavily obfuscated; "mathematically explainable", where either statistics or some numerical reasoning is provided for justification; and "fully explainable", where comprehensive systems provide the user with a break down of how any given solution was reached[23]. The "black-boxes" include any algorithm where the user is prevented from seeing the underlying model for safety reasons such as with self-driving cars[24], or any algorithm sufficiently complex in its nature that it becomes impossible for a human to understand the exact relationship between any given input and output based on the model, such as with extremely large Neural Networks used

in Deep Learning[25]. While some algorithms are inherently "black boxes", there do exist techniques that can add explainability into these algorithms[26]. Possible solutions include adding in explanatory layers that are trained to produce a human understandable explanation for a specific prediction or combinations of traditional statistical techniques that can allow us to comprehend the complex mathematical network that a model might be using[22].

Mathematically explainable models are ones that have understandable reasoning for their decision making, but that reasoning requires extra mathematical steps to parse. An example of this is the process of correlation analysis of underlying data provided by the model, or seeing how comparable different effects are. The "fully explainable" algorithms provide specifics about what types of inputs lead to specific outputs and, in some scenarios, will provide the specific reason for any given particular prediction.

Within those groups, how any given user "accepts", or trusts, the given explanation can depend on several factors, such as the users knowledge of the algorithm and the ability of the algorithm to convey its information to the user[27]. While the mathematical models are often sought by ML architects, the fully explainable models allow for the most widespread use, by providing cognitive value that is can be understood by individuals outside of the model developer and allows for human-in-the-loop decision making[28]. Some algorithms are a natural fit into the definition of Explainable AI, such as the family of algorithms that use decision trees and forests[29].

1.2.2 Random Forests

Leo Breiman published Classification and Regression Trees (CART) in 1984[30], a decision tree based prediction method. However, the process was prone to over-fitting to a given data set. In 2001, he expanded on his work with the publication of Random Forest[31], a method that combines many decision trees into a forest with added random variation in the tree creation. Within a given decision tree, a set of samples are divided at splits, based on a chosen feature, an example data set for this can be seen in Figure 1.1. This feature is chosen from a random subset, based on its ability to subdivide the target into pure groups. An example of a split can be seen in Figure 1.2, where the features are the first four columns and the target is the fifth, orange, column and the samples are the rows.

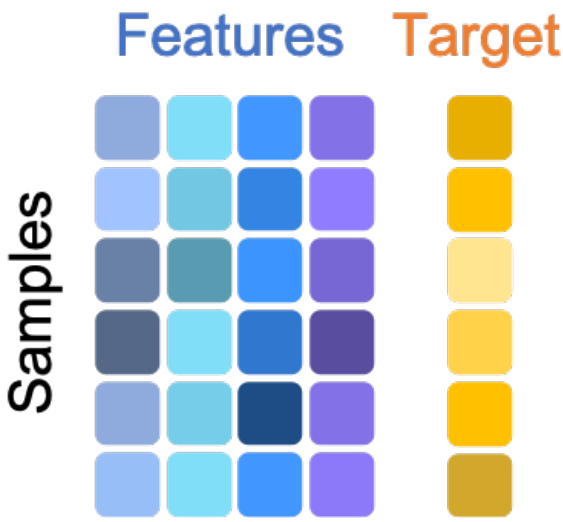

Figure 1.1: This figure shows the general layout of data for machine learning algorithms. Separate samples have measurements for multiple features that can be used to predict a target.

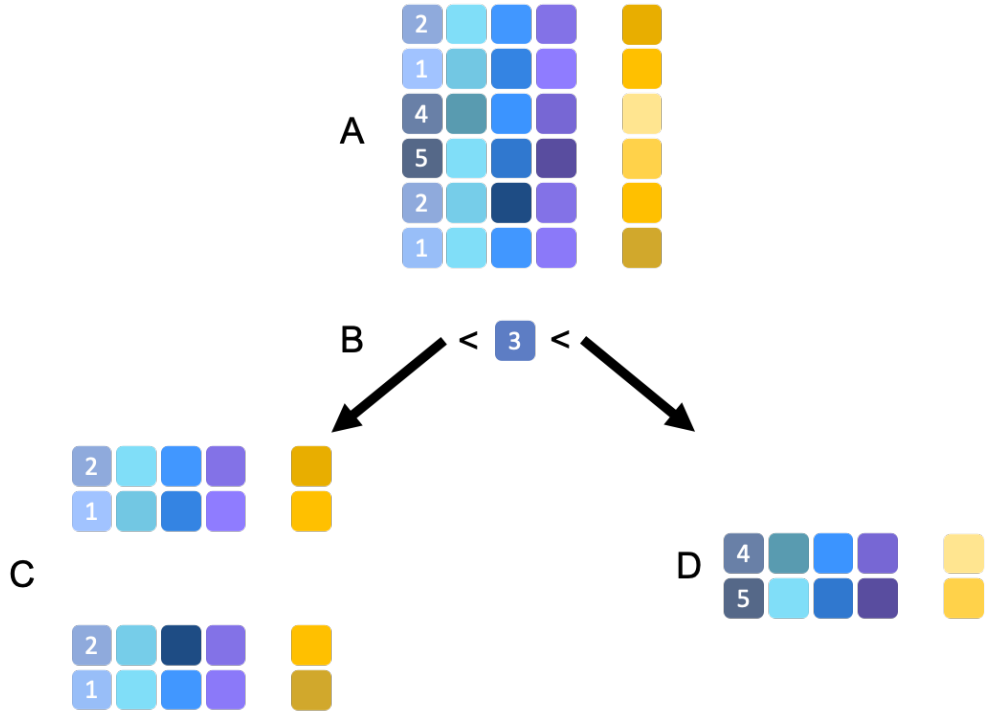

Figure 1.2: In a decision tree the most basic unit of model creation is a single decision that splits a single node. A) The parent node of the split contains all of the samples (and each of there features and targets) available to the split. B) The feature for the split is chosen to be the first column, and the split value is chosen to be 3. C) The Samples that have a value less than the split value for the feature move to the left child. Since the two middle samples have values for the feature greater than the split value they are not present on the left. D) The samples not present in the left child node are present in the right node.

The most basic unit of any decision tree based algorithm is a decision split (see Fig. 1.2). These splits make decisions based on the data to partition the samples from the source node into children nodes. This process is repeated on the children nodes until a stopping parameter is reached. The final resulting children nodes are considered the leaf nodes of the decision tree (see Fig. 1.3).

To make an effective learner, the model looks at all possible features in a given node to find the feature and split value that optimizes some cost function, whether that be GINI, MSE, or something else. In this regard, a single decision tree is a deterministic algorithm; repeatedly given the same data and parameters, it will produce the same result every time. Unfortunately, this leads a single decision tree to overfit to the training data. This was accounted for by Breiman with the introduction of bagging, the process of combining multiple weaker models into a single stronger model[3], and altering each of the individual trees by re-sampling the samples with replacement. This, combined with the addition of subsampling the features to consider at each possible split, resulted in the creation of the Random Forest algorithm[31]. As shown in Figure 1.4, a Random Forest is a collection of decision trees, where each decision tree looks at a subset of the samples, chosen with replacement to allow for duplicates, and each split considering a random subset of features.

Explainability of Random Forests

Random Forest provides a robust method for feature selection [32] by providing a feature importance score for each independent variable used in the model. These feature importances are calculated by summing the contribution of their splits. While there are several cost functions used to determine the best splits, one of the more common methods for classification trees is Gini[31, 33], which calculates the impurity improvement from parent to child nodes. Gini Impurity for a single node is calculated from the proportion of a given class within the total number of samples. If the proportion of category a (p_a) is defined as $p_a = n_a/n$, where n is the total number of samples and n_a is the number of samples in category a, the Gini impurity G for two classes (a and b) at a given node (i) can then be calculated by:

$$G(i) = 1 - p_a^2 - p_b^2$$

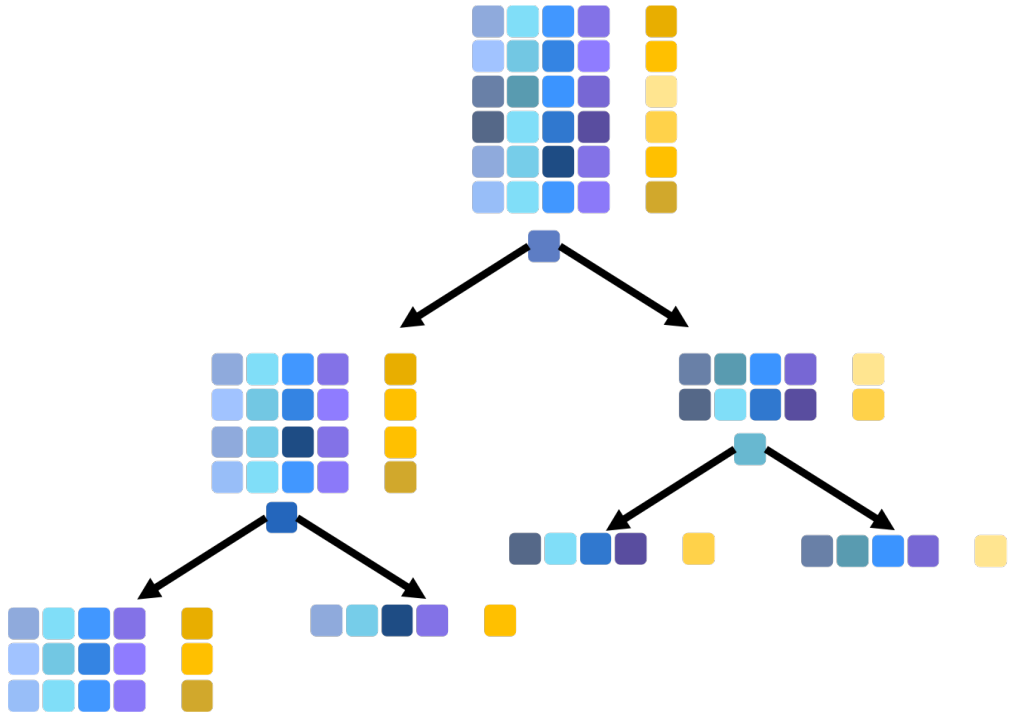

Figure 1.3: An example decision tree that starts with 6 samples at the root node and splits them into 4 leaf nodes with 3 total splits.

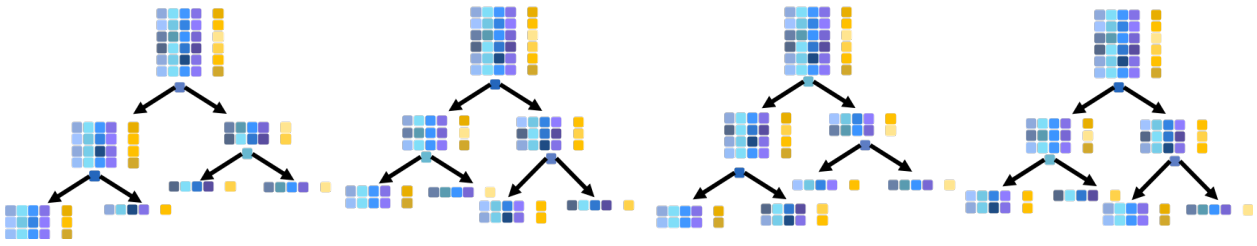

Figure 1.4: An example of grouped decision trees that, when taken in aggregate, form a forest. To be a Random Forest, each tree needs a random subset of the samples, and each split needs to look at a random subset of features to split upon.

where p_b is the proportion of samples within class b. To calculate the improvement to Gini impurity ($\Delta G(i)$) by a split:

$$\Delta G(i) = G(i) - p_l * G(i_l) - p_r * G(i_r)$$

where i_l and i_r are the left and right children of node i and p_l and p_r are the proportion of samples that go to the left or right children nodes from the original node.

For regression problems, one of the common cost functions/importance scores is variance explained. The simple version of this score can be thought of as the difference between the variance as defined by:

$$\sigma^2[X] = E[X^2] - E[X]^2$$

or

$$\sigma^2(i) = \Sigma(x - \hat{x})^2/n$$

where X is a random variable and E is the expectation function. In the node i and the variance in the two children nodes (i_l and i_r) as defined by:

$$\Delta\sigma^2(i) = \sigma^2(i) - \sigma^2(i_l) - \sigma^2(i_r)$$

where the feature for the split is chosen by maximizing $\Delta\sigma^2(i)$. And, as with the classification, the importance of any given feature is the sum of this variance explained for all splits that use that feature.

Iterative Random Forest

Using feature importance, it is possible to address an issue that can occur in Random Forest, namely the random selection of features at each node can cause a majority of the splits to only consider non-causative features. A solution to this is to weight the random feature selection to favour a subset of features that are more likely to be causative. One way to determine which features are likely to be more causative is to use the feature importance from a Random Forest model. By turning this into an iterative process, with each new

iteration using the importance scores from the previous iteration as feature weights, a new algorithm called iterative Random Forest (iRF)[4] was developed.

The iterative process results in more efficient decision trees and more organized decision paths. These organized paths can be mined to find sets of features that co-occur often in the same decision path with an algorithm called Random Intersection Trees (RIT)[34]. With a version of iRF designed to run efficiently on high performance computing (HPC) machines, or supercomputers, it is possible to run on larger and more complex data sets[9].

1.3 Genomics and Phenotypes

The central dogma of genetics states that genetic information is encoded as DNA, which is expressed as RNA, which is then used to produce proteins. A single genotype is an individual in a species with a unique DNA sequence. A phenotype is some measurable quality (quantitative or qualitative) of the organism, such as height of a tree, eye color of a person, number of toes on a cat, RNA expression profile of a fruit fly, etc. While a genotype can be measured with as little as a single cell with some techniques[35], phenotypes have a wide variety of ways they can or need to be measured, with some methods being passive (no interference with organism) and some being destructive (completely destroying the organism in order to measure a specific phenotype). Despite the fact that the central dogma does not account for non-coding regions of the genome, as they do not eventually become proteins, it is possible for genetic differences in those regions to effect a given phenotype[36].

To build an understanding of the relationship between genotype and phenotype, the connection has to be isolated in some way. This usually takes the form of gene knockout/knockdown [37], with a process like CRISPR[38], to isolate the phenotypic function of a specific gene or gene-set, or Genome Wide Association Study (GWAS)[39], that tries to determine which gene, or region, affects a particular phenotype. While linear models are the traditional tool for GWAS, other algorithms, such as iRF and other ML models, can provide improved results by considering non-linear effects.

1.3.1 Genomic Data

One of the most important concepts regarding genomic data is its relationship to characteristics of the individual. In the most simple of terms, this can be done by analyzing variance in phenotypes and assigning that variance to different causes[40]. We know that in general, the phenotype (P), or trait, being measured can be defined by the equation:

$$P = G + E + G * E$$

where G is the genotypic effect, E is the environmental effect, and $G * E$ is the effect of the interaction between the genotypic effect and the environment[41]. The field of quantitative genetics is focused on better understanding this relationship and its values for various species and their phenotypes and how to use populations to better understand the effect of the genetics[42].

One population effect of genetics is any population under specific conditions, such as no evolutionary pressure, will follow the Hardy-Weinberg Principle and will have allelic distributions that are perfectly determined by their allelic frequencies[43]. An allele is a specific DNA code at a specific location in the genome. These studies, such as the development of Quantitative Trait Loci mapping (QTL-mapping)[44], next-generation sequencing, single nucleotide polymorphism (SNP) chips, and genotyping by sequencing have led to the identification of several genes of wheat (*Triticum aestivum*) that are related to resisting spot blotch. Spot blotch is important for agriculture because it can lead to loss of grain quantity and quality[45]. We can see the effects of long term phenotypic selection by analyzing the genetics of the modern cultivars of wheat. The malting quality of grains were selected to produce these cultivars, which was possible due to the relationship between certain grain genes and certain malting traits[46].

Genome Wide Association Studies (GWAS) find statistical connections between genotype variations and phenotypes[39]. A substantial amount of work has gone into optimizing these calculations to keep up with the amount of data increase that comes with improved technology[47, 48]. Most GWAS analyses scale in cubic time with the number of samples

but linear in terms of the number of features, or SNPs, as seen in the big O notation:

$$\mathcal{O}(n^3 + pn^2)$$

where p is the number of single nucleotide polymorphisms(SNPs) and n is the number of individuals who were sampled. According to the authors of GEMMA, the only step that is cubic in nature is determining the relationship of individuals[48]. Unfortunately, when these methods are used to try to detect epistasis, where multiple SNPs effect the same phenotype in a non-additive fashion, the computational expense increases exponentially with each order of interaction, and the above equation becomes:

$$\mathcal{O}(n^3 + p^k n^2)$$

where k is the order of interaction. To feasibly detect high-order interactions, machine learning techniques such as iterative Random Forest (iRF) [4, 9] and Random Intersection Trees (RIT) [34] can be used at a reasonable computational scale because they do not have an exponential term that scales with the size of the interaction to be detected. By using these data-driven X-AI models, the mechanistic understanding of an organism can be accelerated by providing high throughput hypothesis generation models. This human-in-the-loop process can be used to design more efficient experiments to accelerate scientific findings[49].

1.3.2 Epistasis

Epistasis is defined to be any non-additive genetic component or non-multiplicative component on the penetrance scale[50], but, in general, epistasis can be described as inter-genetic interaction, or as defined by G.A. Churchill as a:

> ... genetic phenomenon that is defined by an interaction of genetic variation at two or more loci to produce a phenotypic outcome that is not predicted by the additive combination of effects attributable to the individual loci.

These effects can be qualitative or quantitative, discrete or continuous, and can proved insight into how a given species evolved in relation to the particular phenotype[51].

Another way to look at epistasis is that genes rarely work in isolation; most gene products work with other gene products in biochemical pathways. These 'physiological genetic effects' can be significant contributors to an individual but generally cause only small variations to 'statistical genetic effects', or to a population wide effect. These effects are often only additive changes, on the whole, but gene products do interact in non additive ways. In terms of evolutionary population differentiation, gene interactions can be significant[52]. These epistatic interactions may dictate the interaction of complex molecular pathways and can be modeled onto protein structures to determine their specific molecular function[53]. These same interactions can be analyzed to understand their functional nature, and how that interaction has come to be over time through evolution.

In fact, functional synthesis of molecular evolution can be used to track the evolution and consequences of epistasis[53]. Epistasis helps define and structure functional pathways of gene products and is used by evolution to shape certain phenotypes[54]. Epistasis is shown to have an effect on various human diseases[55] including hypertension[56], diabetes[57], Alzheimers[58], coronary heart disease[59], colon cancer[60], and Crohn's disease[61], as well as plant phenotypes such as flowering timing of soybean[62]. Epistasis has also been shown to have a positive effect in the overall growth of some plants[63].

The basic effect(y) on a phenotype from two loci (labeled 1 and 2), including additive and dominance effects, can be defined with the equation:

$$y = \mu + a_1 x_1 + d_1 z_1 + a_2 x_2 + d_2 z_2$$

when including epistasis the equation becomes:

$$y = \mu + a_1 x_1 + d_1 z_1 + a_2 x_2 + d_2 z_2 + i_{aa} x_1 x_2 + i_{ad} x_1 z_2 + i_{da} z_1 x_2 + i_{dd} z_1 z_2$$

where μ is the population mean effect, a is the coefficient of the additive component of the loci x, d is the coefficient of the dominance component of the loci z, and the four i each represent an epistatic interaction between the two loci[64, 50]. From this equation we can see that epistasis can interfere with traditional QTL mapping by masking or overriding additive

signal[54]. We can also see how complicated of a task it is to be able to detect one of these many types of epistasis between all pairs of loci.

Classic Epistasis Detection

The most simplistic approach to detect epistasis is to evaluate the relationship of every set of SNPs for a two-way relationship to the phenotype, independent from the individual effects from each of the SNPs[65]. However, because of the size of genomes and the exponential scaling of checking all possible sets, this quickly becomes an infeasible task[66]. The potential solution space of all pair-wise relationships grows exponentially with the size of the genome[54]. As more large genomes are sequenced with high quality tools, specialized techniques are needed to effectively search for and evaluate them for epistatic effects. The majority of additive genetic effects are small and require large sample sizes and robust sequencing to appropriately identify the effects. The same is true for interactive effects, but likely even more samples are needed than traditional techniques to accurately assign effects to epistasis[67], further reinforcing the need for specialized methods. The most common initial solution is to implement data reduction and pattern recognition[68].

A classic approach to epistasis is to do combinatorial partitioning, where subsections of the genome are evaluated separately and then their results combined[59]. This is basically the concept of limiting the search space to smaller chunks which are easier problems to solve. However, in the reduction of search space the potential exists to remove correct answers. Some methods look for SNPs that have conditional effects on a phenotype with regard to another target SNP[69]. These function by effectively making a new phenotype that is preconditioned on the given SNP and analysing the association of all other SNPs with this new phenotype. This is feasible if you have prior knowledge of a causative SNP, but would require an exhaustive search of all pairs to find both individuals of the epistatic pair. Some methods reduce the processing required by organizing the problem into two separate problems: with-in gene epistasis, where alleles in the same gene have a non-additive effect on a phenotype, and cross-gene epistasis, where alleles in different genes display epistatic interactions[70].

Some approaches simply focus on scaling the calculations and optimize the traditional brute force approach[71]. These brute force approaches are often designed to be more efficient when detecting multiple phenotypes[72] for the same genetic data set. Another more analytical approach looks for statistical interactions between loci, as evolutionary pressure on epistatic effects should affect the distribution of alleles[58]. An approach to high throughput epistasis detection can be done by strictly filtering the SNPs and the search space prior to analysis, but this can severely reduce the ability to detect small effects[73].

"BOolean Operation-based Screening and Testing" (BOOST) is an algorithm that optimizes applied traditional epistasis detection specifically in case-control tests by optimizing the problem into Boolean operations that are highly optimized on modern computers[57]. Backward genotype-trait association (BGTA) is another feature reduction algorithm designed for case-control studies that removes uninformative markers[74]. Multifactor-dimensionality reduction (MDR) reduces the search space to only include loci likely responsible for interacting with known alleles associated with disease, as it is designed for identifying human susceptibility to disease, specifically interactions among estrogen-metabolism genes in sporadic breast cancer[75]. SNP-SNP filtering algorithms, such as MDR, reduce the space needed to search for epistatsis and can be ensembled together to create an even more powerful filter[76]. Polymorphism Interaction Analysis (PIA) accounts for imbalances in both SNP frequency and phenotypic balance (i.e. more controls than cases in terms of clinical phenotypes) [60]. Restricted partition method (RPM) partitions the genome to find interacting genes or gene-environment interaction[77]. By treating epistasis detection as an optimization problem, efficient algorithms from other fields can be utilized[78]. These methods are often evaluated and compared against each other[79].

Machine Learning for Epistasis Detection

In general, ML approaches can help reduce the search space for genetic interactions, such as in human disease resistance and susceptibility, which can be due to epistasis[55]. One such technique, MegaSNPHunter[80], focuses on feature space reduction by appling gradient boosted[81] trees[30]. Some methods are built on top of traditional methods, such as a synthetic approach for testing epistasis with MDR while incorporating various heritabilities

and imbalanced data sets[82]. Another model built on top of MDR includes transforming the data through techniques like principle component analysis (PCA), "MB-MDR for Structured Populations" (Model-Based Multifactor Dimensionality Reduction-Principal Component (MBMDR-PC))[83]. These techniques rely on the effectiveness of the principle components in capturing the population structure in order to effectively isolate potential epistatic effect.

Linear models can be used to detect epistasis by applying a regularization penalty. One machine learning approach to epistasis discovery relies on using a pairwise encoding of SNPs and then using a linear model L1-regularized (LASSO[84]) on the new encodings. This allows for the complex interactions of epistasis to be simplified to an optimized linear model[70]. A similar approach can be done with L2 penalized logistic regression to detect gene interactions[85], with potentially different results.

Other machine learning methods like Random Forest can be used to detect important SNPs in predicting phenotypes[86] and can also be used to detect genes interacting with other genes or environmental factors when predicting a phenotype[87]. Work in this area has lead to several optimized code bases, including scaling to High Performance Computers[9]. A fast random forest focused on GWAS studies of complex genetic disease, called random jungle[61], has also been developed. A genetic algorithm combined with an ensemble of various classifiers iteratively produces subsets of SNPs to test for interactions and phenotype associations [88]. Neural Networks can be designed to identifying epistasis effects[89]. Other methods include Bayesian inference to detect epistasis[90], Monte Carlo logistic regression to discover epistasis [91], and network guided epistasis detection [92].

1.3.3 *Populus trichocarpa* data

Populus trichocarpa, colloquially known as black cottonwood, is a poplar tree indigenous to the Pacific Northwest of North America. As a main biofuels research plant with a fast growth rate and moderate lignin content, it has been studied for several decades, including a full genomic analysis[93]. Funded by BESC[94] and its successor CBI, both BioEnergy Research Centers, a diverse set of genotypes were collected from nearly the entire species' geographic range and a wide variety of research has been done.

Over 1300 genotypes have been propagated to multiple common gardens in different climates within the natural range of the species and sequenced for association and population studies[95, 96]. At these field sites, a randomized block design was used to spread three replicates of each genotype throughout each of the common gardens. Over the course of several years, these common gardens were monitored and several phenotypes were measured. Externally measured phenotypes like height, diameter (measured as diameter at breast height[DBH], or as diameter at 20 cm(D20) for young trees), bud set, bud flush, presence of diseases (like Melampsora[97] and Septoria[98]), and microbiome on and around leaves, xylem, and roots(Plant-Microbe Interactions (PMI)). Other phenotypes that require destructive sampling were also measured for a few of the common gardens including density, total biomass, gene expression profile of leaf, xylem, and root, metabolite profiles, and sugar content. Many Genome Wide Association Studies (GWAS) have been completed with these data. These include, but are not limited to, wood composition studies[99], other biomass traits[100], and adaption to different climates[101].

1.4 Genomic Selection

Genomic selection (GS) is the process of selectively breeding a species to adjust one or more phenotypes at the population level in a desired direction with the aid of genetic data. The most basic form of trait performance improvement, having been used for hundreds if not thousand of years, is phenotypic selection[102]. This process involves taking the best performers, or individuals with the most desirable characteristics, and having them breed with each other, sometimes called crossing. Over the course of generations, this has resulted in the domestication of crops and livestock[103]. Gregory Mendel proved the process of inheritance of physical traits and showed a way to predict a phenotype of offspring based on the same phenotype in the parent. This included concepts such as dominance, where one gene masks another recessive gene. However, more complex phenotypes, such as height, rely on much more information than can be captured in a simple phenotypic model. Capturing more information by incorporating relatedness, such as full-sibling vs. half-sibling relationships, allows for a better understanding of the genotypic contribution to the phenotype[104].

With the advent of cheaper sequencing technology, this relatedness, structured as a relationship matrix, can be updated to use exact genetic distance. Heritability is the proportion of phenotypic variance that can be described by genetic inheritance and variance, usually measured by matching genetic relatedness to phenotypic relatedness. Heritability only describes the specific population that it was measured in, but is generally very similar across like populations[105]. The application of quantitative genetics relies on several assumptions, including: the phenotype is responsive to genotypic variation, the phenotype of continuous traits rely on a small effect from several Mendelian gene effects, the phenotype data has had appropriate accounting for genetic by environment effect, and the phenotypes of the progeny has been evaluated in the same way as the original population. Depending on the interconnectedness of the genetic effects, the power to improve multiple traits at once can be significantly hampered[106].

Plant breeders generally follow the process of: setting an objective for improvement, assembling genetic variance for measuring and breading, selecting crosses, evaluating the new generation, and, finally, releasing the cultivars or genotypes that improves the selected objective. Specific methods and techniques to follow this process depend on the restraints of the problem, such as specific parameters of mating, the genetic variability, and the available breeding material[41, 107]. The combining ability of individuals within a species is critical for improvement of plant breeding[108]. Some plant species that have have benefited from genomic selection include oil palms[109], and pearl millet[110], which where selected for yield and heartiness. While different specific processes may be used for animal breeding, similar technologies and evaluation techniques as plant breeding are applied[111].

1.4.1 Progeny Trials

The quality of genetic material a specific genotype would pass on to its progeny and descendants is often the most important aspect to consider for genomic selection. A progeny trial is designed to specifically measure the genetic contributions of the parent to its children. These trials can be designed in several different ways, depending on the ability of the parent to self-cross and the expected contribution of reciprocal crosses[112]. Different types of mating trials optimize for different objectives including optimizing specific traits or identifying

specific genotypic or familial qualities[113]. This contribution is generally defined in two parts, specific combining ability (SCA) and general combining ability (GCA) [114].

GCA refers to the value of the specified parent's contribution to the genetic effect responsible for a phenotype in its progeny, given a random partner from the species. SCA is the effect explicitly occurring because of the mating of two specific genotypes, isolated from those parents' individual GCAs[115]. Some species can have SCAs greater than GCAs, especially if they have high Heterosis. Heterosis is the trait of certain cross-pollinated crops that are highly heterozygous, that causes better phenotypic performance with greater heterozygosity within the genotype. This leads to inbreeding depression, due to the creation of homozygotes, within the population if trying to isolate specific genotypic effects with self-crosses[116].

Better understanding of how heritable specific traits are and how much variance within the trait is due to genetic variance can lead to improvements in predictions and biological understanding[117]. By incorporating genetic variables with traditional variables, such as environment and management, breeders were able to build a model that explained 88% of the variability in time to first flower in a population of the common bean, *Phaseolus vulgaris* [118].

1.4.2 Traditional Genomic Selection

Traditional genomic selection focuses on determining what parents will most effectively improve a phenotype within the population for subsequent generations while maintaining genetic diversity[119]. This results in focusing explicitly on determining the GCA, which is composed mainly of the additive genetic effects contributing to a phenotype. A rank order of genotypes that have a high genetic value is produced and the top genotypes on the list are crossed together to produce a new generation. For practical purposes, this means that the total number of crosses needed for a selection of parents increases quadratically with the number of parents. However, by limiting the number of different parents, genetic diversity falls off sharply.

Most common genomic selection techniques are methods that use linear algebra to determine the breeding value of each separate genotype. One of the most commonly used

methods that incorporates genetics is Genomic Best Linear Unbiased Prediction (gBLUP), that uses the genetic relationship of individuals in the population to determine the value each genotype provides to the phenotype in question[120]. A similar method that assigns value to specific SNPs is Ridge Regression[121] Best Linear Unbiased Prediction (rrBLUP) [122]. Identifying specific regions of the genome, or quantitative trait loci (QTLs), that control specific responses to stress, can be used in marker-assisted selection (MAS), improving the speed at which breeding programs can improve stress tolerance or other phenotypes[123].

Application of population genetics, with understanding of quantitative genetics, has been applied to hatcheries of pearl oysters, improving several important commercial traits across the entire population. Molecular genetic techniques were used to target specific genes to provide improvements faster than traditional selection with less risk of inbreeding[124]. MAS has been used to improve barley breeding for several traits, specifically for brewing and yield[125]. QTL mapping of rice stress response has lead to improved crop stability and greater yield[126].

1.4.3 Machine Learning for Genomic Selection

Machine Learning and Artificial Intelligence have a wide variety of uses including the identification of recent evolutionary changes[127] and have been applied to the process of genomic selection[6]. A multilayer perceptron (MLP), a type of neural network, has been shown to be more effective in the genomic selection of wheat and jersey cow over a Pearson's selection model, with each data set providing greater improvement for different phenotypes[128]. Across multiple data sets in wheat, non-linear algorithms, such as various AI models, generally performed better in prediction accuracy than comparable non-linear algorithms[129]. However, a study in maize showed that some of the same AI models only showed a slight improvement over a well-tuned linear model[130].

1.5 SARS-CoV-2 Background

The Covid-19 pandemic has cost the lives of over one million people in the US alone (according to the CDC Covid Data Tracker https://covid.cdc.gov/covid-data-tracker/#datatracker-home), and will likely affect even more as long-term effects are evaluated [131, 132]. The virus responsible, SARS-CoV-2, has been sequenced across the time-course of the pandemic, providing valuable data that could be used to build a real-time mutational history. A strong understanding of the mutational changes would help identify patterns between viral characteristics and symptoms, as well as help predict vaccine escape and severity of new strains. Ongoing research and observation is required as new lineages with variant-of-concern-like mutations continue to be found[133].

Evolutionary time between sequences allows mutations to build up that can be used to distinguish individuals from the same species or species from their ancestors and descendants. However, in the case of SARS-CoV-2 sequences, the difference caused by individual mutations can be tracked over the course of weeks, rather than years or decades. The level of fine grained data can bring to light some of the assumptions that state-of-the-art algorithms make when trying to elucidate the lineage of a virus and the mutations they may contain. The modern calculation of phylogenetic trees can miss subtlety in the data without curation, that becomes increasingly difficult as the size of data increases [134]. The tree structure forces the assumption of purely divergent evolution where different evolutionary paths cannot cross back to each other via convergent evolution or recombination. Network models are better able to include evolution that may include homoplasic loops, an evolutionary pattern implying convergent evolution or mutation transfer between diverged individuals[135].

1.5.1 Structure and Variants

The SARS-CoV-2 virus falls within the viral classification of coronaviruses, which are single-stranded RNA viruses with positive polarity[136]. The genome varies from 29.8kb to 29.9kb across variants and contains the orflabpolyproteins and structural proteins characteristic of coronaviruses. The virus also contains six accessory proteins encoded by ORF3a, ORF6, ORF7a, ORF7b, and ORF8 genes[137].

RNA viruses mutate faster than DNA viruses and single stranded viruses mutate faster than double-stranded[138], leading SARS-CoV-2 to be expected to have a relatively high rate of mutation. At $1.1 - 6.2$ x $10^{-}3$ mutations per site per year, SARS-CoV-2 has a higher mutation rate than the seasonal influenza at $0.6 - 2.0$ x $10^{-}6$[139]. With a higher mutation rate comes a higher quantity of distinct variants with distinct characteristics. Many of the early variants were noted to contain mutations specific to the spike protein, which is a 1,273 amino acid trimeric glycoprotein responsible for virus entry into host cells[140]. The specific changes and mutations to the spike protean may be under different selective pressures[141].

After the early spike protein variants, other variants quickly arose, such as the B.1.351 variant in South Africa and the P.1 variant in Brazil. The pango network has been established as one of the accepted naming schemes for distinct lineages of SARS-CoV-2 and provides information about the distinction between lineages as well as specific variants of concern, such as B.1.1.7 and P.1[142].

1.5.2 Template Switching and Recombination

A specific concern for SARS-CoV-2 is whether or not recombination is the cause for any new variants, because it is known to generate diversity, and create new opportunities to overcome selective pressure and adapt to new hosts and environments. Recombination happens when multiple variants infect the same cell and exchange genetic information, with different types of recombination based on the structure of the crossover site[143]. Copy-choice recombination is caused by template switching, where the transcription complex leaves the donor template and resumes synthesis on the acceptor template[144]. HIV has been shown to use template switching to facilitate recombination across various strands, and if the possibility of multiple template switching is ignored the estimated recombination rate undershoots the measured rate[145]

While the standard phylogenetic techniques ignore recombination because it interferes with divergent evolution and mapping lineages to a tree structure, there has been other evidence reported pointing to the presence of recombination between lineages even from the earliest stages of the pandemic [146]. There has also been evidence that some of the lineages from that time are the result of recombination[147] and that recombination likely caused

some of the spreading of SARS-CoV-2 at the beginning of the pandemic[148]. This early and persistent presence of recombination is potentially caused by strong evolutionary pressure on different regions of the virus's genome[149]. The high prevalence of multiple divergent lineages leads to a more likely scenario of co-infection that can lead to template switching, leading to the most virulent strains with the greatest likelihood of spreading[150]. There has also been evidence to show that recombination has specifically transferred evolutionary mutations within the spike gene region across different lineages[151]. A better understanding of the causative process of recombination could help fight the pandemic[152]; by increasing the rate of mutational accrual within SARS-CoV-2, the virus could face a rapid extinction, by losing necessary components to function, due to the new mutations occurring too quickly and interfering with the normal process of virus[153].

1.5.3 Evolutionary Incompatibility

Current lineage analysis relies on phylogenetic trees that expect divergent evolution away from the original species/sequence. Recombination and convergent mutations are seen as nuisances[154] in this type of model because they interfere with the assumed tree structure. For short time scale evolution in organisms where recombination is possible, these methods can ignore events that are potentially important for tracking evolutionary and functional changes.

Evidence of non-divergent evolution can be found through evolutionary incompatibility between SNPs. A refined incompatibility score is defined as a number of homoplasies in the population between two sites[155], that can only be explained by recombinations, recurrent mutations, or convergent mutations in the population. A quick equation for this can be written as:

$$i(X_i, X_j) = l(X_i, X_j) - (|X_i| - 1) - (|X_j| - 1)$$

which effectively counts the number of total combinatoric states between the two sites (X_i and X_j) not including the ancestral state($l(X_i, X_j)$) minus the total number of individual states at each site minus their ancestral state ($-(|X_i| - 1) - (|X_j| - 1)$) as defined by Bruen

et. al. [156]. Modern phylogenetic tree construction assume perfect divergent evolution even when trying to maximize parsimony[157].

1.6 Conclusion

The sheer amount of possible solutions to many of the problems in genetics can not feasibly be solved by purely linear algorithms. The size of data and the complexity of problems, such as epistasis, suggest that ML models are likely well suited for this field. Many machine learning algorithms are specifically designed to scale more rapidly to meet the needs of large scale data, especially as the linear solutions require brute force techniques for calculating combinatorics, like conditional relationships. These methods can be applied to a variety of use cases for problems across biology. The application of the correct and most valuable method for a problem is just as important as the creation of new methods for the field.

Chapter 2

Genome Wide Epistasis Study with Iterative Random Forest

2.1 Abstract

The relationship between genome and phenotype can be very complex. Most algorithms are designed to find the linear relation between individual SNPs and the phenotype, while ignoring any epistatic effects because they are computationally expensive and are not guaranteed to be present for every phenotype. Here I show that machine learning algorithm Iterative Random Forest (iRF), with several new metrics, can efficiently find epistatic relationships. These new metrics mine the model for information already contained within the results, but which previously lacked transparency. The metrics are evaluated using a synthetic data creation process that provides conditional relationships. The metrics are then tested on the height phenotype for *Populus trichocarpa*, a complex phenotype of biological and biofuel significance thought be partially caused by epistatic interactions.

2.2 Introduction

In the most simplistic view of biological association to a physical trait, the expression of a single gene is directly responsible for changes in the trait. However, biology is not so simple, and many different aspects may influence any given trait. Epistasis, one such aspect, can complicate finding the associations between Single Nucleotide Polymorphisms (SNPs) in DNA and the external traits (phenotypes) that they effect. Epistasis has been assumed to be any non-additive genetic component or non-multiplicative component on the penetrance scale[50], but, in general, epistasis can be described as inter-loci interaction or any effect on a phenotype from multiple loci that is not attributable to their additive components. These relationships can complicate the detection of causation with methods not designed specifically to account for them. Conversely, the development and utilization of methods that account for, and gain value from, epistatic interactions can improve the understanding of gene to phenotype relationships.

Genome Wide Association Studies (GWAS) traditionally use linear models to find statistical connections between genotype variations and phenotypes[39], however they tend to lack the ability to find non-linear interactions, such as epistasis. Work has gone into

optimizing these GWAS calculations to keep up with the amount of data increase that comes with improved technology[47, 48]. Most GWAS analyses scale in cubic time with the number of samples but linear in terms of the number of features, as seen in the big O notation, $\mathcal{O}(n^3 + pn^2)$, where p is the number of SNPs and n is the number of individuals who were sampled. Unfortunately, when these same methods are used to try to explicitly and exhaustively detect epistasis, where multiple SNPs effect the same phenotype in a non-additive fashion, the computational expense increases exponentially with each order of interaction, and the above equation becomes $\mathcal{O}(n^3+p^kn^2)$, where k is the order of interaction, or the number of SNPs working in relation with each other in a non-additive fashion.

To feasibly detect high-order interactions, machine learning techniques such as iterative Random Forest (iRF) [4, 9] and Random Intersection Trees (RIT) [34] can be used at a reasonable computational scale. As non-linear and non-exhaustive methods, they do not have an exponential term that scales with the size of the interaction to be detected, but instead scale only with data size and algorithm parameters. These algorithms fit into the category of Explainable Artificial Intelligence (X-AI), where the internal decision making mechanisms are human-understandable.

Here I show how iRF, already efficiently scaled[9], can be a useful tool when detecting epistasis in genetic data. By understanding the underlying mechanisms of iRF [4], we can both develop new ways to process the results to find more information from feature importance and better select for interacting sets involved in epistasis. To show the robustness of iRF's ability to detect epistasis, a validation approach has been designed. The process uses synthetic genetic data with generated phenotypes to ensure the effect detection of iRF has an exact truth set to be compared against. Some of the information that will be gained from these new techniques include conditional feature importance, sets of features that are more important in conjunction than independently, and features that have overlapping importance (such as the case with SNPs in linkage disequilibrium). Other provided information surrounding the feature importance such as an effective cutoff for identifying causative features and a method to determine the effect the feature has on the target. The GWES methods are then used on genetic data with phenotypes determined to have epistasis, namely

the height phenotype from *Populus trichocarpa*, to find potential epistatic interactions for a phenotype of known biological interest.

2.3 Methods

2.3.1 Iterative Random Forest

The Iterative Random Forest method is based on Random Forest, as published by Leo Breiman in 2001[31]. Breiman designed Random Forest to build on his earlier work with decision trees, CART (Classification and Regression Trees)[30]. A single decision tree subdivides a data set's samples using discrete features, to best split the samples' dependent (target) values. A sample data set is shown in Figure 1.1 and an example decision split is shown in Figure 1.2. The tree model looks at all possible features at each split point (node) to find the feature and split value that optimizes some cost function, whether that be GINI for classification, MSE for regression, or some other cost function for the purity of the set of targets. Multiple trees when trained together form a forest, however the standard decision tree is deterministic, so a forest of decision trees would contain only replicates. Breiman developed Random Forests to solve the deterministic nature of an individual tree and improve the overall value of the model by only allowing the splits for each node to consider a random subset of the full features. Each tree is also provided with a distinct subset of the original samples by re-sampling the data set with replacement. Figure 1.4 shows an example Random Forest.

From this method, not only is a trained learner developed, but the feature importance scores for each feature are also provided. These importance scores are calculated based on how often and how well each feature was used throughout the forest to subdivide the samples. Using feature importance it is possible to address an issue that can occur in Random Forests, where the random selection of features at each node can cause a majority of the splits to only consider non-causative features if non-causative features significantly out-weigh causative features.

A solution is to weight the random feature selection to favour a subset of features that are more likely to be causative, and one way to determine which features are likely to be more causative is to use the feature importance from a Random Forest model. By turning the method into an iterative process, with each new iteration using the importance scores from the previous iteration as feature weights, a new algorithm called iterative Random Forest (iRF) was produced[4]. This results in more efficient decision trees and more organized decision paths. The decision paths contain embedded information on feature interaction because, as each sequential decision is made, the selection process for the node is conditioned on every decision before it in the decision path[4]. The conditionality implies that features that occur within the same decision path have an interaction within the model. An algorithm called Random Intersection Trees (RIT)[34], mines these organized paths to find sets of features that co-occur often, offering insights into possible interactions between features.

While it is extremely useful to pull out these sets from iRF, as shown by Basu et. al., the trained model contains even more information about the relationships between features, as well as between features and the target. It is possible to decompose every tree in the forest into its component parts, the nodes and their splits, allowing the derivation of what the model is considering at a more in-depth level than traditional summary statistics like feature importance or co-occurring feature sets.

2.3.2 iRF Metrics

The main metric used to when discussing explainability in iRF models is the importance score, calculated for each feature in the model. The importance score is most often presented as either the sum of the splitting criteria used for that feature averaged across the trees in the forest, or the normalized version of that value (i.e. the proportion of that value across all features in the model attributable to that specific feature). For iRF models that use variance explained as the splitting criteria, normalization provides the proportion of total variance explained by a particular feature within the overall model.

Evaluating Feature Importance Scores

The most basic decision made by the model is the split within a node. At each split, the model has chosen a feature and split value for that feature that breaks the samples in the current node into two separate children nodes. The split can be defined as a step function:

$$f(x) = \begin{cases} L & x < S \\ R & x \geq S \end{cases} \tag{2.1}$$

where x is the value of the feature, S is the split value, L is the average phenotype value of the samples from the parent node that had a value in x less than S, and R is the average phenotype value of the samples from the parent node that had a value in x greater than S.

For each feature, all splits of that feature are in proportion to the number of samples handled by that split and can be combined into a single function that contains the direct relationship from the given feature to the target within the model:

$$F(x) = \Sigma f_i(x) * w_i \tag{2.2}$$

where $F(x)$ represents the feature's full contribution to the model, $f_i(x)$ is a split from eq. 2.1 with i representing the specific split using the variable and w_i is the weight based on the number of samples effected by that split.

The resulting equation is a combined series of step functions which are sampled at equal distances for the whole range that the feature has splits on within the model, being sampled a number of times equal to the number of unique split values plus one. This sampling produces a single line for the entire relationship and is used to build the linear fit, determining both a slope (or relationship between feature and phenotype) and p-value for the linear fit (showing how well the linear fit captures the information in $F(x)$). The linear fit is calculated using the linear regression package scikit-learn[158] to produce the underlying linear equation[159].

This package provides an R^2 and p-value that determine how well the linear function actually represents the underlying data.

This information is used to define a new metric for iRF called feature effect. It provides both the overall direction of effect that the feature has on the target within the model and provides an appropriate magnitude relating the feature and target regardless of the number of samples the feature splits within the model. A positive feature effect indicates that the relationship of the feature and target values are in the same direction, i.e. as one increases so does the other, whereas a negative feature effect indicates an inverse relationship where the target values decrease as the feature values increase. The directionality relationship between feature and target is an often desired model result that is traditionally lacking from decision tree based methods.

Evaluating the efficacy of the importance scores produced by iRF can be done by looking at the rank order of the feature importances. The feature with the highest importance score is most likely to be truly causative. While rank ordering the important features is useful, it does not provide a cutoff without previously knowing the appropriate amount of causative features. However, by considering a purely random association between a feature and a phenotype, an easy to use cutoff can determined. First, the total variance (TV) of the phenotype can be defined by multiplying the traditional variance $\sigma^2(x)$:

$$\sigma^2(x) = \frac{\Sigma(x_i - \bar{x})^2}{N}$$

of the data set by the total number of samples in the data set:

$$TV = \sigma^2(x) * N = \Sigma(x_i - \bar{x})^2$$

where N is the total number of samples. Considering that any given phenotype can have a random distribution of values (flat, Gaussian, bi-modal, etc.), then it can be determined that on average any single sample in any given data set contributes about $1/N$ to the total variance of the phenotype in the data set.

If a random variable with no relationship to the distribution of the phenotype is taken and the split in the variable isolates a single sample from the full data set (not likely to

be the best possible split), then the variance explained by this split is expected to be the variance contribution of that single sample ($\frac{TV}{N}$). If the split in the variable breaks the set of samples into two sets with both containing at least 2 samples, then the contribution of that split will be either higher or lower than that of the single sample split. The magnitude of the difference between the single sample and multi-sample scenarios can be determined by the difference in the average phenotype value of the samples from the parent node to the children nodes ($\bar{x}$).

Because of the assumption of random distribution of the phenotype, and the assumption of no relationship from feature to phenotype, the average change in $\bar{x}$ is expected to be 0. This leads us to the conclusion that the expected total variance explained by a random *feature* in a single split is the same as the case in which a single *sample* is being split ($\frac{TV}{N}$). When the feature importance is normalized by TV, the expected proportion of importance of a random feature, for a single split, is $\frac{1}{N}$. An example model with 100,000 features and 100 samples would have a cutoff of $\frac{TV}{100}$ or 1% of TV, .01 normalized importance. Effectively any feature with more than or equal to 1% of the total feature importance of the model is identified as most likely being causative. The cutoff in this scenario defines that up 100 features would be able to be identified as important if they all shared the same importance within the model, at 1% TV, and also means that if there are more true causative features in the data set than samples, say 200, the cutoff prevents half of those features from being identified in even a best case scenario. However, in the later scenario, even with 200 features sharing importance, it is plausible that the model favors few enough to elevate them over the threshold, thus providing accurate features, if not the full set of correct features. Thus, the cutoff value of $\frac{1}{N}$ normalized importance can be used as a threshold for separating probable noise from probable signal. The model is going to assign features with true signal a larger proportion of importance, and features with only noise less importance.

Evaluating Sets Reported from iRF

RIT[34] is used to determine the prevalence, or occurrence rate, of sets of features in the feature paths, as well as the expected occurrence of that set based on the features that make it up. Utilizing this method makes it possible to determine the difference from expected

co-occurrence, or overlap, to actual overlap. A higher actual overlap points to some of the importance of the set being conditional as opposed to purely additive. Using the same concept of mining the decision paths, it is possible to determine the portion of the importance that is based on the conditionality of the set and not just of the importance addition of the isolated features. This is possible by looking at the difference of importance from splits with and without the conditionality from the forest.

With a decision tree of two levels, three total splits and four leaf nodes (this same structure can be seen in Fig. 1.3), decision paths are defined as a step function:

$$f(x) = \begin{cases} L_1 & x_i < S_1; x_j < S_2 \\ L_2 & x_i < S_1; x_j \geq S_2 \\ R_1 & x_i \geq S_1; x_k < S_3 \\ R_2 & x_i \geq S_1; x_k \geq S_3 \end{cases} \tag{2.3}$$

where each of the final leaf values (L_1, L_2, R_1, R_2) can be reached by one of the decision paths represented by a multi-part conditional (such as $x_i < S_1 and x_j < S_2$) where each x_i, x_j, and x_k representing different features and S_1, S_2, and S_3 represent the different split values.

RIT, used as part of the previously established iRF method, provides the prevalence of the most common sets of features within the paths. Because of the nature of decision pathways, the maximum number of sets obtainable with RIT from the decision paths is $\binom{x}{d}$, where x is the number of features and d is the depth of the trees, assuming no repeat features are used within the path. Beyond the limitations of the paths, RIT outputs are bounded by the user in two main ways. The first is by limiting the minimum prevalence reported and the second is by limiting the total number of sets reported.

Three additional measurements have been developed and calculated for the sets reported by RIT. The first of these is *RIT_adjusted*, which accounts for the individual prevalence of the features that make up the set. The prevalence, how often a feature occurs throughout the entire forest, indicates whether or not a feature is simply highly occurring versus occurring conditionally with another feature or features. *RIT_adjusted* is calculated as:

$$RIT\_adjusted = RIT - \prod_i \omega(x_i)$$

where RIT is the reported RIT prevalence for a set and $\omega(x_i)$ is the frequency of the feature x_i within the pathways. The equation results in positive *RIT_adjusted* values for sets that occur together more than expected from random overlaps due to just the prevalence of the constituent features and negative values for the set that occur together less than expected. When converting to a classification problem of identifying causative sets of features within a model, a true positive for *RIT_adjusted* is a true set that has a greater than zero value, identifying the set as co-occurring more than a random distribution of the constituent features throughout the forest.

The next value is *set importance*, which is calculated by determining the 'feature' importance for each RIT set by evaluating it as though it were a single feature. For each pathway containing the feature set, an imaginary node is created by treating the splits between the features in the set as a single decision. The total variance leading into this new imaginary node corresponds to the total variance of the from the samples included in the first node split by the first occurrence of a feature in the set. The imaginary node then splits the samples into two new nodes, one child node containing the samples from the earliest node in the original pathway that had been split by all of the features within the set, the second is all the other samples from the original imaginary node. The variance explained from this imaginary node is summed with all other imaginary nodes from the forest that can be constructed for the same set of features, providing the *set importance* of the set. This importance can then be scaled by the TV explained by the model and compared to the expected noise level of a single feature. A true positive for *set importance* is a true set that has a *set importance* greater than $\frac{1}{N}$.

The final measurement is *conditional importance* and it looks at each feature in the set and compares the importance of its splits, when conditioned by the other features of the set, with the rest of the occurrences of that feature throughout the model. To account for a bias in TV from split to split, the TV from each split is divided by the number of samples the given split is accounting for, to effectively normalize the importances. The conditioned

splits and non-conditioned splits are then compared, with a higher unweighted condition split resulting in a positive *conditional importance* for that variable in the set and the opposite resulting in a negative *conditional importance*. *Conditional importance* details the impact of being a part of the set for each feature within the set. A positive value indicates that the feature is more important to the overall model when conditioned on other features within the set as opposed to the rest of its occurrences within the model, and a negative value indicates that the individual feature is less important to the overall model when conditioned on the other features within the set as opposed to the rest. A true positive for *conditional importance* is a true set that has at least one feature in the set with a positive *conditional importance*.

2.3.3 Generating Synthetic Data with Epistatic-Like Properties

To determine if epistatic effects are being captured appropriately and the sensitivity at which they are captured, a synthetic data set can be created with known epistatic effects. When just looking to detect epistatic effects, the data set is comprised of features that have a flat distribution and no shared information amongst features. From a generated set of features, a synthetic phenotype is generated by choosing a random subset of the features to have random parametric (linear) and non-parametric (non-linear and likely multiple features combined) effects. The linear effects of a set of random features on a phenotype is then defined as:

$$f(x) = \Sigma a_i x_i \tag{2.4}$$

where a_i equals 0 when the feature is not in the selected set and is the effect size and direction (positive or negative) if the selected feature is in the causative set. The combined effects are constructed as non-parametric step functions with multiple conditions that all must be met before the random effect is added to the final target. The function of a single-step effect that is determined by two features that both meet specific conditions (in this case $x_i > S_1$ and $x_j < S_2$) is:

$$f(x) = \begin{cases} z & x_i > S_1; x_j < S_2 \\ 0 & \text{otherwise} \end{cases} \tag{2.5}$$

and this conditional, or epistatic, effect is added to the effects from other sources. The samples that meet the conditions will be shifted from the rest of the population by the magnitude of z. Once all of the effects, both linear and conditional, for a phenotype are determined, an amount of random noise is added to each sample in proportion to the total desired amount of noise for the data set. So if the proportion of noise is .25, or 25%, then 75% of the variance of the phenotype is attributable to its causative features, with the rest of the variance only being attributable to randomness. These synthetic data sets are repeatedly created with different parameters such as number of samples, number of features, number of causative linear effect features and number of non-linear effects on the given phenotype. This allows for the measurement of effectiveness of iRF in different scenarios.

2.3.4 Synthetic Trials

To begin validation of the ability to detect epistatic conditions in multiple scenarios, various test conditions are needed. To accomplish this, multiple synthetic data sets were created for testing the efficacy of the algorithm in different scenarios, including number of samples, number of additive causative features, number of epistatic-like sets, and amount of noise contributing to each phenotype. To thoroughly cover difference influences these parameters could have on the results, an exhaustive combination of the parameters listed in Table 2.1 was produced and modeled with iRF.

Table 2.1: An exhaustive combination of each parameter listed was used to to generate synthetic data trials.

Parameter	Values
Feature Count	100, 10000
Sample Count	500, 1000
Contains Correlated Features	no, yes
Linear Causative Features	0,1,10,50
Epistatic Sets	0, 1, 10
Proportion of Noise	25%,50%,75%

The correlated feature data sets are constructed by determining the number of *original* features that will be purely independent of each other and the distance, in number of features, between those *original* features. The *correlated* features are then produced by averaging, in incremental steps, the nearest two *original* features. These new features have diminishing correlations the farther apart they are in the list of features. The *original* features are also now partially correlated to the surrounding new features, resulting in every feature in the data set having several partially correlated features. Each combination of the non-phenotype parameters (Feature Count, Sample Count, and Contains Correlated Features) is used to create 10 unique data sets with those parameters. This results in a total of 80 uniquely constructed data sets. For each of the data sets, 10 unique phenotypes are created with each combination of the phenotype parameters (Linear Causative Features, Epistatic Sets, and Proportion of Noise), creating 360 phenotypes for each synthetic data set. The effect size and direction of the causative linear and epistatic sets are randomly determined with an equal chance of being positive or negative. The combination of all parameters results in a total of 28,800 total synthetic phenotypes with which to test the iRF GWES analysis metrics.

2.3.5 Model Evaluation Methods

The effectiveness of the techniques for the three areas of focus are evaluated by treating each model as a classification problem. The results of the classification problem are the ability to correctly identify a causative feature or set, and identify the direction of the effect with the feature effect value. For the goal of accurately determining the most causative features and sets of features for a given phenotype, converting the correct identification of feature sets into a binary classification allows numeric calculations of how well the model distinguishes between causative and non-causative.

Feature Effect

Because all features in the model are assigned a feature effect value but only causative values can be evaluated, the classification will have a balanced set of results when predicting the

direction of the effect, with about half being positive feature effects and the other half being negative. The balance is fully attributable to the random nature of the direction applied during the data creation process, which used an even distribution between positive and negative. While this may not be true for a real world data set, it allows for a full range of possible feature effects to be evaluated. The most straightforward way to analyze how the model predicts the feature effect then is to calculate the Area Under the Receiver Operator Characteristic curve (AUROC). This value is the integration of the curve mapping the True Positive Rate by the False Positive Rate of the classification. A model producing random results will have an AUROC of 0.5 and a perfect model will have a value of 1.

Balanced Accuracy of Causation

For the process of calculating how accurately the models are able to identifying features and their sets as causative or not, an issue with balance between cases occurs. Non-causative features can outweigh the causative features in the data by up to a ratio of 9,999 to 1, and non-causative sets can outweigh their causative counter parts by up to 49,994,999 to 1. This means that traditional classification assessment techniques that assume balanced data are likely to be either non-informative or misleading. The simplest way to account for the imbalance while providing an informative measurement of accuracy of the classifications is to calculate the balanced accuracy[160]. The balanced accuracy is calculated by averaging the True Positive rate, or how likely a true prediction is actually true, and the False Positive rate, or how likely a false prediction is actually false. This value is equivalent to accuracy if the classes of the model are balanced.

Post Test Probability

The other important factor to identify is the likelihood that the features and sets that the model predicts are actually causative. Traditionally, this is evaluated by looking at the Precision, or True Positive Rate as mentioned previously, of the model. But again the problem is that the non-causative features and sets significantly outweigh the causative ones and outweigh them by different quantities based on the specific data set. This can be accounted for by calculating the Post Test Probability, which is the expected likelihood of a

predicted true value to be a true positive value, as opposed to precision which merely reports the percentage of predicted true values that were True Positive. This calculation is common in medical research when determining the efficacy of a test for diagnosis[161].

The Post Test Probability is calculated by first determining the positive likelihood ratio, or how many times more likely a predicted positive value is to be a True Positive than a negative predicted value is to be a False Negative. It can be calculated by dividing the Sensitivity, or True Positive Rate, by 1 minus the Specificity, or False Positive Rate. To get from this to the Post Test Probability, the probability is converted to odds, multiplied by the odds of a randomly selected value being a True Positive, and then converted back to a probability. While this new value is still partially affected by the balance of classes, it does generalize better than specificity since the odds of randomly selecting a True Positive can be changed based on the underlying model. This calculation also provides a directly understandable value that can be applied to any result: "This predicted value has this percentage chance of being a True Positive as determined by Post Test Probability."

2.3.6 *Populus trichocarpa* data

Populus trichocarpa, colloquially known as black cottonwood, is a poplar tree indigenous to the Pacific Northwest of North America. It has been studied for several decades, including a full genomic analysis[93, 94]. A diverse set of genotypes were collected from nearly the entire species' geographic range. These were then clonally propagated to multiple common gardens in different climates within the natural range of the species. At these field sites, a randomized block design was used to spread three replicates of each of the 848 genotypes throughout each of the common gardens. Over the course of several years, these common gardens were monitored and many phenotypes were measured, including height.

It is known that, in general, the phenotype (P), or trait, being measured can be defined by the equation:

$$P = G + E + G * E$$

where G is the genotypic effect, E is the environmental effect, and $G * E$ is the effect of the interaction between the genotypic effect and the environment[41]. To determine

potential epistatic effects within the overall genetic contribution, and isolate the genetic effect from the environmental effect, the heights from a site with randomized block design are evaluated through a genomic Best Linear Unbiased Prediction (gBLUP) model[120] to assign the variance of height that is directly attributable to each genotype. With this variance specifically attributable to the genotype, the iRF model is used to determine the SNPs that interact with each other, in their prediction of height. Height was chosen as the phenotype for selection both for its availability of data and for the complexities of the phenotype in general because it is a continuous value that is effected by many proprieties[162] and specifically within forest trees because of the different growth and regulation patterns the organisms go through over their life span[163].

To validate the interactions from iRF have functional causation, the interactions are measured in other types of data. This can be simplified by identifying the genes most likely to be effected by the SNPs, where either the SNP falls within the coding region of the gene or in a nearby non-coding region, and through the use of networks that connect genes based on other data types. Also the SNP set used was pruned to remove highly correlated SNPs that are near-by on the genome. This Linkage Disequilibrium pruning process makes SNPs act as markers for changes that could occur in the region of the genome around it. These networks can be built from expression data to produce potential regulation relationships, or data classifications such as Gene Ontology (GO)[164] that describes different types of gene functionality. The regulatory network is constructed from data mined from literature and assembled into a network that connects genes who are involved in the regulatory pattern of other genes [165, 166, 167]. The GO networks are created by connecting genes that share an annotation from the GO label hierarchy.

Epistasis is the genetic effect of separate SNPs interacting to effect a phenotype, but for the organism to express the epistatic interaction from multiple genes there must be some functional relationship between the genetic loci. One way for the functional relationship to change is through differential expression. Both the expression and GO-networks can highlight the likely nature of a specific relationship. A two-step analysis can determine the strength of the relationship within the network. The first step is to determine if there is a path between the genes within the network via either direct connection or indirectly though

connections to other genes. If there is a path between the genes, then the next step is to determine the shortest path between the genes, with the shorter the path meaning a stronger relationship.

2.4 Results and Discussion

2.4.1 Synthetic Data

The overall effectiveness of capturing the explanatory information with iRF can be seen by the average results across the synthetic data and the variations in the conditions within the data set and underlying model generation. The synthetic data generated as described in section 2.3.3 and in the trial format as described in section 2.3.4 was used to train an iRF model for each individual phenotype. These resulting iRF models where then mined with RIT and evaluated by the thresholds and methods listed in section 2.3.2. The evaluations of the models were broken into three main areas of focus: effectiveness of determining feature effect, the ability and efficacy for the $\frac{1}{N}$ variance threshold on feature importance to capture true causative features, and the ability and efficacy of the four set evaluation techniques to capture the true causative sets. The full descriptions for each of the evaluation methods are available in the Methods section. For evaluation purposes, the prevalence requirement for RIT is 0.1, meaning a set must appear in at least 10% of decision paths to be evaluated, and the total number of sets is limited to twenty for each phenotype.

Detecting Feature Effect

By building synthetic data with a prescribed amount of epistasis and additive effects over a known amount of noise, it is possible for iRF to identify the epistatic signal. Analyzing synthetic data can show how effective iRF is at detecting epistatic signals at different strength levels and with different amounts of background noise. An example of detecting causative feature effects can be seen in Figure 2.1 where the y-axis is the phenotype values, the x-axis the feature values, and the horizontal lines each representing the step function of a different split.

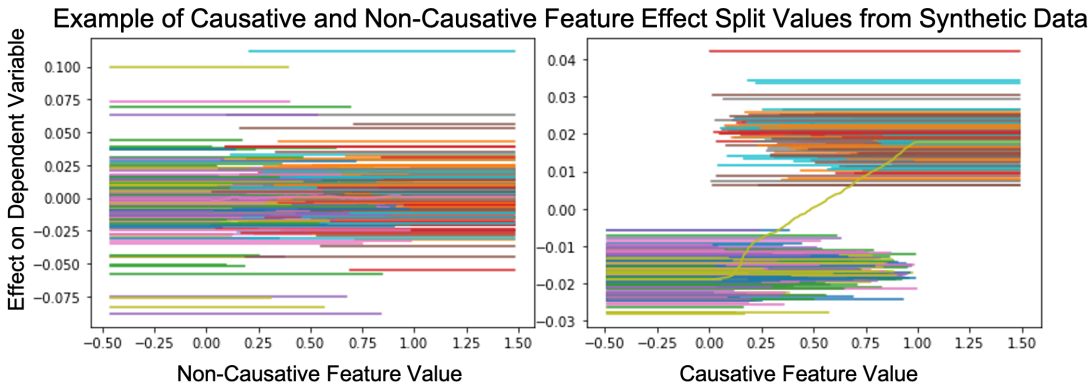

Figure 2.1: By training an iRF model with synthetic data that includes causative and non-causative features with included noise we can see how the split's across the entire forest for a feature behave. The left graph shows splits from a non-causative feature, with no coherent or consistent effect measured. The right graph shows splits from a causative feature with a consistent effect that shows a positive relationship between the feature and the dependent variable.

The left graph shows a plot of the splits for a non-causative feature, that appears to be mainly noise, while the right graph shows how the splits for a causative feature look and produce a linear fit that approximates a straight line. A causative feature clearly shows a more consistent pattern, even if there is some variation, which itself is what provides the indication that a feature is causative.

While this anecdotal example shows a visualization of a causative feature with positive effect, we can examine the effectiveness across the synthetic data set of correctly identifying the causative features and their effects. Because all features that have any amount of importance will also have a feature effect assigned, we can begin by looking at true causative features and evaluating if they appropriately determine the direction of the features effect. The average AUROC of feature effect detection of different subsets of the synthetic data sets can be seen in Figure 2.2. From the figure it can be seen that each type of delineation has an effect on the AUROC of detecting feature effect. By increasing the number of samples in the data set, the more information the model has to work with, the model is better able to attribute the correct direction to the causative features. Increasing the number of features has the opposite effect, where the model is required to sift through more potential features with potentially small amounts of random overlap with the causative features, that cause the model to make inaccurate predictions. Similarly, as the amount of noise in the data increases, the model begins to struggle to appropriately assign the correct direction to feature effect. Correlated features generally cause an overall negative effect on the model but with some inconsistencies, while the correlated feature value is occasionally higher than the no correlated feature comparison. Overall, the average AUROC for the different data parameters were well above the 0.5 threshold, with stronger results for the smaller feature sets and lower noise levels.

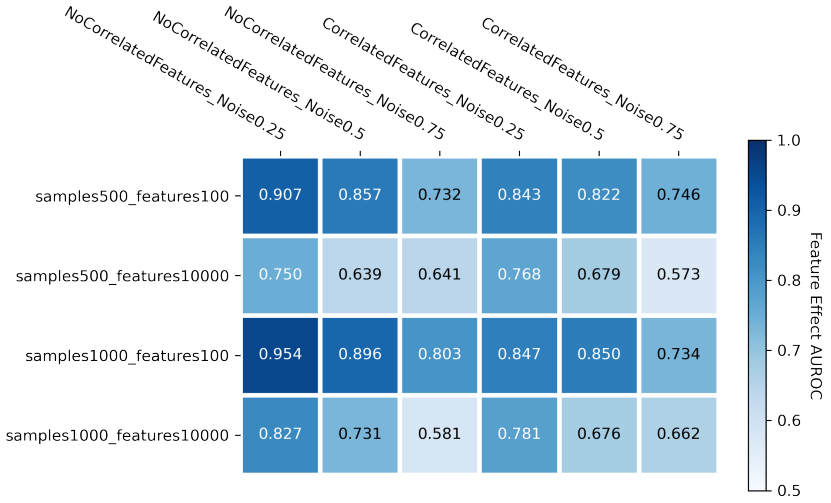

Figure 2.2: This heat map shows the average AUROC of subsets of the iRF models trained on the synthetic data sets in determining the effect direction of causative features. The the columns represent models trained on data sets with an amount of noise, either 25%, 50%, or 75% noise. The first three columns represent the models trained on data sets with no correlated features and last three columns represent the presence of correlated features. The heat map rows represent the size and structure of the data sets with the top two rows containing 500 samples each and the bottom 2 having 1000 samples each. The rows alternate from representing 100 features to 10000. The scale of the heat map begins at .5, which is the AUROC of a random model and tops out at 1 which would be a perfect model. This shows how every axis of change of the data set has some effect on the models. More samples generally mean better AUROC, more total features generally correlate to lower AUROC, more noise in the model leads to lower AUROC, and correlated features generally leads to lower AUROC.

Detecting Individual Causative Features

The balanced accuracy of predicting the correct causative features with the importance threshold of $\frac{1}{N}$ where N is the number of samples used in the model can be see in Figure 2.3. The most accurate models in this heat map are the ones with the fewest features. The relationship is due to the model being more easily able to sort features into the categories of causative vs. non-causative when there are fewer features to consider. With the same number of samples, iterations, and trees, in the data sets with fewer features, each feature will be considered more often than in a data set with significantly more features. While the number of features considered at each node scales with the total number of features, the number of selected features for a node will always be one. This effectively causes data sets with larger feature sets to split on each feature fewer times, preventing the model from evaluating each feature's importance as well as a similar model with fewer features would be able to.

For this smaller feature set at the lowest noise values we see worse performance than the next step in noise for sets with correlated features. This shows, in this specific circumstance, that the model is aided by an amount of noise that helps distinguish between similar features and determine the one that is most likely causative. The models with 10,000 features perform significantly worse and show significantly more impact from increased levels of noise. Interestingly, the number of samples seems to only have a small positive impact.

The other way to evaluate the effectiveness of the feature importance threshold is how likely the features are that pass the threshold to be truly causative. The results of this can be seen in Figure 2.4. Here we see that in every scenario except for large numbers of features and high noise, the chance of an identified feature being causative is over 80%. However, in each case with high noise there is a clear difference between 100 features and 10,000 features, in the direction expected. For the lower noise levels, the larger number of features actually shows the same or even better performance than the lower number of features. The incorrect importance assigned to non-causative features is the result of those random features having spurious correlations to the noise added to the phenotypes.

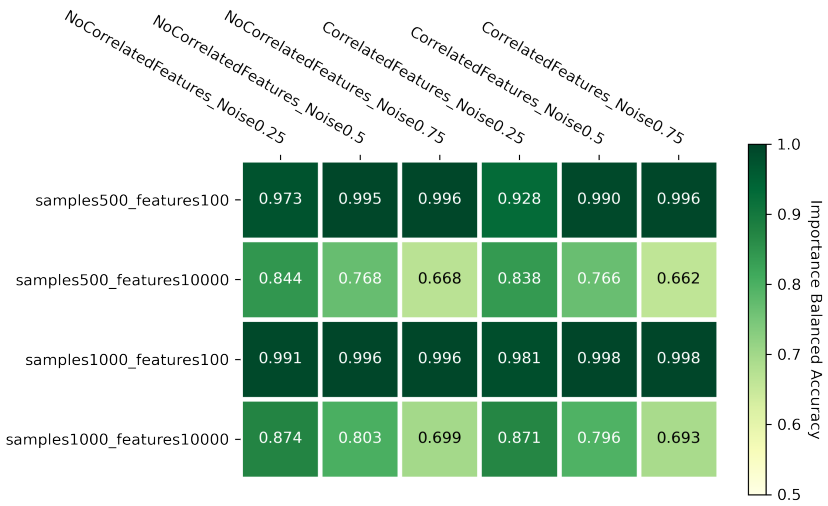

Figure 2.3: This heat map shows the average balanced accuracy of subsets of the iRF models trained on the synthetic data sets in prediction of causative features with the importance threshold of $\frac{1}{N}$ where N is the number of samples used in the model. The the columns represent models trained on data sets with an amount of noise, either 25%, 50%, or 75% noise. The first three columns represent the models trained on data sets with no correlated features and the last three columns represent the presence of correlated features. The heat map rows represent the size and structure of the data sets with the top two rows having 500 samples and the bottom 2 having 1000 samples, and the rows alternate from representing 100 features to 10,000. The scale of the heat map begins at .5, which is the balanced accuracy of a random model and tops out at 1 which would be a perfect model.

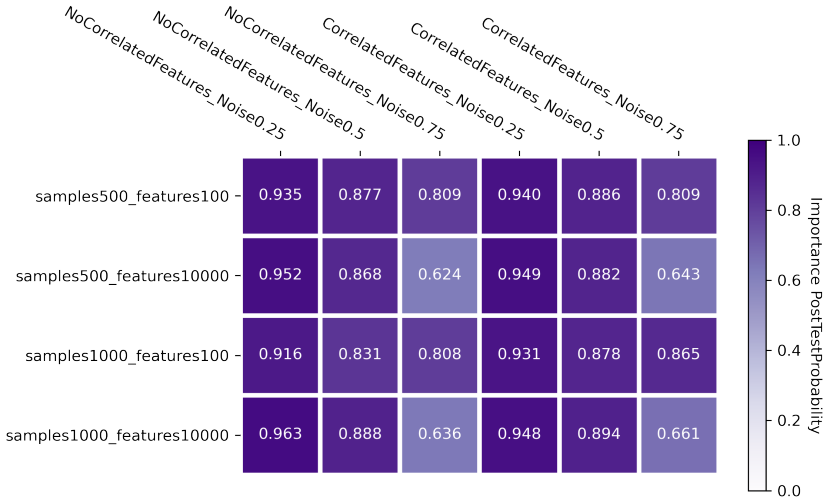

Figure 2.4: This heat map shows the average post test probability of subsets of the iRF models trained on the synthetic data sets in prediction of causative features with the importance threshold of $\frac{1}{N}$, where N is the number of samples used in the model. These values directly correspond to the chance that a feature that meets the importance threshold from that data set is a True causative feature for the phenotype of that model. The the columns have represent models trained on data sets with an amount of noise, either 25%, 50%, or 75% noise. The first three columns represent the models trained on data sets with no correlated features and last three columns represent the presence of correlated features. The heat map rows represent the size and structure of the data sets with the top two rows having 500 samples and the bottom 2 having 1000 samples, and the rows alternate from representing 100 features to 10000. The scale of the heat map begins at 0 which is the probability of never correctly predicting a causative feature and tops out at 1 which would be a every predicted feature being a true causative feature.

When there are more total features the spurious feature importance is spread over more of the non-causative features, reducing the number of spurious correlations above the normalized threshold. This effect disappears at the highest noise levels.

Detecting Synthetic Epistatic Effects

When using iRF to detect epistasis in the synthetic data sets, the overall accuracy is effectively 100% regardless of the specific measurement type due to the sheer number of true negatives. However, with balanced accuracy the True Positive Rate and the True Negative Rate are equally as important, which can be seen in Figure 2.5. Of the four types of measurements (RIT, *RIT_adjusted*, *set importance*, *conditional importance*), all are either RIT or reliant on RIT to produce a set of potential feature sets to evaluate. The two metrics with the highest balanced accuracy are RIT and *set importance* (setimp), which with rounding perform identically across the different data sets. Across all the data sets and measurements, the amount of noise affects the balanced accuracy negatively but the change from 50% noise to 75% noise has a more significant effect than the change from 25% noise to 50% noise, with the *RIT_adjusted* value seeing the smallest, almost negligible effect from the noise. The worst measurement is *conditional importance* (conimp) when evaluating the highest noise data sets. Overall, the best two methods according to balanced accuracy are RIT and *set importance* with each having the same largest value. Because *set importance* is a subset of the set from RIT we know that RIT will provide more overall correct sets but at the same rate.

The upper limit of the other measurements (*RIT_adjusted*, *set importance*[setimp], *conditional importance*[conimp]) are explicitly reliant on the available sets from RIT. However, we can see in Figure 2.6, that while some of these measurements incorrectly prune out true sets, lowering their post test probability (*RIT_adjusted*, *conditional importance*), the *set importance* measure significantly improves the post test probability of RIT, or at worst maintains the same level as with the higher noise data sets. This indicates that *set importance* with the $\frac{1}{N}$ threshold is appropriately identifying causative sets from the ones provided by RIT. The overall worst performing measure is *conditional importance* and *RIT_adjusted* falls short of base RIT.

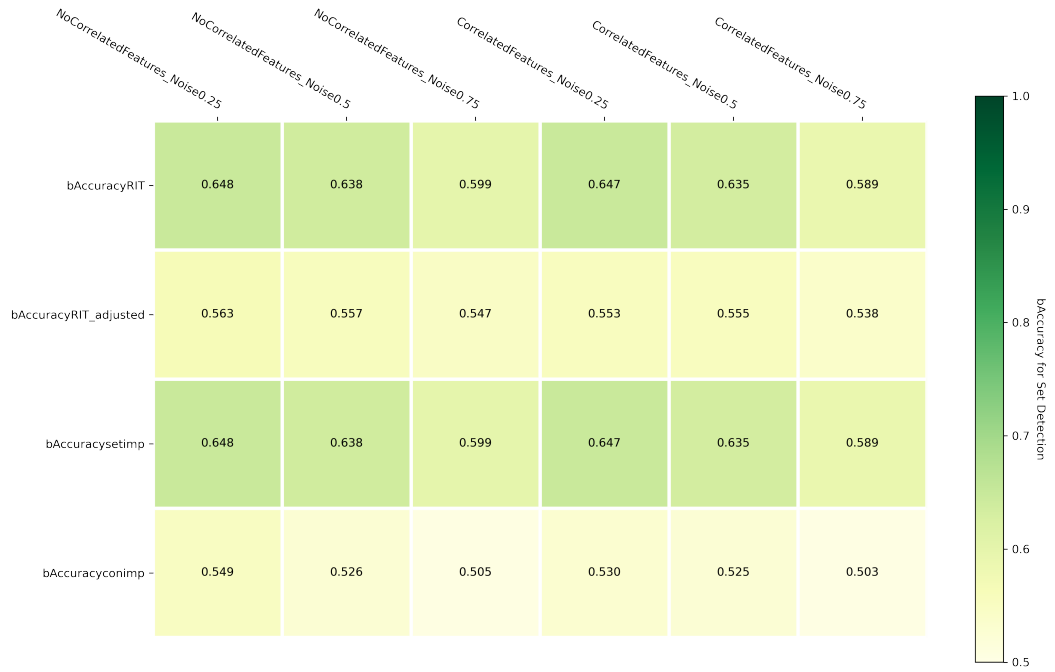

Figure 2.5: This heat map shows the average balanced accuracy of subsets of the iRF models trained on the synthetic data sets in prediction of causative sets of features with the four separate detection methods: RIT, *RIT_adjusted*, *set importance* (setimp), and *conditional importance* (conimp). The columns represent models trained on data sets with an amount of noise, either 25%, 50%, or 75% noise. The first three columns represent the models trained on data sets with no correlated features and last three columns represent the presence of correlated features. The heat map's rows indicate the four metrics being evaluated - RIT, *RIT_adjusted*, *set importance*, and *conditional importance*. The scale of the heat map begins at .5 which is the balanced accuracy of a random model and tops out at 1 which would be a perfect model. Because the sets examined by the other methods are determined by RIT, it makes some sense that it has the highest balanced accuracy in prediction of causative sets. While RIT and *conditional importance* look identical when comparing out to 3 decimal places, they do diverge slightly after considering more places.

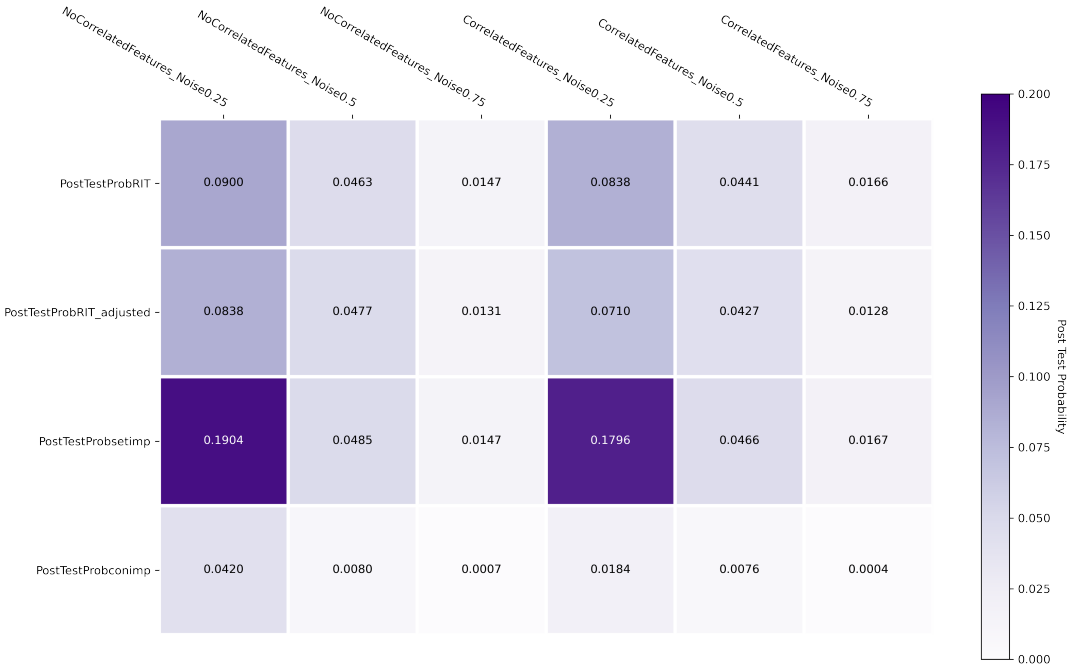

Figure 2.6: This heat map shows the average post test probability of subsets of the iRF models trained on the synthetic data sets in prediction of causative sets of features with the four separate detection methods: RIT, *RIT_adjusted*, *set importance* (setimp), and *conditional importance* (conimp). The color scale has been set to show the range between the best performing metrics and the worst. The columns have represent models trained on data sets with an amount of noise, either 25%, 50%, or 75% noise, also first three columns represent the models trained on data sets with no correlated features and last three columns represent the presence of correlated features. The heat map's rows indicate the four metrics being evaluated - RIT, *RIT_adjusted*, *set importance*, and *conditional importance*. The scale of the heat map begins at 0 which is the probability of never correctly predicting a causative sets and tops out at 1 which would be a every predicted feature being a true causative sets. We can see here that the likelihood of a predicted set being a causative set, is significantly lower than identifying the individual features. However part of this is likely because of the imbalanced nature of true to false sets playing a more significant roll. Despite this, *set importance* does perform the best by having a predicted set being truly causative nearly 20% of the time for the lowest noise subsets.

It should be noted that, while all of these probabilities are significantly lower than the post test probability for the feature sets, this is significantly influenced by the pretest probability of a set being causative, or the balance between causative sets and non-causative sets. The average pretest probability of randomly selecting a causative set across all of the synthetic data sets is .037% or 0.00037, which is determined by averaging the number of causative sets over total possible sets of two features for each data set. This means that, on average, every method here does better than random selection in every scenario with *conditional importance* of the high noise set with correlated features only being slightly better than random. This also shows that while a 19% chance that a selected set is causative for the best case scenario for *set importance* seems low, it represents over a 500 fold increase chance of reporting a causative value from random.

2.4.2 Poplar Data

An iRF model was successfully trained to model the genetic component of height, as determined by GBLUPs, for *Populus trichocarpa* from the most important Linkage Disequilibrium (LD) pruned SNPs and sets of SNPs, as selected by the methods previously discussed. The model identifies 113 SNPs that have an importance greater then the threshold set based on the number of samples, which is to be expected from a complex phenotype like height[163]. Unfortunately, many of the genes identified by this process do not necessarily have annotations within *Populus trichocarpa*, but potentially have orthologs in other species or produce a protein that may have been analyzed in another species. The most important SNP in the model was on chromosome 8 at position 7,083,177. This SNP falls into an intergenic region, not marking a change in a specific gene, however this specific SNP could represent a change in surrounding genes either by being in a promoter region for a nearby gene [168] or by being the marker SNP, due to the LD pruning process, for a change within a gene or a series of changes in that region of the genome[169, 170].

The gene this SNP falls nearest to is Potri.008G111100, which is a gene that produces an enzyme, specifically cytosine deaminase, and is responsible for zinc-ion binding in humans as seen in drug therapy studies[171] and part of pyrimidine metabolism as determined by a study into protein structures[172]. Because the metabolic process is central to growth, it

appears that the iRF model has identified a SNP-level change that may affect the plant's metabolism and eventually the height of the tree[173]. This SNP also has a positive feature effect value indicating that the change denoted by this feature leads to taller trees. The gene corresponding to the SNP with the most importance, but negative feature effect, is Potri.010G114000, which is a MYB transcription factor, whose main purpose in plants is to respond to stimuli and stress[174]. This indicates that the model is potentially identifying the SNP and region causing certain poplar trees to respond differently to a specific stress and as a consequence end up with less height within the population.

The model also produced several sets of SNPs that work together to predict height, suggesting that epistatic interactions might be occurring. Each of the top 50 sets of genes (closest genes from the given SNP from RIT) from the Height BLUPs model all have a connection within the regulation network, with all of those sets having a shortest path of no greater than 5 genes. This indicates that the regulation of all of the sets of genes identified have at least an indirectly related expression profile. Of those 50 sets of genes, 12 sets have a connection in one of the three GO-networks. Of those 12 sets, 3 gene sets have connections with 2 separate GO-networks. To fully verify the relationships identified by iRF, they have been evaluated for closeness of relationship to each other via Random Walk with Restart (RWR)[175] across multiple biological networks constructed for *Populus trichocarpa* [176]. From this process, the genes that most closely associated with each other from the RWR models also matched to the sets with either the top RIT scores or top *set importance* scores with very few exceptions. Effectively the highest performing sets according to RIT and to *set importance* are also the sets that showed the closest biological relationship according to RWR.

One of sets with the highest *set importance* that is verified by the RWR process includes Potri.008G111100, the important gene involved in metabolism, and Potri.015G132500, a gene whose ortholog in *Arabadopsis thaliana* is involved in ionic binding and is specifically an intracellular membrane-bounded organelle[177]. This relationship as identified by iRF could potentially be an epistatic signaling process to trigger a metabolic process, likely having a downstream effect on overall growth and height of the organism, as a part of a stress response of a forest tree[178]. Potri.006G137600, which encodes a flavin-containing

monooxygenase, a protein whose presence in yeast is often responsible for xenobiotics by making suspected foreign objects more soluble[179], and Potri.007G102200, which encodes a proprotein convertase subtilisin/kexin, or an enzyme that activates other proteins, as seen in mice[180], have the highest RIT score among the other sets of genes. They are connected through the regulation network and via the GO Molecular Function network, and the GO Biological Process network via the gene. The relationship of these genes, while not explicitly annotated within *Populus trichocarpa* to be involved in height or growth, may have an epistatic interaction with each other based the iRF and RIT result as well as the relationships they share across other organisms.

2.5 Conclusion

Here I have provided new metrics and methods to complete Genome Wide Epistasis Studies with Iterative Random Forest. The new additions to analyzing the results from iRF have been shown to have some uses in both synthetic and real biologic data sets. The most effective new additions are the importance threshold of $1/N$, the identification of feature effect direction, and the *set importance* measurement being more valuable than RIT prevalence alone. These methods have been shown to produce valid results regardless of conditions of the data set, with regards to size of the data (both in samples and features) as well as noise in the data. Generally the main limitations of the model can be addressed, more noise and more features make it more difficult to produce accurate results, but the inclusion of more samples can significantly improve the results. In the general case, adding more trees to the forest construction process could increase the models overall ability to capture these different effects, but would add to the computational expense. Other meta-parameter tuning could also effect the model's ability to capture information from data set to data set, but again the tuning process would likely come at the cost of more computation time.

With the ability of the method to provide useful results shown, future work can be done to validate the method across a variety of species and conditions. Given the complexities in wet lab validation, further validation may work best if done with model organisms with large amounts of publicly available data and known epistatic interactions. After validation,

the next step would be to use the method as a hypothesis generation tool to identify cases of epistasis that can be evaluated via *in vivo* or *in vitro* tests. As seen in the synthetic data, the chance of a set identified by *set importance* being a true causative set of alleles can be nearly 20%. As the tests to do full *in vivo* or *in vitro* studies are often costly procedures, starting with a higher quality hypothesis from this method should increase the efficiency of the success rate of those tests, saving both time to run the trials and money spent setting them up.

Moving forward, some of the calculations to determine the metrics including feature effect, *set importance*, and *conditional importance*, could be optimized to handle larger data sets. The current implementation of these metrics requires searching through every split in every forest to identify splits that meet the appropriate conditionals for each feature or set of the metric. While individual feature metrics can all be calculated with one pass of the forest, set importances would require an exponentially large amount of memory to store all possible sets. The total number of the calculations for these metrics scales as the size of the trees and forest grows, which itself scales with the amount of samples. As none of the calculations for different conditionals rely on each other, a simple parallelization scheme could be developed to distribute the analyses across multiple compute threads.

Chapter 3

Progeny Prediction with iRF

3.1 Abstract

Genomic selection, the process of making informed crosses between individual trees in order to produce progeny with beneficial phenotypes, is of particular interest to the study and development of biofuels. However, traditional processes of determining the best parents to be used in crosses can be time and resource heavy. By developing methods to accurately replicate progeny trials synthetically, genomic selection processes could be optimized. Here, I introduce a machine learning-based method to predict quantitative phenotypes of progeny which account for both the general combining ability (GCA) of each parent and the specific combining ability (SCA) of each cross. The process is validated using the *Populus trichocarpa* height phenotype, a feature of interest for biofuel production.

3.2 Introduction

Genomic Selection is the process of selectively breeding a species to adjust one or more phenotypes, or traits, in a desired direction with the aid of genetic data. Some selection processes have been important for crop and resource improvement across the broad scope of the history of human civilization. The most basic form of selection being phenotypic selection[102], which takes the best performers, or individuals with the most desirable characteristics, and crosses them with each other. However, it can take multiple generations over the course of several years for this process to produce a significant improvement in a phenotype of interest. Gregory Mendel proved the process of inheritance of some simple physical traits and presented a way of predicting the likelihood of a particular phenotype of an offspring based on the same phenotype present in the parents. He also described dominance, where a single dominant gene or double recessive gene controls the expression of the trait, and additive effects, where multiple genes will have similar effects that are combined into a single phenotype[181].

While Mendel's work could describe inheritance of very simple traits, it was assumed that more complex traits were the result of a large number of genes affecting some small portion of the phenotype, or that there was some environmental effect that changed the

trait. However, a complex phenotype may have portions of the trait that is the result of a non-additive interaction between two genes - a phenomenon called epistasis[8]. These more complex phenotypes, such as height[182], rely on more information than can be captured in a simple Mendelian phenotypic model because they combine the effect of several genes via additive effect, dominance, or epistasis, into one phenotype.

When considering the value of genotypes within a population as potential parents in a genomic selection process, the most valuable parents are generally those who have the highest General Combining Ability (GCA) for the target phenotype. GCA is the genetic variance of the phenotype expected to be transferred by that parent to all of its offspring[40], and is a metric which effectively contains all of the additive genetic effects for the phenotype that the parent will transfer to its offspring. Specific Combining Ability (SCA) is the genetic variance of the phenotype of the offspring not directly attributable to either parent, but specifically to the cross of those two parents[114]. The proportion of GCA to SCA is different for every phenotype, with more complex phenotypes more likely to have more contributions from SCA. Thus, building a model that relies on only GCA has the potential to miss a significant amount of information important to the manifestation of the phenotype.

Traditional genomic selection focuses on determining what parents will most effectively improve a phenotype within the population for subsequent generations while maintaining genetic diversity. This process typically results in a ranked order of genotypes that have the highest value for GCA, as this is guaranteed to be passed on to the offspring regardless of choice for the other parent. The highest ranking genotypes are crossed together to produce a new generation. In terms of complexity, this means that the total number of crosses needed for a selection of parents increases quadratically with the number of parents to do a full crossing process. Importantly, accounting for this increase in complexity by limiting the number of parents will dramatically decrease genetic diversity of the progeny. If the number of parents is not limited but the number of crosses is, GCA alone has no way to influence which of the potential crosses are chosen, which can lead to missing high performing crosses. Completing this process requires a large amount of time and resources where any complications can set back the process by years.

One possible improvement on the traditional genomic selection model is to incorporate machine learning algorithms to better process the data that is already available. Many machine learning methods are inherently capable of processing non-linear and conditional relationships, providing avenues for research into the more complex genetic interactions. Some machine models have already been applied to the problem of efficient and accurate genomic selection, including some non-linear models with better predictive accuracy than linear models[129] and a multi-layer-perceptron (MLP) neural network, which has been shown to be better at predicting the best parents for wheat and jersey cows than a Pearson's selection model[128].

Rather than using the traditional prediction of breeding values, significant improvement could come from predicting the full genetic component of the phenotype from a genotype, in a way that allows for the determination of both GCA and SCA. Here I present a method for progeny prediction using a model that predicts the full genetic component of the phenotype for a given genotype, and combines it with a way to produce *in silico* genotypes for synthetic progeny. The method allows for the prediction of the quality of progeny from a specific set of parental genotypes. It also allows for the assessment of the quality of parents and specific pairs of parents without the need to perform the physical crosses by replacing that component with the computational *in silico* process. This has the potential benefit of increasing efficiency of land use and research time and resources by removing the need for physical test crosses. The method inherently accounts for both GCA and SCA and can be used to predict the best parents for a synthetic cross without the need for extensive real world trials.

3.3 Methods

3.3.1 Heritability

In general, the different effects contributing to a given phenotype (P) can be defined by the equation:

$$P = G + E + G * E \tag{3.1}$$

where G is the genotypic effect, E is the environmental effect, and $G * E$ is the effect of the interaction between the genotypic effect and the environment[41]. The goal of genomic selection is then to optimize either the G component, or occasionally both the G and the $G * E$ component for a particular environment. A method of isolating the genetic effect is in section 3.3.6.

When trying to predict the value of a phenotype for offspring, one needs to know the genetic variation of the phenotype within the species. This is considered the heritability of the phenotype within a population. While only considering genetic factors, the heritability sets the upper bound on the accuracy of the phenotypic prediction, as it only incorporates a portion of the information from equation 3.1. H^2 is used to denote the broad sense heritability, which is the variance of the phenotype explained by the entire genetic component over the variance of the entire phenotype:

$$H^2 = \sigma_G^2/\sigma_P^2 \tag{3.2}$$

where σ_G^2 is the variance of the phenotype attributable to all genetic effects, and σ_P^2 is the total variance of the phenotype. The narrow sense heritability, h^2, only accounts for the additive genetic variance of the phenotype:

$$h^2 = \sigma_g^2/\sigma_P^2 \tag{3.3}$$

where σ_g^2 is the variance of the phenotype attributable to only the additive genetic effects[117]. The narrow sense heritability is generally seen as more important because it allows for the simplest model to calculate the value of a parent for its potential crosses, as the GCA of potential parents can be determined without having to do a full-progeny trial to calculate SCA. In fact, the general predictive ratio (GPR) of a phenotype is defined by the proportion of genetic variance that can be described by additive effects, which can be defined by:

$$GPR = 2\sigma_g^2/\sigma_G^2 = 2\sigma_g^2/(2\sigma_g^2 + \sigma_s^2) \tag{3.4}$$

where σ^2_G is equivalent to $(2\sigma^2_g + \sigma^2_s)$ where σ^2_s is the variance of the phenotype caused by non-additive genetic effects such as those responsible for SCA. The general predictive ratio scales to one as the components of GCA make up the entirety of the genetic variance[183]. This means that traditional breeding techniques are more effective for traits that have a higher narrow sense heritability and a lower influence from non-additive components.

Effects that require information from both parents or multiple interacting parts of the genome are contained in the SCA. Epistatic effects are likely to be contained in the SCA component because they require information of disparate parts of the genome. The more base pairs between the segments of DNA involved in the epistatic effect (if they reside on the same chromosome), the less likely they are to be contained in the same haploid during recombination. If all the determining components of a trait are on different haploids then information from both parents is required. Dominance effects can be found in the GCA component if the parent is homozygous dominant for that gene, all of its children will have the dominant trait regardless of the state of the other parent. All other states of the parent for a dominant or recessive gene would be considered an SCA because of the need for information about the state of both parents.

3.3.2 Progeny Trials

SCA and GCA are normally determined by progeny trials[112], where all potential parents are crossed with each other to build a pedigree and evaluate the real breeding value of each parent. However, this can take a significant amount of time, space, and money depending on the species in question since genomic selection is often focused on overall population improvement[119].

The most common genomic selection techniques utilize methods from linear algebra to determine the breeding value of each separate genotype[107]. A particular method that incorporates genetics is Genomic Best Linear Unbiased Prediction (gBLUP), that uses the genetic relationship of individuals in the population, in place of a pedigree, to determine the value each genotype provides to the phenotype in question[120] or its general combing ability (GCA). A similar method that assigns value to specific SNPs is Ridge Regression Best Linear Unbiased Prediction (rrBLUP) [122, 121]. These methods determine the linear

relationship between the genotypes and the genetic variance of the phenotype and assign a breeding value to each genotype.

3.3.3 Progeny Prediction Pipeline

The pipeline for progeny prediction can be broken down into a six step process, as seen in Figure 3.1. The first step for progeny prediction is to collect a diverse set of genotypes from within the species of interest. The more diverse the individuals and the greater the number of individuals, the better the representation of genetic and phenotypic variation of a trait will be achieved. The second step is to grow these individuals together in a common garden, ideally with clonal replicates of each for more robust data. Phenotypes of interest are then measured for each individual. The third step involves sequencing each genotypes within the population, with more accurate and reliable sequencing providing better results. The fourth step is to build a model (such as iterative random forest section 3.3.4) that can use genotypes as input and predict the phenotype of interest.

At this point the model has been trained to predict a phenotype value from any genotype in the species. The more genetic components from the target genotype that exists within the training population, the more accurate the model, however, the target genotype itself need not be in the model to produce an accurate result. Next, in step five, a synthetic progeny population is produced from a set of potential breeding parents using software such as Sequence-Based Virtual Breeding (SBVB) [184]. These progeny are built by SBVB, by taking each of the chromosomes from each parent and performing a synthetic meiosis process where the chromosomes are internally shuffled and then randomly assigned to progeny. By repeating this process for a number of progeny for each possible cross within the population, what results is a full combinatorial cross of all parents. The crossing produces a number of families equal to the number of mothers times the number of fathers. Each unique pairing of parents produce a full-sibling family of unique genotypes of *in silico* progeny. Finally in step six, the *in silico* progeny are used, as well as the model trained in step four, to produce phenotypes for all of the new genotypes. This information can be used as a prediction itself or it be used as a proxy for a progeny trial and used to calculate GCA and SCA within this population.

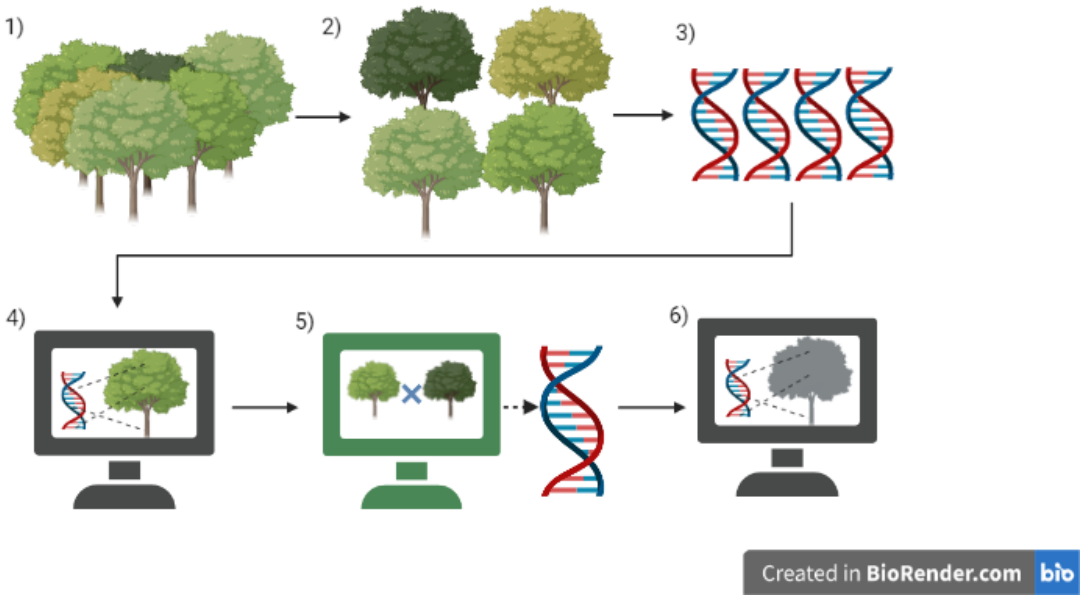

Figure 3.1: This image shows the simple pipeline for Progeny prediction. Step 1 – gather genetically distinct individuals of a species from the general population. Step 2 - grow the collected individuals in a common environment and measure a selection of desired phenotypes. Step 3 - Genotype the collected population. Step 4 - Train a machine learning model to predict the collected phenotype(s) based on their unique genotypes. Step 5 - Computationally produce *in silico* progeny by crossing the genotype of two potential parents in order to create the genotype of a potential offspring. Step 6 -Use the model trained in step 4 to predict the desired phenotype for the *in silico* genotype from step 5.

3.3.4 Iterative Random Forest

Iterative Random Forest (iRF) [4] is an explainable machine learning model that is able to incorporate non-linear and non-parametric effects. By using iRF as the model to predict the genotypic component of the phenotype, the process incorporates an explainable AI model that can include both the GCA and SCA components of the genetic heritability in its prediction.

Random Forest provides a robust method for feature selection [32] by providing a feature importance for each independent variable used in the model. These feature importances are calculated by summing the contribution of their splits. While there are several cost functions used to determine the best splits, one of the more common methods for classification trees is Gini[31, 33], which calculates the impurity improvement from parent to child nodes. Using feature importance, it is possible to address an issue that can occur in Random Forest, the random selection of features at each node can cause a majority of the splits to only consider non-causative features.

One solution involves weighting the random feature selection to favour a subset of features that are more likely to be causative, and one way to determine which features are likely to be more causative is to use the feature importance from a Random Forest model. By turning this into an iterative process, with each new iteration using the importance scores from the previous iteration as feature weights, a new algorithm called iterative Random Forest (iRF) is produced. This results in more efficient decision trees and more organized decision paths[4]. These organized paths can be mined to find sets of features that co-occur often in the same decision path with an algorithm called Random Intersection Trees (RIT)[34]. With a version of iRF designed to run efficiently on high performance computing (HPC) architectures, or supercomputers, it is possible to run on larger and more complex data sets[9].

3.3.5 *Populus trichocarpa* data

Populus trichocarpa, commonly known as black cottonwood, is a poplar tree indigenous to the Pacific Northwest of North America. It has been studied for several decades as a model tree organism and for its use as a biofuel, including a full genomic analysis[93]. A diverse

set of genotypes were collected from nearly the entire species' geographic range. These were then clonally propagated in multiple common gardens in different climates within the natural range of the species. At these field sites, a randomized block design was used to spread three replicates of each genotype throughout each of the common gardens. Over the course of several years, these common gardens were monitored and several phenotypes were measured. Externally measured phenotypes included height, diameter (measured as diameter at breast height (DBH), or as diameter at 20cm (D20) for young trees), bud set, bud flush, presence of diseases (like Melampsora[97] and Septoria[98]), and microbiome profiles on and within leaves, xylem, and roots. Other phenotypes that require destructive sampling, were also measured for a few of the common gardens including density, total biomass, gene expression profile of leaf, xylem, and root, metabolite profiles of leaves, and sugar content.

One of the most important phenotypes for biofuel production from poplar trees is the height of the tree at year three, as this is a significant indicator of the overall biomass available for conversion to fuel. To better understand differences in height at age three between the different collected genotypes, the trees in a common garden with three randomized blocks and clonal replicates of the collected genotypes were phenotyped after three years of growth. The height of these trees represents the differences in nearly the exact same environment because they are grown in close proximity to each other. This proximity still allows for micro-environmental changes to have some effect on the phenotype.

To have an appropriate validation set for the model, the parental sets for this analysis were chosen to match parents of those selected for a field trial, where 49 families were produced - the 7x7 parental population. In the field trial, each cross (containing between 5 and 15 progeny per family) was clonally propagated across the three randomized blocks in the common garden. These were then phenotyped over the course of several years. From this "real" population the GCA for the mothers and fathers and SCA for the pairs was calculated, to compare against the progeny predictions. This common garden also included the clonal replicates of the parents themselves which allowed us to build a genomic selection model were we calculate the genetic component of each parent's phenotype and use that to determine a "traditional" ranking based only on GCA from the parents. This ranking could potentially be considered "cheating" when those same phenotype values are used for

the calculation of the true GCA and SCA values. The data used to create the traditional GCA values were the trees grown along side the full progeny trial and included in the full "Real" calculation, meaning that it is contaminated and not independent of the real data and is also likely to have overfit the GCA values by using the same parental phenotypes.

3.3.6 Data Processing and Environmental Effect Considerations

When discussing a specific phenotype and the overall variance within that phenotype it is important to use equation 3.1 and attempt to assign variance to G, E, or $G * E$. By using a mixed linear model it is possible to assign phenotypic variance from the population mean to each of those individual factors. This is accomplished by using equation 3.1 to define a system of equations that look approximately like:

$$P_{i,j} = \mu + G_i + E_j + H_{i,j} \tag{3.5}$$

where the unique genotype from the data set is identified by i, the unique environment is denoted by j, H represents the effect of G in the specific environment E, or $G * E$ from equation (3.1), and μ is the population average of the phenotype. By having a wide range of genotypes across a wide range of environments it would be possible to accurately determine the amount of variance applicable to each parameter. With only a single environment, the equation effectively becomes:

$$P_{i,j} = \mu + G_i + E_j \tag{3.6}$$

where the total effect from the environment will be based on the micro-environmental differences that exist within the single plot. This can be measured by fitting the phenotype with plot coordinates of the samples to a thin plate spline (TPS). This TPS represents the variations in the phenotype caused by the local environmental variations. By removing the variation due to environment from the phenotype with this TPS, we produce a genotype-specific phenotype value. This value, when mean normalized, produces the genotypic component of the phenotype.

3.3.7 Calculating GCA and SCA

Within a progeny trial (real or predicted), it is known that the genetic component of the phenotypes are set to be determined by three separate factors: the GCA of each parent, the SCA of the specific combination of parents, and individual specific variation. By building a model of linear equations where each individual phenotype is equal to its specific components, we can solve for the value of each of these components. As long as there are more samples collected than total number of components across all genotypes, the set of equations is solvable, giving an exact report of GCA and SCA.

From the phenotypic predictions from iRF, we can then determine the predicted GCA of each of the parents and the predicted SCA of each of the parental pairs. This is accomplished by fitting a linear model to the full *in silico* progeny trial, incorporating features for each parent and each parental pair. The model is able to determine the coefficients for each that best explain the predicted phenotype for each individual progeny. The coefficients for the parents are their GCAs while the coefficients for each pair is the SCA for that pair. This is the same method used to determine the GCA and SCA of genotypes from a full progeny trial. An equation from this set looks like:

$$G_i = FGCA_a + MGCA_b + PSCA_{a,b} + g_i \tag{3.7}$$

where G_i is the genomic component of the phenotype from either equation 3.5 or equation 3.6 for a specific genotype. $FGCA_a$ and $MGCA_b$ are the GCA values, with a and b denoting which father and mother are the parents for this specific for this genotype. $PSCA_{a,b}$ is the SCA value for the specific pairing between Father a and mother b. The value of g_i accounts for the variation of the specific genotype in G_i that is seen between full siblings within a family. Due to the random nature of recombination, the PSCA value is effectively the average SCA value across the entire full sibling family with each independent genotype seeing a potentially different actual SCA value.

When the parental genotypes are included in a progeny trial their equation for G_i can be incorporated into that of the progeny. In the scenario of one of the fathers the equation

would look like:

$$G_n = FGCA_n \tag{3.8}$$

because the only genotypic contribution to the father's phenotype from variables within the model would be its own GCA value. The incorporation of parents in the progeny trial equations can help separate GCA values from the SCA of the offspring.

3.4 Results and Discussion

3.4.1 Pipeline

Following the steps from Figure 3.1, a set of 848 *Populus trichocarpa*[93] distinct individuals were grown in common gardens and measured for a variety of phenotypes, and subsequently underwent whole genome sequencing to characterize their individual genotypes. An iRF model was trained to predict the genotypic TPS adjusted height phenotype at age three from the common garden genotypes. This model was then used as a predictive model where new, outside *Populus trichocarpa* genotypes were fed into the model and predictions of their TPS adjusted height were obtained.

3.4.2 Predictions

The 7x7 parental population, which share no overlapping genotypes with those in the common garden, were synthetically crossed using SBVB,[184] to produce 49 *in silico* progeny full-sibling families, each with 500 unique genotypes. The synthetic genotypes of each progeny were used as inputs to the iRF model to predict the distribution of the phenotypes expected from the progeny of each parental cross. These full sibling families were used to determine genotypic variance broken down into GCA components by treating it like a true progeny trial.

A comparison between the predicted rank order for height of the full-sibling families from the progeny trial, the predicted progeny from the iRF process, and the comparison provided by the true progeny trial can be seen in Table 3.1). This table also includes the comparisons to the traditional ranking that only calculates the genetic components of the parents. The process only allows for the prediction of the GCA component and, thus, the missing values in the table. The correlation between the GCA of the Traditional Estimation and True Progeny trial is so high likely because the parental clonal replicates are used in both calculations. Effectively, the Traditional Estimation uses the same parents but does not include any of the crosses, which is why there can be no prediction for the SCA. Although the iRF Progeny Trial GCA correlations are lower than the Traditional Estimations, it can be seen that, because of the additional ability to calculate the SCA, the overall combined prediction is dramatically closer to the True Progeny Trial than the Traditional Estimation. The last row of the table shows the comparison to the true value of the total prediction of the genetic value for each of the families, or the combination of SCA and GCA. As seen in the table, the predicted *in silico* progeny captured significantly more information of the real total genetic component of the phenotype than the traditional model. This comes even with the traditional model almost perfectly predicting the GCA of the parents. This is likely due to the importance of including SCA in complex phenotypes such as height.

Table 3.1: This table shows the correlation between the predicted genetic components of the progeny trial from iRF and a simple parental trial to the True Full Cross Progeny Trial. The first column indicates which aspect of the genetic component is being compared, either the GCA from one of the sets of parents, the SCA from the crosses, or the combined prediction of GCA and SCA for the full genetic component prediction. The iRF Progeny Trial is the result of dividing the predicted iRF progeny into its genetic components and comparing it to the actual progeny trial. The Traditional Estimation is taken by defining the genetic component of the phenotype of each individual parent grown in clonal replicates at the progeny trial site. The Pearson Correlation is used to compare the effective magnitude similarity between the the Genetic Components and the Spearman Correlation is used to compare the rank order of the Genetic Components.

Genetic Component Prediction correlation to Real Progeny Trial				
Genetic	iRF Progeny Trial		Traditional Estimation	
Component	Pearson	Spearman	Pearson	Spearman
Mother's GCA	0.499	0.607	0.995	0.964
Father's GCA	0.406	0.321	0.985	0.964
Parent's SCA	0.430	0.492	-	-
Combined Prediction	0.400	0.440	0.078	0.0671

As shown in equation 3.1, a portion of the genetic component is confounded with the environment of the individual. This means that even when trying to isolate the genetic component of the phenotype, it is possible that some of the variance will be due to the genetic by environmental effect. This may explain why the iRF model loses accuracy compared to the traditional model, as the parents were grown in the same environment as the full progeny trial. The common garden data used to train iRF was from a different physical location and grown over the course of a different set of years.

To further show that the iRF progeny prediction pipeline is capturing the SCA of the pairs of parents as well as the GCA from the individual parents, we compare the predicted value of the GCA from each of the mothers and fathers from iRF to the actual GCA determined from the real progeny trial (Figure 3.2 and Figure 3.3, Table 3.1). For the mothers, we can see the close correlation between the Real values and the Traditional values. Even though the data points from the parental genotypes exist in both the real calculation and the traditional calculation, the removal of the the progeny from the system of equations for the traditional calculation hampers its predictive accuracy by removing the ability to discern what portion of the variation to assign to SCA. Despite this, we can see that the trend line for the traditional model is similar to that of the real progeny trial, while the only a few of the normalized predictions from iRF are close to their real counterparts.

The comparison of the fathers from predicted *in silico* progeny GCA to the actual GCA determined from the real progeny trial in Figure 3.2 shows a slightly worse story for the iRF model. Again we can see close agreement between the Real GCA and the traditional calculation. The trend line for the iRF model is slightly more separated.

By accounting for these GCA contributions to the iRF predicted phenotypes for *in silico* progeny, the predicted SCA can be determined for each of the parental pairs. In Figure 3.4 we can see the comparison of the predicted and true SCA for each parental pair. As with the GCA graphs, there is some agreement between the Real and Predicted models. We also note again that there is no Traditional data on this graph because without the inclusion of progeny in the model we cannot calculate the SCA values. This shows that the iRF model is, with a few exceptions, matching the overall trend with of the true SCA values.

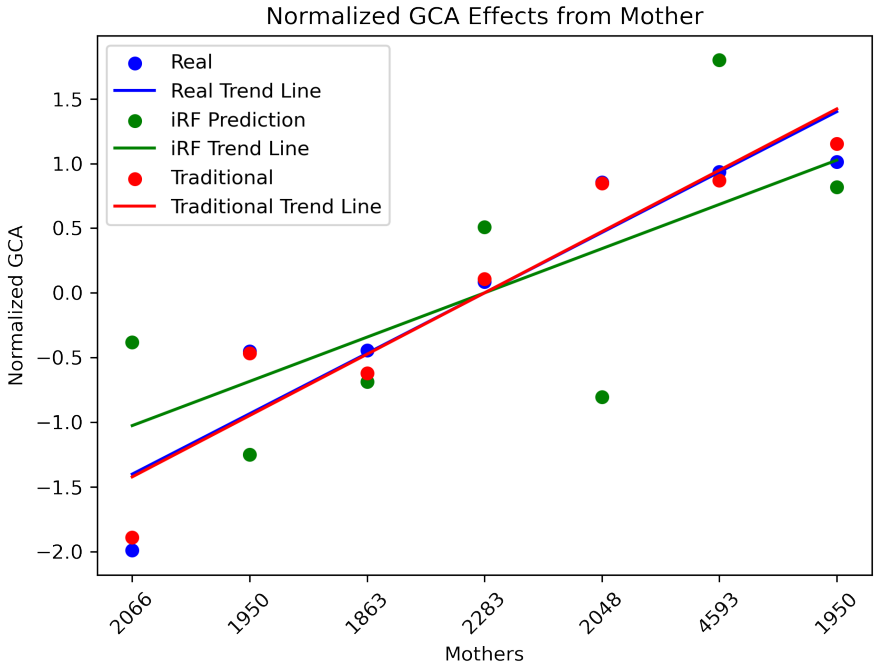

Figure 3.2: A comparison of the normalized values of the coefficients of variance attributable to the mothers from the *in silico* progeny height phenotype (green) to the coefficients of the real progeny trial(blue), and to the traditional calculation requiring the clonal replication of the parents(red). By sorting the Mothers by the Real values we can compare the trend line of the normalized values to the order from the order of the Real values. This graph illustrates how close the traditional method gets to capturing the value of GCA of the mothers but that there is still a decently strong correlation between the trend line of the Real GCA values and the iRF progeny derived Values.

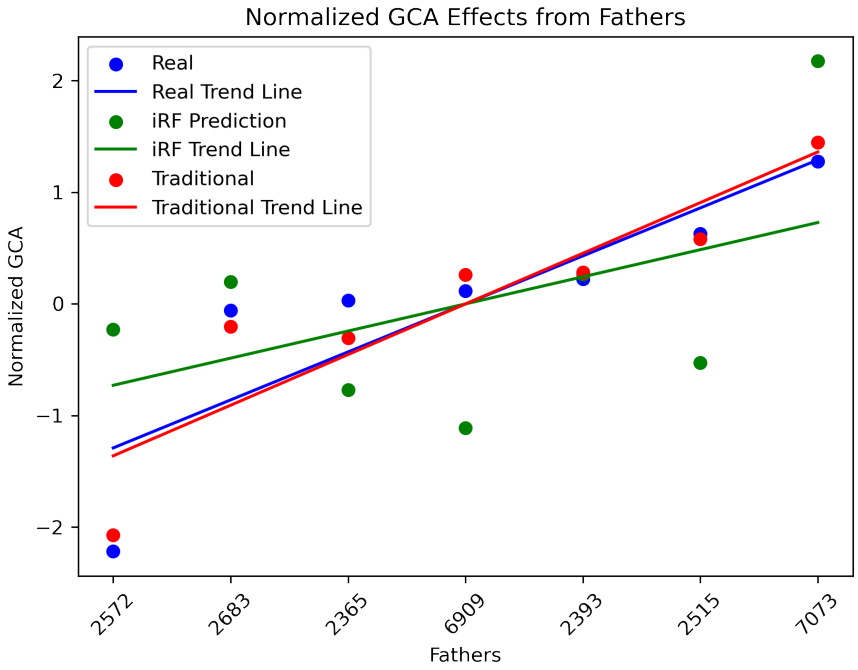

Figure 3.3: A comparison of the normalized values of the coefficients of variance attributable to the fathers from the *in silico* progeny height phenotype (green) to the coefficients of the real progeny trial(blue), and to the traditional calculation requiring the clonal replication of the parents(red). By sorting the Fathers by the Real values we can compare the trend line of the normalized values to the order from the order of the Real values. This graph illustrates how close the traditional method gets to capturing the value of GCA of the mothers but that there is still a reasonably strong correlation between the trend lines of the Real GCA values and the iRF progeny derived Values. By comparing the closeness of the trend lines to figure 3.2 we can see that the fathers from the iRF-based method don't match the real values as well as with the mothers, especially with the value of father 2572.

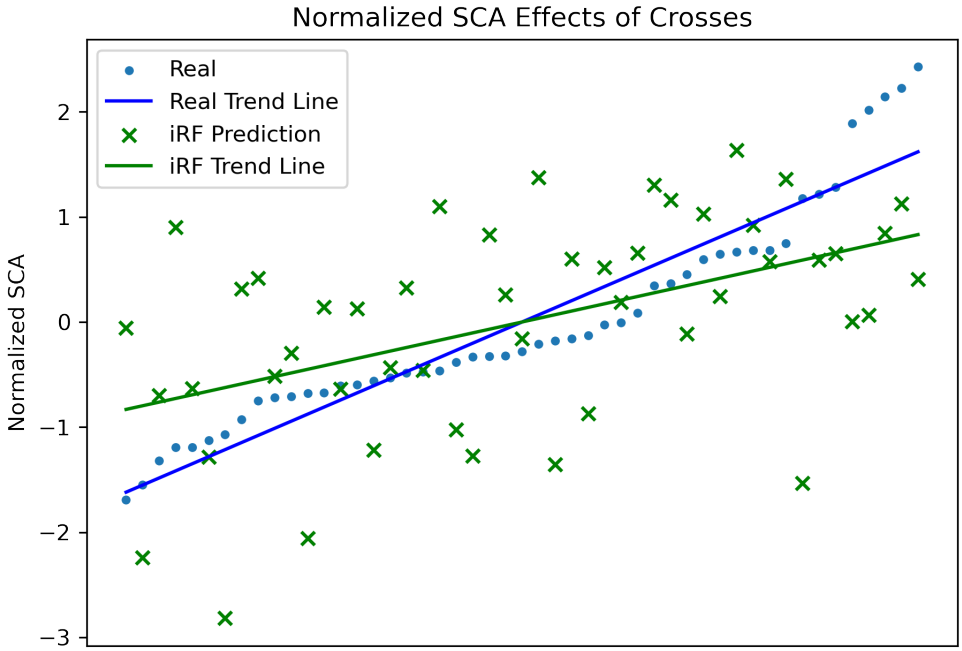

Figure 3.4: A comparison of the normalized values of the coefficients of variance attributable to the Combinations of the parents, or the SCA, from the *in silico* progeny height phenotype (green) to the coefficients of the real progeny trial(blue). Because it is impossible to caluclate the SCA of crosses with a traditional model without including progeny from the population, those values do not exist and are not shown in this graph. By sorting the Pairs by the Real values we can compare the trend line of the normalized values to the order from the order of the Real values. This graph there is a correlation between the trend line of the Real SCA values and the iRF progeny derived Values.

To visualize the overall genetic component of the phenotype, the SCA for each pair is combined with the GCA of each of its parents. Figure 3.5 shows the results of each parental pair with the combination of SCA and GCA for both parents. It can be seen that while the traditional calculation was able to accurately predict the GCAs, the lack of SCAs makes the method look almost random compared to the real results. The iRF progeny trial, while not perfect, is able to capture much more of the information from the real result, as shown by its trend line being much closer to the true line than the traditional method. Because the Traditional estimate only has GCA values, the final prediction of the crosses is just the combination of the parents' GCA values. By lacking the incorporation of SCA values, traditional methods are significantly hampered from capturing the full genetic component of the progeny families.

3.5 Conclusion

Here I have introduced a progeny prediction method that predicts the full genotypic component of a phenotype, which can be used to determine both GCA and SCA. With the availability of a sampling from the general population of the species, with both phenotypic and genotypic values, it is possible to produce predictions for the genetic components of the phenotype for crosses between individuals that did not exist in the original population. For breeders, the method provides a process for generating a progeny trial with resources and time necessary for a traditional trial.

The method and validations shown here indicate that the results of the iRF-based method more closely resembles the genetic component of real progeny than a traditional method that looks only at the potential parents. The acceptability of the accuracies presented must be determined by the individual breeder, however, the availability of SCAs with a positive correlation to true SCAs allows for better decision making. While the GCAs for the iRF-based method are not as good as the traditional method, the iRF-based method has the benefit of not requiring the measured phenotypes of the specific genotypes in the breeding population.

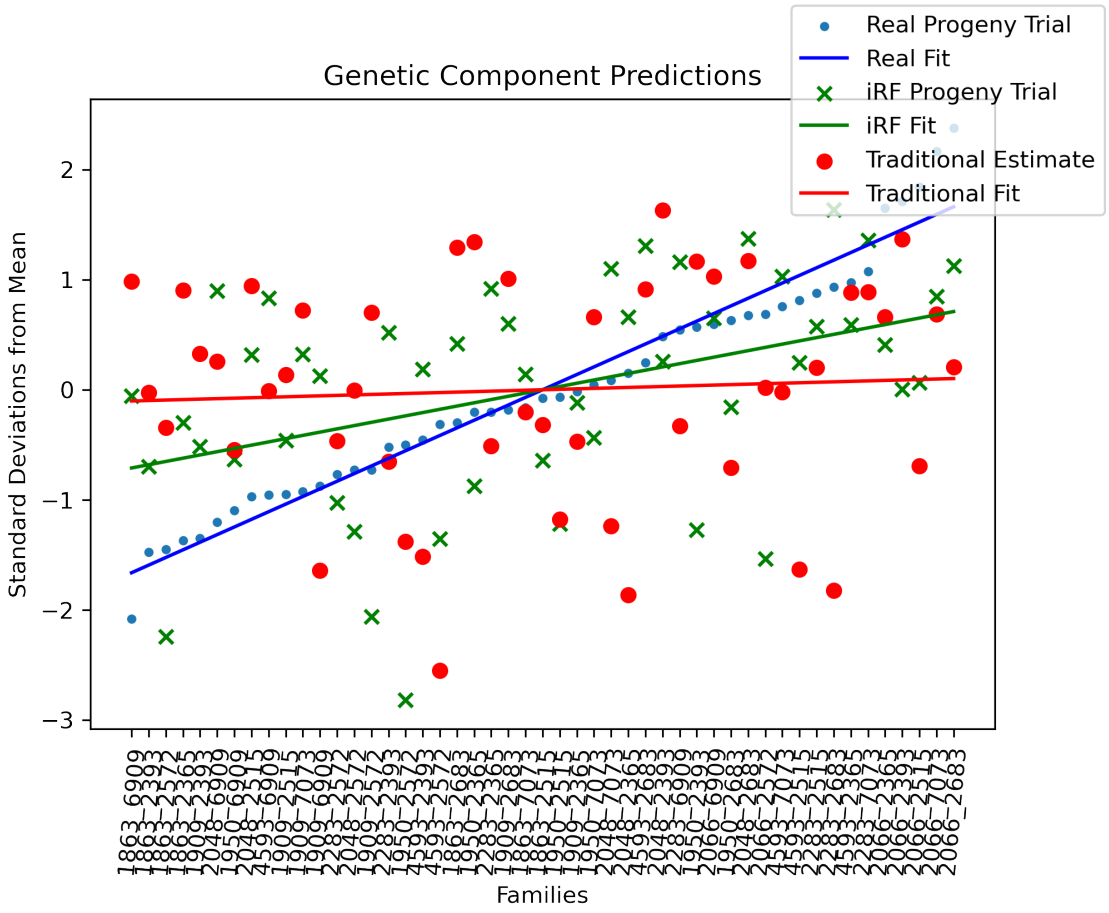

Figure 3.5: By ordering the families by their value in from the real progeny trial we can build trend lines for each of the predictions to illustrate the difference in each prediction. This plot illustrates the comparison of the normalized full genetic components from the Real Progeny trial shown as the blue dots and its trend line as the blue line, the iRF progeny trial shown as the green x's and green line for its trend line, and the Traditional Estimate in red circles and red line for its trend line. With the trend line we can see that the iRF Progeny trial is capturing more information than the traditional estimate when trying to predict the full Genetic component of the phenotype.

The new method also provides an interesting case for acceleration of important projects. With traditional cross testing, the only way to speed up the process is to increase the number of crosses done at a time. The process still requires the full species development time in order to determine the results, and the cumulative results will only be available at the end of that growing cycle. With the introduced progeny prediction method, by increasing the computational power available to the model, more *in silico* progeny can be produced at the same time, speeding up analyses in a way not possible for physical trials. However, due to the necessity of a general population to train the model, this method is limited to studies that contain relatively large amounts of data already available. This limitation hinders the method from being used for data sets with limited ability for population-scale genetic sampling and phenotyping.

With the viability of this method shown, future work includes improvements to the model that predicts the phenotype. The addition of the parents to the phenotype prediction could provide better accuracy for GCA. Also, the iRF model could be replaced with a deep learning model or another model that may have a higher overall predictive accuracy. Another potential way to improve the model would be combining multiple methods that excel at understanding different parts of the genetic components.

Chapter 4

Tensor iterative Random Forest

4.1 Abstract

Machine learning methods that can simultaneously evaluate feature to feature, feature to target, and target to target relationships are few and far between. However, having these types of relationships understood within a comprehensive model would be beneficial for fields such as biology where entire layers of 'omic data are compared against each other. Here, I introduce Tensor Iterative Random Forest (TiRF), a new Explainable Artificial Intelligence (X-AI) method that associates multiple independent features with multiple dependent targets within a single process and provides the information of what relates all combinations of features and targets. I also include model quality evaluations using the DREAM 5 gold standard networks, including two real biological data sets and one *in silico* data set.

4.2 Introduction

Applications for machine learning can be found in almost every scientific field, as seen by the plethora of machine learning methods recently developed and varieties of research being done, including health care[185, 186, 187], drug design[188, 189], agriculture improvements[190], software design[191] and many others[192, 193, 194, 195, 196]. Particularly within the field of biology, with its large quantities of features, machine learning methods have been found to provide novel and useful insights into complex problems[197, 198, 199, 200, 201]. One of the main uses of machine learning within the field of biology is feature selection, or finding the most likely features, or independent variables, responsible for the variation seen in a target, or dependent variable. These algorithms are often able to find patterns and associations that exist within the data much more efficiently than traditional analysis techniques, especially as the size of the data sets grow.

For many feature relationship or feature selection methods, the process involves finding associations between a set of features and a single target. However, biology often contains data sets where there are multiple targets rather than one, such as multi-omic data, where each 'omic layer, such as proteomic, transcriptomic, metabolmic, and genomic layers, each of which contain multiple potential targets. For these targets, simple methods that analyze

each target independently are possible and could provide the necessary insights into the feature to target relationships, but would likely miss out on the ability to learn from any inherent relationships from features to multiple interconnected targets. The types of interconnectedness may vary, for example, two separate phenotypes that are correlated with each other may be the result of pleiotropy, where multiple phenotypes are effected by the same gene or locus[202]. Pleiotropy is just one of the many types of complex biological associations that machine learning methods could provide knowledge about.

Machine learning techniques that predict a set of targets from a set of features exist mostly within the domain of Neural Networks[203]. However, many of these methods only focus on prediction and cannot provide easily interpretable justifications for decisions made and associations reached. It is possible to include explanatory layers in neural network models that are trained to produce human understandable explanations for specific predictions[22], however, these methods require extra steps beyond the original design and creation of the model. Explainable Artificial Intelligence (X-AI) methods inherently contain interpretability, which is important when the model associations are designed to aid in decision making processes, such as fields like medicine[204]. X-AI methods such as Random Forest (RF) and iterative Random Forest (iRF) provide feature relationships with human-understandable decision making[29], but are limited to a single target at a time.

Here, I introduce Tensor iterative Random Forest, a new feature selection method that not only finds relationships between a set of features and a set of targets, but also provides feature to feature interactions and multi-target relationships due to shared features. The method leverages X-AI methods RF and iRF, which innately provide the connection between features and target. RF-based methods also provide associations between features due to the conditional decision making steps inherent to tree-based methods[4]. This new method also relates targets to other targets by finding subsets of targets related by a causative feature. A brief example of what one of the trees from the model would look like can be seen in Figure 4.1. TiRF is evaluated using three DREAM 5 biological data sets, which contain well-studied relationships between transcription factors and the genes that they regulate.

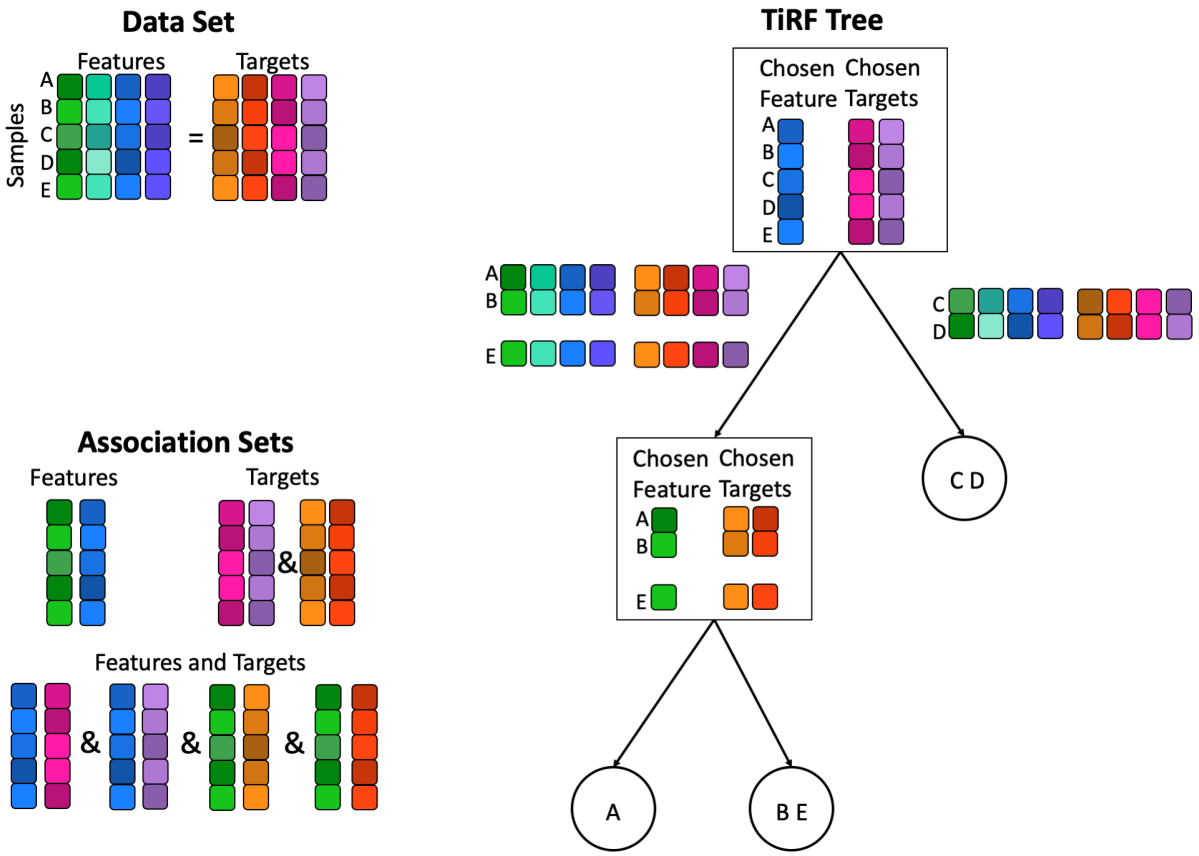

Figure 4.1: The top left corner shows the four columns of features and four columns of targets for an example data set with five samples. The right side of the figure shows a TiRF decision tree, where one feature is chosen at each split based on its ability to best split one or more chosen targets. The bottom left shows the different types of associations that are gathered from the tree, namely feature to feature, target to target, and feature to target. The leaf nodes at the base of each tree path denote the samples that reached that node, defining the sample sets with similar target values.

4.3 Methods

4.3.1 Random Forests

The Random Forest (RF) algorithm was published by Leo Breiman in 2001 [31] as a method for prediction, where a set of independent variables can predict a dependent, and feature selection, where independent variables (features) are evaluated for their effect on the independent variables (targets) have the dependent variable. The method builds on his earlier work of CART (Classification And Regression Trees) [30], which focuses on individual decision tree construction and understanding. By combining the concepts of bagging[3] with more random variation in the individual tree creation, Breiman produced Random Forests, that solved some of the immediate issues with CART, such as over fitting.

A Random Forest is a group of decision trees trained from the same data set. Each tree is provided with all of the features, the target, and a random subset of the total samples, selected with replacement up to the total number of samples in the data set. The most basic unit of the decision tree is the decision split, which chooses features to partition the samples from the parent node into children nodes with the intention of grouping samples with similar target values. This process is repeated on the children nodes until a stopping parameter is reached. To prevent over-fitting, a random subset of the features are looked at at each split.

Random Forest is a robust method for feature selection [32] by providing a feature importance for each independent variable used in the model. These feature importances are calculated by summing the contribution of the splits within the forest using each feature. While there are several cost functions used to determine the best splits, one of the more common methods for classification trees is Gini[31, 33], which calculates the impurity improvement from parent to child nodes, but is limited to classification-based targets.

For regression problems, one of the common cost functions, and therefore also a common importance score value, is variance explained. This score can be thought of as the difference between the total variance of parent to children nodes, where variance is defined as:

$$\sigma^2[X] = E[X^2] - E[X]^2$$

or

$$\sigma^2(i) = \Sigma(x - \hat{x})^2/n$$

where X is a random variable, E is the expectation of the function, or mean of the variable, and n is the number of samples. The total variance(TV) of the node can then be defined as:

$$TV(i) = n\sigma^2(i) = \Sigma(x - \hat{x})^2$$

where the variance is multiplied by the total number of samples in the node i. The variance explained is the difference in total variance(ΔTV) in the parent node (i) and the two children nodes (i_l and i_r) as defined by:

$$\Delta TV(i) = TV(i) - TV(i_l) - TV(i_r)$$

where the feature for the split is chosen by maximizing $\Delta TV(i)$. The importance of any given feature is the sum of the variance explained for all splits that use that feature.

Using feature importance, it is possible to address an issue that can occur in Random Forest; the random selection of features at each node can cause a majority of the splits to only consider non-causative features. This happens when the number of non-causative features far outweigh the causative features, resulting in the random sampling of features at each node to rarely include a true causative feature. A solution to this problem is to weight the random feature selection at each node to favour a subset of features that are more likely to be causative. Because RF already produces feature importances, those values can be used to determine which features are likely to be more causative. An algorithm called Iterative Random Forest (iRF) is produced by turning the model into an iterative process, with each new iteration using the importance scores from the previous iteration as a weighted distribution from which to sample the features viewed at each split[4]. This results in more efficient decision trees and more organized decision paths[4]. These paths each represent a route in a tree from root node to leaf node, and can be mined to find sets of features that co-occur often in the same decision path with an algorithm called Random Intersection Trees (RIT)[34], providing sets of potential interaction.

4.3.2 Tensor Iterative Random Forest

As the name implies, Tensor Iterative Random Forest builds upon the base design of iRF, and seeks to define a set of relationships between and within a set of features and a set of targets. This relationship between features and targets is analogous to the way tensors can define the relationship between two vectors. To connect multiple targets together without disrupting the explainability of the model, multi-dimensionality, or the ability to split on multiple targets at each node, is implemented in iRF. Using the framework of iRF, TiRF can be constructed with a two step addition: first, implement a cost function that defines the split value of multiple targets at once, and second, implement a process to limit each feature to explain a specific subset of targets.

Cost Function

The first step is to define a process that can calculate the gain in purity of the node within the tree for multiple targets, which requires adjusting the way an individual split works within a node. To determine the best split for a node, a cost function defines how to compare different splits with multiple variables. The most effective way to determine the optimal split with multiple targets is to calculate a single score that aggregates the total amount of variance explained for each of the targets of the split.

One such solution is to use the set of targets to define a multi-dimensional space, where each target represents an orthogonal direction. The magnitude of each sample's target variables determines the location of the sample within the space. Comparable to the use of variance within iRF, a value within TiRF is defined as the distance to the center of the group of samples. The overall center of the node is a point within the space with target values equal to the average value for that target within the current subset of samples, defined as:

$$\hat{y}_{in} = \frac{\sum_{j=1}^{N_n} y_{ijn}}{N_n}$$

where $\hat{y}_{in}$ is the average, and therefore center, of target variable y_i for node n, that contains N_n total samples, labeled from $j = 1$ to $j = N_n$. The distance from each sample to the center, D_n, can then be calculated as:

$$D_n = \frac{\sum_i \sqrt{\sum_j (\hat{y}_{in} - y_{ijn})^2}}{\sum_i 1}$$

which is the sum of the total distance for each target i of the square root of the sum of the distances for each sample j to the center point $\hat{y}_{in}$. The summation is divided by the total number of targets evaluated at the given node. The value is the average distance to the center across different targets, normalizing the total distance to be the average distance for each target. This normalization becomes important when comparing splits if they have different numbers of targets.

The optimal split is the one that minimizes the sum of distances, the cost function, to the center of the samples for the two children nodes. Each split in each potential feature is used to create the potential children nodes containing the subset of samples from the current node, and the feature and split that minimizes the cost function is chosen as the split for the node.

Creating Target Subsets for Each Feature

As the model scales up to more targets, the computational cost of calculating the distance increases and the effectiveness of any given split is determined by how it explains variance of all dependent variables at the same time. A negative consequence of the second point leads to the favoring of splits and features that explain, to at least some small extent, all of the targets at once, regardless of overall effectiveness of the split on each target, rather than features that explain a subset of the targets. To address this, a second step of the process is required.

By restricting the targets to a subset that a given feature evaluates its splits against, the feature is allowed to better explain those specific targets. This raises the question of how to select the subset of dependent variables. In order to best accomplish this without disrupting the decision tree structure, a different subset of targets can be selected for each feature within the subset to be analyzed for the split.

To keep the variability high from node to node, as Breiman originally intended for RF, the set of potential targets to address in a node is a random subset of the full set of targets,

where the quantity is set by the square root of the total number of targets in the data set. For each potential feature in a node, several methods can be used to determine the set of targets from that random subset with the closest associations to that feature, such as any algorithm that provides dimension reduction, such as LASSO[84] or even iRF itself. The final number of targets for each feature is determined by either a quantity cut off or an association score threshold.

For the current implementation of TiRF, the absolute value of Pearson's correlation from target to feature is used to determine the association value for each target within the random subset for a given node. These values are sorted to take either the five targets with the greatest association or every target that met the association value threshold of .1, whichever has fewer targets. The set of targets is assigned to the feature for only a single node, as every new node requires a new calculation, and each potential feature receives a unique set of targets.

Using the different target subsets for each feature, an optimal split value can be determined using the method described previously. Because the distance metric is normalized for the number of targets in each subset, it allows for the selection of the feature that provides the best split for the node, regardless of number of targets. Using the normalized distance metric allows the splits that explain fewer variables well to be comparable to splits that explain several variables less well, and insures that splits that explain multiple variables well at the same time will always be favored.

Associations Derived from TiRF

As mentioned previously, TiRF retains the ability from RF and iRF to build associations from feature to target and feature to feature. The feature to feature relationships are determined in a similar way as those methods, by determining the features that co-occur in decision paths most often. For every pair of features, co-occurrence is calculated as the count of the number of pathways where both features of the pair-wise set occur. Due to this, the feature that splits the first node in a decision tree will be counted as having an occurrence with each other feature in all paths for the tree. These pair-wise sets of features can then be ordered by

the number of times they co-occur, with the highest number of co-occurrence sets providing the strongest feature to feature associations.

The feature to target relationship quantification differs slightly from the process used in RF and iRF. Rather than summing the importance scores of each instance that a feature occurs in the model, each feature has an importance score for each target. This importance is the sum of the improvement to the cost function by the feature when the specific target is included in the cost function for that feature at that node. Because the distance calculation is able to determine the improvement for each target independently before normalization, 'target-specific improvement to distance to center' is assigned as the importance for the feature to target relationship.

By tracking the targets selected for the chosen split feature of each node, relationships can be built between targets in a similar, but distinct, way as is done for the feature to feature associations. For every pair of targets, the co-occurrences are counted as the total number of times both targets are in the set of targets chosen for the feature that split each node. Because targets are calculated in this manner, rather than simply targets within the same pathway, the association between targets is specifically dependent on their causative features. The strongest relationships between targets are assigned to the target pairs that co-occur in these subsets the most frequently.

Due to the nature of TiRF, not every target is guaranteed to be predicted by the model, in the same way that RF, iRF and TiRF, do not guarantee that every feature will be used in the model. For a target to be included in the final model, the target must be in the pool of randomly selected targets for a node, the target must be one of the best predictors of at least one feature from the set selected for the node, and that feature must be the chosen feature to split the node. The possible sampling issue is counteracted by building multiple trees, as is used by Random Forest models to balance the features.

Implementation

Tensor Iterative Random Forest was implemented in Python, with influence taken from the implementation of Random Forest in scikit-learn[158], using a new implementation but the same structure of tree information storage. The Numpy package[205] was included

to assist with mathematical functions including the optimized calculation of mean and correlation values, and the Pandas package[206] was included for dataframe management when processing feature importance scores. While the program used to create the TiRF models is relatively straightforward, the steps to process the resulting importance scores, pathways, and relationships is more complex. These post-training processes require tree traversal to construct each of the individual pathways to enable the construction of co-occurrence matrices and calculation of importance scores.

One step that should be noted for the implementation of TiRF is the calculation for the feature weights after the first iteration, as it differs slightly from iRF. For the first iteration, or iteration 0, the total importance for each feature is calculated as the sum of the importances from that feature to all targets. The total importance value is then treated like the iRF importance value: normalized and used as the feature weight for the subsequent iteration. If a feature is never used for any targets, as determined by the model, then the resulting sum of importances will be zero and subsequent iterations will not include it in the feature selection.

4.3.3 DREAM Competition data

The Dialogue on Reverse Engineering Assessment and Methods (DREAM) project, held a competition between multiple network inference methods with data sets derived from *Escherichia coli*, *Staphylococcus aureus*, *Saccharomyces cerevisiae* and *in silico* microarray data from the Gene Expression Omnibus (GEO) database[207, 208]. These data are sets of samples with expression data from the given organism, normalized using Robust Multichip Averaging (RMA)[209]. The expressed genes are divided into transcription factor (TF) and non-transcription factor (non-TF) groups. Each data set is also accompanied by a "Gold Standard" network connecting gene nodes to associated transcription factor nodes based on prior literature. It should be noted that, according to the original dream authors, by building the "Gold Standards" from prior tests and experiments, it is possible that some relationships are missed, indicating that found relationships that do not correspond to associations within the Gold Standard are not necessarily incorrect or false positives. However, they have a high

degree of confidence in the edges present being true, so any evaluation should be viewed as a lower bound of the efficacy.

To analyze the effectiveness of TiRF, data sets 1, 3, and 4, or the *in silico*, *Escherichia coli*, and *Saccharomyces cerevisiae* respectively, of the Dream 5 challenge were used to create models. The *in silico* data set is based on *E. coli* data and contains 195 TFs and 1,448 non-TFs for 805 samples, with the gold network containing 4,012 edges. The prokaryotic model organism *E. coli* data has 334 TFs and 4,177 non-TFs for 805 samples, with 2,066 edges within its gold network. The eukaryotic model organism *S. cerevisiae* has 333 TFs and 5,617 non-TFs for 536 samples, with a total of 3,940 edges in its gold network.

For each model, the TFs were used as the feature matrix while the non-TF genes were used as the target matrix. The resulting feature to target importances from the model define the effect of each TF on every non-TF. With a value 0 from TF to non-TF the model predicts no connection, and there is no guarantee every non-TF is predicted or every TF used for a non-TF. The TF to TF effect is defined by the interacting features, feature to feature co-occurrence, effectively one TF impacting the prediction of another TF. In the TiRF model this is represented by a TF splitting the samples into children nodes and then another TF being the best predictor among the selected set. If the pattern between two TFs occurs often, it implies that there is a relationship between the samples selected by the first TF and the second TF.

Within the scope of the DREAM networks there are TFs that regulate other TFs. The third set of results from TiRF is the non-TF to non-TF relationships, which do not occur in the DREAM Gold Standard networks. Due to the lack of non-TF to non-TF associations within the Gold Standard networks, new gold networks were created from the existing data. As shown in Figure 4.2, for each pair of non-transcription factor genes being regulated by the same transcription factor, the new gold network contains an edge between the two non-transcription factors. By repeating this process across all associations for each of the three networks, three gold standard networks for the non-TF factor to non-TF associations were developed.

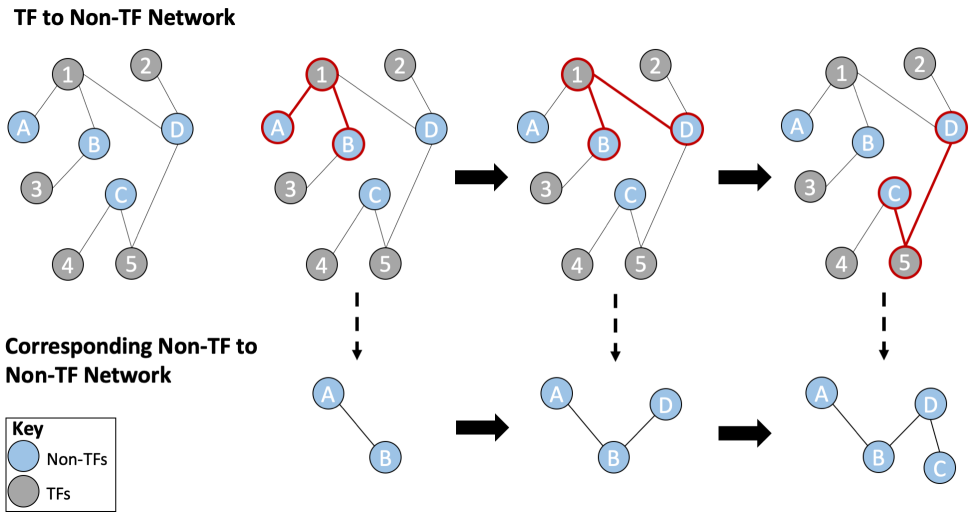

Figure 4.2: The process of creating non-transcription factor to non-transcription factor gold standard networks from corresponding transcription factor to non-transcription factor associations. For each non-transcription factor gene, if it is regulated by the same transcription factor as another non-TF gene, then the two no-TF genes contain an edge associating them in the new network.

4.4 Results and Discussion

4.4.1 TiRF Model

For each of the DREAM 5 networks 1, 3 and 4, a TiRF model was trained to connect the transcription factor (TF) genes as features and the non-transcription factor (non-TF) genes as targets. Tensor Iterative Random Forest was able to successfully capture many of the edges from each of the Gold standard networks. The models where constructed with 5 iterations, 500 trees per iteration, mtry (number of features sampled for each node) of the square root of the number of total features, ytry (number of targets sampled for each node) of 50, and a target leaf node size (the minimum number of samples a node must contain to split) of 50.

4.4.2 DREAM Network Construction

Once the TiRF model for each of the DREAM networks is created, they were compared to the gold networks provided. Using network visualization software Cytoscape[210], an example visualization of the gold standard network can be created. The feature to target associations found with TiRF are then overlaid. The Gold Standard and TiRF associations from Network 1 (the *in silico* network), for the second iteration, are shown in Figure 4.3. The pink edges show the TF to non-TF gold edges that were captured by TiRF, the green edges show the TF to TF gold edges that were capture by TiRF, and the light grey edges show the gold edges that are not captured with TiRF. The blue nodes show the transcription factors and the orange show the non-transcription factors. Note that many of TF nodes are surrounded by groupings of non-TF nodes, indicating a many-to-one relationship where a TF may be responsible for the regulation of multiple non-TF genes. Also a single gene, TF or non-TF, may be regulated by more than one TF.

The general overlap in the figure shows that TiRF is able to capture edges throughout the network and not isolated to one specific region or transcription factor. The way the pink and green edges seem to segregate into different regions of the network potentially indicates areas where one type of relationship, TF to TF or TF to non-TF, is stronger than the other. The

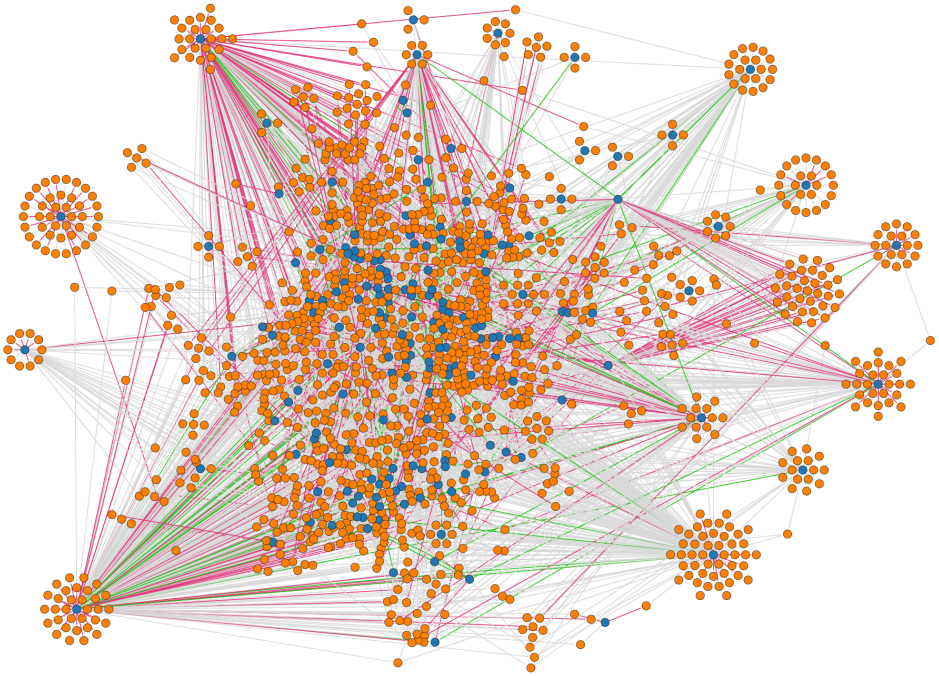

Figure 4.3: The gold standard and TiRF overlap from the from the Dream 5 Network 1. The blue nodes represent the transcription factors, and the orange nodes are the non-transcription factor genes. The pink edges indicate the overlap of TF to non-TF from TiRF to the gold standard, while the green indicate the TF to TF overlap. The gold standard edges not captured by TiRF are indicated in light grey. It can be seen in this image that there are regions of the network that are being better identified by TiRF, specifically some of the transcription factors are much better captured by the model than others.

figure also shows how, for some of the TFs, nearly, if not all, of its connections are captured where other TFs will potentially have only a single edge captured by TiRF. Unfortunately, due to the large size of the created non-TF to non-TF gold network, visualization of the whole network is not feasible, as the network contains approximately 400,000 edges, and the TiRF model provides 67,704 predicted edges.

4.4.3 Dream Network Accuracy

The created networks can be evaluated based on how effectively they capture the information from the gold standard with the use of Receiver Operating Characteristic curves (ROC). For ROC curves, the predicted edges are ordered by the models confidence in the edge, by importance score for TF to non-TF and by co-occurrence count for both TF to TF and non-TF to non-TF. ROC curves allow for quick examination of the model to see how quickly the False Positives increase as predictions are added and the total capture of True Positives increases. An important sign of a good model is an ROC that has a steep incline for small false positive rates, indicating that the highest rated values of the model are more True than False. These curves show how quickly the True Positive Rate rises as the allowance of False Positive Rate increases, indicating how well the model captures the correct information. The area under the ROC (AUROC) provides a numerical score of the quality of the model, where a perfect model would have an AUROC of one. As noted in section 4.3.3, with the DREAM data it is possible to incorrectly identify true positive results as false positive results due the incomplete nature of the gold networks, indicating that the AUROC results represent the lower bound of the model quality.

Figure 4.4 shows the ROC for TiRF on network 1, 3, and 4, measuring the ability to capture TF (feature) to non-TF (target) gene relationships. In this figure, particularly a) and b), it can be seen that the model is not only able to capture the information from the gold standard network, but does so efficiently by having a steep beginning, where the rate of increase in true positive outpaces that of the false positives. This indicates that the highest importance connections between features and targets are more likely to be true positive than false positive by a significant margin. The jaggedness seen in b) and c) is caused by having roughly half the number of true edges in the gold networks as compared to in network 1.

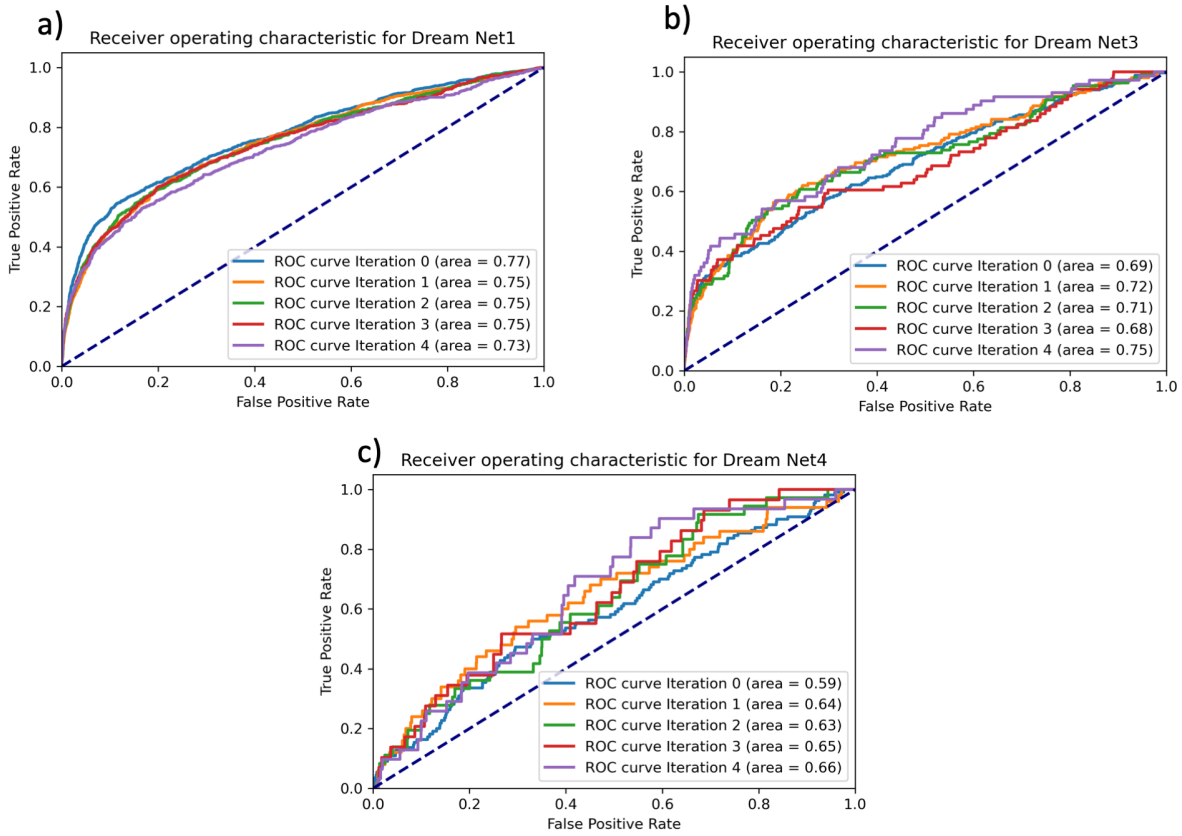

Figure 4.4: The receiver operating curve (ROC) for the first five iterations of TiRF on the Dream 5 Networks 1, 3, and 4, measuring the accuracy of the model at capturing transcription factor genes to non-transcription factor genes as set by the "gold" standard from the Dream competition. It should be noted that the according to the original competition and construction of the "gold" standard the True Positive rate is much more important than the False Positive rate. a) The ROC for DREAM 5 Network 1 shows TiRF is able to capture the Transcription Factor to gene relationship from this synthetic data. However, while the iterative process reduces the total number of edges through refinement the overall AUROC of the model is slightly effected negatively. b) The ROC for Dream 5 Network 3 shows a slightly worse performance than the synthetic data, with the iterative process having a slightly chaotic effect on the over all AUROC, potentially indicating more noise in the data. c)The ROC for Dream 5 Network 4 has a slightly worse result than either of the other two networks, but is still above the baseline shown by the diagonal dashed line. For this network a general improvement can be seen in the AUROC with the iteration process.

Figure 4.4 also shows the slight differences that result from the different iterations across the different data sets, with network 1 showing a steady decrease, network 3 showing variation and network 4 showing a general increase. The effect this refinement has on the model appears to be dependent on the underlying data. While network 1 (a), with *in silico* data, has comparable AUROC values across all iterations, the relationship between iterations in network 3 (b), which contains the original network used to develop network 1, fluctuates more. This may indicate that the relationships developed for the *in silico* network contain less noise than the real data. Network 4, which is also real data, is able to make improvements as the iterations advance, although the total AUROC is lower than either of the other two. It should be noted that the iterations beyond the first, because of the refining of feature selection, contain fewer total connections and fewer unique transcription factors with non-zero importances. As the model goes through its iterations it favors features that explain the most variance across targets potentially losing TF to non-TF edges that aren't as strong. Overall, all three models, across all iterations, provide significant results.

Figure 4.5 shows the ROC graphs for the TF to TF relationships captured by TiRF in networks 1, 3, and 4. When comparing the undirected connections of co-occurrence to the directed edges from the gold network, if either direction of the TF to TF relation exists in the Gold Network it is considered a True Positive prediction. Comparing across the three graphs, it is clear that the model's ability to capture the underlying feature to feature relationships of the data varies for each data set and between each iteration. Network 1 appears to have generally positive capture rate of the underlying information, with reversed relationships to iterations, comparable in ranking to its graph in Figure 4.4. Network 3's scores fluctuate, first increasing and then decreasing with the iteration increase, potentially indicating that the feature pruning process done within the iterations removed TFs that were redundant and causing the feature importance to be split amongst multiple features, while the later decrease may be due to the removal of too many features. Network 4 performs poorly and only gets worse with iterations likely because it had the fewest samples to train from. These factors combined with TiRF's ability to preform well when connecting TF to non-TF likely indicates that the TF to TF relationships from this network were hardest to capture and TiRF was also poor at capturing these connections.

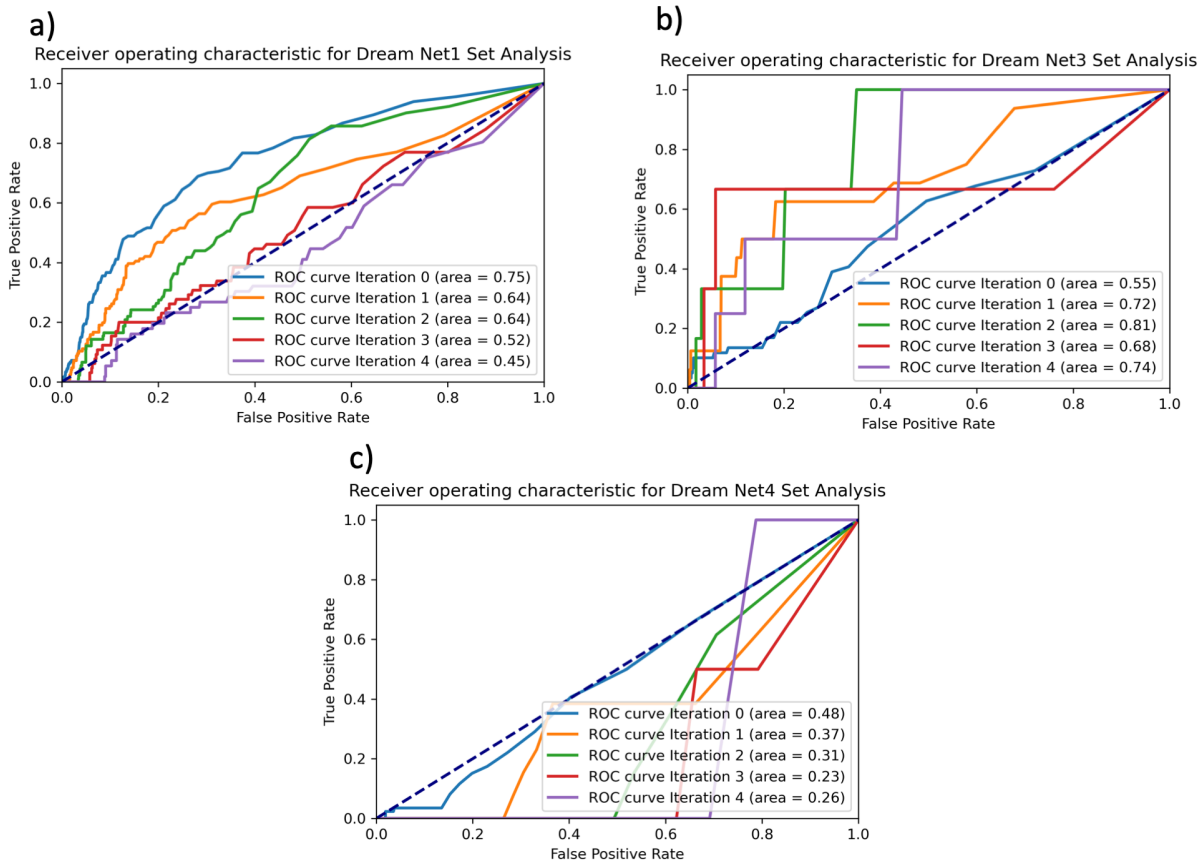

Figure 4.5: The receiver operating curve (ROC) for the first five iterations of TiRF on the Dream 5 Network 1, 3, and 4 measuring the accuracy of the model at capturing Transcription Factor genes to other Transcription Factor genes as set by the "gold" standard from the Dream competition. These results are limited by the co-occurrence of features in the models and by the presence in the "gold" standard of Transcription Factors predicting each other. a) The ROC for Dream 5 Network 1 shows that the First iteration (Iteration 0) captures the feature to feature interaction in this data the best by far. As the iterations continue and the selection of the features becomes more heavily determined by feature weight we can see the AUROC drop off to the point where the final iteration is below the baseline. b) The ROC for Dream 5 Net 3 shows a slightly worse picture for the first iteration than Network 1's first iteration but the subsequent iterations all have much higher AUROCs. c) The ROC for Dream 5 Net 4 has by far the worst overall accuracy, with every iteration being below the baseline.

As with the TF to non-TF ROCs , it should be noted that the iterations beyond the first, because of the refining of feature selection, contain fewer total connections and fewer unique transcription factors with non-zero importances. These graphs highlight that TiRF model may not be the best selection for finding TF to TF relationships because of the limited known true connections that exist from TF to TF within the data set. For the networks 1, 3, and 4, the total TF to TF connections that exist within the gold networks are 246, 143, and 324, respectively. Comparatively, the total number of possible TF to TF connections are 18,915, 55,611, and 55,278, respectively. This limited quantity of connections is also the reason for the discrete step-function style lines for the graphs.

Figure 4.6 shows the ROC for TiRF on networks 1, 3, and 4 measuring the ability to capture non-TF (target) to non-TF (target) gene relationships. The new gold network created for this analysis is described in section 4.3.3. Due the nature of TiRF, a large quantity of target to target associations are provided, with frequency of co-occurrence indicating importance. To limit the results to those most relevant, only the top 25,000 most co-occurring sets of targets for each network were analyzed, out of 33,852 from Network 1, 53,304 from Network 3, and 48,506 from Network 4. All three networks have consistent structures across iterations and all are better than the baseline. Network 1's AUROC has small fluctuations for each iteration and Network 3's AUROC has small improvements for each iteration. Network 4's performance fluctuates with each iteration, but the first iteration performs the best.

The Figure 4.6 shows that TiRF's ability to capture the underlying relationships from targets to other targets of the model is relatively consistent across Network 1 and 3, with Network 4 being the worse than the other two, which has been consistent across the association types. The AUROC for Network 1 are more stable across iterations, only varying by .02, while for Network 3 and 4 the value varies by up to .05, this is likely due to the more deterministic nature of the *in silico* data set used in the creation of Network 1. While not quite as good as some of the feature to target results, all three graphs do indicate the correct capture of true positive target to target results. As the novel contribution for TiRF, these results show that the method is able to deliver on the ability to capture target to target relationships.

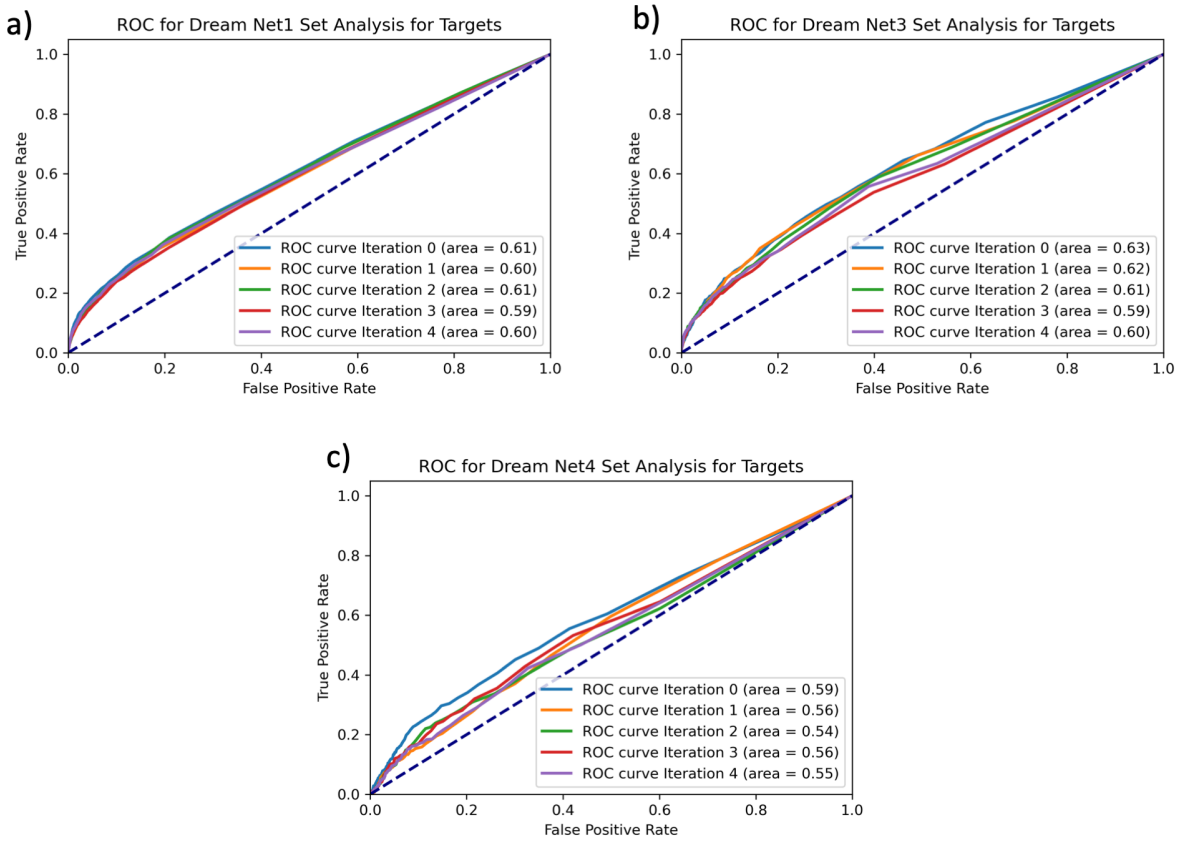

Figure 4.6: The receiver operating curve (ROC) for the first five iterations of TiRF on the Dream 5 Network 1, 3, and 4 measuring the accuracy of the model at capturing non-Transcription Factor genes to other non-Transcription Factor genes as set by the new "gold" standard, derived from the Dream competition. The derivation is performed by connecting all genes that are predicted by at least one common transcription factor. Because of the number of overall connections in the network reaching into the millions, these ROC graphs represent only the top 25,000 most commonly co-occurring target sets from each model. a) The ROC for Dream 5 Network 1 shows that TiRF is able to capture the target to target relationships within the synthetic data set with a small amount of variation over the course of the iterations b) The ROC for Dream 5 Network 3 shows a slightly better capture of the "gold" standard edges than the synthetic data in Network 1, but does show a slight decrease in AUROC with each iteration. c) The ROC for Dream 5 Network 4 has the overall worst results in capturing the target to target relationships, however each iteration is better than the baseline. This network also has the largest fluctuation in AUROC from iteration to iteration, with the first iteration capturing the most information.

4.5 Conclusion

Overall, TiRF has been shown to capture relevant information in three validated data sets for feature to feature, feature to target, and target to target relationships, all within a single model. The network visualization shows that the results of the feature to feature and feature to target associations broadly cover the full truth space, even though they do not capture the entirety of the truth set. The various ROC graphs illustrate that the number of iterations will have different effects on different types of data, as well as on the different types of feature and target relationships. Future work could be done to characterize the intricacies of how these parameters of the data interact. The specific results shown in this chapter illustrate that TiRF is able to capture the information in the relationships between TF and non-TF genes. Because the method is not specially designed for these data sets it should generalize to any omic to omic layer relationship in biology that has true underlying relationships and enough data to adequately train the TiRF model. This technique could be applied to single cell data omics data[211], the relationship between gut microbiome and intestinal health[212], large scale multi-species inter-omic data[213], and even food production for products like cheddar cheese[214].

Future development could also be done for the implementation of the TiRF code. The current version of TiRF is implemented in Python without threading or multi-node communication capabilities and as such is limited in the size and intricacy of the data sets that it can handle. Considering the large quantity of features available in many biological omics data sets, the ability to utilize the computational speed and memory allocation of modern HPC systems would be a significant advancement. The most straightforward solution would likely be to distribute tree creation across separate computational node, parallelizing the tree creation process, as is done in Cliff et. al.[9] with Iterative Random Forest. An implementation with better computational capabilities would also open the door for analysis of higher order interactions, as the current model does only pair-wise associations for each of the association types.

The current implementation of TiRF is also constrained by the the y-reduce and the multi target cost function; both processes use underlying functions that can be replaced with

similar functions that target different information. This can help TiRF focus on different aspects of the information or be able to handle different types of noise for convolution. A better understanding of how the different functions compare across data sets would be a useful metric when implementing the TiRF method.

Chapter 5

Building a Genetic Lineage of SARS-CoV-2

5.1 Abstract

The COVID-19 pandemic has had a worldwide impact and strains are continuing to evolve and change over time. Recombination provides a way for the virus to spread its advantageous mutations across multiple lineages, potentially amplifying the infectiousness of more harmful strains. Here, the potential for recombination and history of recombination of the SARS-CoV-2 virus has been analyzed. With this analysis it is shown that regions of its genome are incompatible from an evolutionary perspective with other regions of its genome, likely the result of recombination. Individual strains of the virus are also examined for evidence of recombination and the prediction of their possible parental strains.

5.2 Introduction

Due to the devastating effects of SARS-CoV-2, it is important to understand the evolution of the virus and how it has adapted to the human host over the course of the pandemic. The global impact of COVID-19 has encouraged a worldwide effort to collect data, which, among other things, has lead to a large amount of viral sequencing data being made available, making it possible to build a close to real-time mutational history of the SARS-CoV-2 genome. This information, if properly mined, can provide valuable information into what viral functionality is changing[215, 216], and how it could change in the near future. Given that homologous recombination is common in coronaviruses[217], research into whether or not virus strains are occurring via recombination is of particular interest, as this process may combine dangerous variants and allow strains to escape current selective pressure constraints[143].

The large amount of data available can also expose bottlenecks in pipelines designed to analyze the evolution of sequences over long periods of evolutionary time[218]. The calculation of phylogenetic trees, the traditional tool used to analyze the evolutionary relationships across species, makes assumptions about the nature of sequence data and evolution[157, 134]. Phylogenetic trees assume purely divergent evolution, where different evolutionary paths cannot cross back to each other, such as with convergent evolution or

recombination. These processes do not fit the model of the assumed tree structure and are thus often assumed to not exist or are purposefully removed from the data[219].

The occurrence patterns of single nucleotide polymorphisms (SNPs) are said to be "incompatible" with each other if it is impossible to build a standard phylogenetic tree without duplicating a mutation, where the mutation has to occur on multiple branches of the tree. An illustration of this can be seen in Figure 5.1, where, if convergent evolution is assumed, a specific mutation has to uniquely occur twice in different branches or, if recombination is assumed, the separate branches must reconnect. Both of these situations are more likely to occur under strong evolutionary pressure for a specific trait. Regardless of the amount of evolutionary pressure, convergent evolution is constrained by the underlying mutation rate of the virus, where recombination is only constrained by the co-occurrence of different sequences with-in a cell. For mapping short timescale evolution in organisms where recombination is possible, phylogenetic tree methods can miss events that are important for tracking evolutionary changes, as well as notable, unexpected functional changes.

To incorporate these possibilities, a network model can be used instead of a tree model to denote evolution. Each node of the network denotes a haplotype, defined as a unique sequence of the virus, connected to each other based on genetic similarity. A network model allows for the occurrence of loops, where there can be multiple possible paths between haplotypes, a pattern that implies convergent evolution or mutations transferred between diverged individuals via recombination [135].

Due to potential interactions among genes within the virus, singular mutations can have wide ranging effects[220]. Ongoing research and observation is required as new lineages with variant of concern-like mutations continue to be found[133]. As the number of mutations continues to increase, the virus may be able to escape the immune response targeted by the initial vaccines (and/or previous infection) and possibly become more efficient at infection and transmission. Specifically, mutations within the spike protein of SARS-CoV-2 will require new vaccines to appropriately combat the new variants[24]. With recombination, it would be possible for a mutation that was adaptive to one specific selective pressure to be combined into a new strain containing other mutations that were adaptive to a separate selective pressure. A visualization of what recombination would look like within a single

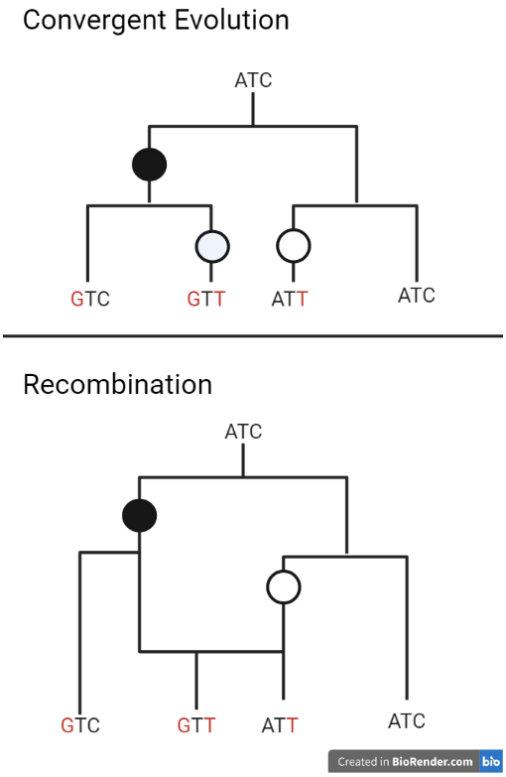

Figure 5.1: The visual difference of Convergent Evolution vs Recombination. For the set of sequences: GTC, GTT, ATT, and ATC, in order to create a phylogenetic tree with purely divergent branches you create a scenario of Convergent Evolution were the first mutation of A to G (black circle) occurs and then the C to T mutation (white circle) occurs twice in the tree in separate locations. However, if recombination is allowed to occur, the same tree can be constructed with each mutation only being required to occur once. It can be seen that, by connecting GTC and ATT with recombination, it is possible to attain the sequence GTT.

stranded virus like SARs-CoV-2 can be seen in Figure 5.2. If two separate viral strains are present in the same cell, in a process known as template switching [221], viral replication could be interrupted and restarted with the RNA from the other viral strain, resulting in a new sequence that contains mutations from both. Template switching has been shown to occur in SARS-CoV-2 [222].

Here, I provide evidence that recombination is likely to be occurring within the SARS-CoV-2 virus via both a population wide measurement of SNP incompatibility and a individual examination of specific haplotypes of interest. By evaluating population-wide incompatibilities, it is shown that mutations are under significant evolutionary pressure to conform to specific traits, either through convergent evolution or recombination. Due to the underlying mutation rate of SARS-CoV-2, convergent evolution is shown to be unlikely. A network-based approach to phylogentic associations is also provided and is used to find haplotypes with potential recombinatorial origins and find likely parents via *in silico* sequence reconstruction.

5.3 Methods

5.3.1 SARs-CoV-2 Data

SARS-CoV-2 sequences from all over the planet have been entered and stored in GISAID with the intent to share with researchers to analyze[223], providing access to 11 million sequences collected at different points during the pandemic. In order to insure that poor quality sequences are not allowed to interfere with the analysis, we filtered these sequences with the following parameters. The set of sequences was limited to include only those that were sampled in the first year. While this does limit the data set, it gives a specific range and clear end point to the information and mutations that occurred. The sequences themselves were checked for thorough completeness, where, if more than 20% of the sequence was unknown or missing, the sequence was removed from the data set. From here, each allelic location was analyzed for variance and removed from consideration if the location did not have at least two unique alleles and the minor allele was not present in at least 0.01% of the sequences. This

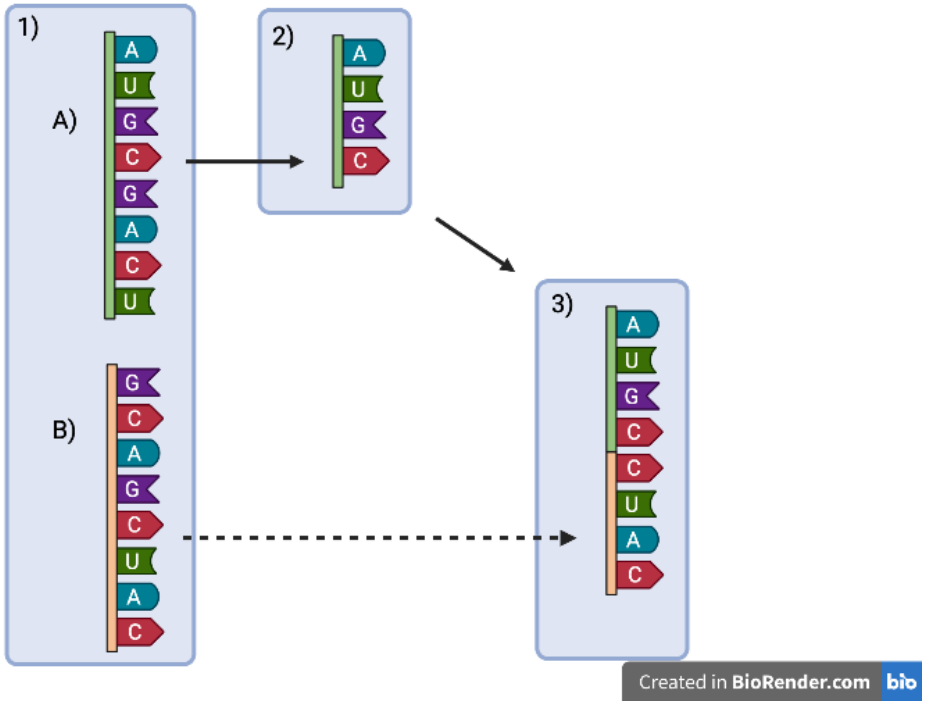

Figure 5.2: A potential process for recombination within SARs-CoV-2. 1) In order for unique haplotype A and B to recombine they would need to be present in the same cell. 2) The replication process starts for haplotype A, and after some section of the sequence is duplicated the process is paused. In this case haplotype A's replication is paused after the AUGC sequence. 3) Replication then restarts using haplotype B as the template. This completes the overall construction of the new haplotype, successfully recombining haplotypes A and B.

produced a set of 9,529 SNP locations for roughly one million sequences. These sequences were then grouped into 18,584 haplotypes which represent the unique sequences based on the parameters.

5.3.2 Incompatibility Between SNPs

To efficiently understand which SNP patterns are likely to be the result of recombination, a value is needed that can numerically represent the presence of homoplasy, or the occurrence of the same trait or mutation in two separate lineages. The only way for homoplasic events to occur is due to convergent evolution, where the two lineages evolve the trait independently, or through recombination. An *incompatibility score*, $i(X_i, X_j)$, can be defined as the number of homoplasic events between two SNP locations i and j within a population[155]. This incompatibility score can be defined as:

$$i(X_i, X_j) = (l(X_i, X_j) - 1) - (|X_i| - 1) - (|X_j| - 1)$$

where $|X_i|$ is the number of unique alleles at SNP location i and $|X_j|$ is the number of unique alleles at SNP location j, as defined by Bruen et. al. [156]. The function $l(X_i, X_j)$ denotes the number of unique combinations of the SNP pairs together at location i and j. In order to discuss the number of events the fact that one of the unique states was the 'origin' must be taken into account, and the events then make some change to that origin. This leads to the inclusion of subtracting one from each of the values.

This incompatibility score indicates the number of incompatibility events that occurred between two SNP locations within this population. The values of the resulting incompatibilities can be viewed as either their numerical values, that represent the number of homoplasic events required to allow for the set of combinations to occur, or as a binary value that represents if any incompatibility exists between the two SNPs. The resulting value is bidirectional, such that if the incompatibility between SNP A to SNP B is 1 then SNP B to SNP A is also 1.

To quantify the incompatibility of the entire SARS-CoV-2 genome, incompatibility scores can be calculated between every pair of SNPs. Individual SNPs that are incompatible

with either other SNPs or genomic regions, i.e. set of SNPs that occur near each other, likely indicates that the individual SNP had a point mutation that occurred in two separate lineages, indicating convergent evolution. However, when entire regions are incompatible with other genomic regions the likelihood of repeated convergent mutation events decreases with each SNP in the region, instead indicative of recombination. Regardless of the reason for the incompatibility, the existence indicates that the regions and combinations highlighted are under evolutionary pressure.

5.3.3 Haplotype Relationship Network

A traditional phylogenic tree attempts to derive the ancestry of each individual in a population to a single ancestor, most often by using a binary tree, where each split in the tree represents one or more evolutionary changes. They can be constructed by a variety of methods, but the underlying concept is the connection of each individual back to the common ancestor through a series of divergent changes[224]. Due to the diverging structure, there is no mechanism to indicate transfer of evolved traits between lineages. In order to visualize the relationship among viral sequences, we can create a network of the haplotypes, or unique sequences, that are connected by their evolutionary distance without specifically requiring a tree structure.

For a network construction, evolutionary distance is defined as the number of SNP differences between two haplotypes. In a network context, SARS-CoV-2 haplotypes are represented as nodes and the evolutionary distance is the edge weight between each pair of nodes. A fully connected network, where every haplotype node is connected to every other node is produced. In order to represent the most parsimonious network, edges between haplotypes are removed if both haplotypes connect to an intermediate haplotype that provides the likeliest evolutionary pathway.

This process of edge removal can be seen in Figure 5.3. Four sequences, ATT, GTT, GTC, and ATC, are represented in the figure, where each is connected to every other sequence by an edge labeled with the mutational distance between the sequences. The larger value edges marked in red across the network represent more changes needing to occur at once for one sequence to mutate into the other than the edges around the outside of the network. These internal edges represent pathways that are less likely evolutionary than following the route of one mutation occurring at a time. From a network connectivity perspective, the edges marked in red could be fully derived from the black edges. The total mutational distance from GTC to ATT can still be determined to be 2 via the edges from GTC to GTT and then GTT to ATT. Therefore, it be can determined that the red edges can be removed from the network without loss of information.

To systematically determine all edges that can be removed for these reasons, an automatic detection method was designed to determine edges that meet the required conditions. For any given edge, to determine if it is 'informative', its value is defined as total mutational distance. If the value is 1, the edge is kept, as it is the most direct relationship possible between two unique sequences. If it is greater than 1, it must be determine if there is a pathway through the network that has an equal total mutational distance.

The process can be done by using an implementation of Dijkstra's algorithm for shortest path determination. By subtracting .001 from each mutational distance, such that a previous mutational distance of 1 is now represented by the value .999, previously equivalent mutational distance pathways are now shorter if more hops between haplotypes are taken, while maintaining the approximate values of each total mutational distance. For example, with the network in Figure 5.3, the direct path from GTC to ATT is now 1.999 while the indirect, but more simple path, is 0.999 plus 0.999, or 1.998, making the simple path slightly shorter. Dijkstra's algorithm can now systematically check the combinations of edges to find the shortest path between every pair of haplotypes.

If the edge connecting two specific nodes is not the shortest path between those two nodes, as would be with the direct GTC to ATT and ATC to GTT connections in Figure 5.3, that edge would be removed. Even after removing the direct edge, there are multiple pathways from sequence GTC to ATT, through ATC or GTT, where a tree structure would

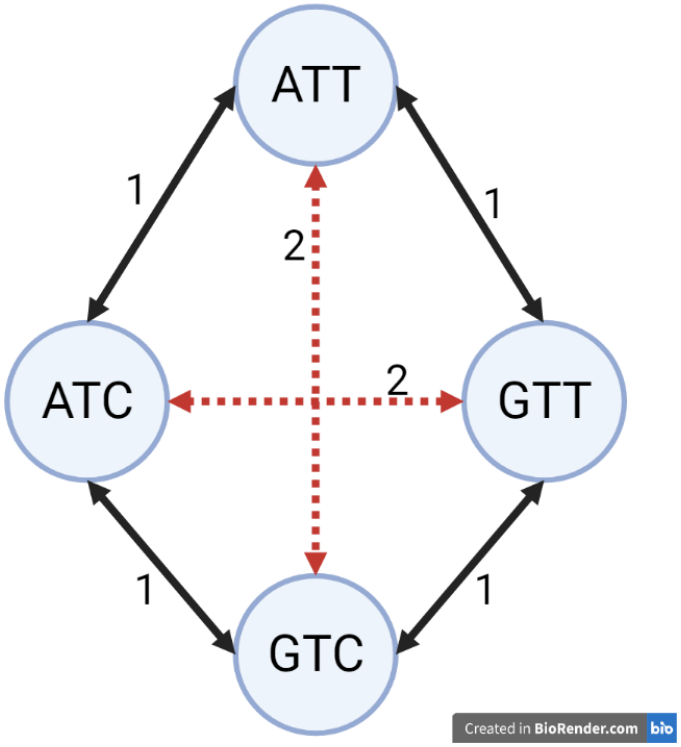

Figure 5.3: Haplotype Network. Haplotypes with the sequences ATC, ATT, GTC and GTT, are all connected with number of SNP differences shown by the edge weight. The distance from ATC to GTT and from ATT to GTC is 2, the distance from ATC to ATT, ATT to GTT, GTT to GTC, and GTC to ATC is 1. Because the edges around the outside are all one and the internal edges are two we can represent the same information in the network by removing the internal edges denoted by the red dotted lines. The remaining edges form a homoplasic loop indicating that there are at least two separate feasible step-wise paths for evolution to take.

have to choose one path or the other. This loop represents a homoplasy within the subset of haplotypes. The network construction also allows for direct edges between haplotypes that have large mutational distances if no other sequence can provide an intermediary step. These edges either represent a large amount of intermediary mutations that are not accounted for in the data set or an event that changed a region of the genome, such as recombination. This process, when used to optimally link haplotypes with their closest neighbors gives an efficient and more accurate model of the lineage of SARS-CoV-2.

5.3.4 Potential Parental Search Method

To try to identify the parents of any organism the first step would be to look for older organisms that share traits with the target of the search. With viral sequences this is done by comparing the specific SNPs from unique sequence to sequence. If divergent evolution is assumed, there should be one parent that is the closest genetically. However, if recombination has occurred the parents of the organism may be slightly more distant with some traits coming from each parent. For SARS-CoV-2, knowing who the parents are of a particular haplotype can help infer the infection rate and severity of illness it can cause. Because the likely method of recombination of SARS-CoV-2 would be template switching during the replication phase of the virus, a similar process to template switching is utilized for the search method.

The outline of the search can be seen in Figure 5.4, with the steps as follows. For a haplotype of interest (A), list of potential "parent" haplotypes to search through (B) is constructed, considering only haplotypes originally sampled prior to the original sampling of our haplotype of interest. These haplotypes (B) are ordered by their original sampling date to prioritize the haplotypes that have existed longer and had more opportunity to be a part of the recombination process. The overlap between two haplotypes, calculated between the haplotype of interest and every potential parent, is determined by calculating the exact SNP locations that occur in both sequences.

Due to the nature of viral evolution, the haplotype of interest is likely to have some amount of overlap with all potential parents. As such, it becomes necessary to identify the region of our haplotype of interest with the least coverage by the haplotypes from our

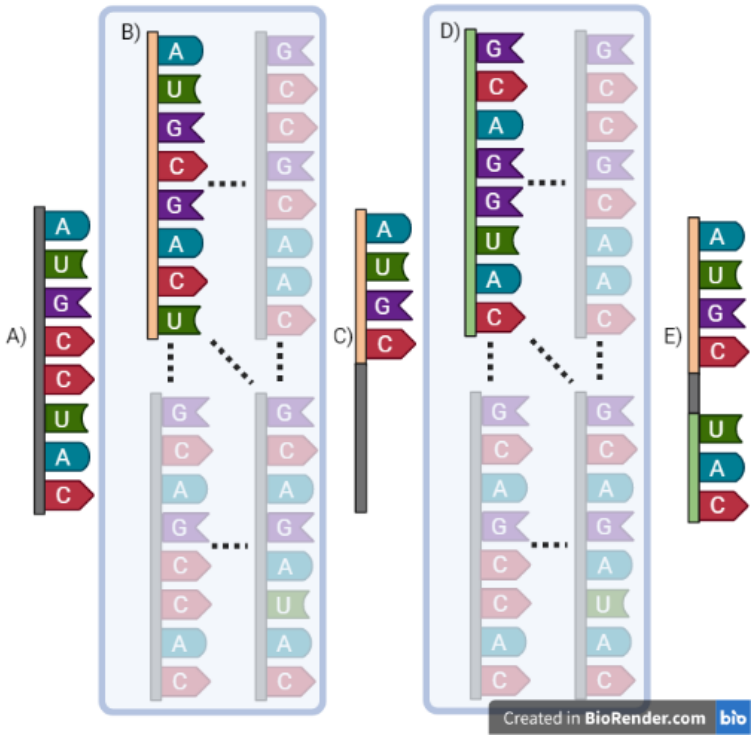

Figure 5.4: The process used to find a potential set of "parents" or ancestors of a SARs-CoV-2 haplotype. A) On selection of a haplotype of interest, all haplotypes that were sampled before it are added to a list of potential parents. B) The first parent is chosen as the haplotype that best covers the region in our haplotype of interest with the least coverage by the potential parents. C) The SNPs that overlap between the selected haplotype from B with the haplotype of interest are then used to create a new partial haplotype. D) The same list of haplotypes used in the search for B) is used to again search for the closest match with the haplotype of interest, excluding the SNPs from partial haplotype from C. E) The overlap from the second parent is added to the partial haplotype. This process of step D and E continues to repeat until the original haplotype is fully reconstructed, the addition of new parents no longer increases the size of the partial haplotype, or a preset number of parents is chosen.

set of potential parents, and find the parent that best covers that region first (B). The overlap between the haplotype of interest and this found parent are used to create a partial haplotype (C). To continue reconstructing the haplotype of interest, the set of potential parents is searched for the haplotype with the overlap locations that best completes the partial haplotype (D). The overlap that exists between the second parent and the haplotype of interest is then added to our partial haplotype(E). The final two processes (D&E) are repeated until one of three stopping parameters is met: the haplotype of interest is fully reconstructed, there are no remaining potential parents that would add a SNP to our partial haplotype, or a predetermined number of parents is reached.

If a haplotype of interest is not the result of recombination, but simply a single divergent mutation from another haplotype, this method will produce that closest haplotype as the single parent. Reporting a single parent is equivalent to this method reporting that the haplotype of interest is not the result of recombination.

5.4 Results and Discussion

5.4.1 Incompatibility of SNPs

For each of the 9,529 SNPs in the SARS-CoV-2 data set, the incompatibility scores were calculated for each pair, resulting in 90,801,841 calculated comparisons with 790,416 pairs of SNPs having an incompatibility greater than zero. Figure 5.5 shows a summarized view of the incompatibility scores. While the value of incompatibility is the number of homoplasic events required to reconcile the two SNP locations, the image treats incompatibility as a binary, either compatible (white pixel) or not (red pixel). This summarized view shows only the areas of highest incompatibility, by including the SNPs (and their immediate neighborhood) that have the largest sum of incompatibility with the rest of the genome. The large number of SNP comparisons shown in the heatmap produces a visual effect that regions that do not have a large concentration of incompatibilities appear lighter in color than the dense regions that appear deep red, however all red used is the same shade.

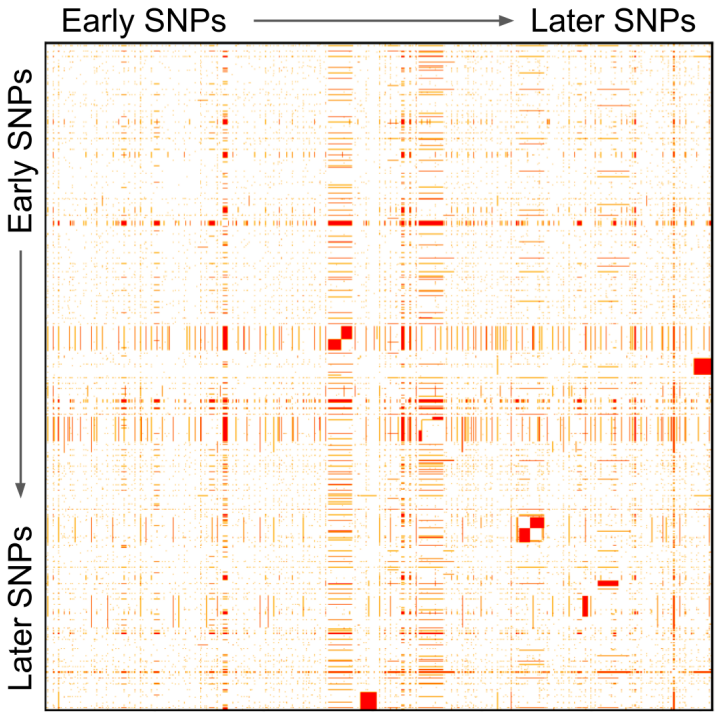

Figure 5.5: Incompatibility Matrix of SARS-CoV-2 Haplotypes shown as a heat map. Each axis is the SNP locations and each red pixel represents incompatibility between the two locations. This represents a summarized view of the genome by removing SNPs from both axes that have very little information (as defined by a total incompatibility with the rest of the genome of less than 500). The darker appearing regions indicate more incompatibility between the two sites and the denser the local area of incompatibility. The lines of darker appearing color highlight sites that have likely had mutations transferred to other parts of the population through recombination resulting in incompatibility with most of the genome.

Since all of the SNPs represented are on both axes, the diagonal represents each SNP location's incompatibility score with themselves, which is always a value of 0. This also causes the upper triangular region to be a transposed copy of the lower triangular region and for the the areas nearest the diagonal to represent the areas adjacent to each other in the haplotypes. A full version appears in Appendix A. Figure 5.5 shows distribution of incompatibilities throughout the genome, but there appears to be specific regions, the solid lines and blocks, that have higher rates of incompatibility, pointing either to recombination or convergent mutation of whole sections of the genome. The mutations at these sites represent the areas in the genome most likely to contain evidence for recombination events, mutations that do not follow a linear evolutionary path. While a single event of incompatibility may easily represent a convergent mutation, whole regions of incompatibility more likely indicate a higher likelihood of recombination in that region.

Focused views for areas of interest can be seen in Figure 5.6. These views highlight specific regions where large sections of the genome has incompatibilities with other large sections. Image (A) shows the heatmap from SNP 18694 to 22000 on both axes containing two separate regions of incompatibility mirrored across the diagonal. Both regions appeared to be fully boxed in by at least one SNP on either side that is compatible with the whole region, potentially representing a SNP at the border of a region of recombination and lines up with the intersection between Beta-CoV-Nsp14 and an Nsp15-like endoribonuclease. Image (B) shows the heatmap from SNP 27382 to 28882 on both axes and contains two separate regions of incompatibility, multiple different shapes of incompatibility, and aligns with the intersecting points of multiple open reading frames, including ORF6, ORF7a, ORF7b, and ORF8. Both part A and B of the figure show a region around the diagonal, such that the image is mirrored showing patterns of incompatibilities that exist within the genome from regions that are near each other. Image (C) shows the zoom in of the incompatibility matrix From SNP 19824 to 22800 on the Y axis and SNP 27925 to 29,904 on the X axis. The region shows both a large block of incompatibility from region of the nsp15 and 2'-O-ribose methyltransferase genes to the region of the Corona Nucleoca and ORF10. Part C of the figure shows a large incompatibility section far off the diagonal, representing two large sets of SNPs from very different regions that are incompatible with each other.

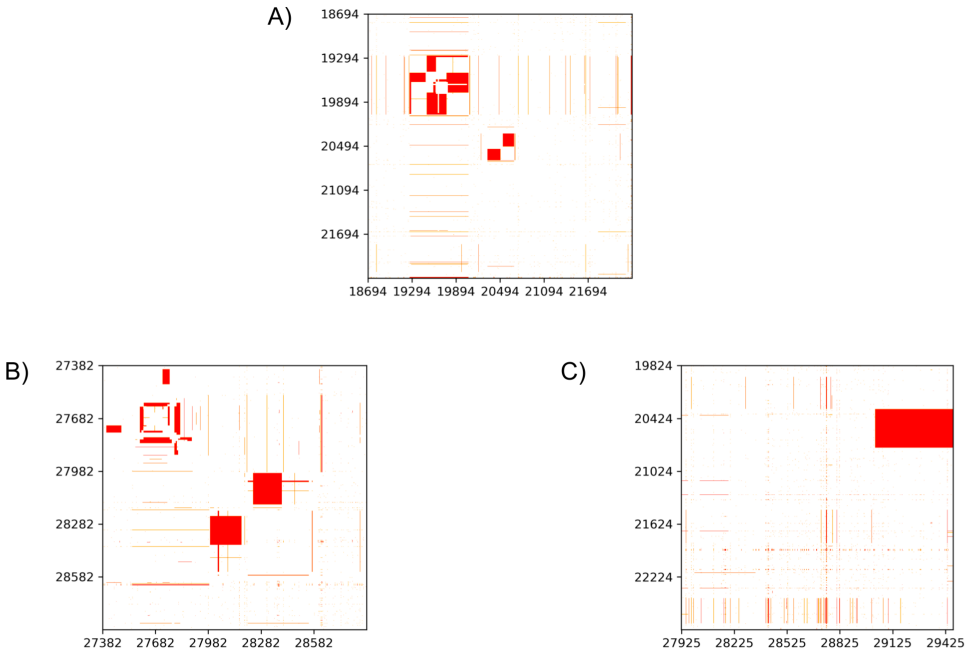

Figure 5.6: Incompatibility Matrix of SARS-CoV-2 Haplotypes shown as a heat map zoomed in to specific regions of the genome. Each axis is the parsimonious SNPs in the order they appear in the sequence. A) From SNP 18694 to 22000 on both axes shows a region along the diagonal that has two separate regions of incompatibility mirrored across the diagonal. Both regions appeared to be fully boxed in by at least one SNP on either side that is compatible with the whole region, potentially representing a SNP at the border of a region of recombination. This region lines up with the intersection between Beta-CoV-Nsp14 and an Nsp15-like endoribonuclease. B) From SNP 27382 to 28882 on both axes shows a region along the diagonal that has two separate regions of incompatibility mirrored across the diagonal. This shows multiple different shapes of incompatibility and aligns with the intersecting points of multiple open reading frames, including ORF6, ORF7a, ORF7b, and ORF8. C) The zoom in of the incompatibility matrix From SNP 19824 to 22800 on the Y axis and SNP 27925 to 29,904 on the X axis. This region shows both a large block of incompatibility from region of the nsp15 and 2'-O-ribose methyltransferase genes to the region of the Corona Nucleoca and ORF10.

In order to better visualize the incompatibilities within the context of the SARs-CoV-2 genome, a Circos plot was created as seen in Figure 5.7. This plot is annotated with the various structures within the virus (annotations were downloaded from NCBI)[225, 226]. The connections shown are chosen by selecting the 100 largest blocks of incompatibility by total size from the heatmap, by summing the total number of pixels in the contiguous region. The near diagonal sections seen in the heatmaps are represented by the lines that connect adjacent sections, with the lines traveling across the genome representing the incompatible sections that are farther away within the genome and farther away from the diagonal within the heatmaps.

5.4.2 Haplotype Relationship Network

The full set of 18,584 haplotypes were connected based on the 9,529 SNPs from the population using the method described in section 5.3.3. Figure 5.8 shows a reduced network connecting 2,662 unique haplotypes, with edges thresholded to represent one to five mutational differences. The lighter and more yellow nodes originated earlier in the pandemic than the darker, more purple nodes. The larger nodes represent more common haplotypes, while the smallest nodes representing haplotypes with only 5 sequences.

The multiple isolated clusters represent groups of haplotypes that are closely related without a close genetic relative between the separate groups. This network is the first step in creating a full lineage of SARS-CoV-2 because each haplotype can only be connected to other haplotypes that were recorded earlier and connections imply an evolutionary relationship. The network shows a hierarchy of haplotypes in a similar way to phylogenetic trees, but because it allows for multiple edges to converge on a single haplotype, it is able to include convergent evolution and recombination. This results in homoplasic loops in the network, where there are multiple possible paths between two connected haplotypes. These homoplasic loops highlight the specific sequences that have the incompatibilities highlighted in the heat maps and circos plot. Because of the nature of evolution, the separation of the haplotypes into distinct clusters could represent the genetic distance caused by multiple mutations and little to no sampling over a period of time, or recombination.

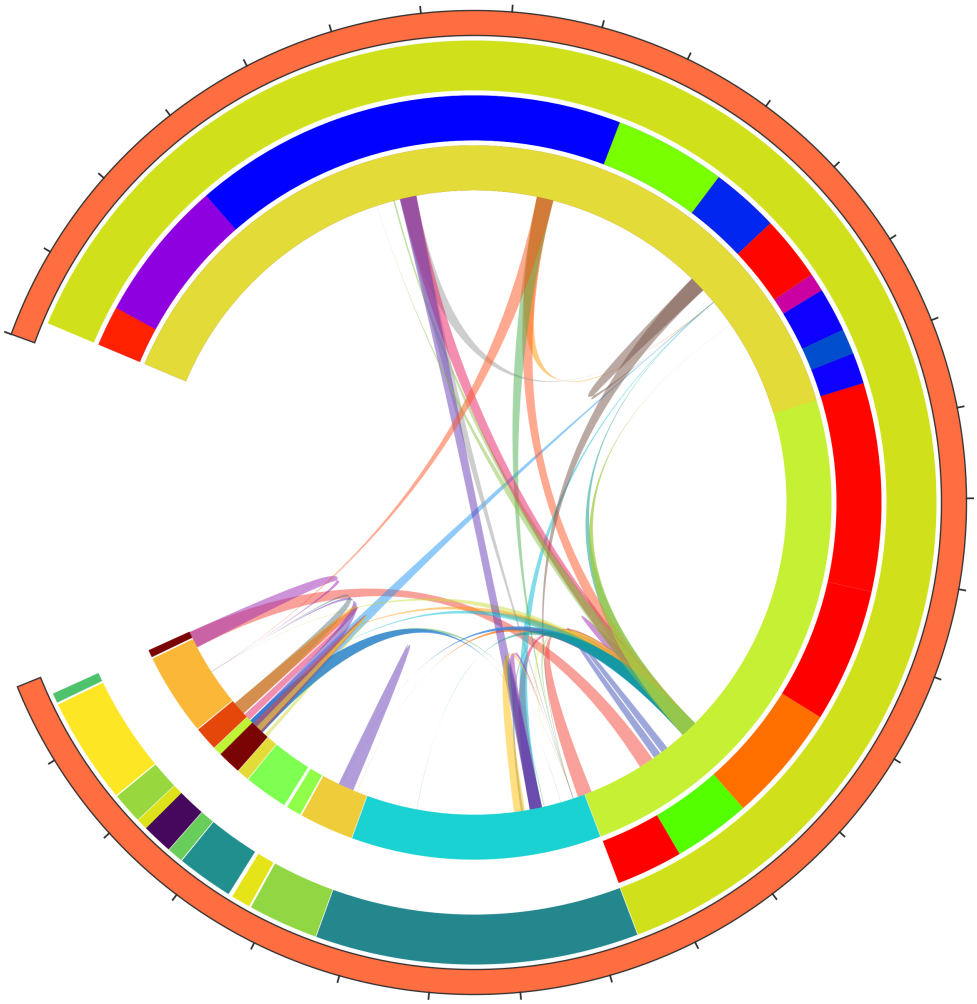

Figure 5.7: A Circos plot showing the location of the largest incompatibility blocks from the heat map in the context of the SARs-CoV-2 genome and annotations. The outer most band represents the 29,904 bases in the full sequence of SARs-CoV-2, with each tick representing a distance of 1,000 bases. The next three tracks are annotated sections of the SARs-CoV-2 reference sequence from NCBI, broken down into three categories with each change in color on the track representing a different structure within the sequence. The first of these tracks is the "genes" , made up mostly of several Open Reading Frames (ORFs), the first ORF making up most of the sequence. The second track shows the peptides in each ORF and the last track showing different conserved protein domains. The connections shown in the center of the circle illustrate the largest 100 sections of the heat map of connected incompatibilities. Most of the largest incompatibility regions are incompatible with multiple other regions, and not just one singular other region. Hot spots for incompatibility favor regions where proteins or genes intersect, but also occasionally appear in the middle of large sequences from a single protein. The teal blue on the inner most track represents the region coding for the SPIKE protein and includes incompatibility hotspots both within the gene and along its intersection with the proceeding gene.

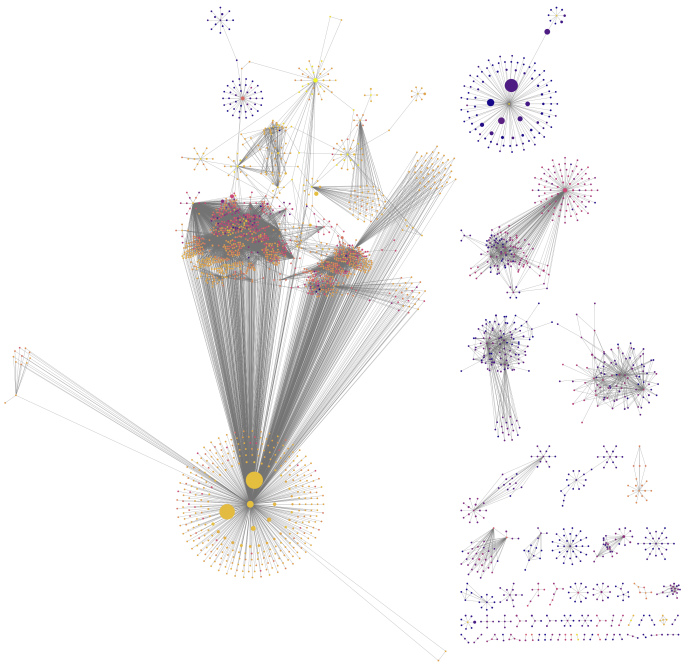

Figure 5.8: The network is constructed by taking unique SARS-CoV-2 haplotypes (the nodes) and connecting them if they are within 5 mutational steps away from each other. The size of each node represents the number of times a sequence was sampled that belonged to that haplotype and the color represents when the haplotype was first sampled, with lighter, yellower colors being earlier and darker, more purple colors being later in the pandemic.

By restricting the network to only show edges that represent one SNP of difference we arrive at the network in Figure 5.9. The figure shows that by limiting the edges significantly, the total number of haplotypes shown is also limited, as the network view only retains haplotypes that have an edge to another haplotype. This figure also shows that the majority of these very similar haplotypes are from roughly the same time period, as denoted by their colors. The limited number of edges represented shows that within the entire set of haplotypes very few are within a single step from each other, with the network only containing 262 haplotypes and 211 edges between them. More examples of the network at different thresholds can be found in Appendix A.

5.4.3 Selection of Potential Parents

Two haplotypes of interested were chosen to search for the potential parents from recombination. The first was a haplotype from the pango lineage B.1.1.7, and the second is P.1 or Gamma. In Figures 5.10 (B.1.1.7), and 5.11 (P.1) the results of these parental searches can be seen on the two separate haplotypes. These haplotypes are shown within the context of the networks from the previous section that connects haplotypes based on SNP differences. In Figure 5.10 the parents of haplotype C are the haplotypes labeled A and B. These haplotypes do not have a direct or indirect connection in this network because a threshold of 5 mutational distance was used to remove edges. The result is that C exists in its own cluster while its parents exist in a separate cluster. The color of the nodes also shows that C was first sampled much closer in time to both of its likely parents than to any of the other haplotypes in its cluster.

In Figure 5.11 no threshold is applied to the edges so every haplotype is connected to every other haplotype that does not have a more efficient intermediate route. So, as seen in the image, all 5 potential parents of P.1 are connected to P.1 because there are no intermediate haplotypes closer with the same types of mutations. This allows us to see that B.1, B.1* (a different haplotype but close to B.1), and B.1.279 are all closely related as denoted by the edges connecting them together.

For B.1.1.7 to directly evolve into P.1 would require a substitution of 82 SNPs within the 29903 bases in the genome of SARs-CoV-2. The evolution rate of SARS-CoV-2 has

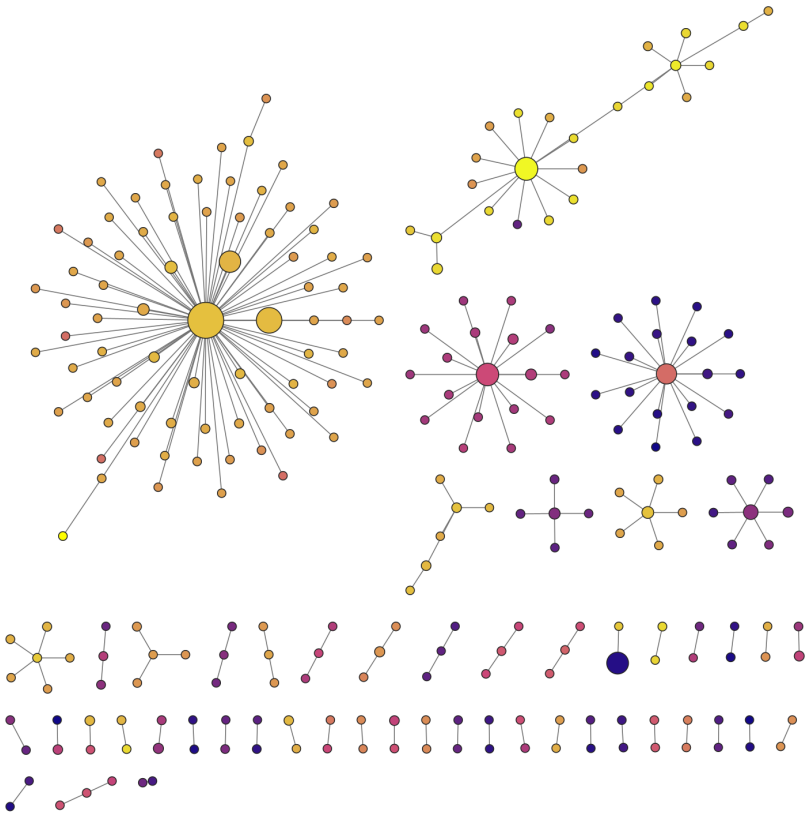

Figure 5.9: The network is constructed by taking unique SARS-CoV-2 haplotypes (the nodes) and connecting them if they are 1 point of mutation away from each other. The size of each node represents the number of samples within that haplotype and the color represents when the haplotype was first sampled, with lighter, yellower colors being earlier and darker, more purple colors being later in the pandemic.

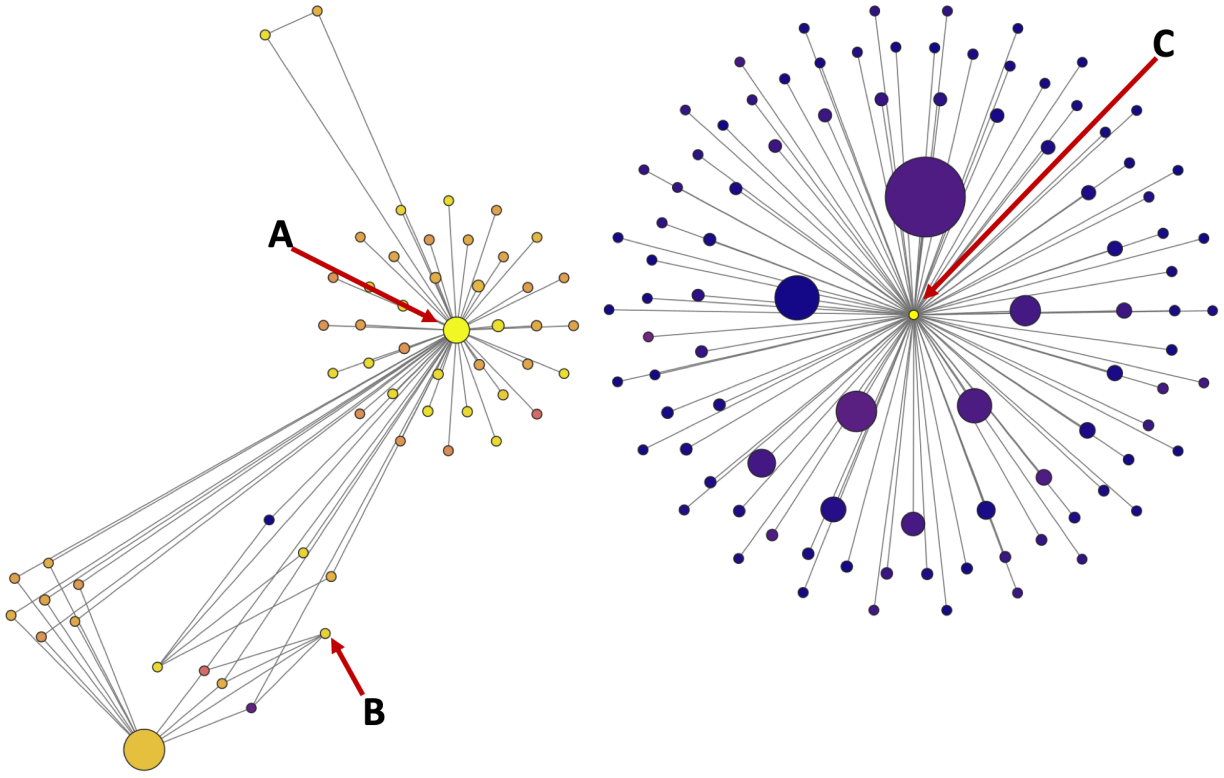

Figure 5.10: The network shows how recombination can be missed by looking at haplotypes connected by mutational distance. By searching through all haplotypes in our data set that had a sample collected before the collection of the haplotype labeled C, B.1.1.7 from the pango lineage, it was identified that the haplotype A and the haplotype B in combination account for all but 5 SNPs that were new mutations at the time, indicating that while their is significant overlap between C and its potential parents it does still introduce new mutations to the population at the same time. This network is constructed by taking unique SARS-CoV-2 haplotypes (the nodes) and connecting them if they are 5 or fewer points of mutation away from each other. The size of each node represents the number of samples within that haplotype and the color represents when the haplotype was first sampled, with lighter, yellower colors being earlier and darker, more purple colors being later in the pandemic.

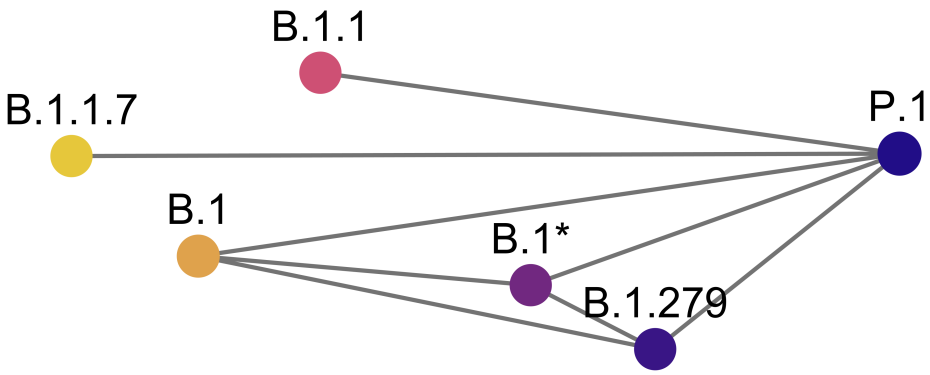

Figure 5.11: The network shows the potential parents of the P.1 strain first detected in Brazil. The P.1 label, as well as the others shown in the graph, are the pango network designations, and allow for small variations in haplotype within a single designation. Using the parental search method, it was identified that haplotype B.1 and the haplotype B.1.1.7, in combination, account for 99.7% of the P.1 sequence, with B.1 providing at least 30 unique mutations not found in B.1.1.7. The other three parents shown in the image, B.1.1, B.1*, and B.1.279 each add only a single additional mutation. Both the color of the node and the relative position left to right indicate date of first sampling of the specific haplotype. Even though B.1.1.7 is a descendant of B.1 it appears before it because the specific haplotype represented by B.1 in the graph was first measured after the B.1.1.7 haplotype. This also accounts for the presence of B.1 and B.1*, with the later being only a few mutations away from the original B.1.

been estimated to be 10^{-3} mutations per site per year[227]. There were 274 days (.75 years) between the sampling of the B.1.1.7 and the P.1 haplotypes. There are 30 SNPs that occur in the B.1 haplotype from the figure that occur in P.1 but not B.1.1.7. At the observed evolution rate for one of those SNPs to have convergent evolution over that time period one would expect a probability of 10^{-3} mutations per site per year multiplied by .75 years. Thus, for each specific site we expect there to be $7.5 * 10^{-4}$ mutations, or effectively a .075% chance of having a mutation, let alone a mutation to the correct base. So the probability that all 30 convergent substitutions would occur within the same time frame is on the order of $(7.5 * 10^{-4})^{30}$ or roughly $1.7858209 * 10^{-94}$ chance of happening. This does not consider the inclusion of selective pressure of the specific mutations or the effect other mutations may have on the sequence in this time period. Nonetheless, the extremely unlikely series of events of the exact loci mutating to the exact SNPs that occurred in another haplotype that existed previously would potentially require only a single recombination event to reproduce.

5.5 Conclusion

Here I have shown the regions of incompatibility that exist throughout the SARS-CoV-2 genome with a population wide measurement and how hot spot regions of incompatibility map to specific regions of the genome potentially under selective evolutionary pressure. One of these regions is the area within and around the spike gene, which has been a hot spot of mutational changes across variants and has prior evidence pointing to possible recombination[151].

Also presented is a haplotype network that illustrates the relationships between haplotypes of interest and their likely parental haplotypes. An example haplotype of interest, haplotype P.1, contains multiple mutations from distant lineages of previously sampled haplotypes, and for those mutations to occur together and to not be the result of recombination would be unlikely to the point of near impossibility. With both the incompatibility throughout the SARS-CoV-2 genome and the homoplasic loops within the haplotype network, the evidence points to evolution using both serial collections of individual point mutations as well as the incorporation of recombinations of mutations that already

exist in the population. By comparing haplotypes of interest such as P.1 in Figure 5.11 to its parents, we can find that large swaths of the sequence come from previously sampled haplotypes from different divergent paths. Upon closer examination we find that, in order to fully recreate this new haplotype, we have to allow for recombination or expect an extremely unlikely set of co-occurring convergent evolution events to have occurred.

These results suggest that, while serial mutations do lead to new haplotypes, recombination can combine different segments from various strains of SARS-CoV-2. This result supports the idea that recombination has been seen since the beginning of the pandemic[146], and that some of the lineages from the first year of the pandemic were likely the result of recombination[147], as is shown with the P.1 lineage. This could indicate that more divergent strains could be created simply by having individuals infected with multiple strains. This knowledge could help track the creation of new strains and help determine which haplotypes need to targeted with new testing. This also allows for further insight into what could be causing recombination and potentially help formulate a way to cause a rapid extinction of the virus SARS-CoV-2[152] by causing an exponential increase in the overall mutation rate within the virus, or imposed mutational meltdown[153].

Future work could be done to evaluate more recent variants, given that the evidence here was limited to the data from the first year of the pandemic. Research could also be done to project out to potential new strains of SARS-CoV-2 that may appear in the future by using same method to reconstruct a haplotype based on potential parents to create new unsampled haplotypes that are the recombination of current haplotypes. The method could use the information from the incompatibility matrix to determine likely template switching points for the recombination process and potentially help prepare for haplotypes that do not currently exist. The more solid the understanding of the current and future viral mutation and recombination patterns within SARS-CoV-2, the more prepared the world is to combat this virus and potential future viruses with similar structures.

Chapter 6

Conclusion

Biological data are known to be complex in nature, requiring tools and methods that can handle and evaluate these complexities. The methods shown here are new techniques that address ways to better take advantage of the inherent conditionalities in the biological data available. These chapters discuss new algorithms to handle these data and new ways of approaching known problems.

Using Iterative Random Forest for Genome Wide Epistasis Studies in chapter 2 shows efficient detection of epistasis with an explainable artificial intelligence (X-AI) algorithm. This process identifies likely sources of epistasis in a genome relative to a given phenotype, and also provides the effect sizes of linear and combinatoric effects and their interactions within the model. This chapter presented a way for iRF models to be mined more efficiently providing new and useful data. Specifically, an importance cutoff value to determine causitive effects that generalizes to any problem or data set was introduced. A method for determining the feature effect and the direction of the feature effect on the phenotype in question was introduced and shown to be robust to almost all tested parameters. Three new set analysis techniques were introduced to work alongside RIT, within the framework of iRF, with set importance having the greatest effect. And while RIT_adjusted and conditional importance did not improve the selection of causative sets from the model, they do provide new insight into how features interact with each other within the framework of iRF.

The progeny prediction method using iRF in chapter 3 introduces a new method for breeders to determine the genetic efficacy of crosses without the time and money necessary for a full progeny trial. By taking advantage of broad population genetic mappings, a machine learning model was built to predict phenotypes from the genotypic information of the organism. By genotyping a set of potential parents, a synthetic set of crosses between parents can be made and the phenotype of the *in silico* progeny can be predicted by the machine learning model. These predictions allow a full evaluation of a progeny trial providing both General Combining Ability and Specific Combining Ability of the parents without requiring the time, space, and money for a full progeny trial. The method can take advantage of a generic training population of the species, grown in a separate location from the intended progeny, and with the genetic information of the parents, to provide accurate predictions on the combining ability of the parents for a given phenotype.

Tensor iterative random forest, as described in chapter 4, provides an X-AI model that can identify the relationships among multiple targets, while maintaining the ability of the model predecessors to identify feature to feature and feature to target relationships. This method was validated in both real and synthetic expression data using transcription factor genes to predict non-transcription factor genes from the DREAM competition. The ability to provide relationships within and across layers of data allows for analysis on full omics-to-omics layers with a single model.

Given the human impact of the SARS-CoV-2 virus, a better understanding of the mechanistic processes can help predict and prepare for future events. Chapter 5 showed how the amount of SARS-CoV-2 sequences could be utilized to detect likely recombination events that have occurred during the pandemic. Multiple metrics are provided that suggest the presence of recombination within SARS-CoV-2, such as incompatibility scores and homoplasic loops within haplotype networks. By providing a method to find potential "parents" of virus haplotypes, this chapter adds another tool in the arsenal for scientists looking to study the full evolutionary history of the virus. This method can also help prepare for potential dangerous possible recombinations of this virus and future viruses.

Together these chapters provide a comprehensive look into the complexities of biological data and provided methods and insights to further elucidate knowledge from genomic data. The quantities of genomic data are going to continue to grow and the tools and methods designed for them must to continue to grow as well.

Bibliography

[1] C. E. Cook, M. T. Bergman, R. D. Finn, G. Cochrane, E. Birney, and R. Apweiler, "The european bioinformatics institute in 2016: Data growth and integration," *Nucleic Acids Research*, vol. 44, no. D1, 2015. 2

[2] J. McDonald, *Handbook of Biological Statistics*. Baltimore, Maryland: Sparky House Publishing, (3rd ed.) ed., 2014. 2

[3] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, pp. 123–140, aug 1996. 2, 8, 83

[4] S. Basu, K. Kumbier, J. B. Brown, and B. Yu, "Iterative random forests to discover predictive and stable high-order interactions.," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 115, pp. 1943–1948, feb 2018. 2, 3, 11, 13, 28, 30, 65, 81, 84

[5] I. Gabur, D. P. Simioniuc, R. J. Snowdon, and D. Cristea, "Machine learning applied to the search for nonlinear features in breeding populations," *Frontiers in Artificial Intelligence*, vol. 5, 2022. 2

[6] O. A. Montesinos-López, A. Montesinos-López, P. Pérez-Rodríguez, J. A. Barrón-López, J. W. Martini, S. B. Fajardo-Flores, L. S. Gaytan-Lugo, P. C. Santana-Mancilla, and J. Crossa, "A review of deep learning applications for genomic selection," dec 2021. 2, 21

[7] A. Monaco, E. Pantaleo, N. Amoroso, A. Lacalamita, C. Lo Giudice, A. Fonzino, B. Fosso, E. Picardi, S. Tangaro, G. Pesole, and R. Bellotti, "A primer on machine learning techniques for genomic applications," *Computational and Structural Biotechnology Journal*, vol. 19, pp. 4345–4359, 2021. 2

[8] A. L. Tyler, F. W. Asselbergs, S. M. Williams, and J. H. Moore, "Shadows of complexity: what biological networks reveal about epistasis and pleiotropy," *BioEssays*, vol. 31, pp. 220–227, feb 2009. 3, 59

[9] A. Cliff, J. Romero, D. Kainer, A. Walker, A. Furches, and D. Jacobson, "A High-Performance Computing Implementation of Iterative Random Forest for the Creation

of Predictive Expression Networks," *Genes*, vol. 10, p. 996, dec 2019. 3, 11, 13, 17, 28, 65, 100

[10] M. O. Riedl, "Human-centered artificial intelligence and machine learning," *Human Behavior and Emerging Technologies*, vol. 1, pp. 33–36, 1 2019. 4

[11] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, pp. 255–260, 7 2015. 4

[12] S. Ray, "A quick review of machine learning algorithms," *Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing: Trends, Prespectives and Prospects, COMITCon 2019*, pp. 35–39, 2 2019. 4

[13] A. Singh, N. Thakur, and A. Sharma, "A review of supervised machine learning algorithms," pp. 1310–1315, 2016. 4

[14] H. Barlow, "Unsupervised learning," *Neural Computation*, vol. 1, pp. 295–311, 9 1989. 4

[15] T. Hastie, R. Tibshirani, and J. Friedman, "Unsupervised learning," pp. 485–585, 2009. 4

[16] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010. 5

[17] S. O'Sullivan, A. Holzinger, K. Zatloukal, P. Saldiva, M. I. Sajid, and D. Wichmann, "Machine learning enhanced virtual autopsy," *Autopsy & Case Reports*, vol. 7, no. 4, p. 3, 2017. 5

[18] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature 2017 542:7639*, vol. 542, pp. 115–118, 1 2017. 5

[19] M. Niazian and G. Niedbała, "Machine learning for plant breeding and biotechnology," *Agriculture 2020, Vol. 10, Page 436*, vol. 10, p. 436, 9 2020. 5

[20] G. Forkuor, O. K. L. Hounkpatin, G. Welp, and M. Thiel, "High resolution mapping of soil properties using remote sensing variables in south-western burkina faso: A comparison of machine learning and multiple linear regression models," *PLOS ONE*, vol. 12, p. e0170478, 1 2017. 5

[21] Q. Bi, K. E. Goodman, J. Kaminsky, and J. Lessler, "What is machine learning? a primer for the epidemiologist," *American Journal of Epidemiology*, vol. 188, pp. 2222–2239, 12 2019. 5

[22] R. Goebel, A. Chander, K. Holzinger, F. Lecue, Z. Akata, S. Stumpf, P. Kieseberg, and A. Holzinger, "Explainable ai: The new 42?," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11015 LNCS, pp. 295–303, 8 2018. 5, 6, 81

[23] D. Doran, S. Schulz, and T. R. Besold, "What does explainable ai really mean? a new conceptualization of perspectives," *CEUR Workshop Proceedings*, vol. 2071, 10 2017. 5

[24] A. Hasan, M. R. Al-Mulla, J. Abubaker, and F. Al-Mulla, "Early insight into antibody-dependent enhancement after sars-cov-2 mrna vaccination," *Human Vaccines Immunotherapeutics*, p. 1, 2021. 5, 104

[25] T. J. Sejnowski, "The unreasonable effectiveness of deep learning in artificial intelligence," *Proceedings of the National Academy of Sciences*, vol. 117, pp. 30033–30038, 12 2020. 6

[26] F. Lecue and J. Wu, "Semantic Explanations of Predictions," may 2018. 6

[27] R. R. Hoffman, S. T. Mueller, G. Klein, and J. Litman, "Metrics for explainable ai: Challenges and prospects," 12 2018. 6

[28] A. Chander and R. Srinivasan, "Evaluating explanations by cognitive value," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11015 LNCS, pp. 314–328, 8 2018. 6

[29] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee, "From local explanations to global understanding with explainable ai for trees," *Nature Machine Intelligence 2020 2:1*, vol. 2, pp. 56–67, 1 2020. 6, 81

[30] L. Breiman, *Classification And Regression Trees*. Belmont, California: Wadsworth International Group, 1984. 6, 16, 29, 83

[31] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001. 6, 8, 29, 65, 83

[32] J. L. Speiser, M. E. Miller, J. Tooze, and E. Ip, "A comparison of random forest variable selection methods for classification prediction modeling," nov 2019. 8, 65, 83

[33] B. H. Menze, B. M. Kelm, R. Masuch, U. Himmelreich, P. Bachert, W. Petrich, and F. A. Hamprecht, "A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data," *BMC Bioinformatics*, vol. 10, p. 213, jul 2009. 8, 65, 83

[34] R. D. Shah and N. Meinshausen, "Random intersection trees," *Journal of Machine Learning Research*, vol. 15, pp. 629–654, 2014. 11, 13, 28, 30, 33, 65, 84

[35] X. Tang, Y. Huang, J. Lei, H. Luo, and X. Zhu, "The single-cell sequencing: New developments and medical applications," jun 2019. 11

[36] D. O. Perkins, C. Jeffries, and P. Sullivan, "Expanding the 'central dogma': the regulatory role of nonprotein coding genes and implications for the genetic liability to schizophrenia," *Molecular Psychiatry 2005 10:1*, vol. 10, pp. 69–78, 9 2004. 11

[37] M. Boettcher and M. T. McManus, "Choosing the Right Tool for the Job: RNAi, TALEN, or CRISPR," may 2015. 11

[38] J. A. Doudna and E. Charpentier, "The new frontier of genome engineering with CRISPR-Cas9," *Science*, vol. 346, p. 1258096, nov 2014. 11

[39] J. S. Witte, "Genome-wide association studies and beyond," apr 2010. 11, 12, 27

[40] C. Eisenhart, "The assumptions underlying the analysis of variance," *Biometrics*, vol. 3, p. 1, 3 1947. 12, 59

[41] G. Acquaah, "Plant breeding, principles," *Encyclopedia of Applied Plant Sciences*, vol. 2, pp. 236–242, 1 2017. 12, 19, 41, 61

[42] D. P. Singh, A. K. Singh, and A. Singh, "Primer on population and quantitative genetics," *Plant Breeding and Cultivar Development*, pp. 77–127, 1 2021. 12

[43] C. Andrews, "The hardy-weinberg principle," *Nature Education Knowledge*, vol. 3, pp. 65–66, 2010. 12

[44] M. J. Kearsey, "The principles of QTL analysis (a minimal mathematics approach)," *Journal of Experimental Botany*, vol. 49, pp. 1619–1623, oct 1998. 12

[45] M. Jamil, N. Ali, A. Ali, and A. Mujeeb-Kazi, "Spot blotch in bread wheat: virulence, resistance, and breeding perspectives," *Climate Change and Food Security with Emphasis on Wheat*, pp. 217–228, 1 2020. 12

[46] P. M. Hayes, A. Castro, L. Marquez-Cedillo, A. Corey, C. Henson, B. L. Jones, J. Kling, D. Mather, I. Matus, C. Rossi, and K. Sato, "Genetic diversity for quantitatively inherited agronomic and malting quality traits," *Developments in Plant Genetics and Breeding*, vol. 7, pp. 201–226, 1 2003. 12

[47] H. M. Kang, J. H. Sul, S. K. Service, N. A. Zaitlen, S. Y. Kong, N. B. Freimer, C. Sabatti, and E. Eskin, "Variance component model to account for sample structure in genome-wide association studies," *Nature Genetics*, vol. 42, pp. 348–354, apr 2010. 12, 28

[48] X. Zhou and M. Stephens, "Genome-wide efficient mixed-model analysis for association studies," *Nature Genetics*, vol. 44, pp. 821–824, jul 2012. 12, 13, 28

[49] A. L. Harfouche, D. A. Jacobson, D. Kainer, J. C. Romero, A. H. Harfouche, G. Scarascia Mugnozza, M. Moshelion, G. A. Tuskan, J. J. Keurentjes, and A. Altman,

"Accelerating climate resilient plant breeding by applying next-generation artificial intelligence," *Trends in Biotechnology*, vol. 37, no. 11, pp. 1217 – 1235, 2019. 13

[50] H. J. Cordell, "Epistasis: what it means, what it doesn't mean, and statistical methods to detect it in humans," *Human Molecular Genetics*, vol. 11, pp. 2463–2468, 10 2002. 13, 14, 27

[51] G. A. Churchill, "Epistasis," *Brenner's Encyclopedia of Genetics: Second Edition*, pp. 505–507, 1 2013. 13

[52] C. J. Goodnight, "Gene interactions in evolution," *Encyclopedia of Evolutionary Biology*, pp. 104–109, 1 2016. 14

[53] A. M. Dean, "Molecular evolution, functional synthesis of," *Encyclopedia of Evolutionary Biology*, pp. 44–54, 1 2016. 14

[54] P. C. Phillips, "Epistasis — the essential role of gene interactions in the structure and evolution of genetic systems," *Nature Reviews Genetics 2008 9:11*, vol. 9, pp. 855–867, 11 2008. 14, 15

[55] H. J. Cordell, "Detecting gene–gene interactions that underlie human diseases," *Nature Reviews Genetics 2009 10:6*, vol. 10, pp. 392–404, 6 2009. 14, 16

[56] J. H. Moore and S. M. Williams, "New strategies for identifying gene-gene interactions in hypertension," *https://doi.org/10.1080/07853890252953473*, vol. 34, pp. 88–95, 2009. 14

[57] X. Wan, C. Yang, Q. Yang, H. Xue, X. Fan, N. L. Tang, and W. Yu, "Boost: A fast approach to detecting gene-gene interactions in genome-wide case-control studies," *The American Journal of Human Genetics*, vol. 87, pp. 325–340, 9 2010. 14, 16

[58] J. Marchini, P. Donnelly, and L. R. Cardon, "Genome-wide strategies for detecting multiple loci that influence complex diseases," *Nature Genetics 2005 37:4*, vol. 37, pp. 413–417, 3 2005. 14, 16

[59] M. Nelson, S. Kardia, R. Ferrell, and C. Sing, "A combinatorial partitioning method to identify multilocus genotypic partitions that predict quantitative trait variation," *Genome Research*, vol. 11, pp. 458–470, 3 2001. 14, 15

[60] L. E. Mechanic, B. T. Luke, J. E. Goodman, S. J. Chanock, and C. C. Harris, "Polymorphism interaction analysis (pia): a method for investigating complex gene-gene interactions," *BMC Bioinformatics 2008 9:1*, vol. 9, pp. 1–12, 3 2008. 14, 16

[61] D. F. Schwarz, I. R. König, and A. Ziegler, "On safari to random jungle: a fast implementation of random forests for high-dimensional data," *Bioinformatics*, vol. 26, pp. 1752–1758, 7 2010. 14, 17

[62] K. H. Kim, J.-Y. Kim, W.-J. Lim, S. Jeong, H.-Y. Lee, Y. Cho, J.-K. Moon, and N. Kim, "Genome-wide association and epistatic interactions of flowering time in soybean cultivar," *PLOS ONE*, vol. 15, p. e0228114, 1 2020. 14

[63] H. Vanhaeren, N. Gonzalez, F. Coppens, L. D. Milde, T. V. Daele, M. Vermeersch, N. B. Eloy, V. Storme, and D. Inzé, "Combining growth-promoting genes leads to positive epistasis in arabidopsis thaliana," *eLife*, vol. 2014, 4 2014. 14

[64] C. C. Cockerham and Z. B. Zeng, "Design iii with marker loci," *Genetics*, vol. 143, p. 1437, 1996. 14

[65] M. D. Ritchie and K. V. Steen, "The search for gene-gene interactions in genome-wide association studies: challenges in abundance of methods, practical considerations, and biological interpretation," *Annals of Translational Medicine*, vol. 6, pp. 157–157, 4 2018. 15

[66] C. Niel, C. Sinoquet, C. Dina, and G. Rocheleau, "A survey about methods dedicated to epistasis detection," *Frontiers in Genetics*, vol. 6, p. 285, 2015. 15

[67] W.-H. Wei, G. Hemani, and C. S. Haley, "Detecting epistasis in human complex traits," *Nature Reviews Genetics 2014 15:11*, vol. 15, pp. 722–733, 9 2014. 15

[68] S. K. Musani, D. Shriner, N. Liu, R. Feng, C. S. Coffey, N. Yi, H. K. Tiwari, and D. B. Allison, "Detection of gene × gene interactions in genome-wide association studies of human population data," *Human Heredity*, vol. 63, pp. 67–84, 2 2007. 15

[69] L. Slim, C. Chatelain, C.-A. Azencott, and J.-P. Vert, "Novel methods for epistasis detection in genome-wide association studies," *PLOS ONE*, vol. 15, p. e0242927, 11 2020. 15

[70] Y.-C. Chang, J.-T. Wu, M.-Y. Hong, Y.-A. Tung, P.-H. Hsieh, S. W. Yee, K. M. Giacomini, Y.-J. Oyang, and C.-Y. Chen, "Genepi: gene-based epistasis discovery using machine learning," *BMC Bioinformatics 2020 21:1*, vol. 21, pp. 1–13, 2 2020. 15, 17

[71] C. C. Chang, C. C. Chow, L. C. Tellier, S. Vattikuti, S. M. Purcell, and J. J. Lee, "Second-generation plink: rising to the challenge of larger and richer datasets," *GigaScience*, vol. 4, 12 2015. 16

[72] T. Schüpbach, I. Xenarios, S. Bergmann, and K. Kapur, "Fastepistasis: a high performance computing solution for quantitative trait epistasis," *Bioinformatics*, vol. 26, pp. 1468–1469, 6 2010. 16

[73] B. Ding, "High-throughput genetic interaction study," *Between the Lines of Genetic Code: Genetic Interactions in Understanding Disease and Complex Phenotypes*, pp. 55–80, 1 2014. 16

[74] T. Zheng, H. Wang, and S.-H. Lo, "Backward genotype-trait association (bgta)-based dissection of complex traits in case-control designs," *Human Heredity*, vol. 62, pp. 196–212, 12 2006. 16

[75] M. D. Ritchie, L. W. Hahn, N. Roodi, L. R. Bailey, W. D. Dupont, F. F. Parl, and J. H. Moore, "Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer," *American Journal of Human Genetics*, vol. 69, p. 138, 2001. 16

[76] P. Yang, J. W. Ho, Y. H. Yang, and B. B. Zhou, "Gene-gene interaction filtering with ensemble of filters," *BMC Bioinformatics 2011 12:1*, vol. 12, pp. 1–10, 2 2011. 16

[77] R. Culverhouse, T. Klein, and W. Shannon, "Detecting epistatic interactions contributing to quantitative traits," *Genetic Epidemiology*, vol. 27, pp. 141–152, 9 2004. 16

[78] L. Wang and M. N. Mehr, "An optimization approach to epistasis detection," *European Journal of Operational Research*, vol. 274, pp. 1069–1076, 5 2019. 16

[79] C. Chatelain, G. Durand, V. Thuillier, and F. Augé, "Performance of epistasis detection methods in semi-simulated gwas," *BMC Bioinformatics 2018 19:1*, vol. 19, pp. 1–17, 6 2018. 16

[80] X. Wan, C. Yang, Q. Yang, H. Xue, N. L. Tang, and W. Yu, "Megasnphunter: a learning approach to detect disease predisposition snps and high level interactions in genome wide association study," *BMC Bioinformatics 2009 10:1*, vol. 10, pp. 1–15, 1 2009. 16

[81] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors)," *https://doi.org/10.1214/aos/1016218223*, vol. 28, pp. 337–407, 4 2000. 16

[82] D. R. Velez, B. C. White, A. A. Motsinger, W. S. Bush, M. D. Ritchie, S. M. Williams, and J. H. Moore, "A balanced accuracy function for epistasis modeling in imbalanced datasets using multifactor dimensionality reduction," *Genetic Epidemiology*, vol. 31, pp. 306–315, 5 2007. 17

[83] AbegazFentaw, V. LishoutFrançois, M. J. M., ChiachoompuKridsadakorn, BhardwajArchana, G. S., WeiZhi, HakonarsonHakon, and V. SteenKristel, "Epistasis detection in genome-wide screening for complex human diseases in structured populations," *https://home.liebertpub.com/sysm*, vol. 2, pp. 19–27, 9 2019. 17

[84] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, pp. 267–288, 1 1996. 17, 87

[85] M. Y. Park and T. Hastie, "Penalized logistic regression for detecting gene interactions," *Biostatistics*, vol. 9, pp. 30–50, 1 2008. 17

[86] A. Bureau, J. Dupuis, K. Falls, K. L. Lunetta, B. Hayward, T. P. Keith, and P. V. Eerdewegh, "Identifying snps predictive of phenotype using random forests," *Genetic Epidemiology*, vol. 28, pp. 171–182, 2 2005. 17

[87] X. Chen, C.-T. Liu, M. Zhang, and H. Zhang, "A forest-based approach to identifying gene and gene–gene interactions," *Proceedings of the National Academy of Sciences*, vol. 104, pp. 19199–19203, 12 2007. 17

[88] P. Yang, J. W. Ho, A. Y. Zomaya, and B. B. Zhou, "A genetic ensemble approach for gene-gene interaction identification," *BMC Bioinformatics 2010 11:1*, vol. 11, pp. 1–15, 10 2010. 17

[89] M. D. Ritchie, B. C. White, J. S. Parker, L. W. Hahn, and J. H. Moore, "Optimization of neural network architecture using genetic programming improves detection and modeling of gene-gene interactions in studies of human diseases," *BMC Bioinformatics 2003 4:1*, vol. 4, pp. 1–14, 7 2003. 17

[90] Y. Zhang and J. S. Liu, "Bayesian inference of epistatic interactions in case-control studies," *Nature Genetics 2007 39:9*, vol. 39, pp. 1167–1173, 8 2007. 17

[91] C. Kooperberg and I. Ruczinski, "Identifying interacting snps using monte carlo logic regression," *Genetic Epidemiology*, vol. 28, pp. 157–170, 2 2005. 17

[92] D. Duroux, H. Climente-González, C.-A. Azencott, and K. V. Steen, "Interpretable network-guided epistasis detection," *bioRxiv*, p. 2020.09.24.310136, 9 2020. 17

[93] G. A. Tuskan, S. DiFazio, S. Jansson, J. Bohlmann, I. Grigoriev, U. Hellsten, M. Putnam, S. Ralph, S. Rombauts, A. Salamov, J. Schein, L. Sterck, A. Aerts, R. R. Bhalerao, R. P. Bhalerao, D. Blaudez, W. Boerjan, A. Brun, A. Brunner, V. Busov, M. Campbell, J. Carlson, M. Chalot, J. Chapman, G. L. Chen, D. Cooper, P. M. Coutinho, J. Couturier, S. Covert, Q. Cronk, R. Cunningham, J. Davis, S. Degroeve,

A. Dŕjardin, C. DePamphilis, J. Detter, B. Dirks, I. Dubchak, S. Duplessis, J. Ehlting, B. Ellis, K. Gendler, D. Goodstein, M. Gribskov, J. Grimwood, A. Groover, L. Gunter, B. Hamberger, B. Heinze, Y. Helariutta, B. Henrissat, D. Holligan, R. Holt, W. Huang, N. Islam-Faridi, S. Jones, M. Jones-Rhoades, R. Jorgensen, C. Joshi, J. Kangasjärvi, J. Karlsson, C. Kelleher, R. Kirkpatrick, M. Kirst, A. Kohler, U. Kalluri, F. Larimer, J. Leebens-Mack, J. C. Leplé, P. Locascio, Y. Lou, S. Lucas, F. Martin, B. Montanini, C. Napoli, D. R. Nelson, C. Nelson, K. Nieminen, O. Nilsson, V. Pereda, G. Peter, R. Philippe, G. Pilate, A. Poliakov, J. Razumovskaya, P. Richardson, C. Rinaldi, K. Ritland, P. Rouzé, D. Ryaboy, J. Schmutz, J. Schrader, B. Segerman, H. Shin, A. Siddiqui, F. Sterky, A. Terry, C. J. Tsai, E. Uberbacher, P. Unneberg, J. Vahala, K. Wall, S. Wessler, G. Yang, T. Yin, C. Douglas, M. Marra, G. Sandberg, Y. Van De Peer, and D. Rokhsar, "The genome of black cottonwood, Populus trichocarpa (Torr. & Gray)," *Science*, vol. 313, pp. 1596–1604, sep 2006. 17, 41, 65, 69

[94] P. Gilna, L. R. Lynd, D. Mohnen, M. F. Davis, and B. H. Davison, "Progress in understanding and overcoming biomass recalcitrance: A BioEnergy Science Center (BESC) perspective Mike Himmel," *Biotechnology for Biofuels*, vol. 10, p. 285, nov 2017. 17, 41

[95] L. M. Evans, G. T. Slavov, E. Rodgers-Melnick, J. Martin, P. Ranjan, W. Muchero, A. M. Brunner, W. Schackwitz, L. Gunter, J.-G. Chen, *et al.*, "Population genomics of populus trichocarpa identifies signatures of selection and adaptive trait associations," *Nature genetics*, vol. 46, no. 10, pp. 1089–1096, 2014. 18

[96] G. Tuskan, G. Slavov, S. DiFazio, W. Muchero, R. Pryia, W. Schackwitz, J. Martin, D. Rokhsar, R. Sykes, M. Davis, *et al.*, "Populus resequencing: towards genome-wide association studies," in *BMC Proceedings*, vol. 5, pp. 1–1, BioMed Central, 2011. 18

[97] M. Miranda, S. G. Ralph, R. Mellway, R. White, M. C. Heath, J. Bohlmann, and C. P. Constabel, "The transcriptional response of hybrid poplar (Populus trichocarpa x P. deltoides) to infection by Melampsora medusae leaf rust involves induction of flavonoid

pathway genes leading to the accumulation of proanthocyanidins," *Molecular Plant-Microbe Interactions*, vol. 20, pp. 816–831, jul 2007. 18, 66

[98] G. Newcombe, " An Epidemic of Septoria Leaf Spot on Populus trichocarpa in the Pacific Northwest in 1993 ," *Plant Disease*, vol. 79, p. 212, 1995. 18, 66

[99] F. P. Guerra, H. Suren, J. Holliday, J. H. Richards, O. Fiehn, R. Famula, B. J. Stanton, R. Shuren, R. Sykes, and M. F. Davis, "Exome resequencing and GWAS for growth, ecophysiology, and chemical and metabolomic composition of wood of Populus trichocarpa," *BMC genomics*, vol. 20, no. 1, pp. 1–14, 2019. 18

[100] A. D. McKown, J. Klápště, R. D. Guy, A. Geraldes, I. Porth, J. Hannemann, M. Friedmann, W. Muchero, G. A. Tuskan, J. Ehlting, *et al.*, "Genome-wide association implicates numerous genes underlying ecological trait variation in natural populations of populus trichocarpa," *New Phytologist*, vol. 203, no. 2, pp. 535–553, 2014. 18

[101] A. D. McKown, J. Klápště, R. D. Guy, Y. A. El-Kassaby, and S. D. Mansfield, "Ecological genomics of variation in bud-break phenology and mechanisms of response to climate warming in Populus trichocarpa," *New Phytologist*, vol. 220, pp. 300–316, oct 2018. 18

[102] C. Darwin and A. Wallace, "On the tendency of species to form varieties; and on the perpetuation of varieties and species by natural means of selection.," *Journal of the Proceedings of the Linnean Society of London. Zoology*, vol. 3, pp. 45–62, 8 1858. 18, 58

[103] J. G. Kingsolver, H. E. Hoekstra, J. M. Hoekstra, D. Berrigan, S. N. Vignieri, C. E. Hill, A. Hoang, P. Gibert, and P. Beerli, "The strength of phenotypic selection in natural populations," *https://doi.org/10.1086/319193*, vol. 157, pp. 245–261, 7 2015. 18

[104] J. G. Kingsolver and D. W. Pfennig, "Patterns and power of phenotypic selection in nature," *BioScience*, vol. 57, pp. 561–572, 7 2007. 18

[105] W. G. Hill, "Heritability," *Brenner's Encyclopedia of Genetics: Second Edition*, pp. 432–434, 1 2013. 19

[106] R. Burdon, "Genetics and genetic resources — quantitative genetic principles," *Encyclopedia of Forest Sciences*, pp. 182–187, 1 2004. 19

[107] R. Bernardo, "Reinventing quantitative genetics for plant breeding: something old, something new, something borrowed, something blue," *Heredity 2020 125:6*, vol. 125, pp. 375–385, 4 2020. 19, 62

[108] P. Fasahat, "Principles and utilization of combining ability in plant breeding," *Biometrics Biostatistics International Journal*, vol. Volume 4, 6 2016. 19

[109] N. N. Godswill, N. E. G. Frank, A. N. Walter, M. Y. J. Edson, T. M. Kingsley, V. Arondel, B. J. Martin, and Y. Emmanuel, "Oil palm," *Breeding Oilseed Crops for Sustainable Production: Opportunities and Constraints*, pp. 217–273, 1 2016. 19

[110] P. P. Jauhar and W. W. Hanna, "Cytogenetics and genetics of pearl millet," *Advances in Agronomy*, vol. 64, pp. 1–26, 1 1998. 19

[111] G. L. Bennett, E. J. Pollak, L. A. Kuehn, and W. M. Snelling, "Breeding: Animals," *Encyclopedia of Agriculture and Food Systems*, pp. 173–186, 1 2014. 19

[112] B. Griffing, " concept of general and specific combining ability in • relation to diallel crossing systems," *Australian Journal of Biological Sciences*, vol. 9, pp. 463–493, 1956. 19, 62

[113] J. Muthoni and H. Shimelis, "Mating designs commonly used in plant breeding: A review," *AJCS*, vol. 14, pp. 1835–2707, 2020. 20

[114] G. F. Sprague and L. A. Tatum, "General vs. specific combining ability in single crosses of corn1," *Agronomy Journal*, vol. 34, pp. 923–932, 10 1942. 20, 59

[115] P. Rukundo, H. Shimelis, M. Laing, and D. Gahakwa, "Combining ability, maternal effects, and heritability of drought tolerance, yield and yield components in sweetpotato," *Frontiers in Plant Science*, vol. 0, p. 1981, 1 2017. 20

[116] D. P. Singh, A. K. Singh, and A. Singh, "Inbreeding depression and heterosis," *Plant Breeding and Cultivar Development*, pp. 291–303, 1 2021. 20

[117] J. W. Dudley and R. H. Moll, "Interpretation and use of estimates of heritability and genetic variances in plant breeding1," *Crop Science*, vol. 9, pp. 257–262, 5 1969. 20, 61

[118] D. Wallach, D. Makowski, J. W. Jones, and F. Brun, "Gene-based crop models," *Working with Dynamic Crop Models*, pp. 445–486, 1 2019. 20

[119] D. P. Singh, A. K. Singh, and A. Singh, "Population improvement," *Plant Breeding and Cultivar Development*, pp. 305–330, 1 2021. 20, 62

[120] S. A. Clark and J. Van Der Werf, "Genomic best linear unbiased prediction (gBLUP) for the estimation of genomic breeding values," *Methods in Molecular Biology*, vol. 1019, pp. 321–330, 2013. 21, 42, 62

[121] P. G. R. Vlaming, "The current and future use of ridge regression for prediction in quantitative genetics," *Biomed Res Int*, vol. 2015, p. 143712, 2015. 21, 62

[122] J. B. Endelman, "Ridge Regression and Other Kernels for Genomic Selection with R Package rrBLUP," *The Plant Genome*, vol. 4, pp. 250–255, nov 2011. 21, 62

[123] D. Paudel, S. Dhakal, S. Parajuli, L. Adhikari, Z. Peng, Y. Qian, D. Shahi, M. Avci, S. O. Makaju, and B. Kannan, "Use of quantitative trait loci to develop stress tolerance in plants," *Plant Life Under Changing Environment*, pp. 917–965, 1 2020. 21

[124] K. T. Wada and D. R. Jerry, "Population genetics and stock improvement," *The Pearl Oyster*, pp. 437–471, 1 2008. 21

[125] A. Cuesta-Marcos, J. G. Kling, A. R. Belcher, T. Filichkin, S. P. Fisk, R. Graebner, L. Helgerson, D. Herb, B. Meints, A. S. Ross, P. M. Hayes, and S. E. Ulrich, "Barley: Genetics and breeding," *Encyclopedia of Food Grains: Second Edition*, vol. 4-4, pp. 287–295, 1 2016. 21

[126] C. Prakash, A. M. Sevanthi, and P. S. Shanmugavadivel, "Use of qtls in developing abiotic stress tolerance in rice," *Advances in Rice Research for Abiotic Stress Tolerance*, pp. 869–893, 1 2019. 21

[127] U. Isildak, A. Stella, and M. Fumagalli, "Distinguishing between recent balancing selection and incomplete sweep using deep neural networks," *Molecular Ecology Resources*, pp. 1755–0998.13379, apr 2021. 21

[128] D. Gianola, H. Okut, K. A. Weigel, and G. J. Rosa, "Predicting complex quantitative traits with Bayesian neural networks: A case study with Jersey cows and wheat," *BMC Genetics*, vol. 12, oct 2011. 21, 60

[129] P. Pérez-Rodríguez, D. Gianola, J. M. González-Camacho, J. Crossa, Y. Manès, and S. Dreisigacker, "Comparison between linear and non-parametric regression models for genome-enabled prediction in wheat," *G3: Genes, Genomes, Genetics*, vol. 2, pp. 1595–1605, dec 2012. 21, 60

[130] J. M. González-Camacho, G. de los Campos, P. Pérez, D. Gianola, J. E. Cairns, G. Mahuku, R. Babu, and J. Crossa, "Genome-enabled prediction of genetic values using radial basis function neural networks," *Theoretical and Applied Genetics*, vol. 125, pp. 759–771, aug 2012. 21

[131] R. C. Becker, "Anticipating the long-term cardiovascular effects of covid-19," *Journal of Thrombosis and Thrombolysis*, vol. 50, pp. 512–524, Oct 2020. 22

[132] S. Lopez-Leon, T. Wegman-Ostrosky, C. Perelman, R. Sepulveda, P. A. Rebolledo, A. Cuapio, and S. Villapol, "More than 50 long-term effects of covid-19: a systematic review and meta-analysis," *Scientific Reports*, vol. 11, p. 16144, Aug 2021. 22

[133] G. Dudas, S. L. Hong, B. I. Potter, S. Calvignac-Spencer, F. S. Niatou-Singa, T. B. Tombolomako, T. Fuh-Neba, U. Vickos, M. Ulrich, F. H. Leendertz, K. Khan, C. Huber, A. Watts, I. Olendraitė, J. Snijder, K. N. Wijnant, A. M. Bonvin, P. Martres, S. Behillil, A. Ayouba, M. F. Maidadi, D. M. Djomsi, C. Godwe, C. Butel, A. Šimaitis, M. Gabrielaitė, M. Katėnaitė, R. Norvilas, L. Raugaitė, G. W. Koyaweda, J. K.

Kandou, R. Jonikas, I. Nasvytienė, Živilė Žemeckienė, D. Gečys, K. Tamušauskaitė, M. Norkienė, E. Vasiliūnaitė, D. Žiogienė, A. Timinskas, M. Šukys, M. Šarauskas, G. Alzbutas, A. A. Aziza, E. K. Lusamaki, J.-C. M. Cigolo, F. M. Mawete, E. L. Lofiko, P. M. Kingebeni, J.-J. M. Tamfum, M. R. D. Belizaire, R. G. Essomba, M. C. O. Assoumou, A. B. Mboringong, A. B. Dieng, D. Juozapaitė, S. Hosch, J. Obama, M. O. Ayekaba, D. Naumovas, A. Pautienius, C. D. Rafaï, A. Vitkauskienė, R. Ugenskienė, A. Gedvilaitė, D. Čereškevičius, V. Lesauskaitė, L. Žemaitis, L. Griškevičius, and G. Baele, "Emergence and spread of sars-cov-2 lineage b.1.620 with variant of concern-like mutations and deletions," *Nature Communications 2021 12:1*, vol. 12, pp. 1–12, 10 2021. 22, 104

[134] E. B. Hodcroft, N. De Maio, R. Lanfear, D. R. MacCannell, B. Q. Minh, H. A. Schmidt, A. Stamatakis, N. Goldman, and C. Dessimoz, "Want to track pandemic variants faster? Fix the bioinformatics bottleneck," *Nature*, vol. 591, pp. 30–33, mar 2021. 22, 103

[135] M. R. Garvin, E. T. Prates, M. Pavicic, P. Jones, B. K. Amos, A. Geiger, M. B. Shah, J. Streich, J. G. Felipe Machado Gazolla, D. Kainer, A. Cliff, J. Romero, N. Keith, J. B. Brown, and D. Jacobson, "Potentially adaptive SARS-CoV-2 mutations discovered with novel spatiotemporal and explainable AI models," *Genome Biology*, vol. 21, pp. 1–26, dec 2020. 22, 104

[136] S. Ludwig and A. Zarbock, "Coronaviruses and sars-cov-2: A brief overview," *Anesthesia and analgesia*, vol. 131, pp. 93–96, Jul 2020. 32243297[pmid]. 22

[137] R. A. Khailany, M. Safdar, and M. Ozaslan, "Genomic characterization of a novel sars-cov-2," *Gene Reports*, vol. 19, p. 100682, Jun 2020. 22

[138] R. Sanjuán and P. Domingo-Calap, "Mechanisms of viral mutation," *Cellular and molecular life sciences : CMLS*, vol. 73, pp. 4433–4448, Dec 2016. 27392606[pmid]. 23

[139] L. Liu, F. Zeng, J. Rao, S. Yuan, M. Ji, X. Lei, X. Xiao, Z. Li, X. Li, W. Du, Y. Liu, H. Tan, J. Li, J. Zhu, J. Yang, and Z. Liu, "Comparison of clinical features and

outcomes of medically attended covid-19 and influenza patients in a defined population in the 2020 respiratory virus season," *Frontiers in Public Health*, vol. 9, 2021. 23

[140] K. Tao, P. L. Tzou, J. Nouhin, R. K. Gupta, T. de Oliveira, S. L. Kosakovsky Pond, D. Fera, and R. W. Shafer, "The biological and clinical significance of emerging sars-cov-2 variants," *Nature Reviews Genetics*, vol. 22, pp. 757–773, Dec 2021. 23

[141] A. J. Greaney, T. N. Starr, P. Gilchuk, S. J. Zost, E. Binshtein, A. N. Loes, S. K. Hilton, J. Huddleston, R. Eguia, K. H. Crawford, A. S. Dingens, R. S. Nargi, R. E. Sutton, N. Suryadevara, P. W. Rothlauf, Z. Liu, S. P. Whelan, R. H. Carnahan, J. E. Crowe, and J. D. Bloom, "Complete mapping of mutations to the sars-cov-2 spike receptor-binding domain that escape antibody recognition," *Cell Host Microbe*, vol. 29, p. 44, 1 2021. 23

[142] A. Rambaut, E. C. Holmes, Áine O'Toole, V. Hill, J. T. McCrone, C. Ruis, L. du Plessis, and O. G. Pybus, "A dynamic nomenclature proposal for sars-cov-2 lineages to assist genomic epidemiology," *Nature Microbiology 2020 5:11*, vol. 5, pp. 1403–1407, 7 2020. 23

[143] M. Pérez-Losada, M. Arenas, J. C. Galán, F. Palero, and F. González-Candelas, "Recombination in viruses: mechanisms, methods of study, and evolutionary consequences," *Infection, genetics and evolution : journal of molecular epidemiology and evolutionary genetics in infectious diseases*, vol. 30, pp. 296–307, Mar 2015. 25541518[pmid]. 23, 103

[144] M.-J. Kim and C. Kao, "Factors regulating template switch ¡i¿in vitro¡/i¿ by viral rna-dependent rna polymerases: Implications for rna–rna recombination," *Proceedings of the National Academy of Sciences*, vol. 98, no. 9, pp. 4972–4977, 2001. 23

[145] T. E. Schlub, R. P. Smyth, A. J. Grimm, J. Mak, and M. P. Davenport, "Accurately measuring recombination between closely related hiv-1 genomes," *PLoS Computational Biology*, vol. 6, p. 1000766, 4 2010. 23

[146] H. Yi, "2019 novel coronavirus is undergoing active recombination," *Clinical Infectious Diseases*, vol. 71, pp. 884–887, 7 2020. 23, 126

[147] D. VanInsberghe, A. S. Neish, A. C. Lowen, and K. Koelle, "Recombinant sars-cov-2 genomes circulated at low levels over the first year of the pandemic," *Virus Evolution*, vol. 7, 12 2021. 23, 126

[148] R. C. C. D. M. P. K. S. M. A. R. R. Juan Ángel Patiño-Galindo, Ioan Filip, "Recombination and lineage-specific mutations linked to the emergence of sars-cov-2," *Genome Medicine*, vol. 13, pp. 1–14, 12 2021. 24

[149] X. Li, E. E. Giorg, M. H. Marichannegowda, B. Foley, C. Xiao, X. P. Kong, Y. Chen, S. Gnanakaran, B. Korber, and F. Gao, "Emergence of sars-cov-2 through recombination and strong purifying selection," *Science Advances*, vol. 6, 7 2020. 24

[150] B. Jackson, M. F. Boni, M. J. Bull, A. Colleran, R. M. Colquhoun, A. C. Darby, S. Haldenby, V. Hill, A. Lucaci, J. T. McCrone, S. M. Nicholls, Áine O'Toole, N. Pacchiarini, R. Poplawski, E. Scher, F. Todd, H. J. Webster, M. Whitehead, C. Wierzbicki, N. J. Loman, T. R. Connor, D. L. Robertson, O. G. Pybus, and A. Rambaut, "Generation and transmission of interlineage recombinants in the sars-cov-2 pandemic," *Cell*, vol. 184, pp. 5179–5188.e8, 9 2021. 24

[151] Y. Turkahia, B. Thornlow, A. Hinrichs, J. McBroome, N. Ayala, C. Ye, N. D. Maio, D. Haussler, R. Lanfear, and R. Corbett-Detig, "Pandemic-scale phylogenomics reveals elevated recombination rates in the sars-cov-2 spike region," *bioRxiv*, p. 2021.08.04.455157, 8 2021. 24, 125

[152] E. Santiago and A. Caballero, "The value of targeting recombination as a strategy against coronavirus diseases," *Heredity 2020 125:4*, vol. 125, pp. 169–172, 6 2020. 24, 126

[153] J. D. Jensen, R. A. Stikeleather, T. F. Kowalik, and M. Lynch, "Imposed mutational meltdown as an antiviral strategy," *Evolution*, vol. 74, pp. 2549–2559, 12 2020. 24, 126

[154] M. H. Schierup and J. Hein, "Consequences of recombination on traditional phylogenetic analysis.," *Genetics*, vol. 156, p. 879, 2000. 24

[155] D. Penny and M. Hendy, "Estimating the reliability of evolutionary trees.," *Molecular Biology and Evolution*, vol. 3, pp. 403–417, sep 1986. 24, 108

[156] T. C. Bruen, H. Philippe, and D. Bryant, "A simple and robust statistical test for detecting the presence of recombination," *Genetics*, vol. 172, pp. 2665–2681, apr 2006. 25, 108

[157] Y. Turakhia, B. Thornlow, A. S. Hinrichs, N. D. Maio, L. Gozashti, R. Lanfear, D. Haussler, and R. Corbett-Detig, "Ultrafast sample placement on existing trees (usher) enables real-time phylogenetics for the sars-cov-2 pandemic," *Nature Genetics 2021 53:6*, vol. 53, pp. 809–816, 5 2021. 25, 103

[158] 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*, vol. 12, pp. 2825–2830, 2011. 31, 88

[159] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İlhan Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, and P. van Mulbregt, "Scipy 1.0: fundamental algorithms for scientific computing in python," *Nature Methods 2020 17:3*, vol. 17, pp. 261–272, 2 2020. 31

[160] K. H. Brodersen, C. S. Ong, K. E. Stephan, and J. M. Buhmann, "The balanced accuracy and its posterior distribution," *Proceedings - International Conference on Pattern Recognition*, pp. 3121–3124, 2010. 40

[161] W. Zhang, M. Doherty, E. Pascual, T. Bardin, V. Barskova, P. Conaghan, J. Gerster, J. Jacobs, B. Leeb, F. Lioté, G. McCarthy, P. Netter, G. Nuki, F. Perez-Ruiz, A. Pignone, J. Pimentão, L. Punzi, E. Roddy, T. Uhlig, and I. Zimmermann-Gòrska, "Eular evidence based recommendations for gout. part i: Diagnosis. report of a task force of the standing committee for international clinical studies including therapeutics (escisit)," *Annals of the Rheumatic Diseases*, vol. 65, p. 1301, 10 2006. 41

[162] P. Muthuirulan and T. D. Capellini, "Complex phenotypes: Mechanisms underlying variation in human stature," *Current osteoporosis reports*, vol. 17, p. 301, 10 2019. 42

[163] M. D. Resende, M. F. Resende, C. P. Sansaloni, C. D. Petroli, A. A. Missiaggia, A. M. Aguiar, J. M. Abad, E. K. Takahashi, A. M. Rosado, D. A. Faria, G. J. Pappas, A. Kilian, and D. Grattapaglia, "Genomic selection for growth and wood quality in eucalyptus: capturing the missing heritability and accelerating breeding for complex traits in forest trees," *New Phytologist*, vol. 194, pp. 116–128, 4 2012. 42, 53

[164] S. Carbon, E. Douglass, B. M. Good, D. R. Unni, N. L. Harris, C. J. Mungall, S. Basu, R. L. Chisholm, R. J. Dodson, E. Hartline, P. Fey, P. D. Thomas, L. P. Albou, D. Ebert, M. J. Kesling, H. Mi, A. Muruganujan, X. Huang, T. Mushayahama, S. A. LaBonte, D. A. Siegele, G. Antonazzo, H. Attrill, N. H. Brown, P. Garapati, S. J. Marygold, V. Trovisco, G. dos Santos, K. Falls, C. Tabone, P. Zhou, J. L. Goodman, V. B. Strelets, J. Thurmond, P. Garmiri, R. Ishtiaq, M. Rodríguez-López, M. L. Acencio, M. Kuiper, A. Lægreid, C. Logie, R. C. Lovering, B. Kramarz, S. C. Saverimuttu, S. M. Pinheiro, H. Gunn, R. Su, K. E. Thurlow, M. Chibucos, M. Giglio, S. Nadendla, J. Munro, R. Jackson, M. J. Duesbury, N. Del-Toro, B. H. Meldal, K. Paneerselvam, L. Perfetto, P. Porras, S. Orchard, A. Shrivastava, H. Y. Chang, R. D. Finn, A. L. Mitchell, N. D. Rawlings, L. Richardson, A. Sangrador-Vegas, J. A. Blake, K. R. Christie, M. E. Dolan, H. J. Drabkin, D. P. Hill, L. Ni, D. M. Sitnikov, M. A. Harris, S. G. Oliver, K. Rutherford, V. Wood, J. Hayles, J. Bähler, E. R. Bolton, J. L. de Pons, M. R. Dwinell, G. T. Hayman, M. L. Kaldunski, A. E. Kwitek, S. J. Laulederkind, C. Plasterer, M. A. Tutaj, M. Vedi, S. J. Wang, P. D'Eustachio, L. Matthews, J. P. Balhoff, S. A. Aleksander, M. J. Alexander, J. M. Cherry, S. R. Engel, F. Gondwe,

K. Karra, S. R. Miyasato, R. S. Nash, M. Simison, M. S. Skrzypek, S. Weng, E. D. Wong, M. Feuermann, P. Gaudet, A. Morgat, E. Bakker, T. Z. Berardini, L. Reiser, S. Subramaniam, E. Huala, C. N. Arighi, A. Auchincloss, K. Axelsen, G. Argoud-Puy, A. Bateman, M. C. Blatter, E. Boutet, E. Bowler, L. Breuza, A. Bridge, R. Britto, H. Bye-A-Jee, C. C. Casas, E. Coudert, P. Denny, A. Es-Treicher, M. L. Famiglietti, G. Georghiou, A. N. Gos, N. Gruaz-Gumowski, E. Hatton-Ellis, C. Hulo, A. Ignatchenko, F. Jungo, K. Laiho, P. L. Mercier, D. Lieberherr, A. Lock, Y. Lussi, A. MacDougall, M. Ma-Grane, M. J. Martin, P. Masson, D. A. Natale, N. Hyka-Nouspikel, S. Orchard, I. Pedruzzi, L. Pourcel, S. Poux, S. Pundir, C. Rivoire, E. Speretta, S. Sundaram, N. Tyagi, K. Warner, R. Zaru, C. H. Wu, A. D. Diehl, J. N. Chan, C. Grove, R. Y. Lee, H. M. Muller, D. Raciti, K. van Auken, P. W. Sternberg, M. Berriman, M. Paulini, K. Howe, S. Gao, A. Wright, L. Stein, D. G. Howe, S. Toro, M. Westerfield, P. Jaiswal, L. Cooper, and J. Elser, "The gene ontology resource: enriching a gold mine," *Nucleic Acids Research*, vol. 49, pp. D325–D334, 1 2021. 42

[165] J. Jin, K. He, X. Tang, Z. Li, L. Lv, Y. Zhao, J. Luo, and G. Gao, "An arabidopsis transcriptional regulatory map reveals distinct functional and evolutionary features of novel transcription factors," *Molecular Biology and Evolution*, vol. 32, pp. 1767–1773, 7 2015. 42

[166] J. Jin, F. Tian, D. C. Yang, Y. Q. Meng, L. Kong, J. Luo, and G. Gao, "Planttfdb 4.0: toward a central hub for transcription factors and regulatory interactions in plants," *Nucleic Acids Research*, vol. 45, pp. D1040–D1045, 1 2017. 42

[167] F. Tian, D. C. Yang, Y. Q. Meng, J. Jin, and G. Gao, "Plantregmap: charting functional regulatory maps in plants," *Nucleic Acids Research*, vol. 48, pp. D1104–D1113, 1 2020. 42

[168] N. Q. K. Le, E. K. Y. Yapp, N. Nagasundaram, and H. Y. Yeh, "Classifying promoters by interpreting the hidden information of dna sequences via deep learning and combination of continuous fasttext n-grams," *Frontiers in Bioengineering and Biotechnology*, vol. 7, p. 305, 11 2019. 53

[169] C. Jackson, N. Christie, S. M. Reynolds, G. C. Marais, Y. Tii-kuzu, M. Caballero, T. Kampman, E. A. Visser, S. Naidoo, D. Kain, R. W. Whetten, F. Isik, J. Wegrzyn, G. R. Hodge, J. J. Acosta, and A. A. Myburg, "A genome-wide snp genotyping resource for tropical pine tree species," *Molecular ecology resources*, vol. 22, pp. 695–710, 2 2022. 53

[170] D. M. Goodstein, S. Shu, R. Howson, R. Neupane, R. D. Hayes, J. Fazo, T. Mitros, W. Dirks, U. Hellsten, N. Putnam, and D. S. Rokhsar, "Phytozome: a comparative platform for green plant genomics," *Nucleic Acids Research*, vol. 40, pp. D1178–D1186, 1 2012. 53

[171] U. K. Sukumar and G. Packirisamy, "Bioactive core-shell nanofiber hybrid scaffold for efficient suicide gene transfection and subsequent time resolved delivery of prodrug for anticancer therapy," *ACS Applied Materials and Interfaces*, vol. 7, pp. 18717–18731, 8 2015. 53

[172] I. Chitrakar, D. M. Kim-Holzapfel, W. Zhou, and J. B. French, "Higher order structures in purine and pyrimidine metabolism," *Journal of Structural Biology*, vol. 197, pp. 354–364, 3 2017. 53

[173] T. Wunder and O. Mueller-Cajar, "Biomolecular condensates in photosynthesis and metabolism," *Current Opinion in Plant Biology*, vol. 58, pp. 1–7, 12 2020. 54

[174] S. Ambawat, P. Sharma, N. R. Yadav, and R. C. Yadav, "Myb transcription factor genes as regulators for plant responses: an overview," *Physiology and Molecular Biology of Plants*, vol. 19, p. 307, 7 2013. 54

[175] L. Lovász and H. J. Prömel, "Combinatorics," *Oberwolfach Reports*, vol. 1, no. 1, pp. 5–110, 2004. 54

[176] A. Valdeolivas, L. Tichit, C. Navarro, S. Perrin, G. Odelin, N. Levy, P. Cau, E. Remy, and A. Baudot, "Random walk with restart on multiplex and heterogeneous biological networks," *Bioinformatics*, vol. 35, pp. 497–505, 2 2019. 54

[177] U. K. Aryal, Z. McBride, D. Chen, J. Xie, and D. B. Szymanski, "Analysis of protein complexes in arabidopsis leaves using size exclusion chromatography and label-free protein correlation profiling," *Journal of proteomics*, vol. 166, pp. 8–18, 8 2017. 54

[178] A. Harfouche, R. Meilan, and A. Altman, "Molecular and physiological responses to abiotic stress in forest trees and their relevance to tree improvement," *Tree Physiology*, vol. 34, pp. 1181–1198, 11 2014. 54

[179] S. Eswaramoorthy, J. B. Bonanno, S. K. Burley, and S. Swaminathan, "Mechanism of action of a flavin-containing monooxygenase," *Proceedings of the National Academy of Sciences*, vol. 103, pp. 9832–9837, 6 2006. 55

[180] A. W. Artenstein and S. M. Opal, "Proprotein convertases in health and disease," *New England Journal of Medicine*, vol. 365, pp. 2507–2518, 12 2011. 55

[181] J. X. Chong, K. J. Buckingham, S. N. Jhangiani, C. Boehm, N. Sobreira, J. D. Smith, T. M. Harrell, M. J. McMillin, W. Wiszniewski, T. Gambin, Z. H. C. Akdemir, K. Doheny, A. F. Scott, D. Avramopoulos, A. Chakravarti, J. Hoover-Fong, D. Mathews, P. D. Witmer, H. Ling, K. Hetrick, L. Watkins, K. E. Patterson, F. Reinier, E. Blue, D. Muzny, M. Kircher, K. Bilguvar, F. López-Giráldez, V. R. Sutton, H. K. Tabor, S. M. Leal, M. Gunel, S. Mane, R. A. Gibbs, E. Boerwinkle, A. Hamosh, J. Shendure, J. R. Lupski, R. P. Lifton, D. Valle, D. A. Nickerson, and M. J. Bamshad, "The genetic basis of mendelian phenotypes: Discoveries, challenges, and opportunities," *American Journal of Human Genetics*, vol. 97, p. 199, 8 2015. 58

[182] T. Näholm, S. Palmroth, U. Ganeteg, M. Moshelion, V. Hurry, and O. Franklin, "Genetics of superior growth traits in trees are being mapped but will the faster-growing risk-takers make it in the wild?," *Tree Physiology*, vol. 34, pp. 1141–1148, 11 2014. 59

[183] R. J. Baker, "Issues in diallel analysis," *Crop Science*, vol. 18, pp. 533–536, 7 1978. 62

[184] M. Pérez-Enciso, N. Forneris, G. de los Campos, and A. Legarra, "Evaluating sequence-based genomic prediction with an efficient new simulator," *Genetics*, vol. 205, pp. 939–953, feb 2017. 63, 69

[185] O. J. Kirtley, K. van Mens, M. Hoogendoorn, N. Kapur, and D. de Beurs, "Translating promise into practice: a review of machine learning in suicide research and prevention," *The Lancet Psychiatry*, vol. 9, pp. 243–252, 3 2022. 80

[186] Q. D. Buchlak, N. Esmaili, J. C. Leveque, C. Bennett, F. Farrokhi, and M. Piccardi, "Machine learning applications to neuroimaging for glioma detection and classification: An artificial intelligence augmented systematic review," *Journal of Clinical Neuroscience*, vol. 89, pp. 177–198, 7 2021. 80

[187] K. Santos, J. P. Dias, and C. Amado, "A literature review of machine learning algorithms for crash injury severity prediction," *Journal of Safety Research*, vol. 80, pp. 254–269, 2 2022. 80

[188] O. C. Adekoya, M. E. Yibowei, G. J. Adekoya, E. R. Sadiku, Y. Hamam, and S. S. Ray, "A mini-review on the application of machine learning in polymer nanogels for drug delivery," *Materials Today: Proceedings*, 2 2022. 80

[189] D. D. Martinelli, "Generative machine learning for de novo drug discovery: A systematic review," *Computers in Biology and Medicine*, vol. 145, p. 105403, 6 2022. 80

[190] A. Khan, A. D. Vibhute, S. Mali, and C. Patil, "A systematic review on hyperspectral imaging technology with a machine and deep learning methodology for agricultural applications," *Ecological Informatics*, vol. 69, p. 101678, 7 2022. 80

[191] I. Batool and T. A. Khan, "Software fault prediction using data mining, machine learning and deep learning techniques: A systematic literature review," *Computers and Electrical Engineering*, vol. 100, p. 107886, 5 2022. 80

[192] K. Kourou, K. P. Exarchos, C. Papaloukas, P. Sakaloglou, T. Exarchos, and D. I. Fotiadis, "Applied machine learning in cancer research: A systematic review for patient

diagnosis, classification and prognosis," *Computational and Structural Biotechnology Journal*, vol. 19, pp. 5546–5555, 1 2021. 80

[193] E. R. Okoroafor, C. M. Smith, K. I. Ochie, C. J. Nwosu, H. Gudmundsdottir, and M. J. Aljubran, "Machine learning in subsurface geothermal energy: Two decades in review," *Geothermics*, vol. 102, p. 102401, 6 2022. 80

[194] R. D. Sandberg and Y. Zhao, "Machine-learning for turbulence and heat-flux model development: A review of challenges associated with distinct physical phenomena and progress to date," *International Journal of Heat and Fluid Flow*, vol. 95, p. 108983, 6 2022. 80

[195] O. Dogan, S. Tiwari, M. A. Jabbar, and S. Guggari, "A systematic review on ai/ml approaches against covid-19 outbreak," *Complex Intelligent Systems 2021 7:5*, vol. 7, pp. 2655–2678, 7 2021. 80

[196] Z. Ullah, F. Al-Turjman, L. Mostarda, and R. Gagliardi, "Applications of artificial intelligence and machine learning in smart cities," *Computer Communications*, vol. 154, pp. 313–323, 3 2020. 80

[197] J. L. Faulon and L. Faure, "In silico, in vitro, and in vivo machine learning in synthetic biology and metabolic engineering," *Current Opinion in Chemical Biology*, vol. 65, pp. 85–92, 12 2021. 80

[198] Y. K. Wan, C. Hendra, P. N. Pratanwanich, and J. Göke, "Beyond sequencing: machine learning algorithms extract biology hidden in nanopore signal data," *Trends in Genetics*, vol. 38, pp. 246–257, 3 2022. 80

[199] B. Yuan, C. Shen, A. Luna, A. Korkut, D. S. Marks, J. Ingraham, and C. Sander, "Cellbox: Interpretable machine learning for perturbation biology with application to the design of cancer combination therapy," *Cell Systems*, vol. 12, pp. 128–140.e4, 2 2021. 80

[200] S. O'Neill, "Machine learning turbocharges structural biology," *Engineering*, 3 2022. 80

[201] E. Arslan, J. Schulz, and K. Rai, "Machine learning in epigenomics: Insights into cancer biology and medicine," *Biochimica et Biophysica Acta (BBA) - Reviews on Cancer*, vol. 1876, p. 188588, 12 2021. 80

[202] F. W. Stearns, "Anecdotal, historical and critical commentaries on genetics: One hundred years of pleiotropy: A retrospective," *Genetics*, vol. 186, p. 767, 11 2010. 81

[203] M. Elhefnawy, M.-S. Ouali, and A. Ragab, "Multi-output regression using polygon generation and conditional generative adversarial networks," *Expert Systems with Applications*, vol. 203, p. 117288, 10 2022. 81

[204] A. Holzinger, C. Biemann, C. S. Pattichis, and D. B. Kell, "What do we need to build explainable ai systems for the medical domain?," 12 2017. 81

[205] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature*, vol. 585, pp. 357–362, Sept. 2020. 88

[206] T. pandas development team, "pandas-dev/pandas: Pandas," Feb. 2020. 89

[207] D. Marbach, J. C. Costello, R. Küffner, N. M. Vega, R. J. Prill, D. M. Camacho, K. R. Allison, M. Kellis, J. J. Collins, A. Aderhold, G. Stolovitzky, R. Bonneau, Y. Chen, F. Cordero, M. Crane, F. Dondelinger, M. Drton, R. Esposito, R. Foygel, A. D. L. Fuente, J. Gertheiss, P. Geurts, A. Greenfield, M. Grzegorczyk, A. C. Haury, B. Holmes, T. Hothorn, D. Husmeier, V. A. Huynh-Thu, A. Irrthum, G. Karlebach, S. Lèbre, V. D. Leo, A. Madar, S. Mani, F. Mordelet, H. Ostrer, Z. Ouyang, R. Pandya, T. Petri, A. Pinna, C. S. Poultney, S. Rezny, H. J. Ruskin, Y. Saeys, R. Shamir, A. Sîrbu, M. Song, N. Soranzo, A. Statnikov, N. Vega, P. Vera-Licona, J. P. Vert, A. Visconti, H. Wang, L. Wehenkel, L. Windhager, Y. Zhang, and R. Zimmer, "Wisdom of crowds for robust gene network inference," *Nature Methods 2012 9:8*, vol. 9, pp. 796–804, 7 2012. 89

[208] T. Barrett, D. B. Troup, S. E. Wilhite, P. Ledoux, C. Evangelista, I. F. Kim, M. Tomashevsky, K. A. Marshall, K. H. Phillippy, P. M. Sherman, R. N. Muertter, M. Holko, O. Ayanbule, A. Yefanov, and A. Soboleva, "Ncbi geo: archive for functional genomics data sets–10 years on," *Nucleic acids research*, vol. 39, 1 2011. 89

[209] B. M. Bolstad, R. A. Irizarry, M. Åstrand, and T. P. Speed, "A comparison of normalization methods for high density oligonucleotide array data based on variance and bias," *Bioinformatics (Oxford, England)*, vol. 19, pp. 185–193, 2 2003. 89

[210] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker, "Cytoscape: a software environment for integrated models of biomolecular interaction networks," *Genome research*, vol. 13, no. 11, pp. 2498–2504, 2003. 92

[211] V. Marx, "How single-cell multi-omics builds relationships," *Nature Methods 2022 19:2*, vol. 19, pp. 142–146, 2 2022. 100

[212] T. Gill, S. R. Brooks, J. T. Rosenbaum, M. Asquith, and R. A. Colbert, "Novel inter-omic analysis reveals relationships between diverse gut microbiota and host immune dysregulation in hla-b27-induced experimental spondyloarthritis," *Arthritis rheumatology (Hoboken, N.J.)*, vol. 71, p. 1849, 11 2019. 100

[213] B. Comte, J. Baumbach, A. Benis, J. Basílio, N. Debeljak, Åsmund Flobak, C. Franken, N. Harel, F. He, M. Kuiper, J. A. M. Pérez, E. Pujos-Guillot, T. Režen, D. Rozman, J. A. Schmid, J. Scerri, P. Tieri, K. V. Steen, S. Vasudevan, S. Watterson, and H. H. Schmidt, "Network and systems medicine: Position paper of the european collaboration on science and technology action on open multiscale systems medicine," *https://home.liebertpub.com/nsm*, vol. 3, pp. 67–90, 7 2020. 100

[214] R. Afshari, C. J. Pillidge, E. Read, S. Rochfort, D. A. Dias, A. M. Osborn, and H. Gill, "New insights into cheddar cheese microbiota-metabolome relationships revealed by integrative analysis of multi-omics data," *Scientific Reports 2020 10:1*, vol. 10, pp. 1–13, 2 2020. 100

[215] M. R. Garvin, C. Alvarez, J. I. Miller, E. T. Prates, A. M. Walker, B. K. Amos, A. E. Mast, A. Justice, B. Aronow, and D. Jacobson, "A mechanistic model and therapeutic interventions for covid-19 involving a ras-mediated bradykinin storm," *eLife*, vol. 9, pp. 1–16, jul 2020. 103

[216] S. Perlman and J. Netland, "Coronaviruses post-SARS: Update on replication and pathogenesis," may 2009. 103

[217] S. Pollett, M. A. Conte, M. Sanborn, R. G. Jarman, G. M. Lidl, K. Modjarrad, and I. Maljkovic Berry, "A comparative recombination analysis of human coronaviruses and implications for the sars-cov-2 pandemic," *Scientific Reports*, vol. 11, p. 17365, Aug 2021. 103

[218] X. Tang, C. Wu, X. Li, Y. Song, X. Yao, X. Wu, Y. Duan, H. Zhang, Y. Wang, Z. Qian, J. Cui, and J. Lu, "On the origin and continuing evolution of SARS-CoV-2," *National Science Review*, vol. 7, pp. 1012–1023, jun 2020. 103

[219] Z. Wang and K. J. Liu, "A performance study of the impact of recombination on species tree analysis," *BMC Genomics*, vol. 17, pp. 11–14, 11 2016. 104

[220] V. A. Ngo and R. K. Jha, "Identifying key determinants and dynamics of sars-cov-2/ace2 tight interaction," *PLOS ONE*, vol. 16, p. e0257905, 9 2021. 104

[221] E. Simon-Loriere and E. C. Holmes, "Why do rna viruses recombine?," *Nature Reviews Microbiology*, vol. 9, no. 8, pp. 617–626, 2011. 106

[222] S. K. Garushyants, I. B. Rogozin, and E. V. Koonin, "Template switching and duplications in sars-cov-2 genomes give rise to insertion variants that merit monitoring," *Communications biology*, vol. 4, no. 1, pp. 1–9, 2021. 106

[223] S. Elbe and G. Buckland-Merrett, "Data, disease and diplomacy: Gisaid's innovative contribution to global health," *Global Challenges*, vol. 1, pp. 33–46, 1 2017. 106

[224] L. Qin, Y. Chen, Y. Pan, and L. Chen, "A novel approach to phylogenetic tree construction using stochastic optimization and clustering," *BMC Bioinformatics*, vol. 7, pp. 1–12, 12 2006. 109

[225] F. Wu, S. Zhao, B. Yu, Y. M. Chen, W. Wang, Z. G. Song, Y. Hu, Z. W. Tao, J. H. Tian, Y. Y. Pei, M. L. Yuan, Y. L. Zhang, F. H. Dai, Y. Liu, Q. M. Wang, J. J. Zheng, L. Xu, E. C. Holmes, and Y. Z. Zhang, "A new coronavirus associated with human respiratory disease in china," *Nature*, vol. 579, pp. 265–269, 3 2020. 118

[226] S. Srinivasan, H. Cui, Z. Gao, M. Liu, S. Lu, W. Mkandawire, O. Narykov, M. Sun, and D. Korkin, "Structural genomics of sars-cov-2 indicates evolutionary conserved functional regions of viral proteins," *Viruses 2020, Vol. 12, Page 360*, vol. 12, p. 360, 3 2020. 118

[227] Y. M. Bar-On, A. Flamholz, R. Phillips, and R. Milo, "Sars-cov-2 (covid-19) by the numbers," *eLife*, vol. 9, 3 2020. 125

Appendices

A Chapter 5 Supplementary Figures

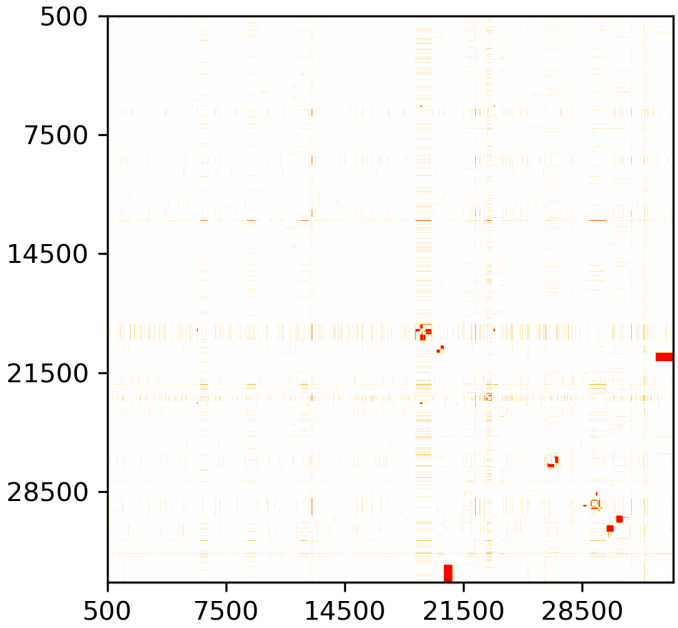

Figure 1: Incompatibility Matrix of SARS-CoV-2 Haplotypes shown as a heat map. Each axis is the parsimonious SNPs in the order they appear in the sequence with the SNPs labeled to roughly what they would be in the full Genome. The darker color means more incompatibility between the two sites. The lines of darker color highlight sites that have likely had mutations transferred to other parts of the population through recombination resulting in incompatibility with most of the genome.

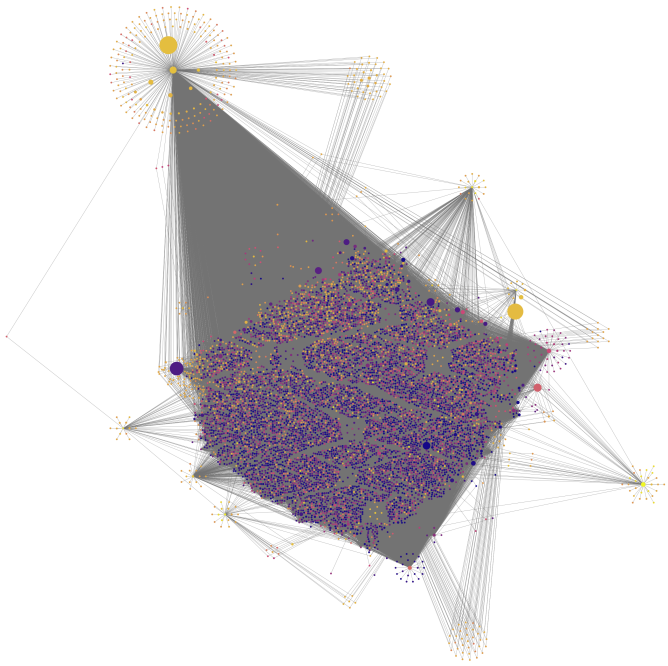

Figure 2: This network is constructed by taking unique SARS-CoV-2 haplotypes (the nodes) and connecting them with an edge whose value is the number of mutational differences between the haplotypes. The size of each node represents the number of samples collected with that haplotype and the color represents when the haplotype was first sampled, with lighter colors being earlier and darker colors being later in the pandemic. Each edge is then pruned if there is an intermediary haplotype whose edge values to the original two haplotypes sum to the same or less value than the distance of the original edge. This is equivalent to saying that haplotype of the intermediary node is also the intermediary evolution point between the two originally connected haplotypes. This network contains 8742 unique haplotypes representing up to 2941 individual samples and 390307 edges connecting them which represent mutational differences from 1 to 722 of 14793 SNPs.

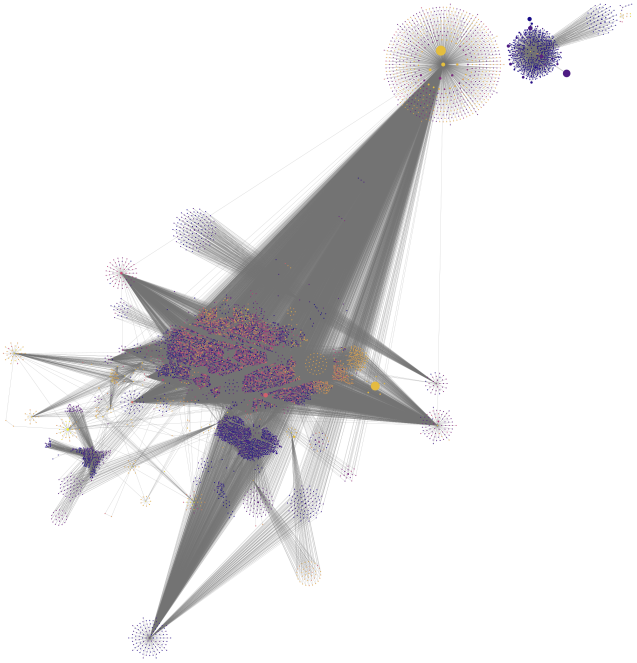

Figure 3: This network is constructed by taking unique SARS-CoV-2 haplotypes (the nodes) and connecting them if they are 15 points of mutation (the value of the edge) away from each other or less. The size of each node represents the number of samples within that haplotype and the color represents when the haplotype was first sampled, with lighter colors being earlier and darker, bluer colors being later in the pandemic. Each edge is then pruned if there is an intermediary haplotype whose edge values to the original two haplotypes sum to the same or less value than the distance of the original edge. This is equivalent to saying that haplotype of the intermediary node is also the intermediary evolution point between the two originally connected haplotypes.

Vita

Jonathon Romero was born in Atlanta, Georgia and moved to Winter Haven, Florida with his mother at age three. He attended Winter Haven High School from 2007 to 2011, graduating Salutatorian of his class. He went on to obtain a Bachelors of Science Degree in Physics and Math from the University of Alabama in Tuscaloosa, Alabama in 2015. Jonathon pursued his doctorate in Energy Science and Engineering through the Bredesen Center for Interdisciplinary Research and Graduate Education at the University of Tennessee Knoxville, originally focusing on the calculation of the electronic structure of super conductors before joining Dr. Daniel Jacobson's Computational Systems Biology group at Oak Ridge National Laboratory. There, Jonathon focused on developing and applying Explainable Artificial Intelligence models to a variety of biological data sets.