

An interdisciplinary approach:  
computational prediction of protein function  
with experimental validation

Hyunjin D. Choi

Dissertation submitted to the Faculty of the  
Virginia Polytechnic Institute and State University  
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy  
in  
Genetics, Bioinformatics, and Computational Biology

Brett M. Tyler, Chair  
T. M. Murali  
Christopher T. Franck  
Ronald M. Lewis

September 26, 2013  
Blacksburg, Virginia

Keywords: Protein function prediction, *Phytophthora sojae*, effectors, support vector machines, functional linkage network, amino acid physical properties  
Copyright 2013, Hyunjin D. Choi

An interdisciplinary approach:  
computational prediction of protein function  
with experimental validation

Hyunjin D. Choi

(ABSTRACT)

Pathogens colonize their hosts by releasing molecules that can enter host cells. A biotrophic oomycete plant pathogen, *Phytophthora sojae* harbors a superfamily of effector genes whose protein products enter the cells of the host, soybean. Many of the effectors contain an RXLR-dEER motif in their N-terminus. More than 400 members belonging to this family have been previously identified using a Hidden Markov Model. Amino acids flanking the RXLR motif have been utilized to identify effector proteins from the *P. sojae* secretome, despite the high level of sequence divergence among the members of this protein family.

I present here machine learning methods to identify protein candidates that belong to a particular class, such as the effector superfamily. Converting the flanking amino acid sequences of RXLR motifs (or other candidate motifs) into numeric values that reflect their physical properties enabled the protein sequences to be analyzed through these methods. The methods evaluated include Support Vector Machines and a related spherical classification method that I have developed. I also approached the effector prediction problem by building functional linkage networks and have produced lists of predicted *P. sojae* effector proteins. I tested the best candidate through gene gun bombardment assays using the beta-glucuronidase reporter system, which revealed that there is a high likelihood that the candidate can enter the soybean cells.

This work was supported in part by grants from the the National Institute of Food and Agriculture, USDA NIFA Awards No. 2009-65109-05990 and No. 2011-670009-30133 and by the Virginia Bioinformatics Institute and by the Virginia Tech Graduate Program in Genetics, Bioinformatics and Computational Biology.

# Dedication

I dedicate my dissertation to:  
Jesus Christ, through whom all things are made (Gospel of John, 1:3)

# Acknowledgments

I want to thank my brilliant adviser, Dr. Brett Tyler, who continues to inspire me with his tireless devotion to research and his care for his students. I am very thankful for his guidance and support through the years. I also want to thank my committee members, Dr. T.M. Murali, Dr. Chris Franck and Dr. Ron Lewis, for their helpful suggestions and understanding. I am grateful for my husband, Vincenzo Antignani, whose encouragement and help pulled me through the process. I thank my parents, who gave me opportunities they never had, and my brother, Joshua Choi, who always believed in me.

# Attribution

Here I acknowledge the contributions from the colleagues who aided in the writing and the research behind each chapter presented in the dissertation.

**Chapters 2, 3, 4 and 5:** Brett M. Tyler, Ph.D. is a professor and director in the Center for Genome Research and Biocomputing at Oregon State University. Dr. Tyler was the principal investigator for the grants supporting this research. He directed the research approach and provided editorial comments.

**Chapters 2 and 3:** T. M. Murali, Ph.D. is an associate professor in the Department of Computer Science at Virginia Tech. Dr. Murali guided the research and contributed editorial comments.

**Chapter 4:** Christopher T. Franck, Ph.D. is an assistant research professor in the Department of Statistics at Virginia Tech. Dr. Franck provided guidance on the statistical methods used and contributed editorial comments.

**Chapter 5:** Felipe Arredondo is a research associate in Dr. Brett Tyler's laboratory at Oregon State University. Mr. Arredondo performed the gene gun bombardment assays and provided the photographs of bombarded soybean leaves.

Ronald M. Lewis, Ph.D. is a professor in the Department of Animal and Poultry Sciences at Virginia Tech. Dr. Lewis provided editorial comments during the writing of the dissertation.

# Contents

<b>1</b>	<b>Literature Review</b>	<b>1</b>
1.0.1	Introduction to oomycete and fungal pathogens . . . . .	1
1.0.2	Effectors enhance pathogen fitness against the plant immune system .	2
1.0.3	RXLR and RXLR-like motifs in pathogen effectors . . . . .	2
1.0.4	Experimental validation of effector function . . . . .	3
1.0.5	Computational prediction of protein function: Homology method and motif discovery . . . . .	4
1.0.6	Computational prediction of protein function: Classification and clustering . . . . .	6
1.0.7	Computational prediction of protein function: Network-based methods of functional prediction . . . . .	8
1.0.8	Conclusion . . . . .	10
<b>2</b>	<b>Using machine learning classification and clustering analysis to predict <i>P. sojae</i> effector candidates</b>	<b>11</b>
2.1	Abstract . . . . .	11
2.2	Background . . . . .	12
2.3	Implementation . . . . .	13
2.3.1	A. Conversion of amino acid sequences to Kidera factors . . . . .	13
2.3.2	B. Building the training set . . . . .	14
2.4	Results . . . . .	15
2.4.1	A. Hierarchical clustering analysis . . . . .	15
2.4.2	B. Spherical classification . . . . .	16

2.4.3	C. Support Vector Machines (SVM) . . . . .	20
2.4.4	D. Logistic Regression with Stepwise Variable Selection . . . . .	23
2.5	Discussion . . . . .	25
<b>3</b>	<b>Functional linkage network approach to predicting <i>Phytophthora sojae</i> effector candidates</b>	<b>27</b>
3.1	Abstract . . . . .	27
3.2	Background . . . . .	28
3.3	Implementation . . . . .	28
3.4	Results . . . . .	29
3.5	Discussion . . . . .	36
<b>4</b>	<b>Analysis of fungal effector cell-entry motifs using statistical methods</b>	<b>40</b>
4.1	Abstract . . . . .	40
4.2	Background . . . . .	41
4.3	Implementation . . . . .	43
4.3.1	Datasets . . . . .	43
4.3.2	MEME motif analysis . . . . .	44
4.3.3	Simple correct call method . . . . .	44
4.3.4	Logistic regression analysis with a covariate . . . . .	44
4.3.5	Markov Clustering . . . . .	45
4.4	Results . . . . .	45
4.4.1	MEME Analysis . . . . .	45
4.4.2	Correct Call Analysis . . . . .	46
4.4.3	Markov Clustering . . . . .	49
4.5	Discussion . . . . .	50
<b>5</b>	<b>Identification and functional testing of novel effector candidates from <i>Phytophthora sojae</i></b>	<b>57</b>
5.1	Abstract . . . . .	57

5.2	Background . . . . .	57
5.3	Methods . . . . .	58
5.3.1	Candidate selection criteria . . . . .	58
5.3.2	Cell entry experiment design using gene gun . . . . .	60
5.3.3	Plasmid construction . . . . .	61
5.3.4	Gene Gun Bombardment Assays . . . . .	62
5.4	Results . . . . .	63
5.5	Discussion . . . . .	67
<b>6</b>	<b>Conclusion</b>	<b>71</b>
	<b>Bibliography</b>	<b>74</b>



# List of Figures

2.1	Heatmap generated from the training set using Kidera factor 1. . . . .	15
2.2	Precision-recall curve resulting from using hypothetical spheres around their geometric center to predict functional effectors. . . . .	18
2.3	Cumulative number of false positives . . . . .	18
2.4	ROC curve for Spherical Classification . . . . .	19
2.5	10-fold Cross Validation for Spherical Classification . . . . .	19
2.6	ROC curve for 10-fold Cross Validation of Spherical Classification. . . . .	20
2.7	Number of true positives and false positives as Sphere Size increases . . . . .	21
3.1	Ranking of Sinksource+ confidence scores of 281 RXLR-like motifs from HMM predicted effectors (red) and 223 permuted negatives (grey), in the validation data network. . . . .	30
3.2	Precision-Recall curve for the validation network (same as Figure 3.1), calculated on the 281 HMM predicted positives that were treated as unknowns and the 223 negatives produced by permutation. . . . .	31
3.3	ROC curve for the <i>P. sojiae</i> validation data network (same as Figure 3.1). . . . .	31
3.4	Null distribution of confidence scores produced by the Sinksource+ algorithm on a random network created from 1000 permuted sequences and 10 randomly assigned seeds. . . . .	32
3.5	Distribution of confidence scores from the <i>P. sojiae</i> test network. . . . .	33
3.6	Distribution of HMM positive RXLR-like motifs in the test data set ranked by confidence score. . . . .	33
3.7	Distribution of permuted (true negative) RXLR-like motifs in the test data set ranked by confidence score. . . . .	34

3.8	Precision-recall curve calculated on 281 HMM predicted positives and 1000 permuted true negatives in the <i>P. sojae</i> test data network . . . . .	34
3.9	ROC curve for <i>P. sojae</i> test data network. . . . .	35
4.1	Motifs found by MEME from 28 cell-entering <i>M. oryzae</i> sequences. . . . .	46
4.2	Motifs found by MEME from 24 <i>P. sojae</i> RXLR effectors. . . . .	47
4.3	MEME discrimination analysis of 28 cell-entering <i>M. oryzae</i> proteins compared to 29 experimentally validated non-entering <i>M. oryzae</i> proteins. . . . .	48
4.4	Correct call analysis of <i>P. sojae</i> RXLR effectors. . . . .	49
4.5	Correct call analysis of permuted positive and negative <i>P. sojae</i> effector sequences. . . . .	50
4.6	Correct call analysis of 28 translocating <i>M. oryzae</i> sequences, compared with 29 non-translocating sequences as negatives. . . . .	52
4.7	Correct call analysis of 28 <i>M. oryzae</i> translocating proteins as positives and permutations of those sequences as negatives. . . . .	53
4.8	Correct call analysis of permuted positive and negative <i>M. oryzae</i> sequences. . . . .	53
4.9	Logistic regression analysis of <i>P. sojae</i> effector sequences as positives and permuted sequences as negatives using an amino acid composition covariate. . . . .	54
4.10	Logistic regression analysis of <i>M. oryzae</i> translocating sequences as positives and non-translocating sequences as negatives using an amino acid composition covariate. . . . .	54
4.11	Amino acid composition of datasets used in the analysis. . . . .	55
5.1	Double barrel gene gun leaf bombardment scheme. . . . .	62
5.2	Schematic diagram of cloning strategy. . . . .	63
5.3	Williams and L77-1863 soybean plants ready for bombardment. . . . .	64
5.4	Double barrel gene gun. . . . .	64
5.5	Representatives leaves from L77-1863 and Williams are shown after bombarding and staining the soybean leaves. . . . .	68
5.6	Ratio of GUS spots for the gene gun bombardment results. . . . .	69

# List of Tables

2.1	Summary of Kidera physical property factors . . . . .	13
2.2	Summary of 10 Kidera factors . . . . .	14
2.3	Support Vector Machine results based on a 10-fold cross-validation of the training set. . . . .	22
2.4	Support Vector Machine predictions on six known fungal effectors. . . . .	22
2.5	Permuted sequences of AvrLm6 are wrongly predicted to be effectors . . . . .	23
2.6	Permutations test on the 6 non-oomycete effectors . . . . .	23
2.7	Final list of 27 features selected for the logistic regression model. . . . .	24
2.8	SVM results from <i>P. sojae</i> with only selected features from logistic regression variable selection. . . . .	24
3.1	Top 25 scoring candidate <i>P. sojae</i> effectors. . . . .	35
3.2	Top 25 scoring candidate effectors from a list that included RXLR motifs from 4 <i>Phytophthora infestans</i> effectors. . . . .	36
3.3	Top scoring RXLR-like motifs (top 28 shown here) predicted using squared correlations as edge weights. . . . .	37
3.4	<i>P. sojae</i> effector prediction list. . . . .	37
4.1	Top 3 clusters from Markov clustering, using 24 <i>P. sojae</i> effector sequences. . . . .	51
4.2	Top 3 clusters from Markov clustering, using full length <i>M. oryzae</i> sequences (including signal peptide) which are known to translocate across the cell membrane. . . . .	51
4.3	Top three clusters from <i>M. oryzae</i> Markov Clustering centered on large hydrophobic residues. . . . .	51

5.1	Examples of high scoring nodes with GAIN confidence scores. . . . .	60
5.2	Expected results from double barrel gene gun bombardment. . . . .	62
5.3	Number of blue spots counted for all four sets of bombardments. . . . .	66

# Chapter 1

## Literature Review

### 1.0.1 Introduction to oomycete and fungal pathogens

Many devastating plant diseases are caused by microbial pathogens, such as oomycetes and fungi. While communities of microbes help to foster and to maintain natural ecosystems, pathogenic microbes can destroy agriculturally important and other economically significant plants. Among the oomycetes, more than 120 species of *Phytophthora* have been identified. An oomycete plant pathogen, *Phytophthora infestans*, was the causative agent of the Irish Potato Famine in the mid-1800s, resulting in tremendous population relocations and disturbing the socio-economic dynamics of Europe at the time [28]. A relative, *Phytophthora sojae*, lives in soil and attacks soybean plants, causing root and stem rots. *Phytophthora sojae* is a model oomycete pathogen and its genome sequence was published in 2006, enabling scientists to discover many clues as to how oomycete pathogens could be so successful at colonizing their hosts [61]. Another *Phytophthora* species, *P. ramorum*, causes the Sudden Oak Death disease and damages plants in natural forest ecosystems and in plant nurseries. This pathogen also damages other plants that have ornamental value, such as Christmas trees and rhododendrons. There are also many fungal pathogens that destroy agriculturally important plants, such as *Magnaporthe oryzae*, the causative agent of rice blast disease. Infected rice seedlings as well as adult rice plants whose stems and roots are affected by the fungus almost completely fail to produce grain [14]. Understanding the molecular mechanisms of plant pathogen attack on hosts is an important aim. Developing targeted control measures for plant pathogens will benefit society on a global scale through economic profit and gains in food security. It will also promote a healthier food supply by reducing chemical usage for protecting crops, and will help to benefit food security by increasing crop production.

### 1.0.2 Effectors enhance pathogen fitness against the plant immune system

Many pathogenic oomycetes and fungi produce effector proteins that aid the pathogens in colonizing their plant hosts [60]. Effector proteins are not only limited to oomycetes and fungi, but are also produced by bacteria, nematodes and viruses [60]. Many effectors are small, hydrophilic proteins that can enter the host cells. Since plant cells are immobile, they must rely on the immune capability of each cell and signals arising from the infection sites [32] to avoid pathogenesis. When plants detect foreign microbial molecules such as microbial/pathogen-associated molecular patterns (MAMPs/PAMPs) or effectors, they can respond by launching two cooperating branches of the plant immune system [32]. After detecting MAMPs or PAMPs, plants can respond by triggering PAMP-triggered immunity (PTI) [32]. During this step, pathogens can release effectors into the plant host. If one of these effectors from the pathogen is recognized by a resistance protein (R protein) from the plant, then this interaction can trigger a successful hypersensitive defense response of the plant [32]. Because some effectors detected by the plant prevent a successful infection, effector genes have historically been termed avirulence genes [60]. Activation of R proteins may be through a direct protein-protein interaction with an effector (for example, AvrPto from *Pseudomonas syringae* and Pto from tomato), but the interaction may also be indirect [12]. This interaction results in effector-triggered immunity (ETI) [32]. Pathogens have a better survival strategy when they have eliminated effectors recognized by the plant and have diversified their effector repertoire to maintain virulence [32]. Plants in turn have better survival when they evolve R proteins that recognize effectors that are indispensable to the pathogen.

### 1.0.3 RXLR and RXLR-like motifs in pathogen effectors

One of the largest families of effectors in oomycetes is called the RXLR-effector family. This family is a group of rapidly diversifying proteins involved in pathogen virulence and entry into the hosts. These proteins share little sequence similarity except the signature sequence RXLR (Arginine, Unspecified, Leucine, Arginine) in their N-terminal regions [30]. A second more loosely conserved motif, located a short distance from RXLR, is the dEER motif (Aspartic acid, Glutamic acid, Glutamic acid, Arginine — small d indicates that the Aspartic acid in the first position is not strictly conserved). Nearly 400 candidate RXLR effectors have been identified in *Phytophthora sojae* and *Phytophthora ramorum* through bioinformatic studies using recursive BLAST and Hidden Markov Modeling [30]. However, the presence of RXLR-motif does not automatically guarantee the proteins membership in the HMM classified effector superfamily. There are many RXLR-containing, secreted proteins that are classified as non-effectors, based on the composition of the surrounding amino acids. There are other significant motifs found near the C-terminus of many RXLR effectors, such as the lysine (K) motif, the tryptophan (W) motif and the tyrosine (Y) motif

[16]. Thirty six percent of all HMM predicted RXLR effectors have all three motifs, and 22% contain only the W motif [30]. Moreover, mutating the W motif and other conserved residues near the W motif from Avr1b abolished its ability to interact with its cognitive resistance gene from the soybean plants [16].

Since many fungal effectors lack an obvious RXLR-dEER motif, a systematic substitution study was conducted using Avr1b of *Phytophthora sojae* to determine which residues would still allow entry of Avr1b into soybean leaf cells [34]. The first residue, arginine, has a positively charged side chain, and substituting arginine with other amino acids with the same property (lysine and histidine) allowed entry, while glutamine could not [34]. The third residue, leucine, has a hydrophobic side chain, and could functionally be substituted with any other amino acids with a large hydrophobic side chain (isoleucine, methionine, phenylalanine, tyrosine or tryptophan) and the last position of the motif tested positive for all substitutions [34]. Based on these substitution mutation results, RXLR-like motifs of a few fungal effectors were identified and experimentally tested. For example, AvrL567 of *Melampsora lini* (flax rust) contains only one RXLR-like motif, composed of arginine, phenylalanine, tryptophan and arginine. When all four of these residues were changed to alanine, the mutated protein was reported to lose its ability to enter plant host cells [34].

#### 1.0.4 Experimental validation of effector function

Many pathogen effectors are capable of entering host cells. The exact mechanism by which pathogen effectors gain entry into the host cells is still an area of active research and intense scrutiny. Bacterial effectors are known to enter host cells using one of several syringe-like secretion injectisomes (type III, type IV or type VI), which are complex nanomachines that allow delivery of effectors across cellular membranes [10]. Among many oomycete effectors, the RXLR-dEER motifs in the N-terminal regions are required for entry, and experimental work has shown that these motifs bind to phosphatidylinositol 3-phosphate (PI3P) and enter the host cells via receptor-mediated endocytosis [31, 34], but there seems to be some debate on this particular point in the literature [20, 65, 66, 70]. For example, an RXLR-like effector HpStp1 from *Saprolegnia parasitica*, an oomycete fish pathogen, was reported to translocate into the host cells through interactions with tyrosine-O-sulfatemedied cell-surface molecules, but not through interactions with phospholipids [66].

To examine how lipids are involved in effector entry of host cells, lipid binding assays have been widely utilized. One assay by which lipid binding by effector candidates can be tested using lipid-filter binding assay, in which lipids dissolved in a solvent are blotted onto filters and proteins of interest are allowed to bind [35]. An alternative to the lipid-filter binding assay is the liposome binding assay, which tests if large vesicles of liposomes incorporating the lipids of interest are able to bind the proteins of interest [35]. An essential virulence effector Avr3a of *Phytophthora infestans* contains the RXLR-dEER motif and binds to phosphatidylinositols (PIPs) in lipid-filter binding assays [70]. When Yaeno et. al. [70] mutated

Avr3as C-terminal PIP binding domain, recognition by R3a of potato was not eliminated, but its ability to suppress programmed cell death suffered significantly. PI3Ps have been found on the outer membrane surface of soybean root cells and also human lung epithelial cells [35]. When exogenous inositol phosphates, which have the potential to bind to the PIP-binding sites of effectors, were incubated with soybean root cells, three PI3P binding effectors (fungal and oomycete) fused with green fluorescent protein were blocked from entering the cells [35].

Many effectors are known to interfere with host cell physiology during infection, for example, by suppressing programmed cell death, which is a part of protective immune responses of the plant. On the other hand, some effectors induce programmed cell death directly to enhance pathogen virulence. In order to test whether an effector candidate induces or suppresses programmed cell death, a reporter gene such as  $\beta$ -glucuronidase (GUS) can be used in particle bombardment assays [16,17,19,64]. Alternatively, Agrobacterium-mediated transient expression of effectors and cell death-inducing proteins can be used [64]. Wang et. al. [64] have examined 169 RXLR effector candidates for their effector function. Using the bombardment assays, they observed that many of the tested effectors could suppress programmed cell death while a few actually triggered cell death by themselves directly [64].

### 1.0.5 Computational prediction of protein function: Homology method and motif discovery

There is tremendous amount of DNA sequence data becoming available. As of April 2013, Genomes Online Database (GOLD) reports that there are 4327 completed sequencing projects whose data have been deposited in a public repository, composed of 187 archaeal, 3957 bacterial and 183 eukaryotic genomes [47]. It is very important to computationally find and assign the functional meaning behind the predicted genes from these large scale biological data sets. An important area of research is the task of computationally searching and identifying motifs or patterns in biological sequences in order to shed light on their potential functions.

In one approach to protein function prediction called the homology method, a protein of unknown function may be related to a homologous protein that has a similar amino acid sequence yet known function. Homologous proteins are assumed to share a similar function and in most cases to have evolved from a common ancestor protein [19]. In new genome sequences, 40% to 70% of predicted genes can be assigned a function through this method [19]. Another approach is to utilize gene expression data and clustering strategies to find genes that display coordinated expression patterns under different conditions or across time points. By analyzing clusters of genes whose expression profiles are similar, one may be able to assign function to genes of unknown functions in a given cluster. This method is attractive because all transcripts can be surveyed at once using microarrays or RNA sequencing, and many proteins participating in the same pathway show similar expression patterns [69].



This method can also be applied to proteins whose sequence has no obvious similarity.

Numerous classes of proteins that share similar functions have been shown to contain recognizable motifs such as the RXLR effectors in the oomycetes. One of the innovative ways that researchers have employed to identify motifs is to represent the biological sequences as numerical values that can then allow mathematical approaches and tools to be applied in searching for motifs embedded in these biological sequences. Kidera et. al. [38] provided a method to represent each amino acid with 10 numerical values dealing with their physical characteristics, that is useful for prediction methods that rely on numerical properties of proteins [38]. They used principal factor analysis to summarize 188 different physical properties of amino acids as 10, orthogonal numerical factors that accounted for at least 86% of the variability present in the physical properties of the 20 different amino acids [38]. In a more recent study, S. Rackovsky [49] utilized the factors from Kidera et. al. [38] and performed Fourier transformation on the numerical values representing strings of amino acids in order to detect underlying global protein structural patterns.

One of the most effective and popular procedures for identifying motifs is based on Hidden Markov Model (HMM) results. For example, HMMs were used to predict RXLR effectors in *P. sojae* and *P. ramorum*. Hidden Markov Models are statistical models in which the current state of a system depends on the previous state that is unobservable or hidden. In a Markov chain, there exists a sequence of distinct states and the system can go through transitions between the states. If between each step, the following state is determined randomly and the the probability distribution from the current state determines the next state, the process can be said to follow a "Markovian behavior" [63]. In order to consider a protein sequence in the context of HMMs, each position of the protein can represent a state, and the amino acid in that position would be an observation from that state. It is assumed that the underlying hidden states follow a Markovian behavior, while the observed sequence of states is not a Markov process [63]. This approach is particularly useful when dealing with biological sequences and has been empirically shown to be successful; while the sequence of interest itself that we observe does not represent the output of a Markov process, we may assume that the hidden sequence of transitions buried underneath is a Markov process. Based on this estimated unobservable Markov process, we can then reverse estimate observable sequences and make predictions on new sequences [63]. The underlying hidden statistical model behind a set of observable protein sequences (the training set) can then be used as a classification tool to predict the class of an unknown protein sequence. For a given unknown protein sequence, we can calculate a probability score to estimate how similar the unknown sequence is to the model derived from the known sequences.

### 1.0.6 Computational prediction of protein function: Classification and clustering

Classification, in machine learning, is an algorithmic process that builds a computational model from a set of classified examples and uses the model to assign classes to newly presented examples [67]. A model here is a statistical or mathematical formula, which is derived from the examples to describe the trend in the dataset. The set of known examples, such as weather conditions in the past or handwriting examples in a handwriting recognition problem, is called the training set, while new samples presented to the model are called the test set. Classification learning can be regarded as a type of supervised learning, because the classification algorithm assumes that a new sample must belong to a class that it has been trained on. On the other hand, unsupervised learning does not assume that the outcomes must belong to pre-defined classes, but operates on the basis of similarity between the members of the set, and aims to learn the intrinsic structure hidden in the data and to group the similar members together into clusters [29]. In bioinformatics, researchers have faced classification problems dealing with large sets of microarray gene expression data since the early 2000s [58]. For example, gene expression profiles of cancer patients allowed reasonable recognition and classification of cancer types [26]. Golub et. al. [26] applied a machine learning classification approach to distinguish two distinct leukemia subtypes—acute lymphoblastic leukemia or acute myeloid leukemia—based on the patients gene expression profiles. They selected the top 50 genes most correlated with the distinction of the two cancer types and performed cross-validation to test prediction accuracy [26]. Their prediction method made 29 correct calls out of 34 previous unknown samples [26], and their study illustrates an example of solving a classification problem using machine learning. Golub et. al. further explored developing methods to discover new classes of cancer automatically using self-organizing clusters of gene expression profiles, and their hypothesis proved correct on their leukemia training dataset [26]. Their computational approach using clustering illustrates how unsupervised learning can be applied practically on gene expression data. Many of the methods described and applied in our study have been utilized in the context of gene expression data. In our study, we use classification tools in order to train a model which would learn from a training set of positives (validated effector protein sequences) and negatives (non-effector protein sequences), and to be able to make predictions about sequences for which we do not have previous knowledge. Here are brief descriptions of the methods we have considered and used in our study.

#### *Hierarchical clustering*

Hierarchical clustering iteratively makes decisions based on some similarity measure about which members of the set belong together to form groups at different hierarchy levels from either a top-down approach (where all data points start out in one cluster and split into separate groups until each element is in its own cluster) or a bottom-up approach (where each

data point is considered a single cluster and the data points are merged until all clusters have merged into one). The bottom-up approach is termed hierarchical agglomerative clustering, and distance calculation between the clusters are generally performed in three different ways. In single-linkage clustering, the distance between two clusters is considered to be the shortest distance between any point in one cluster and any point in the other, while in complete-linkage clustering, the distance is measured by the longest distance. In average-linkage clustering, the distance is the average of all distances between any point in one cluster and any point in the other cluster. Hierarchical clustering results are usually visualized as dendrograms, which show the inter-group distance relationships in a tree diagram. A horizontal line in a dendrogram indicates a merge between two clusters, while a vertical line traces a cluster. Hierarchical clustering is a type of an unsupervised learning tool, because there is no preconceived notion about the resulting clusters unlike in classification. Hierarchical clustering is one of the best known clustering methods, and has been used in gene expression data analysis to find clusters of related genes since the start of the large scale genome studies [57].

### ***Support vector machines***

Support vector machines were introduced by Vladimir Vapnik and his colleagues in 1992. The support vector machine is a type of supervised learning tool, which aims to classify unknown data points into known categories. The support vector machine algorithm first maps the original data into a high dimensional feature space [11]. With the appropriate transformation into this new feature space, a linear hyperplane which separates the data into two classes can be found by solving a quadratic optimization problem [11]. This separating linear hyperplane has the property that the distance to the closest member of either class is as large as possible. This distance is called the margin and such a hyperplane is called the maximal marginal hyperplane [11]. Once this hyperplane is established, the algorithm can be easily applied to assign a label to any new unknown data point depending on which side of the separating hyperplan this point lies. Support vector machines are known to be highly accurate and to suffer less from over-fitting than other classification methods, because the algorithm finds the separating hyperplane with the largest margin between the classes [29]. Over-fitting describes a phenomenon where random noise in the data is so heavily considered in the model that the predictive power suffers. Support vector machines have been applied to many problems from multiple disciplines, such as speech recognition, handwriting recognition, classification of gene expression data and even financial time series forecasting. In this study of effector prediction, amino acid sequences were converted into Kidera factor scores that could be analyzed using computational methods, and the conversion process is described in the Implementation section in detail in Chapters 2. Using these Kidera scores, support vector machines were used to define maximal marginal hyperplanes between *P. sojae* effectors and non-effectors. Fungal sequences were then tested and evaluated against the hyperplane which was built based on *P. sojae* sequences.

## Logistic regression

Logistic regression is a form of statistical regression analysis, which is a type of linear regression model applicable when the dependent variable is dichotomous. When the dependent variable is binary, some of the assumptions behind a linear regression model are violated, such as the assumption of normal distribution of the error terms. We therefore must fit a model that better describes the distribution of the data. Logistic regression takes the logarithm of the odds which is the ratio of probability of an event to probability of non-event. By transforming the dependent variable this way removes the upper and the lower bounds of a binary variable term, allowing the logarithm of the odds to be set equal to a linear function of dependent variables without breaking the integrity of assumptions behind a linear regression. Because the dependent variable in effector prediction (effector or not) is binary, I evaluated logistic regression as an approach to generate a statistical model to separate effector sequences from non-effector sequences. I also examined whether logistic regression could identify which combinations of Kidera factors and amino acid positions had greater influence and significance in determining effector status.

### 1.0.7 Computational prediction of protein function: Network-based methods of functional prediction

Construction and analysis of interaction networks based on functional linkages has been a prominent method employed in protein function prediction [54]. In a functional linkage network, a node represents a protein, while an edge represents a functional linkage, which may be evidence for direct binding or a similar gene expression pattern. An intuitive way to assign function of an unknown protein in a network would be to transfer the knowledge from the neighboring proteins. This process is known as guilt by association [36]. There have been other approaches to meet the challenges of computationally predicting protein functions and interactions using genome-scale networks, using guilt by association methods [36, 42, 44, 54]. These methods largely depend on the principle that proteins can be linked because they have correlated gene expression profiles, or are physically interacting with each other to form a multi-protein complex, or are a part of the same cellular pathway [42]. Based on these relationships, one can infer the function of unknown proteins. Functional linkage networks provide a platform to computationally predict functions for many previously uncharacterized proteins that have no direct sequence similarity to known proteins. One way to formulate guilt by association is a local threshold rule, which states that if a protein of unknown function has over a certain fraction of neighbors all annotated with function  $p$ , then we would assign function  $p$  to the protein [36]. However, many subtle issues complicate the implementation of the local threshold rule, such as how to define an appropriate threshold, how to consider the different sizes of neighborhoods, when to trust the prediction call given the large amount of noise in the network, and the inherent unreliability of propagating small amounts of experimentally validated information across a wide network of uncharacterized

nodes [36]. Thus, many frameworks to propagate evidence through the entire network systematically have been proposed. One example has been provided by Karaoz et. al. by mapping a protein-protein interaction network into a Hopfield network [36]. Conceptually, their method involves repeatedly applying the local threshold rule, until the network reaches a certain equilibrium state which is most consistent with the integrated information sources of protein functions [36].

Another framework for evidence propagation was introduced by Murali et. al. [44] called the Sinksource algorithm, which is a graph-theoretic approach to guilt by association to predict functions of unannotated proteins in a protein-protein interaction network. They investigated six other algorithms and their performance on the task of predicting human immunodeficiency virus (HIV) dependency factors, and found that the Sinksource algorithm and its variant called Sinksource+ produced slightly better and more consistent results [44]. In the Sinksource algorithm, a network can be interpreted as an electrical network. Each node belongs to one of the three categories — positives, negatives and unknowns. Each positive node is a voltage source, and each negative node is an electrical drain to the ground, and currents can flow through the edges between the nodes. When the current flowing into every node in the network is equal to the current flowing out of the node, the system has reached an equilibrium. The resulting voltage in each of the unknown nodes at equilibrium is the amount of confidence the algorithm assigns to that node, i.e., the voltage, which is a measure of how likely it is for a given unknown node to be similar to the positives. However, in some networks, it is difficult to identify nodes that would serve as negative examples. Experimental procedures tend to focus on identification of genes or proteins that carry out a function of interest, and data for nodes without the function is almost always scarce. In order to address the difficulty in identifying negative nodes in networks, an appropriate adaptation has been applied to the Sinksource algorithm. In a modified algorithm called Sinksource+, an artificial node, which serves as a universal negative node, is attached to each and every node in the network except the positives. Adding the artificial negative node to the network eliminates the need for negatives in the algorithm. In Sinksource+, the currents are constantly propagated from the positives, while a portion of the flow is lost to each artificial negative node, and the system eventually reaches an equilibrium. The resulting voltage in each of the unknown nodes is documented and reported as a confidence score, which Sinksource+ assigns as a measure of likelihood that a given unknown shares the same function as the positives.

However, after over a decade of implementing this line of guilt by association approach, large portions of many sequenced genomes still remain to be annotated and the field faces many challenges. In a recent series of journal articles, Gillis and Pavlidis caution that biologically significant functional information is usually concentrated in a very small number of critical interactions, and that such functional properties, such as genetic or physical interactions, cannot be reliably extended to the rest of the network [23–25]. They discuss cross-validation, a process which leaves out a portion of the known part of the network and tests the ability of the algorithm to retrieve the information. They point out that cross-validation assumes

that the guilt by association network possesses certain properties (such as genetic or physical interactions between the nodes) that can be applied to the overall network and overestimates the performance of the algorithm based on this assumption, which is flawed [25]. In some of their case studies, they showed that simply ranking the nodes in the network based on their node degree predicts the protein function better than some other sophisticated algorithms or methods [23]. It seems that studies that involve functional linkage networks should carefully consider whether their method of evaluating the algorithm performance really is based on a property of the network which can be extended to the overall network, not limited only to a small part of it.

### 1.0.8 Conclusion

Computational and statistical methods provide valuable tools for answering important questions in biology, especially regarding host-pathogen interactions. With sequencing technology rapidly evolving and producing large scale data, there are many opportunities to study how different organisms interact on a genomic scale. This thesis is focused on a model organism, *Phytophthora sojae*, interacting with its host, soybean plants and reviewed how the plant immune system responds to the attacks from the oomycete pathogen. In particular the focus is on computational prediction of an important class of proteins, effectors, that plays a key role in pathogen fitness. Many of these effectors contain the RXLR-dEER motif in their N-terminus and other functionally important motifs in the C-terminus. I have introduced a method to convert protein sequences into numeric strings, suitable for computational analysis using the physical properties of amino acids and introduced how to apply existing machine learning tools to build classifiers and to predict protein functions. I have introduced new logistic regression based methods for motif prediction. All the methods described here can easily be extended to studying other genomes and other pathogen-host systems.

## Chapter 2

# Using machine learning classification and clustering analysis to predict *P. sojae* effector candidates

Authors: Hyunjin D. Choi, T.M. Murali and Brett Tyler

### 2.1 Abstract

In recent years, many new methods to predict protein function on a large scale have been introduced in an effort to meet the challenge of annotating newly sequenced genomes [19, 30, 42, 44, 51, 54, 69]. This chapter presents a general workflow that can easily be applied to predicting protein function, not only in *Phytophthora sojae*, but in any other species with a sequenced genome. By converting protein sequences into numerical values using amino acid physical properties, as established by Kidera et. al. [38], classification and clustering tools can be used to make predictions and to estimate their level of accuracy through cross validation. Predictions using Support Vector Machines and Logistic Regression Classification, as well as a novel spherical classifier, performed well on a *Phytophthora sojae* training data set. To explore the application of these approaches to the prediction of effector proteins in fungi and other species, a limited number of known effectors from other species were included with the oomycete training data. However, the predictive power for the non-oomycete effectors was not strong, perhaps due to the large evolutionary differences between these organisms and the very small size of the test set.

## 2.2 Background

As more genomes are sequenced, annotation of the genes and their functions remains a bioinformatic challenge. Automatic or large scale annotation methods are in high demand, but still difficult to develop to meet the specific needs of each particular project. Protein function prediction has recently emerged as its own discipline as a response to this challenge [41, 42, 44, 46]. A major focus for this thesis is the model oomycete called *Phytophthora sojae*, whose draft genome was released in 2006 [61]. The goal was to develop a workflow to identify the members of a particular protein family called effectors, using machine learning approaches. It was desirable for the method to be easily applicable to any other genome. In order to use computational tools for protein function prediction, amino acid sequences were converted into numeric values. Each of the 20 essential amino acids has its own unique physical properties, such as electric charge, conformation and hydrophobicity. Kidera et. al. [38] recognized the usefulness of summarizing and condensing the most important physical properties of amino acids for understanding how protein structures are determined [38]. They considered 188 different physical properties of amino acids as the basis for their analysis. They aimed to capture as much information as possible and assign sets of properties to each of the 20 amino acids, while eliminating properties that were redundant. After clustering their original set of 188 physical properties and applying a factor analysis, they produced 10 numerical factors that accounted for at least 86% of the variability present in all of the physical properties they considered important for protein structure determination (Table 2.1). In a more recent study, S. Rackovsky [49] utilized the factors from Kidera et. al. [38] and performed Fourier transformation on the numerical values representing strings of amino acids in order to detect underlying global protein structural patterns.

Of the machine learning methods, Support Vector Machines (SVMs) was selected for our study because of its potential for finely separating positives (i.e. effectors) from negatives in a multi-dimensional setting. It was known that the BLAST algorithm could not identify all members of the RXLR effector family from the *P. sojae* genome, because sequence similarity between the members of this protein family is low except the RXLR motif [30]. Logistic regression was also evaluated as a strategy to trim down the number of dimensions (200) associated with the use of Kidera factors. Biologically, it seemed likely that some combinations of the amino acid positions and properties under consideration may be more important than others in terms of the protein function. Potentially, variable selection combined with logistic regression could reveal these particular amino acid positions in the flanking sequences, without losing classification performance compared to SVMs.



## 2.3 Implementation

### 2.3.1 A. Conversion of amino acid sequences to Kidera factors

A proteins structure is closely related to its function. Therefore, much predictive information can be gained by converting the linear protein sequences into vectors of numbers which take the amino acid properties into account. Kidera et al. [38] summarized 188 physical properties of 20 naturally occurring amino acids, and summarized them into 10 orthogonal properties (factors) while retaining the inherent information captured by the multiple properties [38]. They achieved their goal in three steps. First, they eliminated physical properties that were not normally distributed, because the goal was to compare different physical properties to each other, and when the properties have different distributions, the comparison becomes difficult. They also applied a standardization procedure on all the properties, by subtracting the mean and dividing by the standard deviation, so that they would all have the mean of 0 and standard deviation of 1 [38]. Secondly, they clustered the remaining properties based on their correlation values. Thirdly, they applied factor analysis to the average of each cluster [38]. Thus, 10 numeric values (1 for each of the 10 orthogonal factors) were derived for each of the naturally occurring 20 amino acids (Table 2.2). These values allowed any given amino acid sequence to be converted into strings of numbers, which in turn enables the application of procedures that require numerical rather than categorical information [49]. In this study, the 10 amino acids flanking each side of each candidate RXLR motif were converted to 20 sets of 10 Kidera factors, yielding 200 numbers, or dimensions, to represent a candidate motif.

Table 2.1: Summary of Kidera physical property factors

Factors	Properties
Factor 1	$\alpha$ -Helix or bend-structure preference-related
Factor 2	Bulk-related (side chain size)
Factor 3	$\beta$ -Structure preference-related
Factor 4	Hydrophobicity-related
Factor 5	Normalized frequency of double bend
Factor 6	Average value of average composition
Factor 7	Average relative fractional occurrence in $E_0$
Factor 8	Normalized frequency of $\alpha$ -region
Factor 9	pK-C
Factor 10	Surrounding hydrophobicity in $\beta$ -structure

Table 2.2: Summary of 10 Kidera factors. 10 Kidera factors, which explain 86% of the original 188 physical properties of amino acids.

Amino Acid	Factor 1	Factor 2	Factor 3	Factor 4	Factor 5	Factor 6	Factor 7	Factor 8	Factor 9	Factor 10
A	-1.56	-1.67	-0.97	-0.27	-0.93	-0.78	-0.2	-0.08	0.21	-0.48
C	0.12	1.27	1.37	1.87	-1.7	0.46	0.92	-0.39	0.23	0.93
D	0.58	-0.07	-0.12	0.81	0.18	0.37	-0.09	1.23	1.1	-1.73
E	-1.45	-0.22	-1.58	0.81	-0.92	0.15	-1.52	0.47	0.76	0.7
F	-0.21	-0.89	0.45	-1.05	-0.71	2.41	1.52	-0.69	1.13	1.1
G	1.46	0.24	0.07	1.1	1.1	0.59	0.84	-0.71	-0.03	-2.33
H	-0.41	0.19	-1.61	1.17	-1.31	0.4	0.04	0.38	-0.35	-0.12
I	-0.73	-1.96	-0.23	-0.16	0.1	-0.11	1.32	2.36	-1.66	0.46
K	-0.34	0.52	-0.28	0.28	1.61	1.01	-1.85	0.47	1.13	1.63
L	-1.04	-0.16	1.79	-0.77	-0.54	0.03	-0.83	0.51	0.66	-1.78
M	-1.4	0	-0.24	-1.1	-0.55	-2.05	0.96	-0.76	0.45	0.93
N	1.14	0.82	-0.23	1.7	1.54	-1.62	1.15	-0.08	-0.48	0.6
P	2.06	0.18	-0.42	-0.73	2	1.52	0.26	0.11	-1.27	0.27
Q	-0.47	0.98	-0.36	-1.43	0.22	-0.81	0.67	1.1	1.71	-0.44
R	0.22	-0.33	-1.15	-0.75	0.88	-0.45	0.3	-2.3	0.74	-0.28
S	0.81	-1.08	0.16	0.42	-0.21	-0.43	-1.89	-1.15	-0.97	-0.23
T	0.26	-0.7	1.21	0.63	-0.1	0.21	0.24	-1.15	-0.56	0.19
V	-0.74	2.1	-0.72	-1.57	-1.16	0.57	-0.48	-0.4	-2.3	-0.6
W	0.3	1.48	0.8	-0.56	0	-0.68	-0.31	1.03	-0.05	0.53
Y	1.38	-0.71	2.04	-0.4	0.5	-0.81	-1.07	0.06	-0.46	0.65

### 2.3.2 B. Building the training set

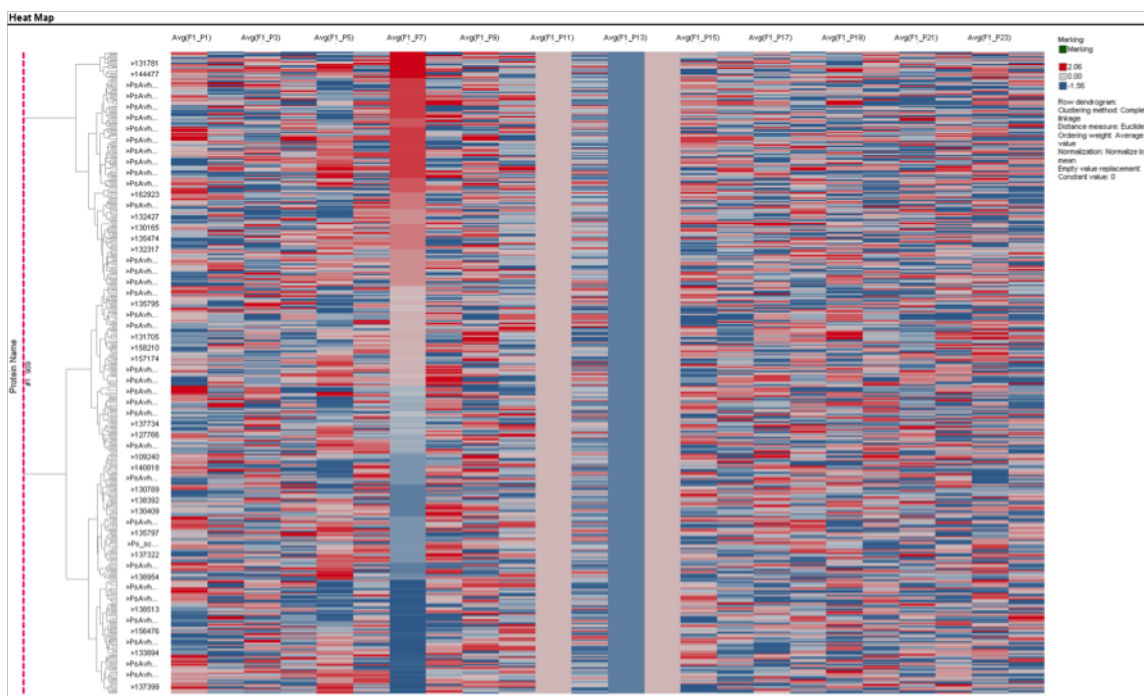
In order to build the training set, we used *P. sojae* effectors predicted with a Hidden Markov Model (HMM) as our positives. Jiang et al. [30] built HMMs from *P. sojae* effector candidates that were found through recursive BLAST sequence similarity searches with Avr1b. They used 10 amino acids flanking the RXLR motif on both sides to build an HMM. They used it for screening the predicted *P. sojae* proteome, and identified additional effector candidates based on the HMM score (E value less than 0.05) and manual inspection. For this study, effector candidates were selected from Jiang et al's list that contained only one RXLR motif so that the identification of the functional RXLR motif was unambiguous. Ten amino acids flanking the RXLR motif on each side were extracted and converted to Kidera factors. A set of negative examples was selected from the pool of 3034 secreted proteins, predicted from the *P. sojae* proteome by SignalP. Of the 647 secreted proteins that contained only one RXLR, 105 did not have 10 amino acids on at least one side, leaving 542 proteins. After removing 20 which included at least one stop codon in the 24 amino acids due to errors in the gene models, 522 remained in the set. Known HMM-predicted effectors were further removed from the reduced set. The remaining 228 proteins were used as the negative data set. From these 228 proteins, 10 amino acids flanking the RXLR motif on both sides were extracted and converted to Kidera factors.

## 2.4 Results

### 2.4.1 A. Hierarchical clustering analysis

Hierarchical clustering was used to discover any underlying hierarchical structure in the training dataset and to reveal any intrinsic pattern that separated the two groups. It was also of interest to determine if hierarchical clustering could select a group of effectors or effector candidates based on similarities in their Kidera factor profiles. Heat maps were generated using single-linkage hierarchical clustering for each of the 10 Kidera factors as well as a composite of all 10 Kidera factors by using the program Spotfire. A representative heat map is shown in Figure 2.1 for Kidera factor 1, which is a score scale related to  $\alpha$ -helix or bend-structure. However, this method proved not to be sensitive enough to detect even a general separation between the effectors and non-effectors in the training set.

Figure 2.1: Heatmap generated from the training set using Kidera factor 1. Rows represent an effector or effector candidate, while columns show the amino acid positions we considered. Red indicates a higher value in Kidera factor 1, while blue means a lower score. The three uniform colored columns show the RXLR motif position. There is no clustering pattern detected here. Positives and negatives were randomly mixed in the dendrogram and no clear distinguishing expression pattern was found. Similar negative results were obtained from using all the other Kidera factors independently and combined.



## 2.4.2 B. Spherical classification

It was hypothesized that the positive examples would cluster around the geometric center of the positives, while the negative examples would be randomly distributed further away from the geometric center. To test this hypothesis, hypothetical spheres of increasing sizes around the geometric center were calculated, and for each sphere size, all the points that fell within the sphere were considered as predicted positives for calculating precision and recall (Equation 2.3). Each protein sequence was considered as a point in the 200 dimension space as described in Implementation A. Then Euclidean distance was used as the distance between two given points,  $X = (x_1, x_2, \dots, x_{200})$  and  $Y = (y_1, y_2, \dots, y_{200})$  (Equation 2.1). The protein sequences in the analysis were classified into two categories, the positive examples (experimentally validated effector proteins and HMM-classified positives) and the negative examples (HMM-classified negatives). The geometric center of the positive examples was found by averaging each of the 200 positions (Equation 2.2). Positive examples that fell within the sphere were considered true positives, while positive examples that fell outside were considered false negatives. Negative examples that fell within the sphere were false positives and negative examples which fell outside the sphere were true negatives. Figure 2.2 shows that more than 80% of the points fell within the hypothetical sphere were true positives all the way up to 80% recall. Cumulative number of false positives is plotted in Figure 2.3, showing that only a small number (about 50 out of over 200) of false positives is reported when 80% of the true positives are recovered (Recall of 0.8). In Figure 2.4, receiver operating characteristic (ROC) curve is plotted, and the area under this curve was 0.854, demonstrating that the spherical classifier performed well. However, since the classifier was tested with the same data that was used to build the classifier, a 10-fold cross validation was performed, in which 10% of the training data was repeatedly left out of the training set, to be used to test the classifier. In order to perform the 10-fold cross validation, the positives and the negatives were first partitioned into 10 sets. Then each set was left out at a time and the center of the positives was calculated from the remaining 9 sets. Then for each set, using the new center calculated from the other 9 sets, we calculated the distance from the center of each member of the test set. The results for all test sets were then combined and used to produce a precision-recall curve and a ROC curve (Figure 2.5 and 2.6). The area under the ROC curve from the 10-fold cross validation was 0.840 and the results confirmed the trend of good classifier performance. Another view of the classifier performance is shown in Figure 2.6, in which the numbers of true positives (effectors classified as effectors by being contained in the separating sphere) and false positives (non-effectors classified as effectors) are shown as the classifying sphere size incrementally increases. When the sphere size is small, mostly true positives are gained, but more false positives are included as the sphere size increases (Figure 2.7). By using this graph, a stringent cutoff size of the sphere can be defined for use as a classifier for test data, at the point before the number of false positives gained began to equal the number of true positives gained (Figure 2.7). A relaxed cutoff could also be defined where the number of false positives gained first begins to exceed the number of true positives gained, in order to include more true positives which are at the

borderline. Overall, the spherical classifier performed well in separating the effectors from the non-effectors. This result suggests that the true positives are indeed clustered around their geometric center, while the negative examples are more widely scattered throughout the parameter space.

$$\text{Distance} = (x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_{200} - y_{200})^2 \quad (2.1)$$

For positive examples  $P_1, P_2, \dots, P_n$  where  $P_n = (p_{n1}, p_{n2}, \dots, p_{n200})$ , the geometric center is defined as  $\bar{P} = (\bar{p}_1, \bar{p}_2, \dots, \bar{p}_{200})$ , where  $\bar{p}_i = \frac{\bar{p}_{1i} + \bar{p}_{2i} + \dots + \bar{p}_{ni}}{n}$  for  $i \leq I \leq 200$ . (2.2)

Positive examples that fell within the sphere were considered true positives, while positive examples that fell outside were considered false negatives. Negative examples that fell within the sphere were false positives and negative examples which fell outside the sphere were true negatives. (2.3)

$$\text{Precision} = \frac{tp}{tp + fp}$$

$$\text{Recall} = \frac{tp}{tp + fn}$$

Since the spherical classification method worked well to separate the *Phytophthora sojae* effector sequences from the negative controls, a similar approach was evaluated for separating fungal or insect effectors from known negatives. The fungal and insect effectors were combined with the oomycete effectors, and their distances from the geometric center were calculated. Potential functional RXLR motifs were identified from six effectors (AvrLm6 and AvrLm4-7 from the fungus *Leptosphaeria maculans*, AvrL567 from the fungus *Melampsora lini*, Avr2 from the fungus *Fusarium oxysporum f. sp. lycopersici*, MiSSP7 from the fungus *Laccaria bicolor*, and vH13 from Hessian fly). In each case, the 20 amino acids flanking the motif were converted to Kidera factors. Then the distance from the center of the sphere of the oomycete positives was calculated for each of the six effector motifs using Equation 2.1. Based on this distance measure, one fungal effector (Ml AvrL567) was recovered when 95%

Figure 2.2: Precision-recall curve resulting from using hypothetical spheres around their geometric center to predict functional effectors.

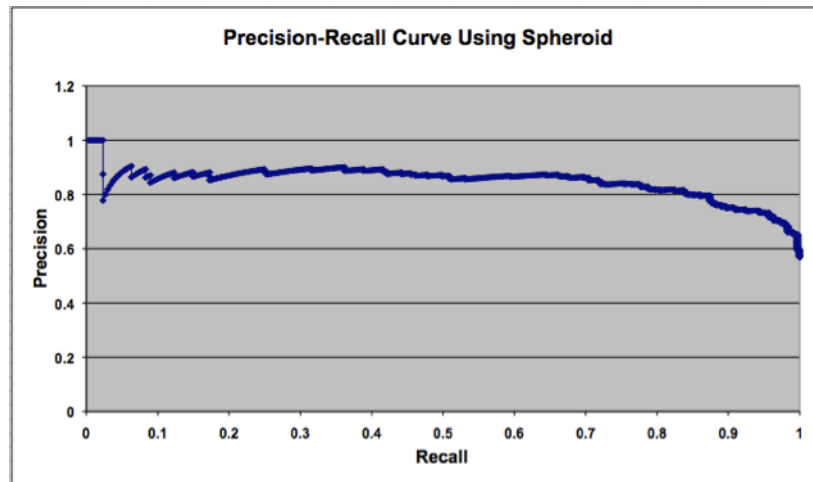
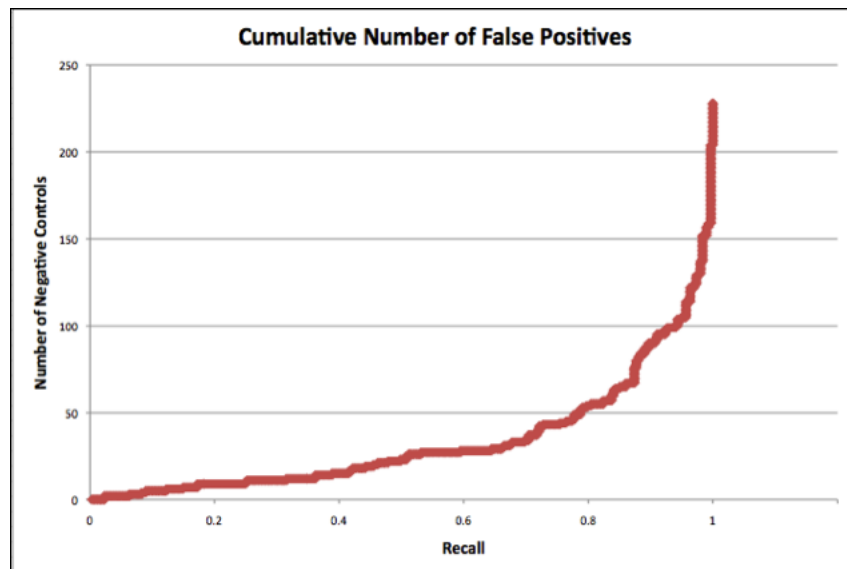


Figure 2.3: The cumulative number of false positives. Another view of the spherical classification result, which shows the cumulative number of false positives (which are the negative controls contained in the hypothetical sphere as its size increases).



of the oomycete positives and 46% of the oomycete negatives were contained in the sphere, while the others (vH13 Avr-1, Fol Avr2, Lm AvrLm6, MiSSP7 and Lm AvrLm4-7) were not recovered until all positives and negatives were contained in the sphere. Thus, the approach of mixing oomycete data with data from other species was not effective in distinguishing

Figure 2.4: ROC curve for Spherical Classification. Receiver operating characteristic curve is plotted for the spherical classification method. When covering 80% of the true positives, only about 20% of the false positives are retrieved. The area under this ROC curve is 0.854.

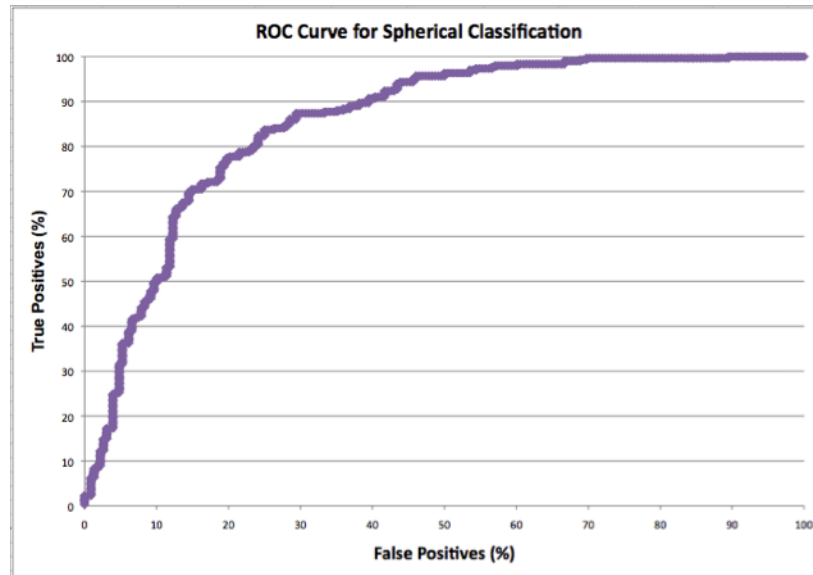
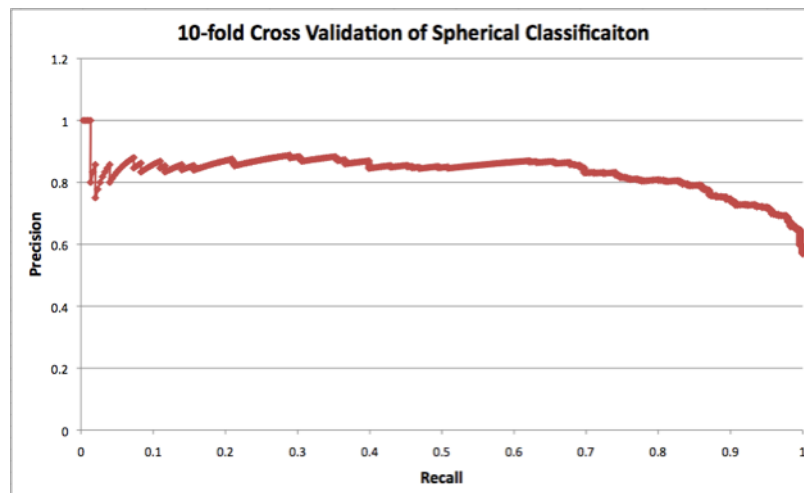
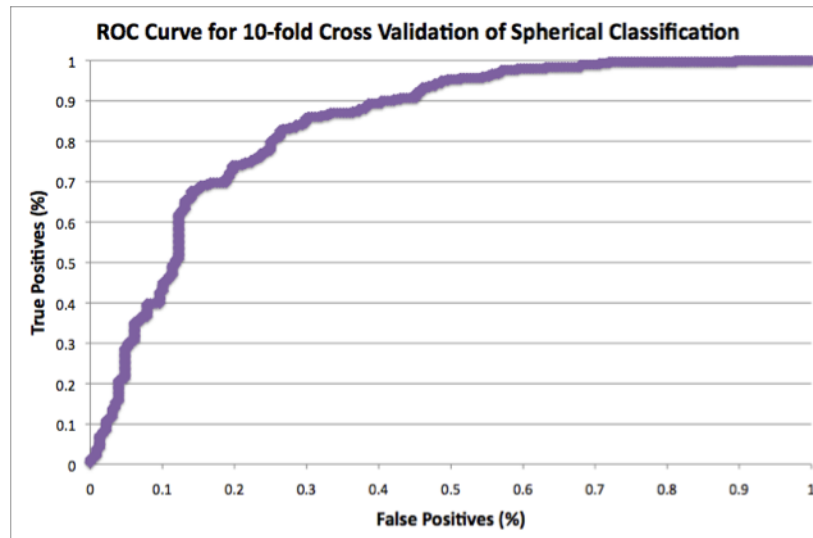


Figure 2.5: 10-fold Cross Validation for Spherical Classification



effectors from the other kingdoms from true negatives.

Figure 2.6: ROC curve for 10-fold Cross Validation of Spherical Classification. The area under this ROC curve is 0.840.

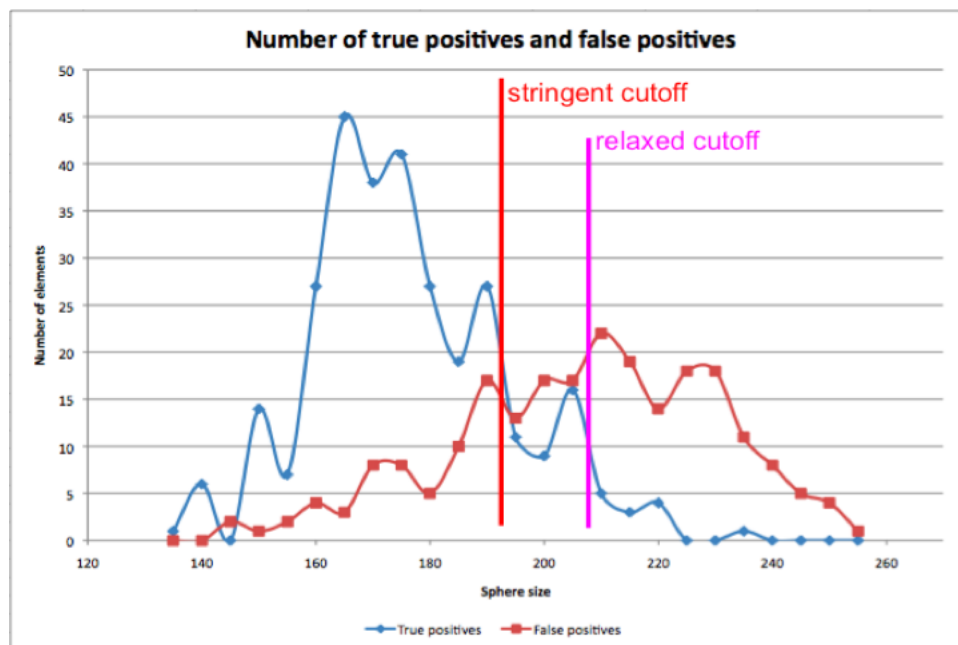


### 2.4.3 C. Support Vector Machines (SVM)

Based on the good performance of the spherical classifier in distinguishing *Phytophthora sojae* effectors, another method which uses a separator that can generalize to a wider variety of shapes was evaluated, namely Support Vector Machines (SVM). Because SVM maps the original data onto a feature space and finds a linear separating plane in that transformed space, the separator could actually be a sphere, an ellipsoid, or a more complex polynomial shape. There are many computational implementations of the SVM algorithm, including versions in R, in C++ and in Matlab among many others [8,27,37]. One of the most useful and well known packages with excellent documentations comes from the University of Waikato in New Zealand called WEKA [67,68]. The software package is written in Java, making it platform flexible. It is free for academic use, bringing together multiple, well-established machine learning algorithms for data mining analyses. The evaluation began with the LibSVM version of SVM in WEKA (version 3.6.2), using the radial basis kernel, which is one of the most popular kernels used in SVM when mapping vectors to another feature space [67]. The radial basis kernel was chosen because the distance between two mapped vectors is calculated as squared Euclidean distance, which could be interpreted as a similarity measure. As an alternative approach, a polynomial kernel was evaluated, because polynomial kernel of degree 2 most resembles a sphere, which was used in the previous spherical classification. Polynomial kernels allows a flexible non-linear mapping of the original features into the feature space [48]. Furthermore two different implementations of the SVM algorithm were evaluated, which differ in how they solve the quadratic programming optimization problem. SMO stands for sequential minimal optimization, which is a popular algorithm for solving



Figure 2.7: Number of true positives and false positives as Sphere Size increases. Number of true positives and false positives as the separating sphere size increases incrementally. When the sphere size is small, true positives are mostly contained in the sphere, while as the size grows, more and more false positives are included.



the quadratic programming optimization problem in SVM, breaking down a very large optimization problem into smaller problems and solving them sequentially [48]. LibSVM is another version of SVM found in WEKA, and the team who developed this library used a slightly different method for solving the quadratic problem compared to SMO [8]. In the evaluation of SVM classifiers (Table 2.3), 10-fold cross validation was used to evaluate the results. In addition to precision, recall and area under the ROC, another indicator of the SVM methods performance, the F-measure was calculated. The F-measure can be considered as a special type of average between precision and recall [45]. Ten fold cross-validation results from the SVM analysis on the *Phytophthora sojae* training set strongly suggested that the effector protein sequences could be effectively separated from the non-effectors with relatively high confidence by all of the methods (Table 2.3). The radial kernel was most effective and performed equally well as the spherical classifier.

Since fungi and oomycetes share similar strategies for entering into host cells, such as releasing effectors that bind to phosphatidylinositol 3-phosphate (PI3P), the potential for differentiating known fungal effectors on the basis of the oomycete training set was explored using the SVM (Table 2.4). The SVM method was used to construct a hyperplane for separating oomycete RXLR effectors from non-effectors, and the fungal sequences were given as a test set. As shown in Tables 2.4 through 2.6, the result suggested that fungal effector

Table 2.3: Support Vector Machine results based on a 10-fold cross-validation of the training set. The *Phytophthora sojae* training set was composed of experimentally validated effectors and HMM classified effectors (positive examples) as well as HMM classified negatives. Only training data results are presented here with 10 fold cross validation. There were HMM predicted effectors and non-effectors, total 529 instances. "Weighted average" shows the adjusted numbers, taking into account the different numbers in the two classes.

SVM Options	Precision	Recall	ROC Area	Class
LibSVM Radial Kernel	0.829	0.9	0.827	Effector
	0.851	0.754	0.827	Negative
	0.839	0.837	0.827	Weighted average
LibSVM Degree 2, Polynomial Kernel	0.82	0.877	0.811	Effector
	0.821	0.746	0.811	Negative
	0.82	0.82	0.811	Weighted average
SMO Degree 2, Normalized Polynomial Kernel	0.834	0.854	0.815	Effector
	0.801	0.776	0.815	Negative
	0.82	0.82	0.815	Weighted average

proteins could not be predicted reliably using the oomycete effector-derived SVM. Only the AvrLm6 protein of *Leptosphaeria maculans* was consistently identified as being an effector (Table 2.4), however permutations of the sequence of this protein also were often wrongly predicted to be effectors (Tables 2.5 and 2.6). Approximately 30% of the time, no matter which kernel was used, permutations of AvrLm6 were wrongly predicted to be effectors (Table 2.6). For the other fungal effectors, the SVM predicts them to be negative based on the oomycete training data (Table 2.4).

Table 2.4: Support Vector Machine predictions on six known fungal effectors. This table shows Support Vector Machine predictions on six known fungal effectors, based on oomycete training data. Three different SVM settings were used for testing as indicated by the column titles. With RXLR denotes that RXLR motif positions were also included in the analysis, while no RXLR signifies that only the flanking regions of the RXLR motif positions were considered.

Kingdom	Protein Name	SMO (no RXLR)	SMO (with RXLR)	LibSVM (no RXLR)	LibSVM (with RXLR)
fungus	Lm AvrLm4-7	Negative	Negative	Negative	Negative
fungus	Lm AvrLm6	Effector	Effector	Effector	Effector
insect	vH13 Avr-1	Negative	Negative	Negative	Negative
fungus	MiSSP7	Negative	Negative	Negative	Negative
fungus	Fol Avr2	Effector	Effector	Negative	Negative
fungus	Ml AvrL567	Negative	Negative	Negative	Negative

Table 2.5: Permuted sequences of AvrLm6 are wrongly predicted to be effectors.

Kingdom	Permuted Sequences				
	Protein Name	SMO (no RXLR)	SMO (with RXLR)	LibSVM (no RXLR)	LibSVM (with RXLR)
fungus	Lm AvrLm4-7	Negative	Negative	Negative	Negative
fungus	Lm AvrLm6	Effector	Negative	Effector	Negative
insect	vH13 Avr-1	Negative	Negative	Negative	Negative
fungus	MiSSP7	Negative	Negative	Negative	Negative
fungus	Fol Avr2	Negative	Negative	Negative	Negative
fungus	Ml AvrL567	Negative	Negative	Negative	Negative

Table 2.6: Permutations test on the 6 non-oomycete effectors. Out of 1000 permutations, the number of times each SVM predicted permuted sequences of the corresponding effector to be an effector.

SVM implementation	AvrLm47	AvrLm6	vH13	MiSSP7	Avr2	AvrL567
LibSVM Radial Kernel	66	326	60	61	161	179
LibSVM Degree 2, Polynomial Kernel	52	343	15	66	207	176
SMO Degree 2, PolyKernel	84	368	42	85	260	215
SMO Degree 2, Normalized Polynomial Kernel	31	251	4	37	137	161
SMO PolyKernel (original option PolyKernel)	222	320	207	181	322	222

#### 2.4.4 D. Logistic Regression with Stepwise Variable Selection

In order to explore the possibility of reducing the number of features (from 200 features per sample) to a subset of potentially more important features variable selection using logistic regression was performed. The purpose of this approach was to potentially reduce noise present in the vast feature space. Stepwise variable selection was chosen, because the algorithm combines both forward selection and backward elimination testing each variable for inclusion or exclusion at each step, perhaps allowing a better evaluation of each variable in the model. We used the statistical program SAS (version 9.2) to perform logistic regression (with proc logistic command) with stepwise variable selection. A column named event was added to the training data matrix to indicate effector status, which served as the binary dependent variable, and all 200 columns, which represent 10 Kidera factors of the selected 20 amino acid positions flanking each RXLR motif, were considered to be included in the logistic regression model. Table 2.7 shows the logistic regression model produced by SAS on the training data, which incorporates 27 of the 200 features. SVM was then implemented using only the selected variables from the logistic regression model, but the algorithm performed worse than when all the features were included (Table 2.8).

Table 2.7: Final list of 27 features selected for the logistic regression model. This table shows the final list of 27 features (which are combinations of position and Kidera factor) selected for the logistic regression model from the training data set (229 negatives and 301 positives). Test statistic labeled Wald Chi-Square and the respective p-values are reported for the null hypothesis that each coefficient is equal to 0.

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-3.2729	0.3668	79.6339	<.0001
F4 P17	1	0.909	0.1854	24.0443	<.0001
F5 P22	1	-0.8461	0.177	22.8488	<.0001
F2 P15	1	-0.7274	0.1636	19.7749	<.0001
F4 P10	1	0.6362	0.1546	16.9325	<.0001
F8 P6	1	0.5373	0.1322	16.5188	<.0001
F2 P7	1	-0.524	0.1401	13.988	0.0002
F2 P16	1	-0.6613	0.1792	13.625	0.0002
F3 P3	1	-0.4683	0.131	12.7821	0.0003
F2 P18	1	-0.5644	0.1622	12.1023	0.0005
F3 P16	1	0.4677	0.1423	10.8017	0.001
F7 P24	1	-0.4898	0.1492	10.7774	0.001
F4 P23	1	0.558	0.1713	10.606	0.0011
F3 P15	1	0.48	0.1551	9.5795	0.002
F4 P6	1	0.5992	0.198	9.1591	0.0025
F7 P21	1	-0.4571	0.1518	9.0664	0.0026
F4 P22	1	0.5163	0.1716	9.0563	0.0026
F7 P1	1	-0.409	0.1372	8.8815	0.0029
F3 P23	1	-0.3677	0.1277	8.2969	0.004
F7 P9	1	-0.3958	0.1378	8.2445	0.0041
F6 P16	1	0.4868	0.1742	7.8104	0.0052
F8 P24	1	0.4673	0.1687	7.6753	0.0056
F9 P16	1	0.6082	0.2205	7.6073	0.0058
F3 P7	1	0.3689	0.1369	7.2615	0.007
F9 P4	1	-0.4836	0.1882	6.6015	0.0102
F3 P24	1	-0.2794	0.1221	5.2333	0.0222
F10 P8	1	-0.3265	0.1489	4.8087	0.0283
F4 P20	1	0.3565	0.1749	4.1553	0.0415

Table 2.8: SVM results from *P. sojae* with only selected features from logistic regression variable selection.

Training only, 10-fold cross validation SVM Implementation	Precision	Recall	F-Measure	ROC Area	Class
LibSVM Degree 2, Radial Kernel, No normalization	0.826	0.884	0.854	0.819	Effector
	0.831	0.754	0.791	0.819	Negative
	0.828	0.828	0.827	0.819	Weighted Average
LibSVM Degree 2, Polynomial Kernel, No normalization	0.813	0.781	0.797	0.772	Effector
	0.725	0.763	0.744	0.772	Negative
	0.775	0.773	0.774	0.772	Weighted Average

## 2.5 Discussion

A variety of machine learning and data mining tools were evaluated to find ways to classify effectors from non-effectors among *P. sojae* secreted proteins. Many of these tools, such as hierarchical clustering and Support Vector Machines, have been used in microarray gene expression datasets to identify clusters of tightly regulated genes [7, 18, 25]. But here these methods were applied in a completely different context, namely for making predictions about new protein sequences by transforming them into numeric Kidera values based on amino acid physical properties. The training dataset was created from *P. sojae* RXLR effectors (positives) and other RXLR-containing sequences (negatives) from the secretome, in order to test the effectiveness of the different approaches. An important qualification is that the positives and the negatives in the training set are derived from using a Hidden Markov Model, rather than based on individual experimental validation. Many of the HMM predicted effector candidates have been proven to possess ability to suppress plant triggered immune response such as programmed cell death [64]. However, there is no experimental evidence at this point to verify that all of the RXLR-containing sequences used as negatives are true negatives. In other words, it is possible that some of the negative examples could potentially belong to the *P. sojae* RXLR effector superfamily. The result from the spherical classification (10-fold cross validation ROC area of 0.840) indicates that this classification method performed very well for the *P. sojae* data. This result from the relatively simple approach is comparable to the results from SVM (10-fold cross validation ROC area ranging from 0.767 to 0.827). The results clearly show, also, that the signal present in the surrounding amino acids of RXLR motifs in *P. sojae* is sufficient for classifying a candidate as an effector or not, even when the exact RXLR motifs themselves are excluded from the analyses. The HMM classification, which is derived from the signal from these flanking amino acids, can be recapitulated using other classification methods, such as the spherical classification and SVM. The results suggest that these flanking amino acids sequences must contribute to the structure and stability of the proteins overall, especially since the analyses used metrics that are derived from physical properties of the amino acids. Unfortunately, the methods could not extend to effective classification of fungal or insect effectors based on the *P. sojae* data as a training set, probably because the evolutionary distances between these organisms (fungi, insects and oomycetes) are so large.

This classification approach, converting that converts amino acid sequences into numeric values and applying classification learning is highly applicable to protein sequences of any organism. Other published studies have used Support Vector Machine predict protein sub-cellular locations [9], or to classify proteins as cell-entering or not in an Avian system [51]. In the case of Avian cell-entering peptide classification using Support Vector Machine, the authors used comprehensive peptide properties, including the length and the net charge of the entire peptide, percent polar amino acids and percent positive amino acids as well as many other factors [51], while this study focused on structural and chemical parameters of each individual amino acid [49]. Sander et al. [51] performed statistical analysis on 61

features and selected the best performing ones, while the 10 Kidera factors [38] were statistically derived from 188 different physical properties of amino acids. The Kidera factors have also been successfully used in combination with Fourier transformation to study the organization of protein sequences [54].

There is a need for a well defined training dataset composed of positive and negative examples in order for this approach to work properly. If there is perfect mathematical separation between the positives and negatives, or if the number of positives and negatives are severely unbalanced, such factors will need careful consideration for these computational methods to perform fairly. In the Avian cell-entering peptide study, the authors created multiple sets of training data, by creating new negative sequences by using 0th order Markov model or randomly selecting a certain number of the negatives to balance the dataset size [51]. In this study, the number of positives and negatives were similar to each other (301 and 228 respectively), and when more negatives were created by permuting the positives in order to create a perfect balance compared to the number of positives, the results were not significantly altered (data not shown). More future studies combining the type of computational approach presented here with experimental validation will continue to reveal previously unknown functions of interesting biological molecules.

# Chapter 3

## Functional linkage network approach to predicting *Phytophthora sojae* effector candidates

Authors: Hyunjin D. Choi, T.M. Murali and Brett Tyler

### 3.1 Abstract

As more and more genomes are sequenced, the challenge of annotating the functions of unknown genes remains. Many computational approaches have been introduced to automate and to accelerate the annotation process. This chapter explores a network based method to predict effector sequences in *Phytophthora sojae*. By converting RXLR-like protein domains using Kidera factors, a functional linkage network of *P. sojae* proteins was built. A network-based algorithm called Sinksource+ was utilized with experimentally validated *P. sojae* effectors as seeds. Among the domains with high confidence scores, more than 80% of the known Hidden Markov Model predicted effector sequences were within the top 4% of all protein domains included in the analysis. Most of the top scoring RXLR-like domains proved to be previously identified effectors. Highly ranked sequences that were not previously identified as effectors were evaluated further to produce a list of predicted new effectors. Experimentally testing of some of the predicted new effectors is described in Chapter 5.

## 3.2 Background

With advancements in sequencing technologies, thousands of new genomes have been sequenced since the publishing of the first complete genome (*Haemophilus influenzae*) using the whole genome shotgun sequencing method in 1995 [22]. The overwhelming amount of available sequencing data underscores the need and challenge of deciphering biological information embedded in the genomes, related to their structures and functions. Traditionally, protein functions have been studied from biochemical laboratory experiments based on individual proteins of interest. New genomic scale sequencing information opens opportunities to contextualize the function of a protein in a network of interacting molecules [41].

One method for studying protein functions involves using protein-protein interaction information to build interaction networks. These networks can be used to predict protein functions based on the assumption that interacting neighbors in the network share common functions [19,36,40,53]. Algorithms for this method of protein function prediction utilize the idea of functional linkage between proteins based on results from molecular experiments, such as yeast two-hybrid genome screens (for protein-protein interactions) or analysis of mRNA expression correlation [19,36]. Other functional linkage relationships between proteins may be inferred from genome sequences, where some interacting proteins in one species may be fused together as a single protein in another species [41].

In this chapter, this conceptual framework of network-based protein function prediction was extended and applied to the effector prediction problem in *Phytophthora sojae*. The Sinksource+ algorithm was selected to explore the secretome of *P. sojae*, with the goal of identifying effector candidates based on network evidence propagation. A network-based approach to effector searches was particularly attractive, because evidence from BLAST search results suggested that RXLR-motif containing effectors of *P. sojae* would form a set of tightly clustered small local networks, rather than a global hierarchical structure. Instead of protein-protein interaction evidence, correlations among the Kidera factor scores of protein segments containing RXLR-like motifs were utilized (see Chapter 2) .

## 3.3 Implementation

Generally, for protein function prediction with the Sinksource+ algorithm, a functional linkage network is created from experimental evidence of protein-protein relationships. A node represents a protein, and an edge a relationship between two proteins based on some experimental evidence. Three different networks were built; the first network was constructed for validation purposes using HMM predicted effectors, the second contained the test data from the *P. sojae* secretome, and the third containing 1000 randomly permuted sequence segments was constructed to establish significance levels. After defining the elements of the three functional linkage networks, an appropriate set of positive nodes for seeding each



network was established. For the purpose of building a validation network, each node was created using 20 amino acids flanking an RXLR motif. The data set for the validation network was identical to the training set used for the spherical classification and the Support Vector Machine analysis in Chapter 2, namely 301 HMM predicted positives and 228 HMM predicted negatives. Among the HMM predicted positives were 20 experimentally validated effectors, which were used to seed the network. For the purpose of effector candidate prediction, a network composed of data from *P. sojae* secreted proteins was created. The nodes for this test network were assigned to be a protein segment of 20 amino acids, 10 each side of an RXLR-like motif (defined as [R|K|H], any residue, [L|I|M|F|Y|W]). From the *P. sojae* secretome containing about 3000 proteins, more than 25,000 such nodes were identified because most secreted proteins had many RXLR-like motifs. For seeding the test network, a single RXLR-containing segment was selected from each of 25 experimentally-validated effectors. Because only one segment was expected to be functional per effector, each seed effector was manually curated to identify the one RXLR segment most likely to be functional. In order to assess the Sinksource+ algorithm's performance in the test network, 1000 permuted nodes were included in this network. These 1000 nodes were derived from flanking sequences of RXLR motifs occurring in 1000 randomly selected permuted sequences of the *P. sojae* secretome. The last network was purely composed of the same 1000 permuted sequences, and 10 of those nodes were randomly selected as seeds. The selection of random seeds was done only once. For all three networks, every pair-wise Pearson correlation was calculated among the Kiderra factor vectors of all the nodes in each network. This was computationally the most intensive step. To simplify the three networks, only the top 20 neighbors of each node were included in the functional linkage networks, because there was evidence that the number of true homologs for a given effector might average about 20 in *P. sojae*, based on BLAST results. Since the Sinksource+ algorithm was used for the analysis, no negative nodes were assigned.

### 3.4 Results

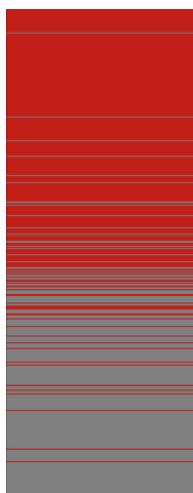
Before the network approach was used to search for new effectors, the approach was validated using the network composed of the HMM positives and negatives. It was useful to run the Sinksource+ algorithm on the validation network first, because this dataset was already analyzed using different classification methods in Chapter 2, and separation between the two classes of nodes (HMM predicted positives and HMM predicted negatives) was well established in this dataset. The prediction was that the HMM predicted positives would rank higher than the HMM predicted negatives.

After the Sinksource+ algorithm was run over the validation network, the confidence values of the motifs from the HMM predicted positives were ranked much higher than those from the HMM predicted negatives (Figure 3.1). When precision and recall were calculated, over 90% precision was achieved at 90% recall (Figure 3.2). The area under the ROC curve

was 0.933 (Figure 3.3). These results provided strong validation that the functional linkage network would be as effective in identifying functional RXLR motifs as the HMM.

In preparation for analyzing the complete set of RXLR motifs from the *P. sojae* secretome (test data set), SinkSource+ was run once using the set of 1000 negative sequences created by permuting sequences from *P. sojae* secretome. Figure 3.4 shows the distribution of confidence scores that resulted. 99% of the confidence scores were smaller than 0.111 and 95% of the scores were less than 0.082.

Figure 3.1: Ranking of Sinksource+ confidence scores of 281 RXLR-like motifs from HMM predicted effectors (red) and 223 permuted negatives (grey), in the validation data network.



Sinksource+ was then run on the *P. sojae* test network containing about 25,000 RXLR-like motifs from the *P. sojae* secretome, the 1000 RXLR-like motifs from permuted sequences (as negative controls), and seeds from 25 experimentally validated effectors. The distribution of local confidence scores from the *P. sojae* test network was compared with those from the random network alone (Figure 3.5). The results indicated that the test network produced local confidence scores which were larger than the scores from the random network, consistent with the expectation that the test network contains tighter clusters of more closely related nodes than the random network.

Next, the distribution of confidence scores of the HMM predicted candidate effectors were compared to the scores of the permuted negatives within the *P. sojae* test network. More than 80% of the HMM candidate effectors ranked within the first 1000 nodes out of more than 25,000 nodes in the network (Figure 3.6), while the scores of the permuted sequences were uniformly distributed throughout the ranked list of candidates (Figure 3.7). The precision and recall were calculated on the 281 of HMM predicted effectors as positives and the 1000 permuted segments as negatives. Over 90% precision was achieved at 90% recall (Figure 3.8). The area under the ROC curve was 0.975 (Figure 3.9). Furthermore, among the 132 nodes

Figure 3.2: Precision-Recall curve for the validation network (same as Figure 3.1), calculated on the 281 HMM predicted positives that were treated as unknowns and the 223 negatives produced by permutation.

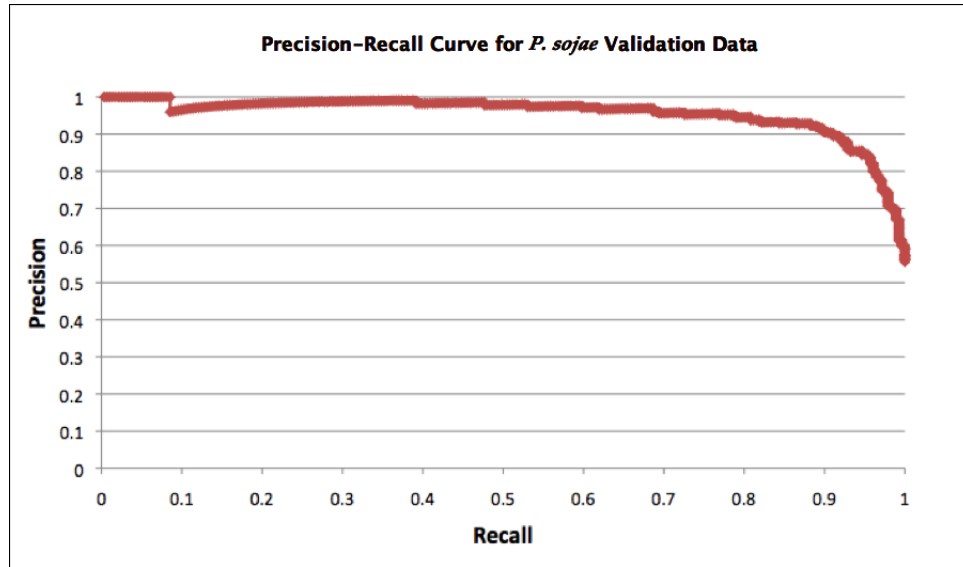


Figure 3.3: ROC curve for the *P. sojae* validation data network (same as Figure 3.1). Area under the ROC curve was 0.933.

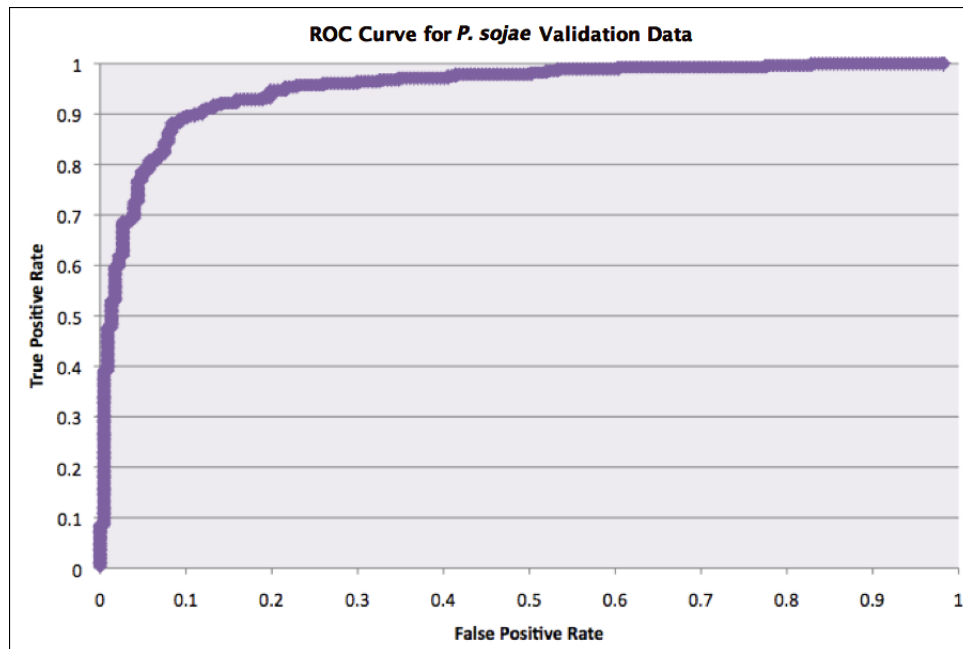
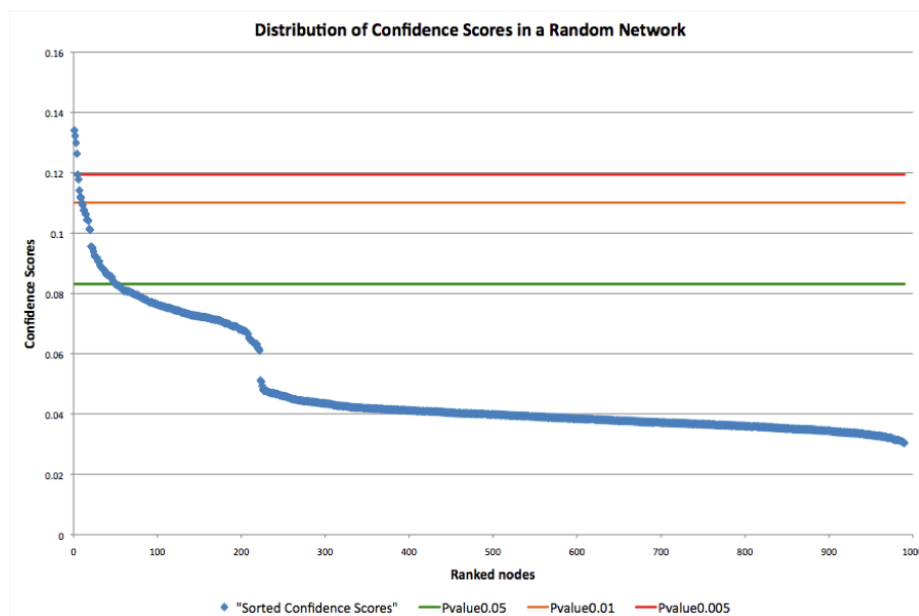


Figure 3.4: Null distribution of confidence scores produced by the Sinksource+ algorithm on a random network created from 1000 permuted sequences and 10 randomly assigned seeds. 99.5% of the confidence scores were less than 0.12, 99% were less than 0.111 and 95% were less than 0.082.



with p-values better than 0.01, 33% were HMM predicted positives and among the 356 nodes with p-values better than 0.05, 25% were HMM predicted positives (Figure 3.5). Among the top 25, 21 of the HMM classified positives were predicted to be effector sequences (Table 3.1). Considering that the *P. sojae* test network only contained 281 HMM positives from more than 25000 nodes (about 1% of all nodes), the percentages of HMM positives among the nodes with small p-values were very high. These results affirmed that the Sinksource+ algorithm was effective in predicting functional RXLR-like motifs.

To further evaluate the performance of the network in predicting effector sequences, experimentally validated effector sequences from two other oomycete species, namely *Phytophthora infestans* and *Hyaloperonospora arabidopsidis* were included in the analysis. Many of these sequences ranked highly based on the local confidence scores (Table 3. 2), showing that effector sequences from other similar oomycete species were in close proximity to effector nodes in the *P. sojae* network.

As shown in Table 3.1, 5 of the top 25 predictions appeared to be false positives, based on the fact that they contained transmembrane domains, and the RXLR-motifs were located far from the N-terminus (see Chapter 5 more a more detailed discussion of these evaluation criteria). In an effort to reduce the frequency of false positives, an alternative approach was evaluated that more strongly emphasized the highest correlation values. The correlation

Figure 3.5: Distribution of confidence scores from the *P. sojae* test network. P-values were calculated from the null distribution (Figure 3.4). 132 nodes (0.53%) had confidence scores with p values better than 0.01 and 356 (1.42%) had scores better than 0.05.

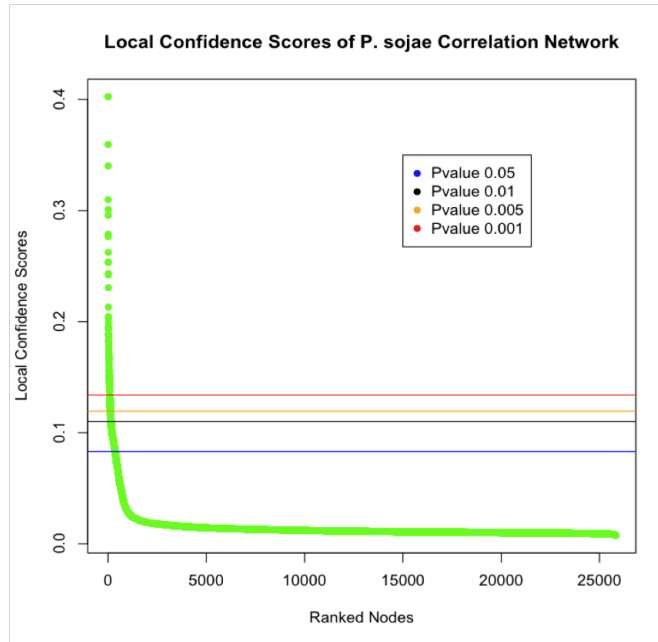


Figure 3.6: Distribution of HMM positive RXLR-like motifs in the test data set ranked by confidence score.

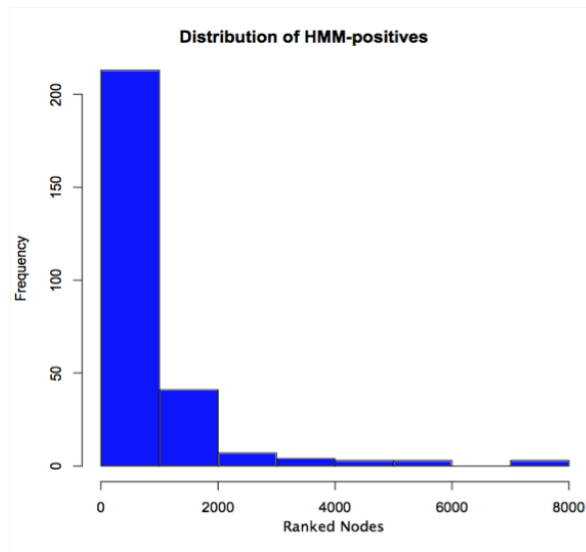


Figure 3.7: Distribution of permuted (true negative) RXLR-like motifs in the test data set ranked by confidence score.

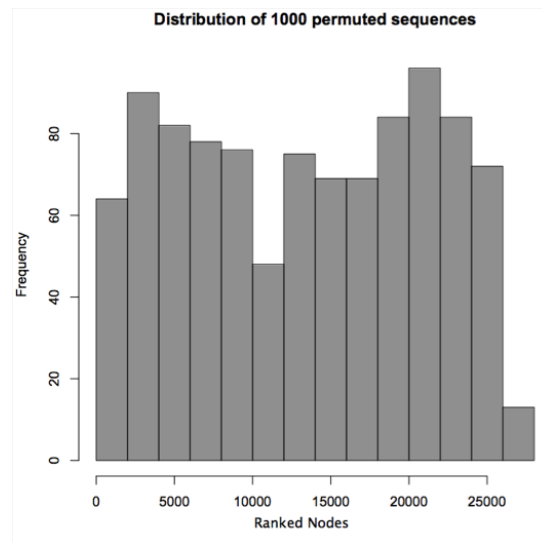
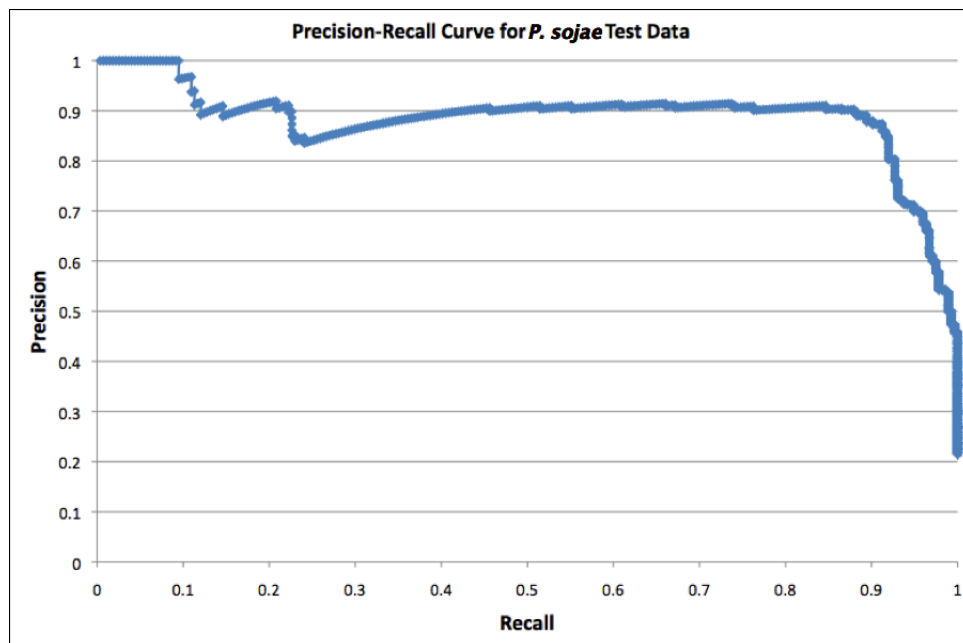


Figure 3.8: Precision-recall curve calculated on 281 HMM predicted positives and 1000 permuted true negatives in the *P. sojae* test data network



values used for weighting the edges were squared and the analysis was repeated. The top scoring candidates from the previous analysis shifted slightly in order, and other HMM

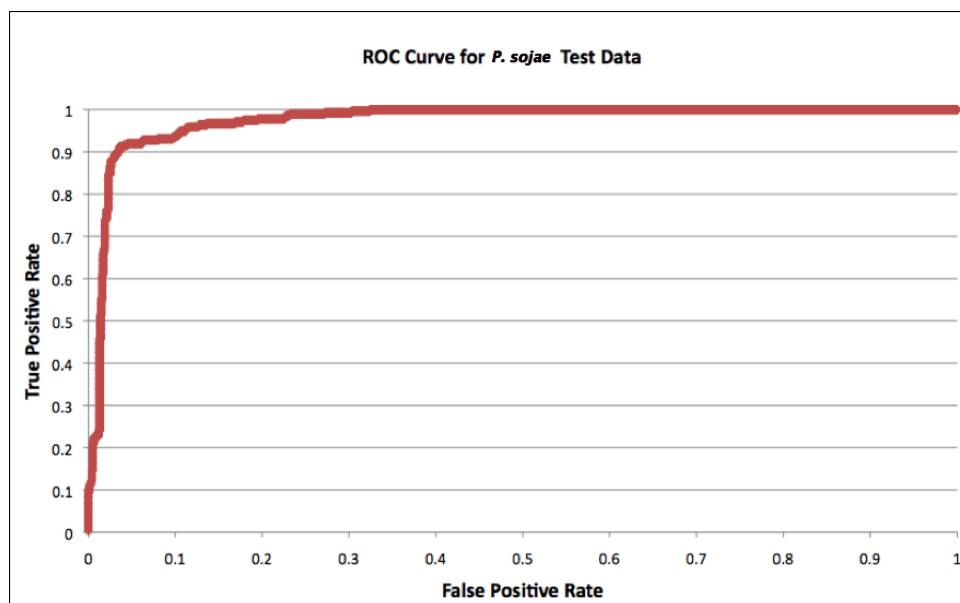
Figure 3.9: ROC curve for *P. sojae* test data network. Area under the ROC curve was 0.975.

Table 3.1: Top 25 scoring *P. sojae* candidate effectors. Red color indicates HMM predicted effectors. The six-digit number indicates *P. sojae* gene/protein ID; the second single digit shows the order of RXLR-like motif occurrence within the protein (i. e. 1 indicates that it is the first RXLR-like motif to occur in the protein). The start position indicates the starting position of the 20 amino acids with respect to the first amino acid of the protein after the signal peptide has been removed.

Rank	Gene ID	Occurrence order	Start position	RxLR-like Motif	HMM Predictions	local confidence	NOTES
1	158998	1	40	RMLR	PsAvh 7c	0.402	
2	139921	1	40	RMLR	Copy of PsAvh 7c	0.402	Hypothetical protein Avh1b-7c
3	158995	1	40	RMLR	PsAvh 7a	0.359	
4	134429	1	40	RMLR	Copy of PsAvh 7a	0.34	Hypothetical protein Avh1b-7a
5	127824	3	43	RMLR	PsAvh 22	0.31	Hypothetical protein Avh1b-22
6	159010	3	43	RMLR	PsAvh 22	0.301	
7	159009	1	39	RMLR	PsAvh 21	0.296	
8	159032	1	36	RMLR	PsAvh 46	0.279	
9	159063	1	39	RQLR	PsAvh 91	0.277	
10	158991	2	41	RFLR	PsAvh 1	0.262	Elicitor Avh1b
11	109104	2	41	RFLR	PsAvh 1	0.254	Inactive elicitor Avr1b (pseudogene)
12	159116	2	95	RHLR	PsAvh 160	0.253	
13	137978	7	254	KHIE		0.243	Hypothetical protein, Transmembrane domain
14	159002	1	84	RMLR	PsAvh 14	0.242	
15	159275	2	77	RMLR	PsAvh 319	0.231	
16	159038	3	65	RLLR	PsAvh 53	0.213	
17	133987	10	389	RRIQ		0.204	Hypothetical protein, Transmembrane domain
18	131783	25	602	RMLV		0.203	Hypothetical protein, Transmembrane domain
19	159035	1	51	RLLR	PsAvh 50	0.198	
20	134001	1	39	RLLR	PsAvh 50	0.195	
21	158992	2	44	RFLR	PsAvh 4	0.193	
22	142101	2	44	RFLR	Copy of PsAvh 4	0.193	
23	133979	16	452	RTLQ		0.189	Transmembrane domain
24	134724	15	538	KQWI		0.188	Transmembrane domain
25	159236	1	31	RHLR	PsAvh 280	0.188	

predicted effectors scored more highly (Table 3.3). In particular, all of the top 28 predictions were HMM positives, and 30.2% of the HMM positives ranked among the top 500 predictions.

Table 3.2: Top 25 scoring candidate effectors from a list that included RXLR motifs from 4 *Phytophthora infestans* effectors. Two of the effectors from *P. infestans* (highlighted yellow) ranked highly.

Rank	Gene ID	Occurrence	order	Start position	RxLR-like Motif	HMM Predictions	Local Confidence
1	158998	1		40	RMLR	PsAvh 7c 158998	0.39
2	139921	1		40	RMLR	PsAvr1b 7c	0.387
3	158995	1		40	RMLR	PsAvh 7a 158995	0.343
4	134429	1		40	RMLR	PsAvr1b 7a	0.326
5	127824	3		43	RMLR	PsAvh 22	0.299
6	159010	3		43	RMLR	PsAvh 22 159010	0.291
7	159009	1		39	RMLR	PsAvh 21 159009	0.287
8	PiAvr2	1		18	RFLR	Phytophthora infestans effector	0.286
9	159032	1		36	RMLR	PsAvh 46 159032	0.275
10	159063	1		39	RQLR	PsAvh 91 159063	0.265
11	158991	2		41	RFLR	PsAvh1 b	0.259
12	159116	2		95	RHLR	PsAvh 160	0.252
13	109104	2		41	RFLR	PsAvh1 b	0.251
14	137978	7		254	KHIE		0.227
15	159275	2		77	RMLR	PsAvh 319 159275	0.225
16	159002	1		84	RMLR	PsAvh 14 159002	0.223
17	159038	3		65	RLLR	PsAvh 53 159038	0.21
18	131783	25		602	RMLV		0.201
19	133987	10		389	RRIQ		0.2
20	159035	1		51	RLLR	PsAvh 50 159035	0.196
21	134001	1		39	RLLR	PsAvh 50	0.193
22	158992	2		44	RFLR	PsAvh 4 158992	0.192
23	142101	2		44	RFLR	PsAvh 4	0.192
24	PiAvr3a	2		11	RRLR	Phytophthora infestans effector	0.192
25	159236	1		31	RHLR	PsAvh 280 159236	0.187

By comparison, 26.8% of the HMM-positives ranked among the top 500 predictions when un-squared edge weights were used.

Given that the network approach could retrieve many of the HMM predicted results with high confidence, the 150 top ranked candidates that were not previously predicted by the HMM approach were retrieved. Some of these are shown in Table 3.4. Other criteria (such as transmembrane domain prediction, quality of signal peptide, length of the candidate, position of the motif, etc) were used to filter this list further as described. The best candidate from this process was experimentally tested and the results are discussed in Chapter 5.

## 3.5 Discussion

The Sinksources+ algorithm was used to predict effector candidates from *P. sojae* protein sequences and to compare the predictions with those from Hidden Markov Models. In order to evaluate the algorithm performance, HMM predicted RXLR effectors were used as positive controls and permuted sequences were included as true negatives. The positive controls ranked highly (Figure 3.6), while the negative controls were uniformly randomly distributed (Figure 3.7), affirming the validity the network approach.



Table 3.3: Top scoring RXLR-like motifs (top 28 shown here) predicted using squared correlations as edge weights.

Rank	Gene ID	Occurrence order	Start position	RxLR-like Motif	Local Confidence	HMM predictions
1	139923	1	40	RMLR	0.411	Similar to PsAvh 7 family
2	158998	1	40	RMLR	0.403	PsAvh 7c 158998
3	139921	1	40	RMLR	0.403	PsAvh 7c
4	158995	1	40	RMLR	0.35	PsAvh 7a 158995
5	134429	1	40	RMLR	0.339	PsAvh 7a
6	135585	2	43	RMLR	0.32	PsAvr1b
7	139995	2	32	RLLR	0.311	Similar to PsAvr1b
8	159009	1	39	RMLR	0.302	PsAvh 21 159009
9	131013	2	27	RHLR	0.294	Similar to PsAvh 240
10	159286	2	75	RSLR	0.276	PsAvh 330
11	158991	2	41	RFLR	0.273	PsAvh 1
12	129558	1	28	RSLA	0.268	Inactive PsAvr1b
13	140525	1	38	RRLK	0.266	Avh 260
14	109104	2	41	RFLR	0.265	Similar to PsAvh1
15	140196	2	43	RHLR	0.252	Similar to PsAvh 158
16	127824	3	43	RMLR	0.249	Same as PsAvh 22
17	135627	2	32	RFLR	0.249	PsAvh 5
18	159010	3	43	RMLR	0.245	PsAvh 22
19	135621	1	40	RFLV	0.244	PsAvh 109
20	136286	2	35	RHLR	0.239	PsAvh 115
21	136214	2	36	RLLR	0.229	PsAvh 94
22	136215	2	39	RSLR	0.227	PsAvh 180
23	130725	1	39	RYLR	0.226	PsAvh 171
24	159066	6	98	RLLR	0.225	PsAvh 95
25	159275	2	77	RMLR	0.222	PsAvh 319
26	159038	3	65	RLLR	0.221	PsAvh 53
27	159032	1	36	RMLR	0.22	PsAvh 46
28	159063	1	39	RQLR	0.208	PsAvh 91

Table 3.4: *P. sojae* effector prediction list. Selected list of the top ranking 150 RXLR-like motif segments were predicted to be potential effector candidates, which were not identified previously using the HMM approach. The ranks are shown from within the list of 25000.

Rank	Gene ID	Occurrence order	Start position	RxLR-like Motif	Local Confidence
29	133896	2	34	HEWN	0.183
41	127602	2	71	HSYL	0.172
55	128329	17	338	KMLA	0.163
71	128540	4	136	KPLR	0.147
74	132864	4	193	KTIF	0.146
77	130496	8	308	KLLS	0.143
85	140303	4	147	KNIT	0.138
89	131366	9	230	HAI	0.132
95	138913	12	350	RCLD	0.130
111	131653	5	114	RIYQ	0.124
117	139012	10	336	KQIR	0.121
136	134355	4	102	KFLQ	0.111
139	129333	1	21	KLWR	0.110
146	131168	4	107	RLLI	0.110
149	133851	2	13	KLLR	0.108

Overall the prediction results indicated that the network approach to effector candidate prediction compared well with the results from Hidden Markov Model predictions, based on the high precision values even in high recall values (90% precision at 90% recall) (Figure 3.8) and the high value of area under ROC curve (0.975) (Figure 3.9). Among the RXLR-like motif containing segments with high confidence scores from the Sinksource+ approach, almost all nodes in the top 30 ranked were HMM predicted positives (Table 3.1), and the performance of the approach was improved even further by using squared correlations as weights (Table 3.3). Also, when effectors from *P. infestans* were included, several of them scored highly among the *P. sojae* effectors (Table 3.2).

Based on these strong performance metrics, the ranked lists of nodes from the Sinksource+ algorithm should form the basis for identification of previously unknown effectors, especially when combined with other biological information (described in detail in Chapter 5). It is possible that among the highly ranked nodes, there may be effector candidates that have RXLR-like motifs rather than strict RXLR motifs (e.g. those shown in Table 3.4).

The Sinksource algorithm has been successfully utilized in prediction of Human Immunodeficiency Virus (HIV) dependency factors (HDFs), which are human proteins that HIV requires for infection, but are not essential for human cell growth [44]. The authors combined multiple sources of protein-protein interaction data to build a network and used known examples of HDFs derived from three different large scale small interfering RNA studies in order to propagate information through the network in order to discover new HDFs [44]. The authors tested a variety of other algorithms from the literature, including the Hopfield network algorithm and the FunctionalFlow algorithm and compared the precision-recall curves from the different algorithms to evaluate their performance [44]. It would be interesting to test different algorithms such as the ones used in the HDFs study and to compare the outcomes for *P. sojae* effector predictions in the future.

The network-based prediction of *P. sojae* protein function has more improvements and benefits compared to other previous computational prediction methods introduced in the literature. A clustering analysis-driven prediction of yeast genes from microarray data reported success in predicting new members of multiple functional categories [69]. In that study, the authors mainly utilized different clustering methods, such as hierarchical clustering and k-means clustering, in order to assign confidence values to new proteins of unknown function according to the biological function of neighbors in the same cluster [69]. As discussed in Chapter 2, clustering methods for *P. sojae* effector data did not perform very well, while other machine learning algorithms improved the predictive performance significantly. Proteins that could potentially share the same function may not cluster together using the clustering methods mentioned above, but may be neighbors in a network, as observed in the *P. sojae* effector data. The approach described in this Chapter aimed to capture the local information present in the different parts of the network by creating the network from close neighbors of each node, rather than using a global threshold to select the members of the network. This approach proved useful because the effectors tend to form groups of more closely related members.

A significant limitation in the analysis presented here included the fact that in the *P. sojae* test data network, the number of seeds, which are derived from experimentally proven effectors, was very small compared to the large number of unknown nodes, resulting in a large number of nodes with very low local confidence scores (Figure 3.5). Since all RXLR-like motifs were included in the initial search of candidates from the *P. sojae* proteins, the number of unknown nodes in the network was always very large, diluting the evidence present in the network. As more validation experiments are performed in the future, there should be more proven effectors to guide network-driven predictions better. Also, this method of network based prediction is highly transferable to other classes of proteins or to studying protein functions of another organisms. This study provided a solid platform for predicting effector candidates of *P. sojae*, and experimental validation of some of these candidates is discussed in Chapter 5.

# Chapter 4

## Analysis of fungal effector cell-entry motifs using statistical methods

Authors: Hyunjin D. Choi, Chris Franck and Brett Tyler

### 4.1 Abstract

Understanding how plant pathogen molecules enter host cells has great potential to lead to new disease control measures. However, discovering the exact mechanisms involved has been a challenge. Recently, it has been shown that many plant pathogen oomycete effectors utilize RXLR and dEER motifs in cell entry. The presence of conserved motifs among cell-entering effector proteins in oomycetes suggested that cell-entering effectors in fungi may also share a motif. This chapter is focused on identifying potential cell entry motifs in effectors from *Magnaporthe oryzae*. The analysis used sequences of two sets of secreted *M. oryzae* proteins, one set validated to enter host cells and one set which failed validation. Well characterized *Phytophthora sojae* effector sequences were used as a training or validation set. In an attempt to identify potential motifs, many statistical methods, including well established algorithms such as "Multiple Em for Motif Elicitation" (MEME) motif search analysis and Markov clustering were applied. A new enumeration method also was developed for motif detection. No statistically significant motifs were found among the fungal sequences using any the methods we applied in this study, although in every case the known motifs in the oomycete sequences could be recovered. The methods applied and developed in this study could readily be extended to sequences of other species to detect motifs, if they are present in the dataset.

## 4.2 Background

The discovery of the N-terminal RXLR and dEER motifs in the effectors of *Phytophthora* species has been instrumental in identifying the members of this effector superfamily in these genomes [30]. Many of the Hidden Markov Model predicted RXLR effector candidates have been shown to bind to phosphatidylinositol 3-phosphate and to possess cell-entry capabilities. Some of these effector candidates have suppress or trigger programmed cell death in host plants [64]. The finding of entry motifs in oomycete effectors prompted an important question about the potential presence of similar motifs in effectors of fungi. Dr. Barbara Valent from Kansas State University kindly agreed to share a set of *Magnaporthe oryzae* protein sequences that experimentally showed cell entry capabilities as well as a set of non-entering sequences from the same experimental study. These sequences provided the basis for a search for one or more motifs common to the cell entering proteins, that were absent from the non-entering proteins.

Many previous motif discovery research projects focused on identifying recognizable patterns in DNA or protein sequences that regulate gene expression [15, 52, 55, 56]. Since then, the development of high throughput technologies has allowed larger scale gene expression studies. Researchers have analyzed transcription factors that target promoter or enhancer regions of co-regulated genes and identified statistically significant DNA binding sites using an enumerative algorithm [55]. Given a set of genes that have similar expression patterns, it may be possible to find a short motif upstream of the start site [15].

Motif discovery algorithms generally fall into three broad categories: enumeration methods, deterministic optimization, and probabilistic optimization [15]. Enumeration-based algorithms cover the entire motif search space by creating words with every possible combinations of amino acids or nucleotides. The advantage of this approach is that the algorithm is less likely to become trapped at a local minimum [15]. However, because the algorithm looks for exact word matches, it is generally inflexible and may not allow subtle variations that could be true hits [13]. Expectation maximization (EM) is an example of deterministic optimization, which iteratively computes maximum likelihood estimates of the unknown model parameters. MEME is a popular implementation of an EM algorithm, which calculates the probability that a segment of the sequence is generated by the motif model rather than the background model [15]. The statistical algorithm (called the MM algorithm) used by MEME software assumes a finite mixture model of two disparate subpopulations, one that gives rise to the motif, and the other to the rest of the sequences. Then, the EM algorithm is applied to estimate the parameters of the two component density and the mixing parameter [5]. An example of probabilistic optimization method is the Gibbs sampling method. This method has been used extensively in motif search algorithms, and software tools such as AlignACE, BioProspector and GibbsST are based on this method [13]. In this application Gibbs sampling also assumes that there are two separate probability distributions under each given sequence, one for the motif and the other for the background. The goal of the algorithm is to find the most similar pattern of a particular length among the given sequences by locating

an alignment that maximizes the ratio between the probability of the motif occurrence to the probability of the background frequencies [13]. The disadvantage of Gibbs sampling is that the algorithm sometimes may become trapped at a poor local maximum [13]. Given the stochastic nature of the Gibbs sampling method, sufficient number of iterations are required to ensure that the best fitting combinations of motif models can be found [15].

A more detailed description of expectation maximization is as follows; this algorithm, implemented in MEME, was used in this study to make a preliminary assessment of the fungal sequences. Given a set of sequences, maximum likelihood estimation is performed using the MM algorithm with the solutions for the parameters of the mixture model being those can best describe the dataset. The learns a finite mixture model that fits the data as closely as possible [5]. Since the finite mixture model found by the MM algorithm consists of two subpopulations, different parameter values are estimated for each of the two subpopulations. The motif model, which fits the motif instances in the sequences, is composed of independent samples drawn from a multinomial trial random variable, where parameters are estimated from the given sequence dataset [5]. Each position in the motif model has an independent probability associated with the occurrence of a particular letter. The background model, which fits the non-motif portions of the sequences, assumes that each position of the sequence is an independent sample from a single multinomial random distribution with different parameters from the motif model [5]. The MM algorithm can find more than one motif present in the given dataset by using an erasing factor for each letter, which is iteratively updated as the algorithm converges after finding a motif. These recurring erasing factor adjustments allow the algorithm to re-estimate the parameters of the finite mixture model and find new potential motif [5].

The study described in this chapter also evaluated an enumerative approach for potential motif discovery, combining previous motif knowledge from *Phytophthora sojae* effector studies with logistic regression as a tool for discriminating positives, such as cell-entering sequences, from negatives. Experiments on RXLR *Phytophthora sojae* effectors showed that histidine (H) or lysine (K), both of which have positively charged side chains like arginine (R), could functionally replace the first arginine of the RXLR motif [34]. Also, the third position in the motif, which is leucine (L), could be functionally replaced by any of the large hydrophobic residues, namely isoleucine (I), methionine (M), phenylalanine (F), tryptophan (W) or tyrosine (Y), while the fourth position, which is arginine (R), seemed very flexible [34]. Therefore, a search was conducted for RXLR-like motifs ([R|K|H], any, [L|I|M|F|Y|W]) as the core of potential motifs in *Magnaporthe oryzae* sequences. The analysis was also expanded to allow any of the 20 amino acids for the first position or for the third position (thus 400 possible individual motifs) to exhaust the motif search space. Each of the motifs in the motif search space was tested for significant association with the translocating capability of *Magnaporthe oryzae* effectors. As discussed in Chapter 2, logistic regression models allow appropriate regression analysis when the dependent variable is binary. The resulting probability predictions associated with each analyzed motif are constrained between zero and one. The motif search tested for the association of a particular motif would be associated with the

positive sequences (translocating proteins), together with the absence of the motif from the negative sequences (non-translocating proteins or permuted sequences) at the same time.

Markov Clustering (MCL) was also explored as a tool for discovering intrinsic structures among the sequences [21]. Kidera factors (as discussed in Chapter 1) were utilized to convert the protein sequences to real numbered vectors, and then Markov clustering was applied to identify clusters of sequences that shared regions of physico-chemical similarity. This method was applied to *Phytophthora sojae* effector sequences as well as the *Magnaporthe oryzae* sequences for comparison purposes. The MCL process relies on a graph composed of nodes (sequences) and edges with weights (similarity score between the sequences) and a square matrix which stores these weights [21]. Through an iterative rounds of matrix multiplication and matrix inflation, the algorithm continues processing the matrix until it converges, which means that there are no more changes in the matrix [21]. The inflation parameter controls the tightness of the clusters; a high inflation value causes the clusters to be small and closely related, while a low inflation value tends to give rise to larger clusters of less similar sequences.

This chapter describes a search for cell entry motifs in *Magnaporthe oryzae* effectors using the methods introduced above, utilizing a data set composed of experimentally validated cell-entering proteins and non-entering ones. The search was based on the hypothesis that such a motif may exist, by analogy with cell entry motifs found in oomycete effectors. The discovery of a cell entry motif in the effectors of a fungal species could lead to the discovery of new important classes of effector proteins and improved understanding of the pathogenic mechanisms that these species use to attack their hosts.

## 4.3 Implementation

### 4.3.1 Datasets

The study utilized sequences of 28 experimentally verified *Magnaporthe oryzae* proteins that translocate across the cell membrane and of 29 non-translocating proteins, which were provided by Dr. Barbara Valent at Kansas State University. SignalP was used to predict the cleavage site of signal peptides, which were removed prior to the sequence analysis. The translocating protein sequences were termed positives, while the non-translocating sequences were termed negatives. The positive sequences were permuted to create an additional set of random negatives, which would most likely be non-translocating if tested in an experimental setting. For comparison with a set of sequences with a known motif, 24 experimentally confirmed *P. sojae* RXLR effector protein sequences, were utilized together with permuted negatives generated from those sequences.

### 4.3.2 MEME motif analysis

The online implementation of the MEME algorithm was used [3] to perform the expectation maximization estimate sequence analysis [4] on the *P. sojae* and *M. oryzae* sequence data sets.

### 4.3.3 Simple correct call method

A simple correct call method was devised to find any motif associated with cell entry capability. The method identifies motifs that occur more often in the positive set compared to the negative set. The method searches looked for amino acid motifs in the form of (Unknown).(Any).(Unknown), where the (Unknown) positions could take any of the 20 amino acids. For example, if the first and the third positions took the amino acid Alanine (A), then the method would search for motifs in the form of A.X.A, where X would be any amino acid. This amino acid arrangement reduced the search space to 400 possible combinations. This motif format was selected based on knowledge of RXLR-like motifs (described above), which specify the first and third positions in the motif [34]. A score for each motif was calculated as follows. For a given motif and a dataset composed of positives and negatives, if the motif was contained in a given positive sequence or if the motif was not contained in a given negative sequence, a correct call was counted. The number of correct calls was then divided by the total number of sequences (positives and negatives combined) to produce a correct call score for a given motif. Using this scoring scheme, each of the 400 motifs was assigned a score and then the motifs were ranked based on the scores. High ranking motifs were examined for similarity to RXLR-like motifs and subjected to more rigorous statistical analysis.

### 4.3.4 Logistic regression analysis with a covariate

This approach was used to evaluate highly ranked motifs against the background distribution of amino acid residues. The approach can account for motifs that occur often because of a skewed amino acid composition. The method takes an enumerative approach to estimate the probability of a motif being associated with cell entry capability. Logistic regression was used since cell entry status is binary. A covariate term was added to the model, that estimates the probability of a given motif appearing in the sequence by chance. To calculate the value of this covariate, each sequence was permuted multiple times (for example, 1000 or 10000 times) and the frequency of the motif was determined among the permuted sequences. The logistic regression model is as follows:

$$\log\left(\frac{p(\text{cell entry})}{1 - p(\text{cell entry})}\right) = \text{Intercept} + p(\text{chance appearance}) + \text{motif presence}$$



The null hypothesis for this logistic regression model is that the presence of the motif is not associated with the probability of the given sequence having cell entry capability. When the p-value associated with motif presence is small using this model, it would mean that the motif is likely to be associated with cell entry after taking into consideration the actual amino acid composition of the effectors. After calculating the probability of association with entry for each of the 400 possible motifs, the top-ranked motifs based on this probability score were evaluated.

### 4.3.5 Markov Clustering

Twenty amino acids flanking every instance of arginine (R) were extracted from 24 *Phytophthora sojae* effector sequences and 24 permuted sequences, and the amino acids were converted into Kidera factors. Markov Clustering (MCL) was applied with inflation value of 2.0 for relatively tight clusters. Likewise, 20 amino acids flanking every instance of arginine (R), lysine (K) and histidine (H) were extracted from 28 positive sequences and 29 negative sequences of *Magnaporthe oryzae* and converted into Kidera factors and were subject to MCL algorithm with inflation value of 2.0.

Markov clustering was used to search for possible motifs in a way that could accommodate subsets of *M. oryzae* effectors that had different motifs. Using the previously described RxLR-like motifs as a guide, 20 amino acids flanking every instance of arginine (R), lysine (K) and histidine (H) were extracted from the 28 positive sequences and 29 negative sequences of *M. oryzae* sequences and converted into Kidera factors. For comparison, the 20 amino acids flanking every instance of arginine (R) were extracted from 24 *P. sojae* effector sequences and 24 permuted sequences, and the amino acids were converted into Kidera factors. In *P. sojae* the search was restricted to arginine as all the RxLR motifs started with that residue. In each case, MCL was applied with inflation value of 2.0 for relatively tight clusters.

## 4.4 Results

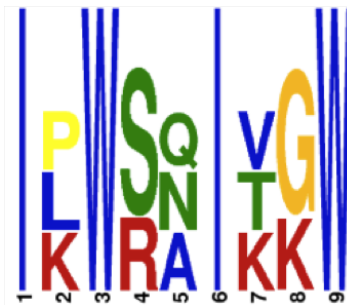
### 4.4.1 MEME Analysis

MEME was used initially to screen for *M. oryzae* effector cell entry motifs, focusing on the full length sequences from proteins that were positive for translocation (Figure 4.1). The only significant motif identified by MEME was the signal peptides (data not shown). The other motifs identified had large e-values, which suggested that they were not very significant (Figure 4.1). For comparison, MEME was applied to the *P. sojae* RxLR effector sequences, and as expected, the RxLR motif was found among almost all positive sequences with an e-value close to zero, as well as the C-terminal tryptophan (W) motif and the dEER motif (Figure 4.2). This result from *P. sojae* highlighted the fact that the motifs which MEME

identified from the *M. oryzae* sequences were most likely not significant. When the negative sequences from *M. oryzae* were included as input into MEME to contrast with the positive sequences, the results (Figure 4.3) were very similar to the results without the negative sequences. The motifs discovered by MEME were only shared by small number of positive sequences each (only 2 or 3) and the e-values were quite large.

Figure 4.1: Motifs found by MEME from 28 cell-entering *M. oryzae* sequences. Signal peptides were removed before analysis.

- (a) Only 3 sequences out of 28 shared this motif with a large e-value (2.9).



- (b) Only 2 sequences shared this short motif with a large e-value (6.1).



- (c) Only 3 sequences out of 28 contained this motif (e-value 6.2).

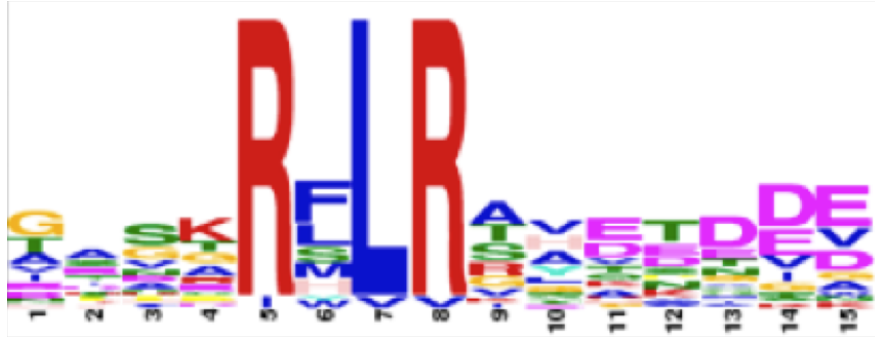


#### 4.4.2 Correct Call Analysis

The correct call analysis was first applied to the *P. sojae* effector sequences to establish a baseline for this new method (Figures 4.4 and 4.5). The purpose of developing this simple method was to examine if there were any short motifs highly associated with the positives, but not with the negatives. When the method was applied to the known effectors of *P. sojae*, the top ranking motifs contained expected motifs, such as S.X.R, R.X.L and F.X.R

Figure 4.2: Motifs found by MEME from 24 *P. sojae* RXLR effectors.

(a) 23 of the 25 effector sequences shared the RXLR motif with a very significant e-value of  $3.19 \times 10^{-19}$ .



(b) 19 of the 25 sequences also shared the known W motif with an e-value of 0.00018.



(c) 22 of the 25 sequences shared the dEER motif with an e-value of 1.2.



(from the RXLR motif). However many dipeptides that did not appear to be associated with known motifs were also highly ranked (Figure 4.4). When all the sequences were permuted (both positives and negatives), the expected motifs disappeared from the top ranks (Figure 4.5), but the scores of the highest ranked dipeptides were similar to the non-permuted sequences, suggesting the method lacked power. Next the correct call method was applied to the *M. oryzae* sequences. The results obtained when experimentally validated positives and negatives, were compared using the method are shown in Figure 4.6. Next, the positive sequences were permuted to create negatives with the same amino acid composition (Figure 4.7). Finally both the positives and negatives were permuted as a negative control (Figure

Figure 4.3: MEME discrimination analysis of 28 cell-entering *M. oryzae* proteins compared to 29 experimentally validated non-entering *M. oryzae* proteins.

(a) Only 3 sequences contained this motif with a large e-value of 2.8.



(b) Only 2 sequences shared this motif with a large e-value of 11.



(c) E-value was 13 and only 2 sequences contained this motif.

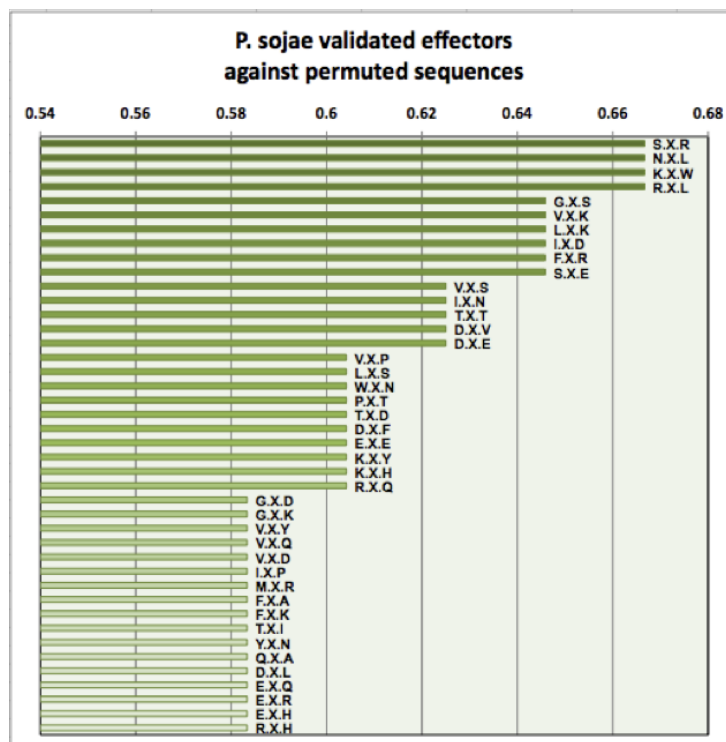


4.8). Dipeptides characteristic of RXLR-like motifs such as H.X.I, and R.X.V, were highly ranked among the unpermuted positives and low among the permuted sequences. However, their scores were lower than those of real motifs found among *P. sojae* effector sequences and upon further investigation (e.g. when their positions in the proteins were mapped), they did not seem to be highly significant (data not shown).

## Logistic Regression

Logistic regression analysis with a covariate was first applied to a *P. sojae* data set to validate the approach. The expected motifs, such as E.X.R ( $p < 0.001$ ), R.X.L ( $p < 0.01$ ) and F.X.R ( $p < 0.02$ ) were identified with high significance (Figure 4.9). The p-values were established by repeated permutation of the effectors sequences followed re-running of the algorithm. The number of permutations (1000 or 10000) for calculating the covariate value did not change the p-value associated with each motif significantly, suggesting that 1000 permutations was sufficient for calculating the covariate (not shown). The results from the

Figure 4.4: Correct call analysis of *P. sojae* RXLR effectors. 24 validated effector sequences were used as positives, and permutations of those 24 sequences were used as negatives. Top ranking motifs are shown. Known motifs such as R.X.L , F.X.R and D.X.E are among the highest ranked.

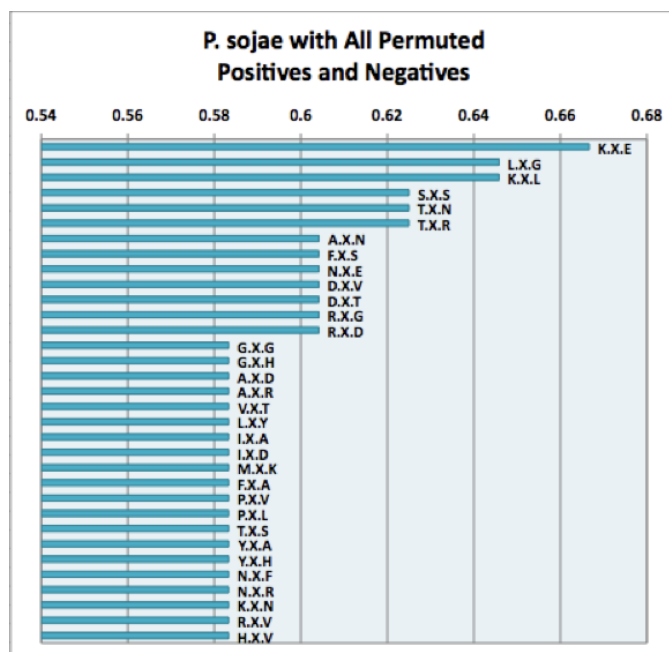


same logistic regression analysis for *M. oryzae* sequence dataset are shown in Figure 4.10. However, the p-values for the top-ranked motifs were larger compared to those from *P. sojae*. Furthermore, the top-ranked *M. oryzae* motifs did not shown a consistent position within the sequences. The amino acid percent compositions of the positive sequences from both *P. sojae* and *M. oryzae*, were examined to determine if the motifs with small p-values are simply ones containing frequently occurring amino acid residues (Figure 4.11), but there were no major associations.

### 4.4.3 Markov Clustering

Markov clustering results from *P. sojae* effectors and permuted sequences revealed that among the sequences flanking arginine (R) residues, the MCL algorithm could correctly identify clusters of RXLR motifs from functional effectors (Table 4.1). Next, the approach was applied to *M. oryzae* sequences, either focusing on the sequences flanking positively-charged residues (R, K, H) (Tables 4.2) or large hydrophobic residues (L, M, I, F, Y, W)

Figure 4.5: Correct call analysis of permuted positive and negative *P. sojae* effector sequences. Recognizable motifs disappeared from the top ranking motifs compared to Figure 4.4.



(Table 4.3). Those residues were chosen as they corresponded to the two main elements of the RXLR-like motif. Only one cluster centered on positively-charged motifs was enriched in non-permuted sequences. It contains five effector sequences with the short motif [R|K|H].G. Its significance is not clear. In three of the sequences, the motif occurred near the N-terminus. The top-ranked clusters centered on the large hydrophobic residues contained several common extended motifs, e.g. G.A.[L|I|F].A.G in cluster 1 and L.A.[L|M].[L|V].P in cluster 2 and [A|T].[T|A].[L|F].[A|S|T].[L|M] in cluster 3. However, all of the latter three motifs were derived from the signal peptide.

## 4.5 Discussion

Discussion Many of the statistical methods employed in this study have been routinely applied to motif searches in DNA or protein sequences. For example, MEME motif searches have become a standard motif search tool in transcription factor binding site studies [33,62]. Valouev et. al. [62] produced CHIP-seq data and determined the location of DNA binding to protein complexes, then successfully used MEME to identify several specific DNA sequence motifs where the transcription factors were binding. In a similar study, Jothi et. al. [33]

Table 4.1: Top 3 clusters from Markov clustering, using 24 *P. sojae* effector sequences. 10 amino acids flanking each side of each arginine (R) were included in the analysis. The 6 digit numbers represent effector IDs, the next numeral shows the order of arginine occurrence in the sequence, the numeral and letter shows the position of the arginine.

CLUSTER 1	CLUSTER 2	CLUSTER 3
Psojae-Avr1b-108861-4-41-R	Psojae-Avr1b-108861-5-44-R	Psojae-Avr3a-159064-159325-2-35-R
Psojae-Avh52-159037-1-39-R	Psojae-Avh238-159194-1-5-R	Psojae-Avr3c-159015-3-34-R
Psojae-Avh94-159065-2-36-R	Psojae-Avh52-159037-2-42-R	Psojae-Avr46-159127-4-42-R
Psojae-Avh240-159196-1-16-R	Psojae-Avh172-159128-3-39-R	Psojae-Avh6-158994-2-40-R
Psojae-Avh240-159196-2-27-R	Psojae-Avh94-159065-3-39-R	Psojae-Avh238-159194-4-36-R
Psojae-Avh23-159011-1-43-R	Psojae-Avh23-159011-2-46-R	Psojae-Avh8-158999-4-39-R
Psojae-Avh109-159076-5-100-R	Psojae-Avh7b-158996-158997-2-43-R	Psojae-Avh29-159017-2-49-R
Psojae-Avh180-159136-2-39-R	Psojae-Avr1a-159231-Permuted-5-96-R	Psojae-Avr3c-159015-Permuted-15-197-R
Psojae-Avh7b-158996-158997-1-40-R	Psojae-Avh29-159017-Permuted-2-12-R	

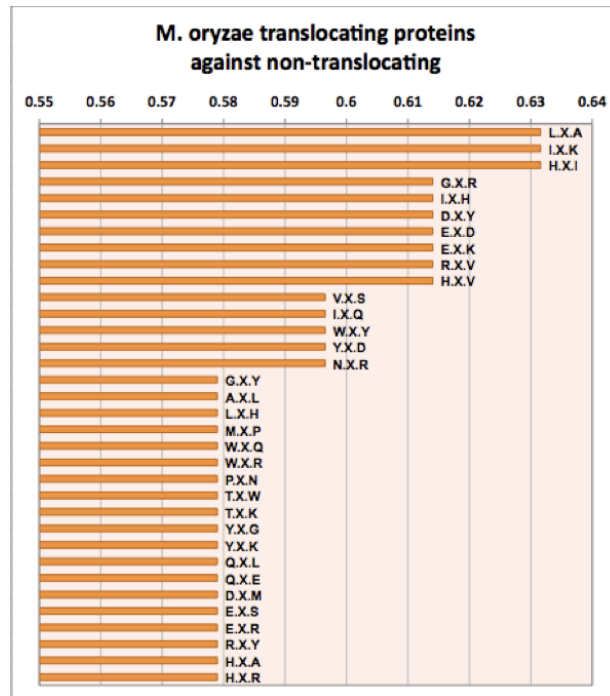
Table 4.2: Top 3 clusters from Markov clustering, using full length *M. oryzae* sequences (including signal peptide) which are known to translocate across the cell membrane. 10 amino acids flanking each lysine (K), histidine (H) or arginine (R) were included in the analysis. The letter represents the effector ID, the next numeral shows the order of K, H or R occurrence in the sequence, the numeral shows the position of the K, H or R, and the final letters indicate the 5 residue motif centered on K, H or R.

CLUSTER 1	CLUSTER 2	CLUSTER 3
Moryzae-F-3-69-KDKGG	Moryzae-G-Permuted-2-50-TVKRR	Moryzae-A-Permuted-8-114-TGKGG
Moryzae-E-1-42-VAHGQ	Moryzae-B-Permuted-1-74-TAHRQ	Moryzae-I-5-79-RGRGG
Moryzae-D-8-116-SARGQ	Moryzae-D-6-92-TGRRV	Moryzae-I-12-121-HRRGG
Moryzae-A-1-56-AKRQK	Moryzae-I-13-132-TGRRQ	Moryzae-D-Permuted-2-65-APRSG
Moryzae-C-11-131-PGRGS	Moryzae-H-Permuted-3-96-GGRRN	
Moryzae-D-Permuted-3-78-FLRGG		

Table 4.3: Top three clusters from *M. oryzae* Markov Clustering centered on large hydrophobic residues. 10 amino acids flanking each leucine (L), isoleucine (I), methionine (M), phenylalanine (F), tyrosine (Y), or tryptophan (W) were included in the analysis. The letter represents the effector ID, the next numeral shows the order of L, I, M, F, Y or W occurrence in the sequence, the next numeral shows the position of the L, I, M, F, Y or W, and the final letters indicate the 5 residue motif centered on L, I, M, F, Y or W.

CLUSTER 1	CLUSTER 2	CLUSTER 3
Moryzae-K-3-21-GALAA	Moryzae-S-2-15-LSLAP	Moryzae-L-1-14-AALAM
Moryzae-V-4-21-GALAG	Moryzae-K-2-15-AALVP	Moryzae-S-1-13-ATLSL
Moryzae-R-4-21-GILAG	Moryzae-V-2-15-LALLP	Moryzae-V-1-13-ATLAL
Moryzae-Q-2-66-GVIQG	Moryzae-R-2-15-LALLP	Moryzae-R-1-13-TALAL
Moryzae-T-1-16-GAIAG	Moryzae-K-Permuted-6-32-FGLGP	Moryzae-P-1-16-ATFTA
Moryzae-U-Permuted-5-130-EKIKE	Moryzae-J-Permuted-4-30-KSIAD	Moryzae-O-Permuted-2-132-ALFQP
Moryzae-S-1-21-GAFAG	Moryzae-L-1-16-LAMLP	
Moryzae-M-Permuted-4-110-GIFGP	Moryzae-Q-3-60-FDYRP	
Moryzae-V-Permuted-2-49-GEWGA		

Figure 4.6: Correct call analysis of 28 translocating *M. oryzae* sequences, compared with 29 non-translocating sequences as negatives.



developed a new method to detect DNA binding sites of human transcription factors using CHIP-seq and utilized MEME to confirm that a large majority of the motifs they found were previously known DNA binding site motifs. By combining a wet lab method (CHIP-seq) and a computational approach (MEME motif analysis), both studies showed that molecular interactions could be linked to sequence patterns.

The search of *M. oryzae* sequences did not yield any convincing motifs by any method that had a statistical significance comparable to those found in oomycetes. The only motifs found were those associated with the signal peptides of the *M. oryzae* proteins. The small set of sequences we analyzed was hand-picked, experimentally validated and studied. The question of motif presence still stands open among other sequences that we have not included in our analysis. There are yet other methods that could be tried, such as Gibbs sampling based methods, which rely on probabilistic optimization rather than a deterministic one such as MEME algorithm [13]. Most likely however, if there are motifs to be found, a much larger set of experimentally validated positives and negatives will be needed. On the other hand, in the case of *M. oryzae*, it is possible that cell entry is not determined by the amino acid sequence of the effectors, but by the promoters of their genes (B. Valent, personal communication).

In a recent review of 13 different motif finding algorithms, it was found that enumerative methods they evaluated performed extremely well among sequences with known motifs [15].



Figure 4.7: Correct call analysis of 28 *M. oryzae* translocating proteins as positives and permutations of those sequences as negatives.

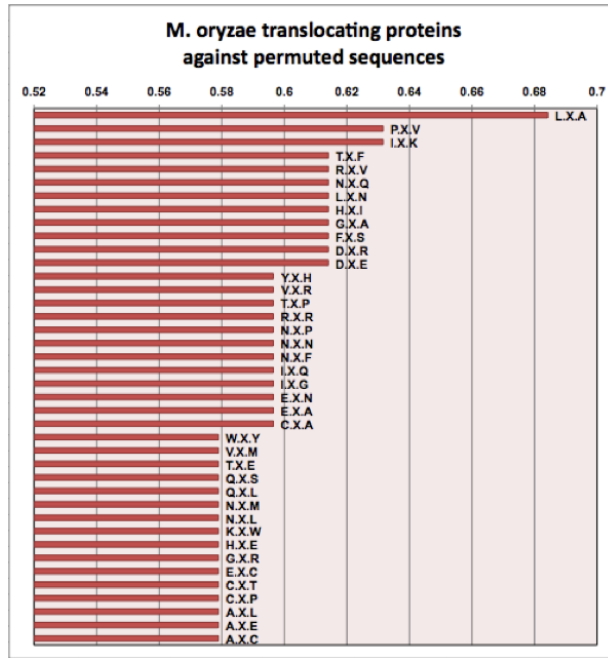


Figure 4.8: Correct call analysis of permuted positive and negative *M. oryzae* sequences.

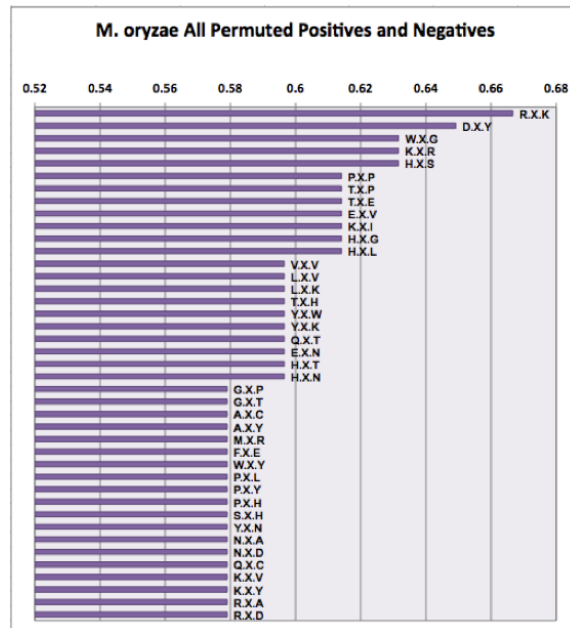


Figure 4.9: Logistic regression analysis of *P. sojae* effector sequences as positives and permuted sequences as negatives using an amino acid composition covariate. 10000 permutations were used to determine the p-values.

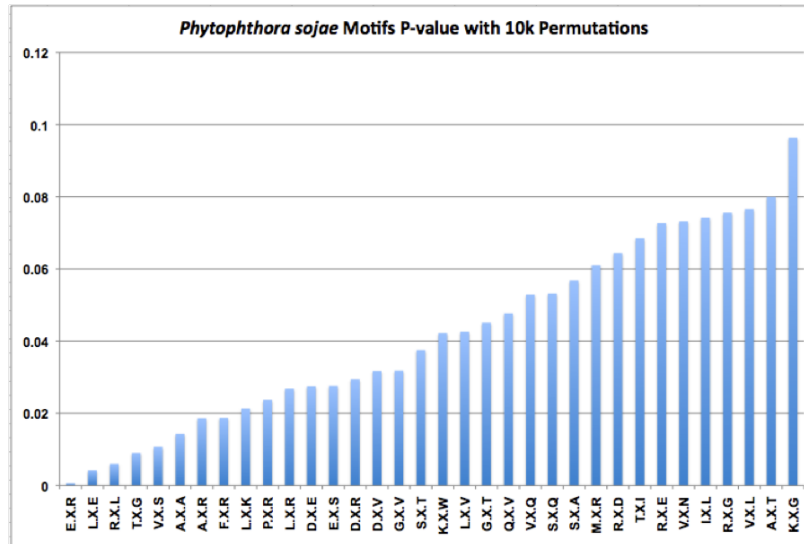
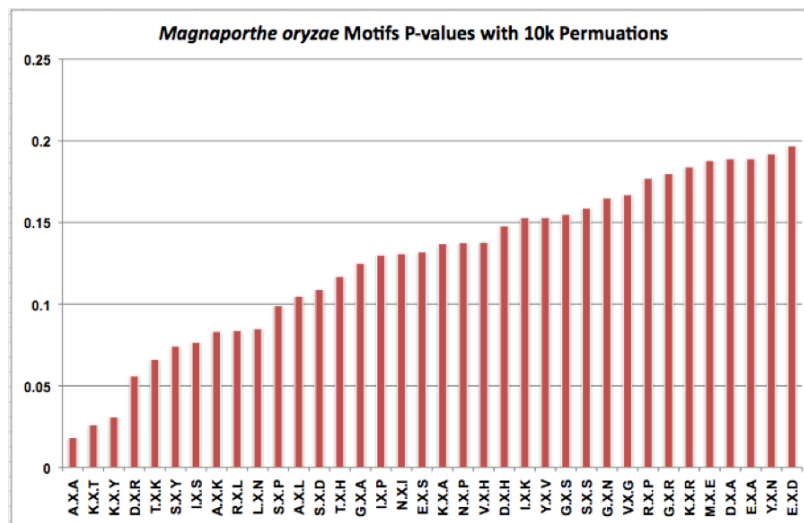
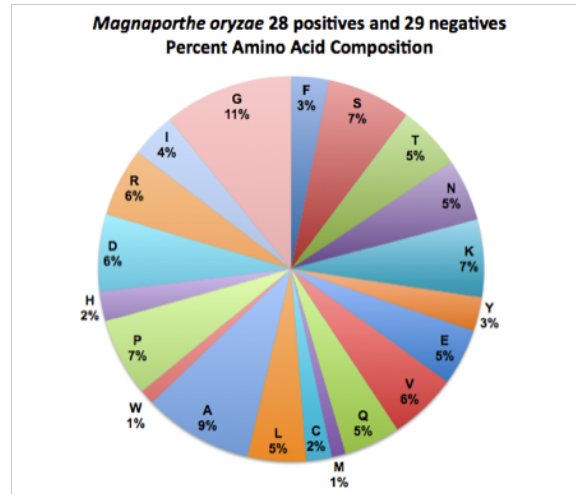


Figure 4.10: Logistic regression analysis of *M. oryzae* translocating sequences as positives and non-translocating sequences as negatives using an amino acid composition covariate. 10000 permutations were used to determine the p-values.

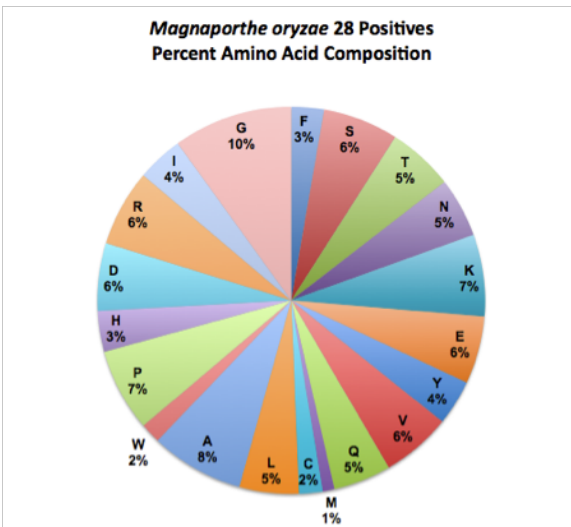


The authors also found that different algorithms encompassed a subset of the known binding sites (in the case of transcription factors) without much overlap [15]. Therefore, it appeared

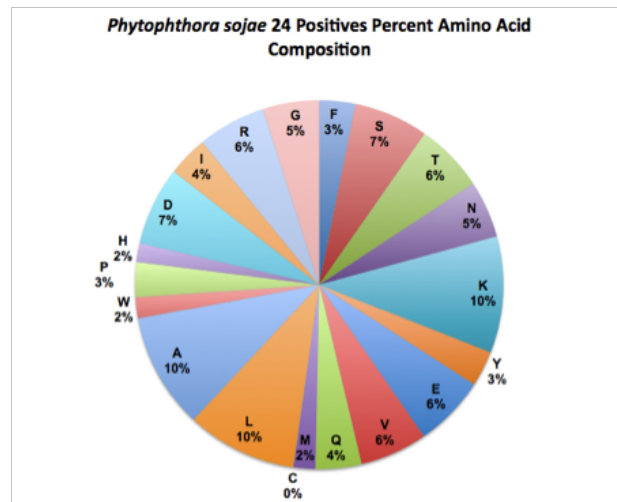
Figure 4.11: Amino acid composition of datasets used in the analysis.



(a) *M. oryzae* 28 translocating positives and 29 non-translocating negatives.



(b) *M. oryzae* 28 translocating positives only.



(c) *P. sojae* 24 RXLR effectors only.

advisable and beneficial to combine results from multiple different motif algorithms [15].

In this chapter, the data set was analyzed with many different approaches. We formulated new statistical methods for short motif discovery because existing methods such as MEME perform better for longer motifs. Also, no method precisely reflected the previous knowledge gained from identifying *P. sojae* effector motifs. If a motif is very short, MEME would have difficulty estimating the parameters for the motif model.

This study would certainly benefit from using a larger dataset from *M. oryzae* with experimental results. The negative results found here for *M. oryzae* do not necessarily provide insights on the presence or absence of cell-entry motifs from other fungal species. Fungi employ a diverse set of strategies to colonize their hosts — some enter directly into the plant epidermal cells, while others penetrate between the intercellular spaces [32]. Some use haustoria while others do not. Fungi have evolved separately for a longer period of time than oomycetes and thus fungal symbionts may have evolved diverse mechanisms for delivery of effectors. Also, fungi are much more diverse in their strategy to penetrate the host cells and in their molecular mechanisms to suppress the hosts defense mechanisms.

# Chapter 5

## Identification and functional testing of novel effector candidates from *Phytophthora sojae*

Authors: Hyunjin D. Choi and Brett Tyler

### 5.1 Abstract

Understanding how molecules released from plant pathogens enter host cells is essential to increasing crop production yield and to securing food supply. Plant pathogens, such as oomycetes and fungi, release effector proteins that enter host cells. Chapters 2 and 3 presented computational methods for predicting effectors from *Phytophthora sojae* using flanking sequences of RXLR-like motifs and explored the potential for applying similar approaches in fungi. This chapter describes detailed biological criteria for selecting a small number effector candidates for experimental testing from a set of computational predictions. The chapter also describes use of a double barrel particle bombardment system to experimentally test one candidate, 133851, in soybean leaves. Statistically significant evidence for cell entry was observed for this candidate.

### 5.2 Background

Since large scale genomic studies have become routine for over a decade, there has been a lot of effort to interpret these data to produce specific hypotheses that can be tested and validated regarding gene or protein function. It is simply impossible to test the functional

activities of all the genes of a sequenced genome. Historically, gene or protein function prediction would involve extensive molecular biology experimentation, unless there were well characterized homologues [42]. However, many new computational strategies for gene or protein function prediction have been introduced in the recent years. In Chapters 2 and 3, I discussed the use of machine learning approaches and network analysis to predict protein function. The protein function prediction efforts presented in this thesis have focused on a specific class of proteins called effectors produced by the oomycete plant pathogen, *Phytophthora sojae*. Over 400 members of this protein family have been predicted in *Phytophthora sojae* and many of them have been experimentally validated [30, 64].

Despite the advancements in computational gene or protein function prediction, experimental validation remains essential. Computational methods necessarily produce false positive predictions. Furthermore, confounding biological factors may produce erroneous predictions. For example, a strong prediction that a protein contains a signal peptide for export outside of the cell may be confounded if the protein is localized on the membrane. This chapter presents additional criteria for identifying good candidates for experimental testing following computation predictions.

One useful technique for validating cell entry activity by candidate effectors is the double-barrel gene gun bombardment assay. The double-barrel gene gun allows two samples of DNA to be shot into a plant leaf side by side simultaneously, enhancing the reproducibility of the results [16]. When plasmids containing foreign DNA are shot into a plant leaf using a gene gun, the plant tissue is transformed and expresses the plasmid construct. This system can be utilized to evaluate a potential interaction between plasmid DNA and proteins that are naturally present inside the plant cells. We can also shoot multiple DNA samples and expect to have them all expressed in the plant and observe how they interact. For example, when a plasmid DNA containing intact full length Avr1b mixed with  $\beta$ -glucuronidase (GUS) is bombarded into soybean leaves that express Rps1b, the leaves will show signs of hypersensitive defense response [17].

## 5.3 Methods

### 5.3.1 Candidate selection criteria

The first step in filtering the effector candidates was based on the confidence scores from GAIN analysis. As discussed in Chapter 3, a functional linkage network was utilized to identify potential effector candidates. The network was composed of more than 25,000 nodes which represent candidate RXLR-like motifs found within predicted *P. sojae* secreted proteins. Each motif received a vector of Kidera scores from the 20 amino acids flanking the motif. The edges represent correlation measurements between the vectors of each node. For each node, only the top 20 best correlated neighbors were considered, rather than neighbors

with edges over a fixed threshold. The GAIN algorithm was initiated with about 25 seeds derived from experimentally validated functional RXLR effector motifs. After the GAIN algorithm was applied, each node in the network received a confidence score (except the seeds), which was used to rank each node in the network. The 150 nodes with the highest confidence scores were considered for further examination and potential experimental validation.

Although the proteins in the network had been selected on the basis of a positive secretion prediction, the latest version of the SignalP prediction algorithm (version 3, HMM version) [6] was used to eliminate any candidates with a score less than 0.9. A Kyte-Doolittle hydropathy plot and the TMHMM algorithm were used to predict the presence of transmembrane domains. If a protein is predicted to have one or more significant membrane-spanning domains, then there is a lesser chance that the protein would be transported outside the cell. By taking into consideration the hydrophobicity and hydrophilicity of each amino acid in a protein, a Kyte-Doolittle hydropathy plot takes a moving-segment approach across a protein and calculates the average hydropathy within each segment of a given length, and displays the consecutive scores along a given protein [39]. These plots can identify potential membrane-crossing segments of a protein and provide a quick method to qualitatively disqualify a protein from being an effector candidate. However, these are also computational predictions for transmembrane domains, and so these predictions were carefully combined with other information about each candidate.

The length of the protein was also used as a selection criterion. The median length of the RXLR effectors is 126 amino acids, while the length ranges from 28 up to 835. However, most of the RXLR effectors tend to be on the shorter side, often less than 200 amino acids. Therefore, the candidate list was further prioritized based on the length of the proteins and with a low priority as potential effectors given to proteins of length greater than 600 amino acids. The position of the RXLR-like motif was also an important consideration. For the majority of RXLR effectors, the RXLR motif occurs near the N-terminus of the protein. The median position of the RXLR motifs is 27 amino acids from the signal peptide, while the motif position can range from 10 up to 67. Proteins whose motif position was closer to the N-terminus were ranked more highly in the context of other information available about each candidate. The presence of introns was also considered, since the majority of validated RXLR genes do not contain introns. Also, if a candidate contained an intron or multiple introns, a more complex experimental strategy for cloning those effector candidate genes would be needed.

Another aspect considered was whether the candidates have homologues in other *Phytophthora* species. Many RXLR effectors, with the exception of the RXLR motif segment, are not very well conserved across different species because of their rapid evolution rate. High conservation across multiple species could imply that the protein in question would most likely participate in basic functions related to viability of the cells, such as metabolism or growth. Thus, candidates which were highly conserved in other *Phytophthora* species, such as *P. infestans*, *P. ramorum*, *P. capsici* and *P. parasitica* were given a reduced priority. On the other hand, since the RXLR motif itself is conserved across different *Phytophthora*

Table 5.1: Examples of high scoring nodes with GAIN confidence scores. The six digit numbers are VMD IDs of *P. sojae* genes, then the next digit represents the order of RXLR-like motif occurrence in the translated protein, and the last number represents the starting position of the RXLR-like motif within the protein.

Rank	Candidate IDs	GAIN confidence score
1	158998-1-40-RMLR	0.402451
2	139921-1-40-RMLR	0.40245
3	158995-1-40-RMLR	0.359449
4	134429-1-40-RMLR	0.34024
5	127824-3-43-RMLR	0.309714
6	159010-3-43-RMLR	0.300865
7	159009-1-39-RMLR	0.29572
8	159032-1-36-RMLR	0.278627
9	159063-1-39-RQLR	0.276598
10	158991-2-41-RFLR	0.262376
11	109104-2-41-RFLR	0.253795
12	159116-2-95-RHLR	0.253113
13	137978-7-254-KHIE	0.243321
14	159002-1-84-RMLR	0.241977
15	159275-2-77-RMLR	0.230545
16	159038-3-65-RLLR	0.213157
17	133987-10-389-RRIQ	0.204459
18	131783-25-602-RMLV	0.202774
19	159035-1-51-RLLR	0.198321
20	134001-1-39-RLLR	0.195346

species, candidates that contained conserved RXLR-like motifs was given elevated priority.

The last piece of information used for prioritization was RNA-seq data on the levels of *P. sojae* transcripts in cultured mycelia and during infection. Since effector genes are likely to be more highly expressed during infection, a higher priority was given to candidates that had a higher expression level during infection.

### 5.3.2 Cell entry experiment design using gene gun

A double barrel gene gun was used for shooting DNA samples into soybean leaves following a detailed published protocol [35]. Thanks to its unique design, the double barrel gene gun allows two samples to be shot at once, a control sample in one barrel and a test sample in the other, significantly reducing the number of replicate sets required for statistical testing. A reporter gene,  $\beta$ -glucuronidase (GUS), is used which produces a small greenish blue spot when expressed in a living cell. DNA carrying the GUS gene is mixed in both a control sample as well as a test sample. If the test sample contains DNA of an effector gene which interacts



with a resistance gene present in the leaf, fewer cells express GUS because of programmed cell death, resulting in fewer blue spots compared to the control.

Experimental validation of selected effector candidates involved two DNA samples. The first consisted of an empty vector mixed with GUS which served as a control in all the bombardments. The second, experimental, construct encoded a fusion protein consisting of the effector candidate with its signal peptide linked to the C-terminus of Avr1b; this was also mixed with GUS DNA. Two different genotypes of soybean leaves were chosen as the targets for the bombardments, namely cultivar Williams which lacks the Rps1b gene that interacts with Avr1b, and L77-1863 which expresses Rps1b and has the same genetic background as Williams. The empty vector control would produce a consistently high number of blue spots on both cultivars, because no interaction between an effector and a resistance gene is expected. For the effector candidate, if the protein was able to carry the Avr1b portion of the fusion into the cell, then a reduced number of blue spots would be observed on L77-1863 leaves, due to cell death triggered by the Avr1b-Rps1b interaction.

### 5.3.3 Plasmid construction

#### A. Cloning Avr1b C-terminus for fusion constructs

In order to attach the Avr1b C-terminus to the candidate protein to trigger programmed cell death in the event of cell entry mediated by the candidate, the Avr1b C-terminus was cloned out of pUCmAvr1b, using gttgtctagaACCTTCAGCGTACTGACC as forward primer including an XbaI site, and cgttgtaccTCAGCTCTGATACCGGTGA as reverse primer with an KpnI site.

#### B. Cloning the candidates into pUC19 vectors

Candidate genes were cloned from purified genomic DNA of *P. sojae* (P6497; race 2). For candidate 133851, forward primer sequence was tatacccgggATGCAGCTCCTCTCAACGA with a terminal XmaI site, and reverse primer sequence was gcgctctagaCGTCGACGAGGTCTGCG with a terminal XbaI site to facilitate cloning into pUC19.

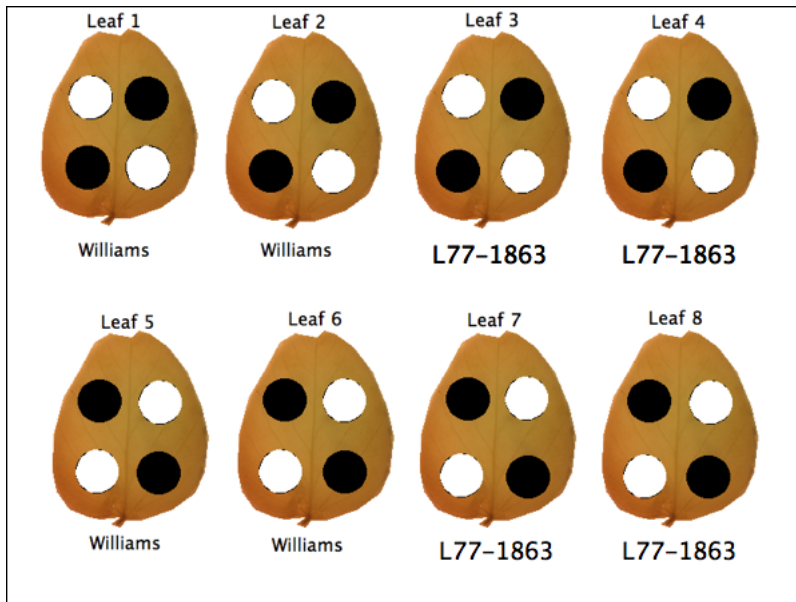
#### C. Preparation of plasmid DNA

The Qiagen Plasmid Maxi Kit was used to prepare plasmid DNA from the clones with correct inserts, using the manufacturers protocol.

Table 5.2: Expected results from double barrel gene gun bombardment. PCD stands for programmed cell death. Rps1b in capital letters indicate L77-1863, and in lower case letters Williams.

Experiment Type	Reagents	RPS1B leaves	rps leaves
Control	Empty Vector + GUS	More blue spots (No PCD)	More blue spots (No PCD)
Experimental	Fusion Plasmid + GUS	Less blue spots? (PCD)	More blue spots (No PCD)

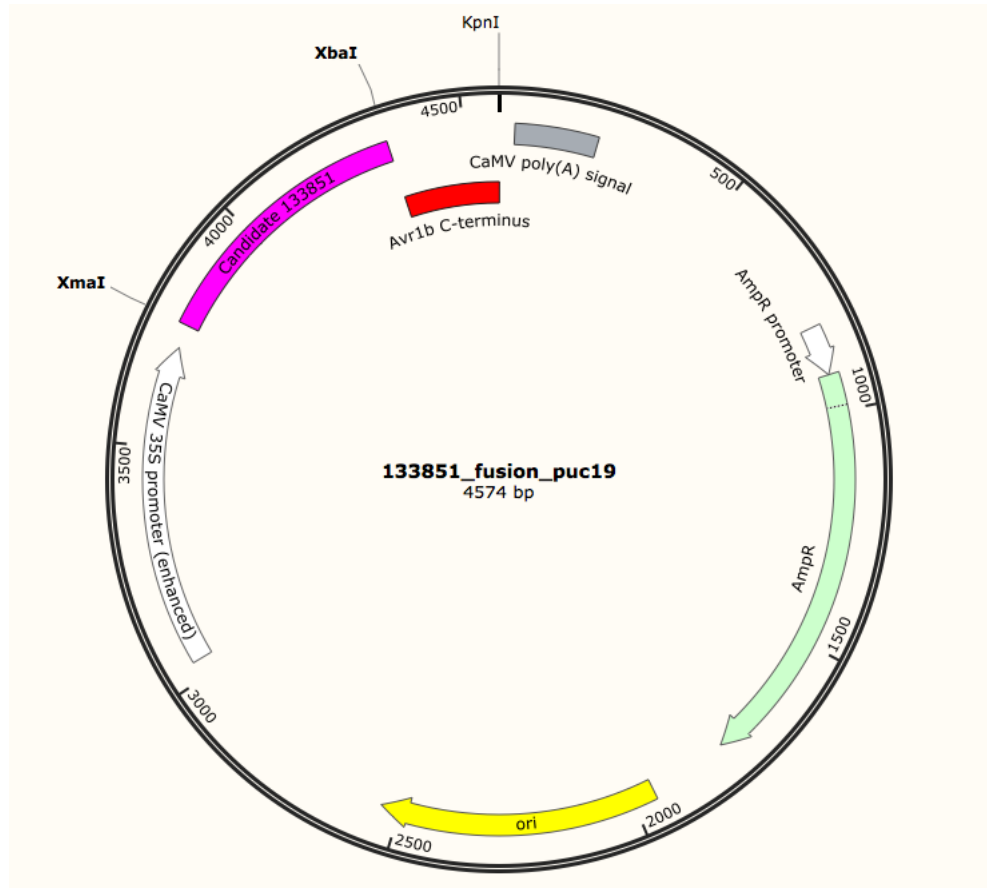
Figure 5.1: Double barrel gene gun leaf bombardment scheme. This layout is adapted from Kale and Tyler [35]. Black dot: test sample, white dot: control sample.



### 5.3.4 Gene Gun Bombardment Assays

Williams and L77-1863 soybean seeds were planted with 5 seeds per pot, then put into a growth chamber (12 hours of day at 28C and 12 hours of night at 25C) for approximately 7 days, and watered daily. By the time trifoliolate leaves of less than 1 inch appeared on the plants, the primary leaves were ready for bombardment.

Figure 5.2: Schematic diagram of cloning strategy.



To prepare the plasmids for gene gun bombardment, plasmid DNA was purified and mixed with water and other reagents, such as tungsten powder, which provides the particles to carry the DNA, and spermidine, which functions as an adhesive between the DNA and the tungsten powder, according to the protocol for the double barrel gene gun [35]. A vacuum centrifuge was used to achieve high enough concentration of the DNA samples before mixing with water and GUS. Soybean leaves were shot using the double barrel gene gun, following the layout displayed in Figure 5.1.

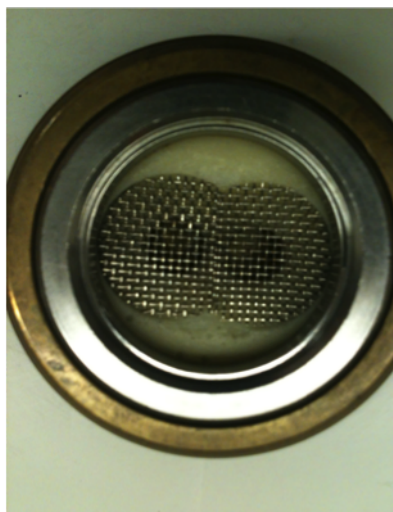
## 5.4 Results

The top-ranked candidate was protein 133851. The protein was relatively short (190 aa), the RXLR-like motif (KLLR) was located at position 24 and the gene was highly expressed during infection. There was a good SignalP prediction (probability 1) and no transmembrane

Figure 5.3: Williams and L77-1863 soybean plants ready for bombardment. It was very important to grow the plants until they were the correct size for shooting. If they were too young and tender, they would not be thick enough and holes would be made after shooting the DNA samples. If they were too old and overgrown, they would not express GUS efficiently.



Figure 5.4: Double barrel gene gun. DNA samples are projected down each barrel through the mesh screens.



domain was predicted. The KLLR motif was well conserved in *Phytophthora ramorum*, while the rest of the protein was not well conserved.

Another candidate considered was 133896. The length of the protein (187 amino acids) and the motif position (45) fit within the range of a typical effector profile. Hydropathy plots and HMM-based transmembrane prediction supported that this protein does not have transmembrane domains, and the SignalP probability was 0.997. The protein showed high similarity to other proteins which are highly expressed during infection, although the gene itself was not highly expressed during infection. However, because candidate 133896 contained an intron, cloning this candidate involved extracting RNA from *P. sojae* mycelium and creating cDNA from RNA. Due to the low expression level, several attempts to clone this gene failed, and validation was focused on the other candidate, 133851.

As described above, double barrel bombardments involved a control in one barrel, and the experimental construct (effector candidate 133851 fused to the C-terminus of Avr1b) in the other barrel. Four sets of experiments were performed, each set following the scheme shown in Figure 5.1. The number of blue spots observed revealed that there were fewer blue spots with the experimental construct compared to the control on L77-1863 leaves (Figure 5.5) while the control and the experimental construct produced comparable numbers of blue spots on Williams leaves in which Rps1b is absent (Figure 5.5).

The differences between Williams and L77-1863 were evaluated using the Wilcoxon rank sum test. This test is a nonparametric test, and does not depend on any underlying distribution normality in the data is not assumed. Also, this test is highly effective when the sample size is fairly small, because the ranks of the data are used for calculating the test statistic rather than the sample means based measures. The raw count of blue spots from the 4 sets of experiments are shown in Table 5.3. The test was conducted on the log ratio of the number of spots produced in the presence of the test construct compared to the control from the same bombardment ( $\log[(1+\text{test})/(1+\text{control})]$ ). The null hypothesis in our statistical analysis was that there is no difference in the ratio of blue spots between L77-1863 and Williams leaves. The mean and the standard deviation of the Wilcoxon rank sum test statistic were calculated and the standardized Z score was calculated (Equation 5.1).  $n_1$  was the number of observed ratios (fusion/control) for L77 leaves and  $n_2$  was the number of observed ratios (fusion/control) for L77-1863 leaves and  $n_2$  was for Williams leaves ( $n_1 = 29, n_2 = 31$ ). Based on the Z score, Wilcoxon rank sum test p-value was less than 0.00001, the null hypothesis was rejected with high confidence. Thus there was a statistically significant evidence that the 133851 fusion construct produced a smaller ratio of blue spots when bombarded into L77-1863 leaves compared to Williams leaves, which in turn supports the prediction that 133851 may be able to enter plant cells.

Table 5.3: Number of blue spots counted for all four sets of bombardments. Each bombardment scheme followed the diagram shown in Figure 5.1, using 4 leaves of L77-1863 and 4 leaves of Williams. NA indicates unavailable data due to technical difficulties, such as a torn portion of a leaf.

1st Experiment							
L77		Williams		Log Ratio		Rank	
Fusion+GUS	Control	Fusion+GUS	Control	L77	Williams	L77	Williams
83	163	113	140	-0.669	-0.213	42	23
91	165	172	117	-0.590	0.383	36	8
141	93	62	181	0.413	-1.061	6	51
71	137	84	178	-0.651	-0.745	40	46
0	30	14	21	-3.434	-0.383	59	27
15	46	NA	NA	-1.078	NA	53	NA
0	39	53	97	-3.689	-0.596	60	37
3	11	10	18	-1.099	-0.547	54	33
2nd Experiment							
L77		Williams		Log Ratio		Rank	
Fusion+GUS	Control	Fusion+GUS	Control	L77	Williams	L77	Williams
91	156	129	193	-0.534	-0.400	31	28
258	316	69	59	-0.202	0.154	22	11
87	191	197	85	-0.780	0.834	47	4
117	241	162	198	-0.718	-0.200	44	21
76	154	231	110	-0.700	0.737	43	5
79	133	142	178	-0.516	-0.225	30	24
127	304	281	357	-0.868	-0.239	49	25
160	298	239	204	-0.619	0.158	39	10
3rd Experiments							
L77		Williams		Log Ratio		Rank	
Fusion+GUS	Control	Fusion+GUS	Control	L77	Williams	L77	Williams
44	86	214	367	-0.659	-0.537	41	32
109	189	NA	213	-0.547	NA	33	NA
79	243	60	111	-1.115	-0.608	55	38
11	154	141	135	-2.559	0.043	58	14
32	117	79	102	-1.274	-0.253	57	26
55	163	94	40	-1.075	0.840	52	3
100	179	189	185	-0.578	0.021	35	16
NA	106	135	91	NA	0.391	NA	7
4th Experiment							
L77		Williams		Log Ratio		Rank	
Fusion+GUS	Control	Fusion+GUS	Control	L77	Williams	L77	Williams
85	74	237	195	0.137	0.194	12	9
102	109	373	379	-0.066	-0.016	19	17
NA	NA	234	264	NA	-0.120	NA	20
NA	NA	133	219	NA	-0.496	NA	29
168	163	52	124	0.030	-0.858	15	48
103	327	141	53	-1.149	0.967	56	1
NA	128	162	150	NA	0.076	NA	13
95	197	107	41	-0.724	0.944	45	2

$$\begin{aligned}
 E(W) &= \frac{n_1(n_1 + n_2 + 1)}{2} \\
 Var(W) &= \frac{n_1 n_2 (n_1 + n_2 + 1)}{12} \\
 Z &= \frac{W_{obs} - E(W)}{\sqrt{Var(W)}}
 \end{aligned}
 \tag{5.1}$$

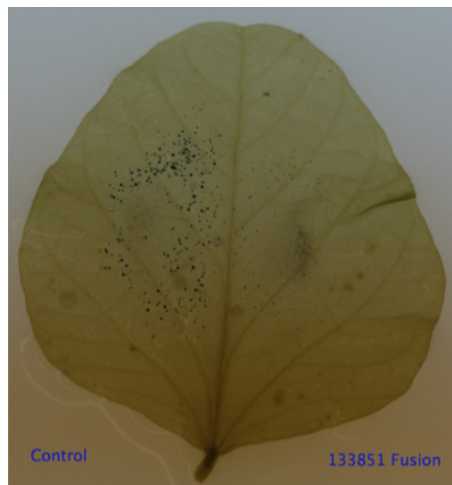
## 5.5 Discussion

### Discussion

Candidate effector constructs were created by fusing the effector with the C-terminus of Avr1b. The Avr1b C-terminus acts as a reporter for effector cell entry because interaction between Avr1b and Rps1b triggers a measurable cell death response (HR). An intact C-terminus of Avr1b from *Phytophthora sojae* is required for interaction with Rps1b in the cytoplasm of soybean host cells [16], especially the conserved motifs, such as the W or Y motifs. The RXLR-dEER motif is not required for functional interaction between Rps1b and Avr1b, but rather is involved in the entry of the effectors into the host cells [17]. The signal peptide of Avr1b targets the effector for secretion outside the cell (either the *P. sojae* hyphae or the soybean cell depending on where it is expressed). Since the interaction between Avr1b and Rps1b occurs in the cytoplasm of the plant cells, if either one is not present in the cytoplasm, no cell death is triggered. Thus, if Avr1b is bombarded into soybean leaves which do not express Rps1b, such as Williams, there is no interaction and no cell death. When full length Avr1b (with its signal peptide) is bombarded into Rps1b expressing leaves, the signal peptide of Avr1b initially directs secretion of the effector outside of the cell, and then the RXLR-dEER motif of Avr1b mediates re-entry of the protein back inside the cytoplasm, resulting in the interaction of Avr1b and Rps1b producing cell death [17]. When the N-terminal domain of Avr1b was deleted, the C-terminal domain alone was sufficient to induce cell death [17]. Candidate effector constructs were created by fusing the C-terminus of Avr1b, because we would be able to observe a hypersensitive response (HR) in the soybean leaves if there was interaction between Avr1b and Rps1b.

Based on the principle of the assay, it can be inferred that candidate effector 133851, fused to Avr1b C-terminus, was exported outside of the cells with its own signal peptide, but was able to gain entry back inside the cells to interact with Rps1b to cause HR. Candidate 133851 was predicted to contain a signal peptide with a very high confidence score from the SignalP software, and the predicted functional RXLR-like motif was KLLR. It is important to note that this prediction derived from a network-based analysis of the sequences flanking KLLR. Thus, these flanking sequences may play a significant role in the entry mechanism. The remaining question that needs to be addressed is how the re-entry of this protein was

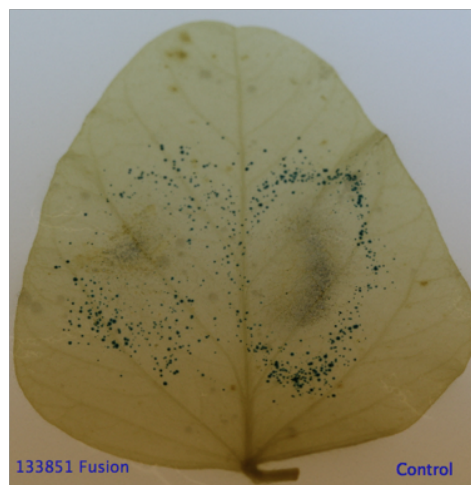
Figure 5.5: Representative leaves from L77-1863 and Williams are shown after bombarding and staining the soybean leaves. Each leaf was shot with a GUS control and a 133851-Avr1b-Ct fusion construct. Labels on the leaves show which side received which DNA sample. (a,b) Representative L77-1863 leaves showing the GUS spots from the bombardment. (c,d) Representative Williams leaves.



(a)



(b)



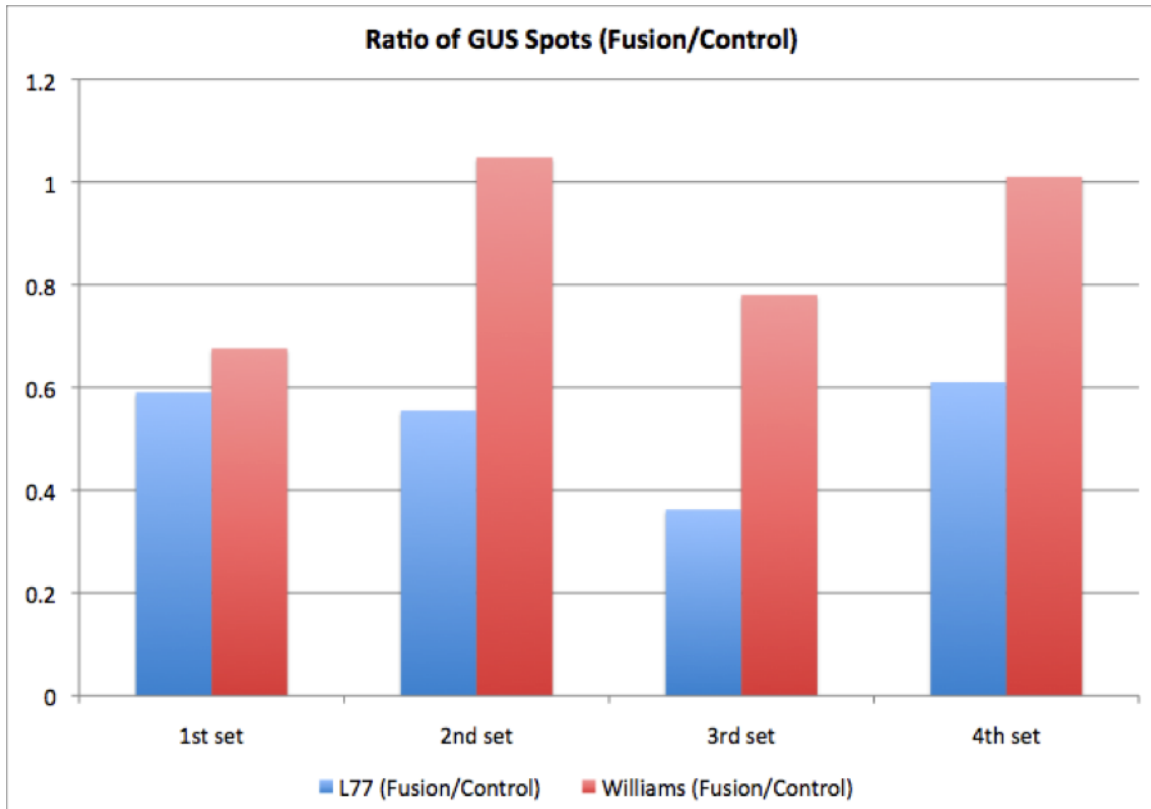
(c)



(d)



Figure 5.6: Ratio of GUS spots for the gene gun bombardment results. For each set of experiments, soybean leaves (L77-1863 and Williams) were bombarded with the control (GUS mixed with empty vector) and the fusion (GUS mixed with 133851 fused to Avr1b-Ct). The ratio between the fusion and the control is shown for each set. There tended to be less number of blue spots compared to the controls on L77-1863 leaves. Geometric average ratios are shown (calculated from the averages of the log ratios).



facilitated and what role KLLR may have played. To address this question, first it must be confirmed that 133851 does actually exit the cells by fusing it to a fluorescent protein such as GFP and observing whether it accumulates outside of the cells. Next, a mutation analysis (such as changing the KLLR to AAAA) would be needed to confirm whether re-entry requires the KLLR sequence as predicted.

Another aspect of RXLR effectors is that many of them bind to the cell surface lipid phosphatidylinositol 3-phosphate (PI3P) and enter the host cells via receptor-mediated endocytosis [34]. For example, the RXLR-dEER domains of Avr1b, Avh331 or Avh5 all have shown binding to PI3P in lipid binding assays [34]. The specific domains of these effectors were fused to a fluorescent protein like GFP and expressed in *E. coli* and purified. Then they were exposed to a nitrocellulose membrane where 14 different lipids were spotted. In

this assay, they showed binding specificity to PI3P and PI4P [34].

Since lipid filter binding assays use a filter medium where the lipids are attached, which is not a physiological context for the lipids, this assay can produce false positive or false negative results. An alternative method to test lipid binding is to use liposomes, which are artificial vesicles composed of a defined composition of lipids. A protein of interest can be added to a buffer solution containing the liposomes and after centrifugation, the pellet will contain liposome-bound proteins while the supernatant will hold non-bound proteins. Liposome binding assays confirmed the results from the lipid filter binding assays for the three effectors mentioned above [34].

Both lipid filter assays and liposome binding assays could be used to test if candidate 133851 can bind to lipids like PI3P or PI4P [34]. It is important to note that PI3P-binding is not the only mechanism available for entry into host cells. So candidate 133851 may not necessarily bind PI3P for entry.

Entry of the effector candidate protein into soybean cells could also be tested directly. Root cell uptake assays have been utilized for this purpose previously [17,34]. First, the candidate effector would be fused to a fluorescent protein like GFP and the fusion protein expressed in *E. coli* and purified. The fusion protein would then be incubated with soybean roots. After incubation, excess proteins would be washed away from the roots, and a confocal microscope could be used to observe if the fluorescently tagged fusion proteins had entered into the root cells. If the candidate showed positive results from lipid binding assays, the roots could be pre-incubated with the appropriate inositol diphosphates or PI3P-blocking proteins to determine if entry is inhibited [34]; inhibition would establish that PI3P-binding is required for entry.

In depth studies of plant pathogen effectors are important because knowledge from those studies can potentially translate to protection of crops from effector-producing pathogens. Using computational methods to produce a small list of candidates for experimental validation is an extremely efficient and economical approach to effector identification. This interdisciplinary approach of computational prediction of protein function combined with experimental validation can be applied to a wide variety of problems in biology.

# Chapter 6

## Conclusion

Despite the recent advancements made in bioinformatics and computational biology, the scientific community needs more studies that closely integrate gene function predictions *in silico* and detailed molecular analysis of the genes and the proteins they encode. Production of genome sequence data is continuing to expand exponentially. Functional genomics assays such as microarrays and other large scale biological experimental platforms are also generating massive data sets. With Next-Generation Sequencing technology, it is now possible to amplify thousands of single transcript fragments to produce a million copies of each fragment in just 8 hours, and sequence the entire transcriptome of an organism with unparalleled precision [43]. Non-coding RNAs have been discovered using this type of sequencing technique and the field of metagenomics is rapidly expanding [43].

Many studies have heavily focused on developing methods in dealing with how to handle and to visualize the vast amount of new data in a coherent manner and how to ensure sound mathematical basis to their analysis methods. However, many recently published computational prediction and modeling methods generate hypotheses and theories that are not feasible to test experimentally in labs. Significant improvement is necessary to develop computational tools that perform well enough to directly produce biological predictions that are readily verifiable in the lab. Accurate, fully computational gene function annotation still remains out of reach today [50]. Large scale evaluations and surveys of the current protein function prediction approaches are emerging to characterize the gap between prediction methods and experimental validation of protein functions [50]. An international effort involving 30 research teams have recently published their findings [50] and concluded that many algorithms used currently in protein function prediction far outperform first generation approaches such as simple BLAST searches and homology based methods.

All computational methods have advantages and disadvantages, and it is worth considering the basis for each method carefully as it may influence the biological interpretations of the results. One of the most robust computational classification methods used here for effector candidate classification is Support Vector Machines (SVM) in Chapter 2. Because SVM

calculates the separating hyperplane between the two classes in the training data in a way that ensures the greatest margin, SVM is known to be relatively robust to bias present in the data (the problem of over-fitting). Also, since kernels are used to transform the training data into a potentially non-linear feature space, no assumptions about linear separability of the training data are necessary, unlike logistic regression. The disadvantage of SVM is the non-transparency of the results [2]; it is difficult to interpret how the classifier was produced, especially given the high number of input data dimensions (there were 200 dimensions in the training data in Chapter 2). In this regard, other classification methods used in Chapter 2, such as spherical classification or logistic regression, have an advantage over SVM. Because the spherical classification used a simple sphere as the separator, this method provided a more straight forward interpretation for how the classifiers were built, which was based on the distance from the geometric center of the positive examples (Equations 2.1 and 2.2), as well as a flexible approach to choosing an appropriate cutoff. The main disadvantage of logistic regression compared to SVM is that the statistical model is based on the assumption that there is a linear relationship between the parameters of the predictor variables (such as the 200 dimensions) and the outcome (for example, effector status). The characteristic of logistic regression can be an advantage, if the assumption of linearity is true, since a readily interpretable model (composed of particular linear combinations of amino acid positions and Kidera factors [38] in Chapter 2) can be generated. Functional linkage networks have an advantage in the case where the relationship between the parameters and the outcome is not clear or cannot be explicitly modeled [59]. If the predictor variables are interacting with each other, the GAIN algorithm used in Chapter 3 would be able to find those relationships without explicitly stating them mathematically, while explicit modeling required by methods like logistic regression. Algorithms like GAIN [36] that perform predictions using functional linkage networks have a further important advantage that functional assignments are made based on local information present in the neighborhood of nodes. There functional linkage networks can be effective when the rules for predicting positives vary across different neighborhoods of the network, whereas methods such as SVM and logistic regression assume that there is a single rule that is consistent across the entire data set. The disadvantage of using a functional linkage network may be that it may be difficult to identify important predictor variables that influence the prediction outcome. The best method of choice for functional prediction will depend on the inherent structure of the training data, such as the relationship between the dimensions of the predictor variables, and relationship between the predictor variables and the outcome.

Another difficulty in large scale genome annotation comes from how the molecular function of a protein is defined. In order to automate any computational process, a machine-readable schema is needed to represent this information. A free-text paragraph describing the function of a protein is not so useful in this regard. The Gene Ontology Consortium [1] has tackled this challenge and remains as a dominant source of controlled vocabulary to provide the platform for automation, but it is still difficult to accurately capture the subtleties of protein function. This is because the function of a protein heavily depends on its context - where and when the protein is expressed, whether it directly or indirectly interacts with other

molecules, etc. Depending on this context, the protein may function differently. However, a particular molecular experiment may uncover only one aspect of the proteins function, or the protein may be associated with a whole-organism level-response, such as a disease, that is difficult to capture in a simple experiment. For example, Chapter 5 examined one candidate effector's ability to enter host cells using gene gun bombardment assays. But no conclusions could be made about the exact mechanism of entry, such as the candidate effector protein's ability to interact with lipids, without a different set of experiments.

As new platforms for high throughput functional analysis of genes and gene products are developed, accurate protein function prediction will likely improve. Undoubtedly, this will provide more opportunities to use machine learning tools as well as network-based methods to integrate this information. The need for developing experimental platforms for targeted, verifiable hypotheses testing is clearly a future need, so that rapidly developing large scale technology can be fully leveraged.

# Bibliography

- [1] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, and J. T. Eppig. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.
- [2] L. Auria and R. A. Moro. Support vector machines (svm) as a technique for solvency analysis. [http://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=1424949](http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1424949). Accessed: 2013-10-01.
- [3] T. Bailey. The meme suite: Motif-based sequence analysis tools. <http://meme.nbcr.net/meme/cgi-bin/meme.cgi/>. Accessed: 2011-06-30.
- [4] T. L. Bailey, M. Boden, F. A. Buske, M. Frith, C. E. Grant, L. Clementi, J. Ren, W. W. Li, and W. S. Noble. Meme suite: tools for motif discovery and searching. *Nucleic acids research*, 37(suppl 2):W202–W208, 2009.
- [5] T. L. Bailey and C. Elkan. Fitting a mixture model by expectation maximization to discover motifs in bipolymers, 1994.
- [6] J. D. Bendtsen, H. Nielsen, G. von Heijne, and S. r. Brunak. Improved prediction of signal peptides: Signalp 3.0. *J Mol Biol*, 340(4):783–795, 2004.
- [7] M. P. Brown, W. N. Grundy, D. Lin, N. Cristianini, C. W. Sugnet, T. S. Furey, M. Ares, and D. Haussler. Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proceedings of the National Academy of Sciences*, 97(1):262–267, 2000.
- [8] C.-C. Chang and C.-J. Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.
- [9] K.-C. Chou and Y.-D. Cai. Using functional domain composition and support vector machines for prediction of protein subcellular location. *Journal of Biological Chemistry*, 277(48):45765–45769, 2002.
- [10] G. R. Cornelis. The type iii secretion injectisome. *Nature Reviews Microbiology*, 4(11):811–825, 2006.

- [11] N. Cristianini and J. Shawe-Taylor. *An introduction to support Vector Machines: and other kernel-based learning methods*. Cambridge Univ Pr, 2000.
- [12] J. L Dangl and J. D. Jones. Plant pathogens and integrated defence responses to infection. *Nature*, 411(6839):826–833, 2001.
- [13] M. Das and H.-K. Dai. A survey of dna motif finding algorithms. *BMC bioinformatics*, 8(Suppl 7):S21, 2007.
- [14] R. A. Dean, N. J. Talbot, D. J. Ebbole, M. L. Farman, T. K. Mitchell, M. J. Orbach, M. Thon, R. Kulkarni, J. R. Xu, and H. Pan. The genome sequence of the rice blast fungus *magnaporthe grisea*. *Nature*, 434(7036):980–986, 2005.
- [15] P. D’haeseleer. How does dna sequence motif discovery work? *Nature biotechnology*, 24(8):959–961, 2006.
- [16] D. Dou, S. D. Kale, X. Wang, Y. Chen, Q. Wang, R. H. Y. Jiang, F. D. Arredondo, R. G. Anderson, and P. B. Thakur. Conserved c-terminal motifs required for avirulence and suppression of cell death by *phytophthora sojae* effector *avr1b*. *The Plant Cell Online*, 20(4):1118, 2008.
- [17] D. Dou, S. D. Kale, X. Wang, R. H. Y. Jiang, N. A. Bruce, F. D. Arredondo, X. Zhang, and B. M. Tyler. Rxlr-mediated entry of *phytophthora sojae* effector *avr1b* into soybean cells does not require pathogen-encoded machinery. *The Plant Cell Online*, 20(7):1930, 2008.
- [18] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863–14868, 1998.
- [19] D. Eisenberg, E. M. Marcotte, I. Xenarios, and T. O. Yeates. Protein function in the post-genomic era. *Nature*, 405(6788):823–826, 2000.
- [20] J. G. Ellis and P. N. Dodds. Showdown at the rxlr motif: Serious differences of opinion in how effector proteins from filamentous eukaryotic pathogens enter plant cells. *Proceedings of the National Academy of Sciences*, 108(35):14381–14382, 2011.
- [21] A. J. Enright, S. Van Dongen, and C. A. Ouzounis. An efficient algorithm for large-scale detection of protein families. *Nucleic acids research*, 30(7):1575, 2002.
- [22] R. D. Fleischmann, M. D. Adams, O. White, R. A. Clayton, E. F. Kirkness, A. R. Kerlavage, C. J. Bult, J.-F. Tomb, B. A. Dougherty, and J. M. Merrick. Whole-genome random sequencing and assembly of *haemophilus influenzae* rd. *Science*, 269(5223):496–512, 1995.
- [23] J. Gillis and P. Pavlidis. The impact of multifunctional genes on ”guilt by association” analysis. *PloS one*, 6(2):e17258, 2011.

- [24] J. Gillis and P. Pavlidis. The role of indirect connections in gene networks in predicting function. *Bioinformatics*, 27(13):1860–1866, 2011.
- [25] J. Gillis and P. Pavlidis. "guilt by association" is the exception rather than the rule in gene networks. *PLoS computational biology*, 8(3):e1002444, 2012.
- [26] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, and M. A. Caligiuri. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537, 1999.
- [27] S. R. Gunn. Support vector machines for classification and regression. *ISIS technical report*, 14, 1998.
- [28] B. J. Haas, S. Kamoun, M. C. Zody, R. H. Jiang, R. E. Handsaker, L. M. Cano, M. Grabherr, C. D. Kodira, S. Raffaele, and T. Torto-Alalibo. Genome sequence and analysis of the irish potato famine pathogen *phytophthora infestans*. *Nature*, 461(7262):393–398, 2009.
- [29] J. Han, M. Kamber, and J. Pei. *Data mining: concepts and techniques*. Morgan kaufmann, 2006.
- [30] R. H. Y. Jiang, S. Tripathy, F. Govers, and B. M. Tyler. Rxlr effector reservoir in two *phytophthora* species is dominated by a single rapidly evolving superfamily with more than 700 members. *Proceedings of the National Academy of Sciences*, 105(12):4874, 2008.
- [31] R.H. Jiang and B. M. Tyler. Mechanisms and evolution of virulence in oomycetes. *Annual review of phytopathology*, 50:295–318, 2012.
- [32] J. D. G. Jones and J. L. Dangl. The plant immune system. *Nature*, 444(7117):323–329, 2006.
- [33] R. Jothi, S. Cuddapah, A. Barski, K. Cui, and K. Zhao. Genome-wide identification of in vivo protein-dna binding sites from chip-seq data. *Nucleic acids research*, 36(16):5221–5231, 2008.
- [34] S. D. Kale, B. Gu, D. G. S. Capelluto, D. Dou, E. Feldman, A. Rumore, F. D. Arredondo, R. Hanlon, I. Fudal, and T. Rouxel. External lipid pi3p mediates entry of eukaryotic pathogen effectors into plant and animal host cells. *Cell*, 142(2):284–295, 2010.
- [35] S. D. Kale and B. M. Tyler. Assaying effector function in planta using double-barreled particle bombardment. *Methods Mol Biol*, 712:153–172, 2011.



- [36] U. Karaoz, T. Murali, S. Letovsky, Y. Zheng, C. Ding, C. R. Cantor, and S. Kasif. Whole-genome annotation by using evidence integration in functional-linkage networks. *Proceedings of the National Academy of Sciences of the United States of America*, 101(9):2888, 2004.
- [37] A. Karatzoglou, D. Meyer, and K. Hornik. Support vector machines in r. *Journal of Statistical Software*, 15(9):1–28, 2006.
- [38] A. Kidera, Y. Konishi, M. Oka, T. Ooi, and H. A. Scheraga. Statistical analysis of the physical properties of the 20 naturally occurring amino acids. *Journal of Protein Chemistry*, 4(1):23–55, 1985.
- [39] J. Kyte and R. F. Doolittle. A simple method for displaying the hydropathic character of a protein. *Journal of molecular biology*, 157(1):105–132, 1982.
- [40] S. Letovsky and S. Kasif. Predicting protein function from protein/protein interaction data: a probabilistic approach. *Bioinformatics*, 19(suppl 1):i197–i204, 2003.
- [41] E. M. Marcotte, M. Pellegrini, H.-L. Ng, D. W. Rice, T. O. Yeates, and D. Eisenberg. Detecting protein function and protein-protein interactions from genome sequences. *Science*, 285(5428):751–753, 1999.
- [42] E. M. Marcotte, M. Pellegrini, M. J. Thompson, T. O. Yeates, and D. Eisenberg. A combined algorithm for genome-wide prediction of protein function. *Nature*, 402(6757):83–86, 1999.
- [43] E. R. Mardis. Next-generation dna sequencing methods. *Annu. Rev. Genomics Hum. Genet.*, 9:387–402, 2008.
- [44] T. Murali, M. D. Dyer, D. Badger, B. M. Tyler, and M. G. Katze. Network-based prediction and analysis of hiv dependency factors. *PLoS Comput Biol*, 7(9):e1002164, 2011.
- [45] D. R. Musicant, V. Kumar, and A. Ozgur. Optimizing f-measure with support vector machines. In *FLAIRS Conference*, pages 356–360, 2003.
- [46] N. Nariai, E. D. Kolaczyk, and S. Kasif. Probabilistic protein function prediction from heterogeneous genome-wide data. *PloS one*, 2(3):e337, 2007.
- [47] I. Pagani, K. Liolios, J. Jansson, I.-M. A. Chen, T. Smirnova, B. Nosrat, V. M. Markowitz, and N. C. Kyrpides. The genomes online database (gold) v. 4: status of genomic and metagenomic projects and their associated metadata. *Nucleic acids research*, 40(D1):D571–D579, 2012.
- [48] J. Platt. *Sequential minimal optimization: A fast algorithm for training support vector machines*. Microsoft Research, 1998.

- [49] S. Rackovsky. Global characteristics of protein sequences and their implications. *Proceedings of the National Academy of Sciences*, 107(19):8623, 2010.
- [50] P. Radivojac, W. T. Clark, T. R. Oron, A. M. Schnoes, T. Wittkop, A. Sokolov, K. Graim, C. Funk, K. Verspoor, and A. Ben-Hur. A large-scale evaluation of computational protein function prediction. *Nature methods*, 2013.
- [51] W. S. Sanders, C. I. Johnston, S. M. Bridges, S. C. Burgess, and K. O. Willeford. Prediction of cell penetrating peptides by support vector machines. *PLoS computational biology*, 7(7):e1002101, 2011.
- [52] J. W. Schwabe and A. Klug. Zinc mining for protein domains. *Nature Structural and Molecular Biology*, 1(6):345–349, 1994.
- [53] B. Schwikowski, P. Uetz, and S. Fields. A network of protein-protein interactions in yeast. *Nature biotechnology*, 18(12):1257–1261, 2000.
- [54] R. Sharan, I. Ulitsky, and R. Shamir. Network-based prediction of protein function. *Molecular systems biology*, 3(1), 2007.
- [55] S. Sinha and M. Tompa. Discovery of novel transcription factor binding sites by statistical overrepresentation. *Nucleic acids research*, 30(24):5549–5560, 2002.
- [56] G. D. Stormo. Dna binding sites: representation and discovery. *Bioinformatics*, 16(1):16–23, 2000.
- [57] A. Sturn, J. Quackenbush, and Z. Trajanoski. Genesis: cluster analysis of microarray data. *Bioinformatics*, 18(1):207–208, 2002.
- [58] T. Srlic, C. M. Perou, R. Tibshirani, T. Aas, S. Geisler, H. Johnsen, T. Hastie, M. B. Eisen, M. van de Rijn, and S. S. Jeffrey. Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. *Proceedings of the National Academy of Sciences*, 98(19):10869–10874, 2001.
- [59] J. V. Tu. Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. *Journal of clinical epidemiology*, 49(11):1225–1231, 1996.
- [60] B. M. Tyler. *Phytophthora sojae*: root rot pathogen of soybean and model oomycete. *Molecular plant pathology*, 8(1):1–8, 2007.
- [61] B. M. Tyler, S. Tripathy, X. Zhang, P. Dehal, R. H. Y. Jiang, A. Aerts, F. D. Arredondo, L. Baxter, D. Bensasson, and J. L. Beynon. *Phytophthora* genome sequences uncover evolutionary origins and mechanisms of pathogenesis. *Science*, 313(5791):1261, 2006.

- [62] A. Valouev, D. S. Johnson, A. Sundquist, C. Medina, E. Anton, S. Batzoglou, R. M. Myers, and A. Sidow. Genome-wide analysis of transcription factor binding sites based on chip-seq data. *Nature methods*, 5(9):829–834, 2008.
- [63] R. van Handel. Hidden markov models. *Lecture Notes*, 2008.
- [64] Q. Wang, C. Han, A. O. Ferreira, X. Yu, W. Ye, S. Tripathy, S. D. Kale, B. Gu, Y. Sheng, and Y. Sui. Transcriptional programming and functional interactions within the phytophthora sojae rxlr effector repertoire. *The Plant Cell Online*, 23(6):2064–2086, 2011.
- [65] S. Wawra, M. Agacan, J. A. Boddey, I. Davidson, C. M. Gachon, M. Zanda, S. Grouffaud, S. C. Whisson, P. R. Birch, and A. J. Porter. The avirulence protein 3a (avr3a) from the potato pathogen phytophthora infestans, forms homodimers through its predicted translocation region and does not specifically bind phospholipids. *Journal of Biological Chemistry*, 287(45):38101–38109, 2012.
- [66] S. Wawra, J. Bain, E. Durward, I. de Bruijn, K. L. Minor, A. Matena, L. Lbach, S. C. Whisson, P. Bayer, and A. J. Porter. Host-targeting protein 1 (sphtp1) from the oomycete saprolegnia parasitica translocates specifically into fish cells in a tyrosine-o-sulphatedependent manner. *Proceedings of the National Academy of Sciences*, 109(6):2096–2101, 2012.
- [67] I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.
- [68] I. H. Witten, E. Frank, L. E. Trigg, M. A. Hall, G. Holmes, and S. J. Cunningham. *Weka: Practical machine learning tools and techniques with Java implementations*. Morgan Kaufmann Publishers, 1999.
- [69] L. F. Wu, T. R. Hughes, A. P. Davierwala, M. D. Robinson, R. Stoughton, and S. J. Altschuler. Large-scale prediction of saccharomyces cerevisiae gene function using overlapping transcriptional clusters. *Nature genetics*, 31(3):255–265, 2002.
- [70] T. Yaeno, H. Li, A. Chaparro-Garcia, S. Schornack, S. Koshiba, S. Watanabe, T. Kigawa, S. Kamoun, and K. Shirasu. Phosphatidylinositol monophosphate-binding interface in the oomycete rxlr effector avr3a is required for its stability in host cells to modulate plant immunity. *Proceedings of the National Academy of Sciences*, 108(35):14682–14687, 2011.