Air Temperature Prediction using Support Vector Regression and GENIE: The Georgia Extreme-weather Neural-network Informed Expert

by

Robert F. Chevalier

(Under the Direction of Walter D. Potter)

Abstract

Several studies have focused on comparisons between Support Vector Regression (SVR) and Artificial Neural Networks (ANNs). However, few have involved domains with massively large data sets. This research led to a methodology for reducing the number of SVR training patterns without a need to pre-process the data set. Using this methodology SVR models were created for air temperature prediction from one to twelve hours ahead. These models were more accurate than ANN models that were trained on data sets of 300,000 patterns and competitive with ANN models that were trained on 1.25 million patterns. A fuzzy expert system was also developed which incorporates the knowledge of local agrometeorologists in order to assess the risk of frost. Wind speed, as well as ANN models of air temperature and dew point temperature, enabled the expert system to make frost predictions from one to twelve hours ahead. This tool will be made available to Georgia farmers through a web-based interface that was created for The University of Georgia's Automated Environmental Monitoring Network website (`http://www.georgiaweather.net`).

Index words: Support Vector Regression, Artificial Neural Networks, Fuzzy Logic, Expert System, Air Temperature, Frost, Weather Modeling, Decision Support

Air Temperature Prediction using Support Vector Regression and GENIE: The Georgia Extreme-weather Neural-network Informed Expert

by

Robert F. Chevalier

B.S., Baldwin-Wallace College, 1999

A Thesis Submitted to the Graduate Faculty
of The University of Georgia in Partial Fulfillment
of the
Requirements for the Degree

Master of Science

Athens, Georgia

2008

Air Temperature Prediction using Support Vector Regression and GENIE: The Georgia Extreme-weather Neural-network Informed Expert

by

Robert F. Chevalier

Approved:

Major Professor: Walter D. Potter

Committee: Ronald W. McClendon
Gerrit Hoogenboom
Joel A. Paz

Electronic Version Approved:

Maureen Grasso
Dean of the Graduate School
The University of Georgia
August 2008

Acknowledgments

I would like to thank all of my committee members for their time and the numerous suggestions that they provided throughout my research. I would especially like to thank Dr. McClendon for his guidance and words of encouragement during my time at UGA. I owe a great debt to Brian Smith and Daniel Shank whose work was essential to my own. I would also like to thank my sister, the smartest person I know, for her constant advice. And finally, I would like to thank my parents for believing in me and reminding me that "It's never too late."

This work was funded in part by a partnership between the USDA-Federal Crop Insurance Corporation through the Risk Management Agency and the University of Georgia and by state and federal funds allocated to Georgia Agricultural Experiment Stations Hatch projects GEO00877 and GEO01654.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Extreme air temperatures are responsible for reduced crop production and loss of livestock in nearly all geographical regions across the U.S. In addition to creating dangerous working conditions for farmers, heat stress has been shown to decrease dairy production in cows by 20 to 30% (Jones and Stallings, 1999). In 1990, frosts in California resulted in $500 million in fruit losses and damages to 450,000 ha of trees (Attaway, 1997). The citrus industry in Florida has been devastated by frost damages on several occasions, resulting in billions of dollars in losses (Cooper et al., 1964; Martsolf et al., 1984; Attaway, 1997). Freezing temperatures in early April of 2007 caused 50% of Georgia's peach crop and 80% of its blueberry crop to be lost (Fonsah et al., 2007). If given sufficient warning, agricultural producers can minimize many of these damages through irrigation or by deploying orchard heaters and wind machines. Accurate air temperature prediction is therefore vital to the agricultural sector.

The University of Georgia's Automated Environmental Monitoring Network (AEMN) is a collection of over 75 weather stations (Hoogenboom, 1993). It was created as a means of collecting meteorological data in mainly rural areas across the state of Georgia. Artificial Neural Networks (ANNs) were created using this data for the purpose of air temperature prediction (Jain et al., 2006; Smith et al., 2006, 2008). Since the AEMN weather stations record observations every 15 minutes, massive amounts of data were available for training these ANNs. The models developed by Smith et al. (2008) made use of over 1.25 million training patterns.

Support Vector Regression (SVR) is a supervised learning technique which can be used in many of the same domains as ANNs. In instances where comparisons have been made between SVR and ANNs, SVR has often been shown to outperform ANNs (Müller et al., 1997; Qin et al., 2005; Khan and Coulibaly, 2006; Chen and Wang, 2007; Mori and Kanaoka, 2007; Zhang et al., 2007). However, training times for the SVR algorithm scale somewhere between quadratic and cubic with respect to the number of training patterns so applying SVR becomes infeasible with massively large data sets (Dong et al., 2005).

Frost damage is responsible for more economic losses than any other weather related phenomenon in the United States (White and Haas, 1975). The severity of a frost is dependent not only on air temperature but also dew point temperature and wind speed. Methods used by experts to determine the relationship between these variables and the risk of frost are not easily quantifiable. Although, the National Weather Service (NWS) issues three levels of frost warnings, they do not consider dew point temperature and with only three levels, the warnings are not very granular. Fuzzy expert systems offer a means of reasoning about the vague concepts which are often associated with meteorological domains. The MEDEX system was developed to predict gale-force winds in the Mediterranean Sea (Hadjmichael et al., 2002). The SIGMAR system was used to assist meteorologists with marine forecasts (Hansen, 1997). Additionally, more traditional expert systems have been used to predict the occurrence of frost on bridges and roadways (Takle, 1990; Temeyer et al., 2003).

Chapter 1 discusses the need for accurate air temperature and frost prediction. Artificial Neural Networks, Support Vector Regression, and Fuzzy Expert Systems, which are used for this purpose, are all introduced. This chapter also provides a brief review of background information and related work. Chapter 2 proposes a methodology for discovering an appropriate reduced training set that enables SVR to be competitive with ANNs while using much smaller training sets. This methodology is used to develop SVR models for air temperature prediction and the results are compared to those obtained using ANNs in previous studies. Chapter 3 describes the development of GENIE, a web-based tool for assisting Georgia

farmers in assessing the risk of frost. This tool is a neural network-based, fuzzy logic expert system. It uses wind speed as well as ANN predictions of air temperature and dew point temperature to provide granular frost warnings from one to twelve hours ahead. Chapter 4 summarizes the research conducted in this study and suggests possible areas for future work.

Chapter 2

Support Vector Regression with Reduced Training Sets for Air Temperature Prediction: A Comparison with Artificial Neural Networks[1]

[1]Chevalier, R. F., R. W. McClendon, G. Hoogenboom and J. A. Paz. To be submitted to *Artificial Intelligence in Agriculture*

2.1 Introduction

Accurate air temperature prediction is an essential component in decision making related to many areas of agricultural production. Extreme air temperatures can cause crop loss and lead to reduced yields as well as causing dangerous working conditions for farmers. In 2007, 50% of Georgia's peach crop and 87% of its blueberry crop were lost due to devastating freezes in early April (Fonsah et al., 2007). Extreme temperatures have also been shown to have an adverse effect on livestock. Heat stress can cause dairy cows to produce 20 to 30% less milk and in some cases can result in death (Jones and Stallings, 1999). Air temperature prediction is an important factor in providing early warnings for both dangerous frost and heat stress conditions. With sufficient notice, crop loss due to frost can be minimized through the use of preventative measures such as orchard heaters and irrigation. During times of extreme heat, proper precautions can also be taken to ensure the safety of workers and livestock.

The Georgia Automated Environmental Monitoring Network (AEMN) was created in 1991 (Hoogenboom, 1993). The purpose of the AEMN is to collect weather data across the state of Georgia and to disseminate the data in near real-time via the World Wide Web. The AEMN is comprised of over 75 stations that are located in mainly rural areas where the National Weather Service does not have a presence. One of the services currently offered on the AEMN website (`http://www.georgiaweather.net`) is a short-term air temperature prediction tool. This tool provides hourly predictions ranging from one to twelve hours ahead. These predictions are generated using Artificial Neural Network (ANN) models (Jain et al., 2006; Smith et al., 2006, 2008). An ANN is a popular data modeling tool that mimics the operation of neurons in the human brain. It is used to express complex functions which perform a non-linear mapping from $\mathbb{R}^I$ to $\mathbb{R}^K$, where I and K are the dimensions of the input and output space, respectively (Engelbrecht, 2002). An ANN is comprised of interconnected nodes, or neurons, which are themselves simple processing elements. Weights are associated with each of the inputs to a given neuron, which control the strength of that input. By adjusting these weights an ANN is able to model the data that are presented to it as patterns

of input-output pairs. The original ANN air temperature models of Jain et al. (2006) were developed using NeuroShell (Ward Systems Group, 1993) and were limited to 32,000 training patterns due to software constraints. Subsequent work by Smith et al. (2006) overcame this limitation by implementing a custom ANN suite in Java. Additionally, several enhancements were made to the original approach which included the addition of seasonal variables, an extended duration of prior variables, and the use of multiple instantiations for parameter selection. As a result of these enhancements, the prediction accuracy of these subsequent models was increased (Smith et al., 2006).

Another supervised learning method that has been used for estimation tasks is Support Vector Regression (SVR). The SVR algorithm is an extension of the popular classification tool Support Vector Machines (SVM). A SVM is a machine learning tool that has its roots in statistical learning theory (Cortes and Vapnik, 1995). In its original form, the SVM was designed to be used as a classification tool. Upon its introduction, researchers applied it to classification problems such as optical character recognition (Schölkopf et al., 1996) and face detection (Osuna et al., 1997). Subsequently, the algorithm was extended to the case of regression, or estimation, and termed SVR (Vapnik, 1995). Intuitively, SVR works by performing a non-linear mapping of the data from the input space to a higher dimensional feature space where linear regression can be performed. In the case of a linear SVR model, if the training data are of the form:

$$\{(\bar{x}_1, y_1), (\bar{x}_2, y_2), \cdots, (\bar{x}_l, y_l)\} \tag{2.1}$$

$$\bar{x}_i \in \mathbb{R}^d,\ y_i \in \mathbb{R},\ \text{and } l = \text{ number of examples}$$

then the solution function takes the form:

$$f(\bar{x}) = \sum_{i=1}^{l} (\alpha_i - \alpha_i{}^*)\langle \bar{x}_i, \bar{x} \rangle + b \tag{2.2}$$

where $\langle \cdot, \cdot \rangle$ represents the dot product of two points and the variables α_i,$\alpha_i{}^*$, and b are calculated by the SVR algorithm (Smola and Schölkopf, 2004). It is important to note that the model is described in terms of dot products between the data. Furthermore, the term

$(\alpha_i - \alpha_i{}^*)$ will be zero for many of the patterns in the training data. This means that only some of the patterns will actually have an impact on the final solution. These patterns are referred to as the support vectors. In the case of a non-linear SVR model, the data must first be transformed from the input space into a higher dimensional feature space using a non-linear mapping function, Φ. As the data are mapped into increasingly higher dimensions, calculating the term $\langle \Phi(\bar{x}_i), \Phi(\bar{x}) \rangle$ becomes infeasible. The solution is to take advantage of Mercer's Theorem, shown in Equation (2.3).

$$\langle \Phi(\bar{u}), \Phi(\bar{v}) \rangle \equiv k(\bar{u}, \bar{v}) \tag{2.3}$$

This theorem states that for certain mappings, Φ, and any two points, $\bar{u}$ and $\bar{v}$, the dot product of these two points can be evaluated using a kernel function, k, without ever explicitly computing the mapping (Schölkopf and Smola, 2001). Substituting a kernel into Equation (2.2) results in the following solution function:

$$f(\bar{x}) = \sum_{i=1}^{l} (\alpha_i - \alpha_i{}^*) k(\bar{x}_i, \bar{x}) + b \tag{2.4}$$

This solution is obtained by solving the convex optimization problem shown below:

$$\begin{aligned} \text{maximize} \quad & \begin{cases} -\frac{1}{2} \sum_{i,j=1}^{l} (\alpha_i - \alpha_i{}^*)(\alpha_j - \alpha_j{}^*) k(\bar{x}_i, \bar{x}_j) \\ -\varepsilon \sum_{i=1}^{l} (\alpha_i + \alpha_i{}^*) + \sum_{i=1}^{l} y_i (\alpha_i - \alpha_i{}^*) \end{cases} \\ \text{subject to} \quad & \sum_{i=1}^{l} (\alpha_i - \alpha_i{}^*) = 0 \text{ and } \alpha_i, \alpha_i{}^* \in [0, C] \end{aligned} \tag{2.5}$$

where $\bar{x}_i$ and y_i are the input-output pairs of the training data, α_i and $\alpha_i{}^*$ are the variables to be discovered, and ε and C are constants. The constant ε represents the SVR algorithm's tolerance for errors. The area within $\pm\varepsilon$ of the learned function is referred to as the SVR regression tube and any errors that fall within this tube are ignored. The constant C is referred to as the penalty factor and it controls the tradeoff between the complexity of the function and the frequency with which errors are allowed to fall outside of the SVR regression tube.

Comparisons between SVR and ANNs have been made in several domains. Khan and Coulibaly (2006) used SVR for lake water level prediction and demonstrated a slightly better performance than that of a multilayer perceptron. The training set for this work was comprised of approximately 850 patterns. Least squares support vector machines were used by Qin et al. (2005) to model water vapor and carbon dioxide fluxes over a cropland. Using training sets that ranged in size from 390 to 1,561 patterns, they were able to achieve better results than those obtained using radial basis function networks. SVR has also been used to predict the daily maximum air temperature (Mori and Kanaoka, 2007). Models were developed using data collected during the summer months from 1999 to 2001 in Tokyo, Japan and tested on data collected in 2002. Predictions obtained using SVR were shown to reduce the average error by 0.8% when compared to a multilayer perceptron. Again, comparisons were drawn between models that were trained on a relatively small training set size of 366 patterns. The performance of SVR in these instances can be explained by a few key characteristics that differentiate it from the ANN approach. Artificial Neural Networks use empirical risk minimization as a basis for model selection (Engelbrecht, 2002). Since empirical risk is simply a measure of the error over a given set of data, ANNs attempt to fit the data as closely as possible. This often leads to a solution function which performs extremely well on the training data but performs poorly on patterns not used in training. This is referred to as *overfitting*. SVR is much less susceptible to overfitting because it is based on the principle of structural risk minimization. Structural risk minimization seeks to minimize not only empirical error but also model complexity (Vapnik, 1995). The less complex a model is, the more likely it will be to fit patterns not used in training. Another advantage of the SVR algorithm over ANNs is that it always converges to a solution which is globally unique and optimal as a result of being framed as a convex optimization problem. Backpropogation ANNs, on the other hand, are susceptible to local minima and will discover different solutions depending upon the initial weights and the order of the training patterns. One critical advantage that ANNs have over SVR is that they are less computationally expensive. It has been shown

that training times for SVR scale somewhere between quadratic and cubic with respect to the number of training patterns (Dong et al., 2005). For this reason, they are typically not considered a practical alternative to ANNs for problems involving massively large data sets (more than 100,000 training patterns).

When working with massively large data sets, SVR is limited by both the speed and memory constraints of the available hardware. Several attempts have been made to address both issues. In the original SVR algorithm, the results of kernel calculations were cached in memory. As the number of training patterns increases, the memory requirements become excessive. The Kernel Adatron algorithm addresses this problem by removing the cached data and re-evaluating kernel components as necessary (Friess et al., 1998). Training time, however, is increased with this algorithm. *Chunking* (Osuna et al., 1997) and *decomposition* (Joachims, 1999; Platt, 1999) are methods that address both issues to some degree by incrementally solving subsets of the overall optimization problem. Even with these enhancements to the original algorithm, SVR is generally not applied to massively large data sets.

One approach to dealing with this limitation involves simply removing a portion of the training data and relying on the SVR algorithm's ability to generalize from the remaining training patterns. Since the decision function obtained by the SVR algorithm is determined exclusively by the support vectors, all non-support vector patterns could be removed from the training set without affecting the final solution. However, it is not possible to definitively identify the support vectors without first solving the optimization problem. Nevertheless, several successful heuristic methods for reducing the number of training patterns have been suggested for use with both SVM and SVR (Rychetsky et al., 1999; Almeida et al., 2000; Yang and Ahuja, 2000; Guo and Zhang, 2007). The drawback to these heuristic methods is that they involve pre-processing the data, which can be quite time consuming when working with massively large data sets.

The ANN models previously created for air temperature prediction by Smith et al. (2006; 2008) were developed using massively large data sets. Twelve models were developed to

predict air temperatures during the winter months from one to twelve hours ahead, using a training set of 300,000 patterns. Additionally, twelve models were developed to predict year-round air temperatures from one to twelve hours ahead using a training set of 1.25 million patterns. The study presented herein used these same data sets to develop models with SVR. The overall goal of this study was to evaluate the capability of SVR in predicting air temperature for horizons from one to twelve hours. The specific objectives included: (1) to develop a methodology for reducing the number of SVR training patterns that does not require that the data be pre-processed, (2) to develop SVR models for air temperature prediction for winter-only and year-round scenarios using the data collected by the AEMN, and (3) to compare the accuracy of these models with those developed in previous studies that used ANNs.

2.2 Model Development

2.2.1 Data Sets

All of the data used in this research were obtained from the AEMN weather stations located across the state of Georgia. Each of these weather stations monitors air temperature, relative humidity, vapor pressure deficit, precipitation, wind speed and direction, solar radiation, and soil temperature at 5, 10, and 20 cm. Some stations are also equipped to monitor open pan evaporation, water temperature, leaf wetness, and soil surface temperature. Of these weather variables, only air temperature, solar radiation, wind speed, rainfall, and relative humidity were used for ANN model development by Smith et al. (2006; 2008). These same inputs were used for SVR model development herein. Individual patterns were comprised of the current values and a 24-hour duration of prior observations for air temperature, solar radiation, wind speed, rainfall, and relative humidity. Additionally, the hourly rates of change in each of these variables over the previous 24-hour period was included. In keeping with the previous work, each of these 250 inputs was then scaled in the range [0.1, 0.9]. Eight additional inputs, representing the time of day and the day of year were also included. These were cyclic

variables that were created using fuzzy logic-type membership functions as described in (Smith et al., 2006). The total number of inputs for each pattern was therefore 258.

The data sets used in this research were partitioned the same as those used by Smith et al. to develop winter-only ANN models (2006) and year-round ANN models (2008). The first data set was used to create models for predicting air temperature during the winter months only. The second data set was used to create models for predicting air temperature throughout the year. The winter-only data set was merely a subset of the year-round data set. It consisted of patterns restricted to the first 100 days of the year and the days for which the recorded temperature was 20°C or below.

Both the year-round data and the winter-only data were partitioned into the following data sets: training, selection, and evaluation. The training sets were used for model development. The year-round training set consisted of approximately 1.25 million patterns, while the winter-only training set consisted of approximately 300,000 patterns. Both training sets were comprised of meteorological data collected during the years 1997-2000 from the following sites: Alma, Arlington, Attapulgus, Blairsville, Fort Valley, Griffin, Midville, Plains, and Savannah. These locations were selected to provide patterns which are both geographically and agriculturally diverse.

The selection sets were used to choose between alternate parameter settings as well as candidate *reduced* training sets. The year-round selection set consisted of approximately 1.25 million patterns, while the winter-only selection set consisted of approximately 300,000 patterns. The evaluation sets were reserved for a final measure of each model's accuracy and the patterns were not used in model development or selection. The year-round evaluation set consisted of approximately 800,000 patterns while the winter-only evaluation set consisted of approximately 200,000 patterns. Selection sets and evaluation sets were comprised of meteorological data taken from the following sites: Brunswick, Byron, Cairo, Camilla, Cordele, Dearing, Dixie, Dublin, Homerville, Nahunta, Newton, Valdosta, and Vidalia. The selection

set patterns were collected during the years 2000-2003, while the evaluation set patterns were collected during the years 2004-2005.

2.2.2 A Method for Reducing the Number of Training Patterns

The major factor that contributed to long SVR run times was the number of training patterns. Since not all patterns are used by the SVR algorithm, an appropriate reduced training set can decrease the run time without a loss of accuracy. The ideal reduced training set would contain only those patterns which the SVR algorithm would eventually determine to be support vectors. Without prior knowledge of which patterns comprise this set, a methodology was developed for creating an effective reduced training set which does not require lengthy pre-processing of the data. This methodology was based on the observation that training time is strongly dependent upon both the size of the training set and the width of the SVR regression tube. The width of the regression tube is controlled by the ε parameter of the SVR algorithm. While smaller values of ε prolong training time, they typically result in improved model accuracy as shown in Figure 2.1. This figure plots the MAE over a selection set for various models developed for the four-hour prediction horizon using a reduced training set of 60,000 patterns. Each line represents the MAE produced while using a different ε value and varying the value of γ from 0.01 to 0.06. It can be seen that regardless of the γ value, the MAE consistently improved as the width of the regression tube was decreased. Although optimal parameter settings cannot be discovered unless all parameters are tuned simultaneously, considerable time savings can be gained by keeping ε at a moderately high value while tuning the remaining parameters. For the SVR algorithm, the only other parameters are the penalty factor, C, and any kernel parameters. The kernel used with the SVR algorithm in these experiments was the radial-basis function. This kernel, which is shown below, was arbitrarily selected because it has been shown to be a good general purpose kernel (Hsu et al., 2000).

$$\exp(-\gamma\|\bar{u} - \bar{v}\|^2) \tag{2.6}$$

For the purposes of SVR, $\bar{u}$ represents a support vector and $\bar{v}$ represents the pattern that is being evaluated. The only free parameter in this kernel is γ, which controls the width of the kernel. Since training time is less of a concern with high values of ε, several reduced training sets can be sampled from the complete training set and evaluated using the selection set. The SVR algorithm need only be applied with smaller values of ε once one of these candidate reduced sets has been selected. The basic steps in this methodology are summarized below:

1. Using a small subset of the complete training data and a moderately high value of ε, perform initial parameter tuning of the penalty factor, C, and any kernel parameters.

2. Find a favorable training set by repeatedly sampling from the complete set, applying the SVR algorithm with a moderately high ε value, and evaluating on the selection set.

3. Reduce the ε and apply the SVR algorithm to produce the final model.

All experiments for this research were conducted in a University of Georgia computer lab containing 30 computers, the fastest of which had Intel Core Duo processors and 2 GB of RAM. Support Vector Regression run times varied depending on the training set size and parameter settings. The longest runs took up to 72 hours on these machines. The SVR algorithm was implemented using SVMlight (Joachims, 1999).

2.3 Results and Discussion

2.3.1 Year-Round Models

The initial experiments were carried out on the year-round data. Parameter tuning was performed using a reduced training set of 15,000 randomly sampled patterns for the four-hour prediction horizon. The results were then applied to all other prediction horizons. While tuning the C and γ parameters, ε was held at a moderately large value of 0.05. A series of progressively finer grained grid searches were performed with values of C from 0

to 800 and values of γ ranging from 0 to 1. The grid searches involved applying the SVR algorithm with varying parameter combinations and then evaluating the resulting model with the selection set. The resulting MAEs ranged from 1.48°C, for the best model, to 110.26°C, for an extremely poor model. Since the models had been built using a training set of only 15,000 patterns, the accuracy was not expected to be remarkable. Of interest was each model's accuracy relative to the other models in the grid search. This relative performance was used as a measure of the effectiveness of the corresponding parameter combinations. While this approach did not guarantee that optimal parameter settings would be found for the final model, it offered a quick method for discovering generally good values for C and γ. Based on these grid searches, values of 25 and 0.0104 were selected for C and γ, respectively. These parameter values were used for all remaining experiments.

The tradeoff between model accuracy and training time was the major factor in determining the size of the reduced training sets to be used for final model development. Larger training sets typically produced more accurate models, as demonstrated in Figure 2.2. This figure shows the MAE for models created by randomly selecting training patterns from the complete year-round training set of 1.25 million patterns. Thirty separate reduced training sets were created for different data set sizes, including 15,000, 30,000, 60,000, and 120,000 patterns. It can be seen that model accuracy generally increases with larger training set sizes. This is because the larger data sets are more likely to contain a greater number of support vectors from the complete training set. However, this improved accuracy had to be balanced with the increased training time that is associated with larger training sets. A run-time limit of three days was arbitrarily placed on any single SVR run. This limitation became a factor when building final SVR models which used smaller regression tube widths. As a result, training set sizes were capped at 60,000 patterns.

For each year-round prediction horizon, 30 candidate reduced training sets were created by randomly sampling 60,000 patterns from the complete training set of over 1.25 million patterns. A model was then created from each candidate set using the C and γ parameter

values that were determined earlier. The ε parameter was again kept at a moderately high value of 0.05. The candidate set whose model achieved the lowest MAE when applied to the selection set became the training set for that particular prediction horizon. Finally, the ε parameter was adjusted for each prediction horizon in order to obtain the highest possible accuracy on the selection set. This was accomplished by applying the SVR algorithm with values of ε ranging from 0.003 to 0.01 with a step size of 0.001. The model with the lowest MAE for each prediction horizon was selected and a final performance estimate was obtained by applying it to the evaluation set.

The MAE over the year-round evaluation set for the one-hour prediction horizon was 0.513°C. The MAE increased monotonically for each subsequent prediction horizon up to the twelve-hour prediction horizon which had an MAE of 1.922°C. This trend in model performance can be seen in Figure 2.3, which shows the predicted and observed air temperatures for all evaluation set patterns from Byron, GA. The distribution of prediction errors was centered near zero for all prediction horizons, while the variance of the errors increased relative to the prediction horizon.

The accuracy of the year-round SVR models was compared to that of the ANN models created by Smith et al. (2008) and is shown in Table 2.1. For the selection set, the SVR models performed slightly better than the ANN models on every prediction horizon with improvements in MAE of up to 0.052°C, or 3.7%, when compared to the ANN models. For the evaluation set, however, the ANN models performed slightly better than the SVR models for all but three of the prediction horizons. Again, there was only a slight disparity between the methods. The largest difference in MAE occurred for the nine-hour prediction horizon where the ANN model had an MAE that was 0.057°C, or 3.4%, lower than that of the corresponding SVR model.

2.3.2 Winter Models

The methodology used to develop year-round air temperature prediction models was applied to the winter-only data set in order to create models specific to the cold weather months of the year. Instead of being drawn from a complete set of 1.25 million patterns, these candidate reduced sets were randomly selected from the winter-only complete set of 300,000 patterns. After tuning the ε parameter, the resulting SVR models were applied to their respective winter-only evaluation set. The MAE over the evaluation set for the one-hour prediction horizon was 0.514°C. The MAE of each subsequent hourly prediction horizon increased monotonically up to the twelve-hour prediction horizon which had an MAE of 2.303°C. These MAE values were slightly higher than those obtained with the year-round models, which was consistent with the findings of Smith et al. (2008) when using ANNs. The accuracy of the winter-only SVR models was compared to that of ANN models created by Smith et al. (2006) and is shown in Table 2.2. For both the selection set and the evaluation set, the SVR models performed slightly better than the ANN models. The largest improvement in MAE for the evaluation set was on the three-hour prediction horizon where the SVR model had an improvement in MAE of 0.027°C, or 3.1%, when compared to the ANN model.

These results comparing SVR to ANNs demonstrate the advantage that ANNs have over SVR in terms of their ability to handle massively large data sets. However, they also demonstrate the ability of SVR to generalize from comparatively smaller training sets. For the year-round data set, the ANNs were able to take advantage of more than 20 times as many training patterns as the SVR algorithm. Yet the SVR models were still competitive with the ANN models. The MAEs of the SVR models for the evaluation set were not greater than 0.057°C, or 3.4%, of the ANNs MAEs on any horizon and they achieved slightly lower MAEs for three of the prediction horizons, as shown in Table 2.1. For the winter-only data set, despite the fact that the SVR algorithm used fewer training patterns than the ANNs, the SVR models achieved slightly greater accuracy on all prediction horizons.

2.4 Summary and Conclusions

Support Vector Regression was shown to be a viable alternative to Artificial Neural Networks for the problem of air temperature prediction. The method proposed in this study offers a quick means of producing a reduced training set from a massively large training set. It works by repeatedly and randomly sampling from the complete training set. Candidate training sets are quickly evaluated by applying the SVR algorithm with relaxed parameter settings, which reduces training time. The more time consuming SVR experiments are only performed after an appropriate reduced set has been identified. Even with computational limitations on the number of training patterns that the SVR algorithm was able to handle, it produced results that were comparable and in some cases more accurate than those obtained with ANNs. With a data set of 300,000 patterns, the SVR algorithm outperformed ANNs for all prediction horizons. With a larger data set of 1.25 million patterns, ANNs appeared to have a slight advantage over SVR. As more processing power becomes available, the advantage could likely shift to SVR.

The SVR models described in this research were developed using the radial-basis function kernel. Evaluating other kernels was beyond the scope of this work, however, alternate kernels may prove more effective. Additionally, parameter tuning was performed using only the four-hour prediction horizon data. It is likely that model accuracy would be increased if parameters were tuned for each individual prediction horizon. Future work may also involve exploring the effects of some limited pre-processing of the data. It is possible that a more intelligent method for creating a reduced training set would result in better performance. The method described by Guo and Zhang (2007) takes advantage of the observation that target values for support vectors are typically local extremes. Pre-processing is performed on the data using a k-nearest neighbor approach in order to find extremes in each neighborhood of size k. The remaining patterns are then filtered out of the training set. As stated earlier, this pre-processing step will be rather lengthy when dealing with massively large data sets, however, the extra investment may prove worthwhile.

Bibliography

Almeida, M. B., A. P. Braga, and J. P. Braga (2000). Speeding svms learning with a prior cluster selection and k-means. In *Proceedings of the 6th Brazillian Symposium on Neural Networks*, Rio de Janeiro, pp. 162–167.

Cortes, C. and V. Vapnik (1995). Support vector networks. *Machine Learning 20*(3), 273–297.

Dong, B., C. Cao, and S. E. Lee (2005). Applying support vector machines to predict building energy consumption in tropical region. *Energy and Buildings 37*, 545–553.

Engelbrecht, A. P. (2002). *Computational Intelligence An Introduction*. West Sussex: John Wiley and Sons.

Fonsah, E. G., K. C. Taylor, and F. Funderburk (2007). Enterprise cost analysis for middle Georgia peach production. Technical Report AGECON-06-118, The University of Georgia Cooperative Extension.

Friess, T.-T., N. Cristianini, and C. Campbell (1998). The kernel adatron algorithm: A fast and simple learning procedure for support vector machines. In *15th International Conference on Machine Learning*, Madison, pp. 188–196. Morgan Kaufman Publishers.

Guo, G. and J.-S. Zhang (2007). Reducing examples to accelerate support vector regression. *Pattern Recognition Letters 28*, 2173–2183.

Hoogenboom, G. (1993). The Georgia automated environmental monitoring network. In K. J. Hatcher (Ed.), *1993 Georgia Water Resources Conference*, Orlando, pp. 398–402. The University of Georgia.

Hsu, C.-W., C.-C. Chang, and C.-J. Lin (2000). A practical guide to support vector classification. Technical report, National Taiwan University.

Jain, A., R. W. McClendon, and G. Hoogenboom (2006). Freeze prediction for specific locations using artificial neural networks. *Transactions of the ASABE 49*(6), 1955–1962.

Joachims, T. (1999). Making large-scale SVM learning practical. In B. Schölkopf, C. Burges, and A. Smola (Eds.), *Advances in Kernel Methods - Support Vector Learning*, Chapter 11, pp. 169–184. Cambridge, MA: MIT Press.

Jones, G. M. and C. C. Stallings (1999). Reducing heat stress for dairy cattle. Technical Report 404-200, Virginia Cooperative Extension.

Khan, M. S. and P. Coulibaly (2006). Application of support vector machine in lake water level prediction. *Journal of Hydrologic Engineering 11*(3), 199–205.

Mori, H. and D. Kanaoka (2007). Application of support vector regression to temperature forecasting for short-term load forecasting. In *IEEE Joint Conference on Neural Networks*, Orlando, pp. 1085–1090. IEEE.

Osuna, E., R. Freund, and F. Girosi (1997). An improved training algorithm for support vector machines. In *Neural Networks for Signal Processing VII. Proceedings of the 1997 IEEE Workshop*, New York, pp. 276–285. IEEE.

Platt, J. C. (1999). Fast training of svms using sequential minimal optimization. In B. Schölkopf, C. Burges, and A. Smola (Eds.), *Advances in Kernel Methods - Support Vector Learning*, pp. 185–208. Cambridge, MA: MIT Press.

Qin, Z., Q. Yu, J. Li, Z. yi Wu, and B. min Hu (2005). Application of least squares vector machines in modelling water vapor and carbon dioxide fluxes over a cropland. *Journal of Zhejiang University 6*(6), 491–495.

Rychetsky, M., S. Ortmann, M. Ullmann, and M. Glesner (1999). Accelerated training of support vector machines. In *IJCNN '99 International Joint Conference on Neural Networks*, Volume 2, pp. 998–1003.

Schölkopf, B., C. Burges, and V. Vapnik (1996). Incorporating invariances in support vector learning machines. In C. von der Malsburg, W. V. Seelen, J. Vorbruggen, and B. Sendhoff (Eds.), *ICANN '96 International Conference on Artificial Neural Networks*, Volume 1112, Berlin, pp. 47–52. Springer.

Schölkopf, B. and A. J. Smola (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge: MIT Press.

Smith, B. A., G. Hoogenboom, and R. McClendon (2008). Artificial neural networks for automated year-round temperature prediction. Submitted for publication to *Computers and Electronics in Agriculture*.

Smith, B. A., R. W. McClendon, and G. Hoogenboom (2006). Improving air temperature prediction with artificial neural networks. *International Journal of Computational Intelligence 3*(3), 179–186.

Smola, A. J. and B. Schölkopf (2004). A tutorial on support vector regression. *Statistics and Computing 14*, 199–222.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. New York: Springer-Verlag.

Ward Systems Group (1993). *Manual of NeuroShell 2*. Ward Systems Group.

Yang, M.-H. and N. Ahuja (2000). A geometric approach to train support vector regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Hilton Head, SC, pp. 430–437.

Table 2.1: Mean Absolute Errors (MAE) of the year-round models for the selection and evaluation sets.

Horizon length (hours)	Year-Round Selection Set MAE (°C)			Year-Round Evaluation Set MAE (°C)		
	ANN model*	SVR model	Percent improvement	ANN model*	SVR model	Percent improvement
1	0.525	0.516	**1.6%**	0.516	0.513	**0.5%**
2	0.834	0.811	**2.8%**	0.814	0.806	**1.0%**
3	1.046	1.036	**0.9%**	1.015	1.027	-1.2%
4	1.226	1.214	**1.0%**	1.187	1.203	-1.4%
5	1.404	1.352	**3.7%**	1.356	1.344	**0.9%**
6	1.483	1.466	**1.1%**	1.432	1.463	-2.2%
7	1.577	1.570	**0.4%**	1.532	1.569	-2.4%
8	1.669	1.659	**0.6%**	1.623	1.664	-2.5%
9	1.734	1.727	**0.4%**	1.686	1.743	-3.4%
10	1.801	1.798	**0.2%**	1.755	1.810	-3.1%
11	1.865	1.855	**0.5%**	1.815	1.875	-3.3%
12	1.908	1.906	**0.1%**	1.873	1.922	-2.6%

*Smith et al. (2008)

Table 2.2: Mean Absolute Errors (MAE) of the winter-only models for the selection and evaluation sets.

Horizon length (hours)	Winter Selection Set MAE (°C)			Winter Evaluation Set MAE (°C)		
	ANN model*	SVR model	Percent improvement	ANN model*	SVR model	Percent improvement
1	0.534	0.528	**1.1%**	0.527	0.514	**2.4%**
2	0.884	0.861	**2.7%**	0.864	0.837	**3.1%**
3	1.167	1.110	**4.9%**	1.118	1.110	**0.8%**
4	1.401	1.386	**1.1%**	1.338	1.329	**0.7%**
5	1.624	1.592	**2.0%**	1.546	1.518	**1.8%**
6	1.811	1.768	**2.4%**	1.715	1.692	**1.3%**
7	1.987	1.929	**2.9%**	1.874	1.832	**2.3%**
8	2.126	2.072	**2.6%**	2.007	1.964	**2.1%**
9	2.243	2.205	**1.7%**	2.091	2.078	**0.6%**
10	2.362	2.315	**2.0%**	2.191	2.173	**0.8%**
11	2.443	2.405	**1.5%**	2.250	2.245	**0.2%**
12	2.526	2.481	**1.8%**	2.333	2.303	**1.3%**

*Smith et al. (2006)

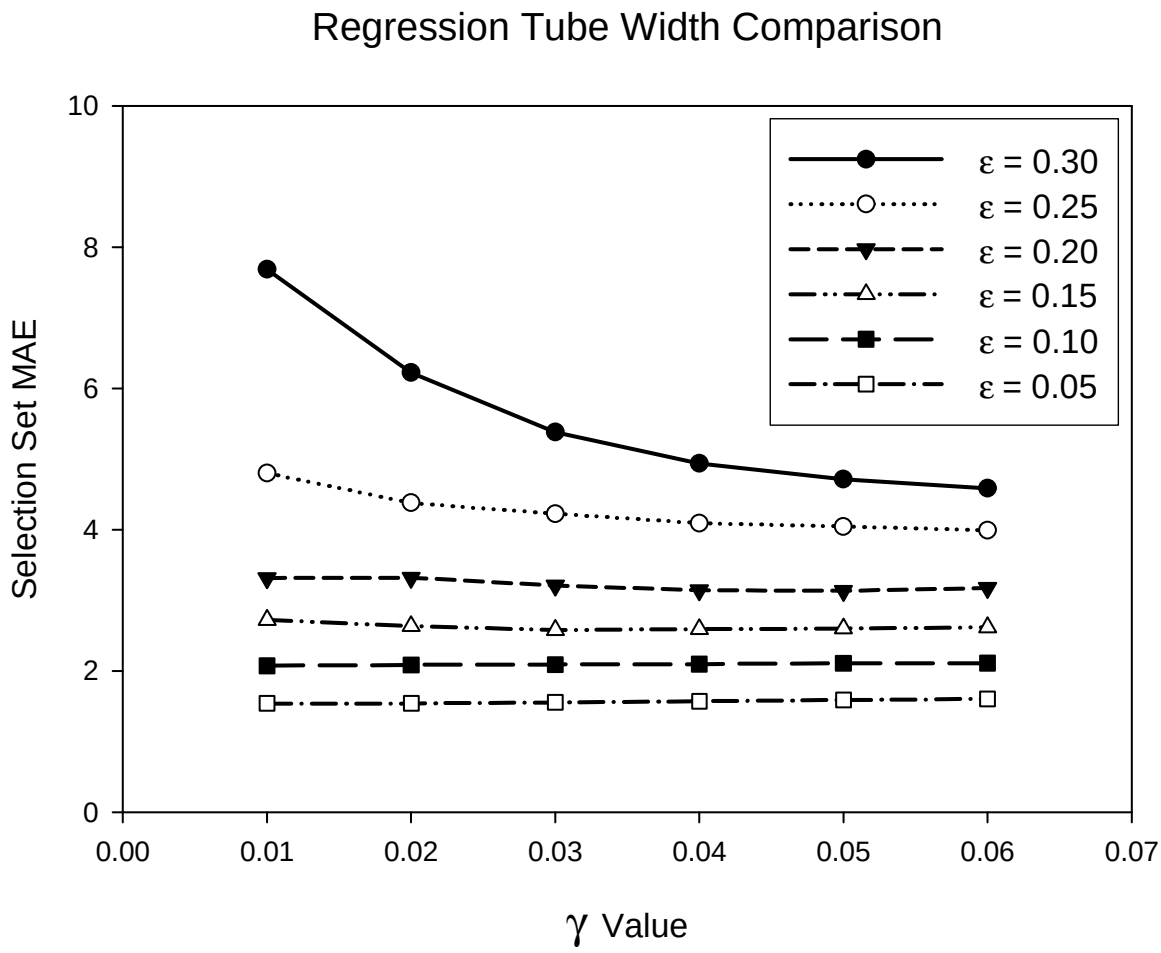

Figure 2.1: Selection set Mean Absolute Error (MAE) for various combinations of ε and γ. All results were obtained using a penalty factor, C, of 25.

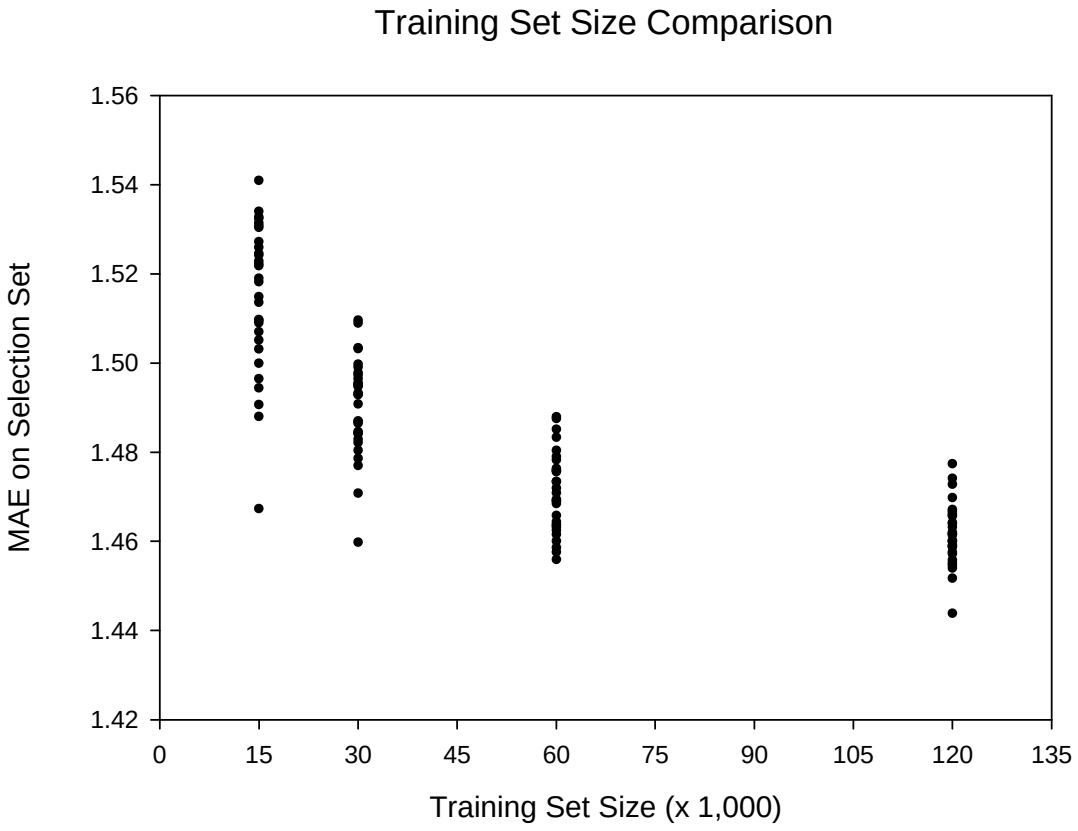

Figure 2.2: The Mean Absolute Error (MAE) for various training set sizes using the year-round, four-hour prediction horizon data. The following SVR parameters were used to create all models: $\varepsilon = 0.05$, $C = 25$, $\gamma = 0.0104$.

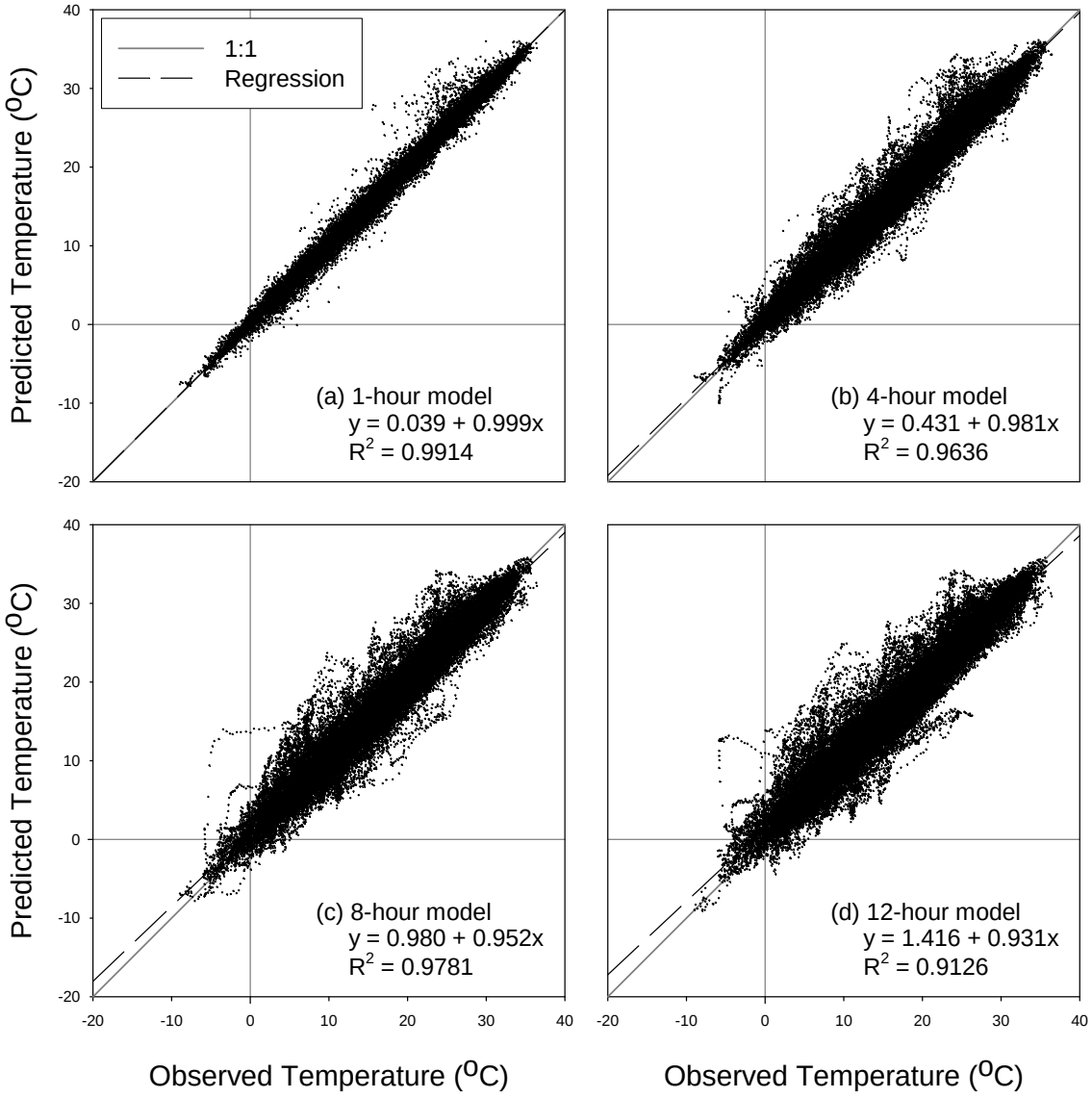

Figure 2.3: A comparison of predicted and observed temperatures for evaluation patterns specific to Byron, GA. (a) one-hour model, (b) four-hour model, (c) eight-hour model, and (d) twelve-hour model.

Chapter 3

GENIE: The Georgia Extreme-weather Neural-network Informed Expert[1]

[1]Chevalier, R. F., R. W. McClendon, J. A. Paz and G. Hoogenboom. To be submitted to *Computers and Electronics in Agriculture*

3.1 Introduction

Agricultural crops in most geographical regions of the U.S. are susceptible to damage from frost and freeze events during the growing season. For instance, in the state of Georgia, USA, a freeze occurred in early April of 2007 which resulted in estimated losses of 50% of the peach crop and 87% of the blueberry crop (Fonsah et al., 2007). The 2007 Georgia Farm Gate Value Report notes that the values for fruits and nuts for 2007 were down by $65 million compared to the previous year as a result of this late freeze event (Boatright and McKissick, 2008). These damages can be mitigated with preventative measures such as irrigation and orchard heaters provided that producers are given sufficient notice of impending frost and freeze events.

The terms frost and freeze are often used synonymously, however, they actually refer to two different conditions. Although there are no universally agreed upon definitions for these terms, it is generally accepted that a frost event becomes a freeze event when extracellular water within a plant freezes. Freeze events are particularly damaging to crops because this extracellular ice formation draws water out of a plant's cells causing dehydration (Snyder and de Melo-Abreu, 2005). Frost events are further classified based on the particular atmospheric conditions causing the loss of heat. Radiation frosts occur when heat is lost through radiation, causing the warm air close to the surface to be inverted with the colder air in the atmosphere. These frosts are characterized by clear skies, no wind, and a low dew point temperature. Advective frosts occur when wind brings cold air into a region replacing warmer air. Advective frosts are characterized by cloudy skies, moderate to strong winds, no temperature inversion, and low humidity.

The National Weather Service (NWS) currently issues three levels of frost and freeze warnings (Perry, 1994). A frost warning is issued when forecasts call for air temperatures near, but not below 0°C and wind speeds below 16 km/h. A frost/freeze warning is issued when forecasts call for air temperatures below 0°C and wind speeds below 16 km/h. Finally, a freeze warning is issued when forecasts call for air temperatures below 0°C and wind

speeds greater than 16 km/h. While the NWS warnings take into account air temperature and wind speed, they do not consider the dew point temperature. Additionally, there is little granularity in the NWS warnings since there are only 3 distinct warnings issued. The utility of the NWS warnings for Georgia producers is further limited by the fact that the NWS focuses on urban areas and does not collect weather data from many of the rural areas where agricultural production is prevalent.

The Georgia Automated Environmental Monitoring Network (AEMN) was created for the purpose of collecting weather data from areas across the state of Georgia, especially for those regions where the NWS does not have a presence (Hoogenboom, 1993). The AEMN consists of over 75 stations whose data are disseminated in near real-time via the AEMN website (`http://www.georgiaweather.net`). Using data collected from AEMN stations, Artificial Neural Network (ANN) models were created to predict air temperature (Jain et al., 2006; Smith et al., 2006, 2008) and dew point temperature (Shank et al., 2008a, 2008b). These models were implemented on the website and provide hourly predictions from one to twelve hours ahead for the areas surrounding the AEMN stations. Given the proximity of the stations to Georgia's croplands, these predictions provide invaluable information to producers across the state. A natural extension to this service would be to interpret these predictions for use in a decision support system. This would be a useful tool for producers who are concerned with frost and freeze events.

Fuzzy Logic is derived from Fuzzy Set theory and is a superset of the more conventional Boolean Logic (Zadeh, 1965). Fuzzy Logic is a method of reasoning that allows for information that is approximate, rather than precise. It is based on fuzzy sets as opposed to the two-valued sets which are the basis of Boolean Logic. In both forms of logic, an input variable's membership in a given set can be determined by applying a membership function to the input variable. In Boolean logic, there are only two possible outputs: a zero, which signifies that the input variable is not a member of the set, or a one, which signifies that the input variable is a member of the set. In Fuzzy Logic, partial membership in a set is

possible because output values can be any number in the continuous set [0, 1]. The output value assigned by a fuzzy membership function, therefore, represents the degree to which an input variable is a member of the set. The process of applying a fuzzy membership function to an input variable is referred to as "fuzzifying" the "crisp" input value (Engelbrecht, 2002). These fuzzy inputs are then used to define fuzzy if-then rules, and ultimately, to produce crisp output values. Formally implementing fuzzy logic involves the following five steps: fuzzification of the inputs, application of fuzzy operators in the rule antecedents, implication from the rule antecedents to the rule consequents, aggregation of the consequents, and defuzzification to a crisp output value (MathWorks, 1999).

Expert systems based on Fuzzy Logic have proved useful in domains involving meteorological data. The MEDiterranean EXpert system (MEDEX) was developed to predict specific gale-force wind events in the Mediterranean Sea (Hadjmichael et al., 2002). The inputs were supplied by various numerical models and this expert system was shown to be competitive with the US Navy's regional forecasting center in Rota, Spain. The Significant Information Generated from Marine Area Reports (SIGMAR) system was used to alert meteorologists when the accuracy of their marine forecasts began to deteriorate (Hansen, 1997). In this application, a fuzzy expert system was used for diagnostic rather than predictive purposes. More traditional expert systems have been used to provide frost related decision support. Takle (1990) used an expert system to predict the occurrence of frost on bridges and roadways. The expert system relied upon maximum and minimum recorded temperatures from the previous day as well as several estimated variables which included cloud cover, air temperature, dew point temperature, precipitation, and average wind speed. These estimated variables were not produced by a numerical model. Rather the user of the expert system was expected to estimate these values. Similar work was performed in which a model for frost deposition on Iowa bridgeways was used as a predictive tool by providing it with predictions from an ANN (Temeyer et al., 2003). Inputs for this model included current air temperature, dew point temperature, wind speed, and surface temperature.

The overall goal of the research reported herein was to develop a decision support tool to assist Georgia producers in assessing the risk of impending frost and freeze events. The specific objectives were: (1) to develop an expert system which encompasses the knowledge of agrometeorologists using fuzzy rules, (2) to evaluate the performance of the fuzzy expert system using historical data collected from the Georgia AEMN, and (3) to develop a web-based interface for disseminating the information to producers across the state of Georgia.

3.2 Methodology

A panel of expert agrometeorologists provided baseline information for the design of the expert system based on user needs and preferences. They determined that while the expert system should produce granular results, these results would be most useful if they were also translated into one of five general warnings: no warning, possible frost, mild frost, severe frost, and hard freeze. The crisp outputs of the fuzzy expert system, which represent the relative level of risk associated with the given weather conditions, were arbitrarily restricted to the continuous set [0, 1000]. Warnings were generated based solely on the value of these outputs as follows:

- Output $\in$ [0, 200] = No Warning
- Output $\in$ [201, 400] = Possible Frost
- Output $\in$ [401, 600] = Mild Frost
- Output $\in$ [601, 800] = Severe Frost
- Output $\in$ [801, 1000] = Hard Freeze

By providing producers with both the warning and the crisp output value, they would be able to differentiate between conditions such as a possibly non-threatening mild frost (output = 405) and a mild frost that is dangerously close to a severe frost (output = 595).

Extracting the knowledge of the local agrometeorologists was accomplished by presenting them with possible scenarios involving various combinations of air temperature, dew point temperature, and wind speed. The wind speed values in each table were determined through preliminary discussions with the agrometeorologists. The experts were asked to label each of these scenarios using one of the five warnings discussed earlier. All scenarios involving low wind speeds are shown in Table 3.1, while medium wind speeds scenarios are shown in Table 3.2, and high wind speed scenarios are shown in Table 3.3. Note that some combinations of air temperature and dew point temperature are labeled with an asterisk. These combinations are unrealistic because the air temperature never drops below the dew point temperature. It is possible to infer from these tables a set of fuzzy rules which cover, not only the specific scenarios described in the tables, but also all of the possible scenarios involving the continuous set of values between these discrete values. As an example, the following rule can be inferred based on the information presented in Table 3.1:

> "If the air temperature is between 0.5°C and 4.5°C, the dew point temperature is between 0.5°C and 2.5°C, and the wind speed is under 8 km/h, then possible frost conditions exist."

In this example, the decision boundaries were selected such that they are located half way between the points in the table where the labels change from one warning to another. For instance, an air temperature of 0.5°C is half way between 0.0°C, which is labeled as *no warning* for the given dew point temperature range, and 1.0°C, which is labeled as *mild frost* for the given dew point temperature range. Overall 75 fuzzy rules of this nature were inferred. These rules covered all possible combinations of air temperature, dew point temperature, and wind speed.

Multiple membership functions were used in each of the 75 rules. For instance, in the example above, six total membership functions were required. Two air temperature membership functions were required to express the degree to which the air temperature is "over 0.5°C" and "under 4.5°C". Similarly, two dew point membership functions were required to

express the degree to which the dew point temperature is "over 0.5°C" and "under 2.5°C". One membership function was required to express the degree to which the wind speed is "under 8 km/h" and one output membership function was required to express the degree to which a *possible frost* warning results. The 75 rules used by the expert system required a total of 25 unique air temperature membership functions, 28 dew point temperature membership functions, 3 wind speed membership functions, and 5 output membership functions. Several types of membership functions were evaluated , including triangular, Gaussian, and sigmoid functions. The suitability of each function was determined through a process of trial and error. The functions were evaluated by implementing them and then determining if they were able to produce outputs that matched the experts' warnings given the inputs found in Tables 3.1, 3.2, and 3.3. In general, concepts which are not clearly defined can be difficult to express using triangular membership functions. Triangular functions have discrete bounds, which means that their appropriateness is dependent upon being precisely fit to particular values. Smooth functions which approach limits are more forgiving in this regard. Therefore a mixture of Gaussian and sigmoid functions were considered for the input membership functions. The air and dew point temperature membership functions were defined exclusively using sigmoid functions. When necessary, the functions were tailored to each concept by adjusting the slope of the membership function. The slope is controlled by the γ term in the following sigmoid equation:

$$y = -\frac{1}{1 + e^{-\gamma(x-m)}} \tag{3.1}$$

The term m refers to the midpoint of the curve. Two of the membership functions from the dew point temperature fuzzy set are shown in Figure 3.1. The membership functions pictured, describe the concepts of "dew point over 0.5°C" and "dew point under 2.5°C". A rule may use these membership functions in conjunction to express the concept of "dew point between 0.5°C and 2.5°C". The three wind speed membership functions are shown in Figure 3.2. A left-facing sigmoid function was used for the *low wind* concept and a right-facing sigmoid function was used for the *high wind* concept. The midpoints of these functions

were centered at 8 km/h and 16 km/h respectively. The *medium wind* concept is defined by a Gaussian function. Notice that the width of the Gaussian curve has been manipulated so that it intersects with the sigmoid curves at their midpoints. This is accomplished by adjusting the constant, c, of the Gaussian equation shown below:

$$y = e^{-\frac{(x-12)^2}{2c^2}} \tag{3.2}$$

The result is that these intersections become decision boundaries for the expert system. Since the output concepts were arbitrarily defined with clear boundaries, it was possible to express them using the triangular membership functions shown in Figure 3.3. Note that although, the *no warning* and *hard freeze* membership functions actually extended beyond the arbitrarily selected bounds, their peaks were centered at 0 and 1000 respectively which were the effective bounds of the expert system's overall crisp output values. This is because the crisp output value was a function of the x-coordinate of each triangle's peak.

When multiple rules exist containing the same output function in the consequent, these output functions must be aggregated. Several aggregation methods exist, including the *Max-Min* method, the *Root-Sum-Square* method, and the *Sum* method (Engelbrecht, 2002). All three of these methods were evaluated using the scenarios provided by the experts. Preliminary work found that all of these methods were capable of producing satisfactory results for the majority of the scenarios. However, the Max-Min method was slightly more consistent in its ability to match the experts' warnings, as shown in Figure 3.4. Each method was able to produce outputs in the appropriate range for the *no warning* and *possible frost* scenarios. However, some of the *mild frost* and *severe frost* scenarios were misclassified when using both the Root-Sum-Square and Sum methods. The Max-Min method of aggregation was therefore selected for use in the model.

Crisp output values were obtained during the defuzzification step using a centroid calculation. Typically the area underneath each curve must be calculated in order to determine the centroid of a complex object. However, since the output membership functions used in this study took the form of identically sized equilateral triangles, their areas did not need

to be considered during the centroid calculation. The simplified centroid equation is shown below:

$$crisp\ output = -\frac{o_i(0) + o_2(300) + 0_3(500) + o_4(700) + o_5(1000)}{o_1 + o_2 + o_3 + o_4 + o_5} \tag{3.3}$$

The terms o_1, o_2, o_3, o_4, and o_5 represent the aggregate output values for each membership function and the constants represent the x-coordinate of the centroid of each triangle. These constants remain the same regardless of the scaled size of each triangle. Note that the crisp output is equal to the x-coordinate of the centroid. This coordinate will shift between 0 and 1000 depending on the size of the triangle associated with each output function.

The expert system fuzzy inferencing process is summarized in Figure 6. The process pictured is a simplified version of the actual inferencing process because only two rules are shown. Crisp input values of air temperature, dew point temperature, and wind speed are fuzzified and then used as inputs for each rule. The order of the rules is irrelevant. Fuzzy operators are applied to the antecedents of a rule to obtain the overall degree of support for that particular rule. This value is used to scale the output fuzzy set. Finally, the aggregate fuzzy output set is defuzzified by calculating the x-coordinate of its centroid.

To extend the capabilities of the fuzzy expert system to produce advance warnings, it must be supplied with predicted values of the weather variables. The work of Smith et al. (2008) and Shank et al. (2008) was used for this purpose. Smith et al. (2008)and Shank et al. (2008) developed ANN models for air temperature and dew point temperature based on weather data collected by the Georgia AEMN. These models can provide predictions from one to twelve hours ahead. Currently no model for wind speed prediction using the AEMN data exists. The expert system, therefore, assumes that the wind speed remains at the current level while assessing the risk of frost or freeze events for each hourly prediction horizon.

The fuzzy expert system was written in Java and its rules incorporate knowledge of critical temperatures that are specific to blueberries. However, the rules are implemented in such a way that the knowledge base can be easily modified to include new scientific knowledge or to

be used with another crop. The rules are constructed from implication operator methods and their arguments. The antecedents of the rules are layered using overloaded methods for each of the implication operators. This means that the existing blueberry rules do not need to be re-written when rules are added specific to other crops. They are simply augmented with an additional layer. A rule can be augmented by simply appending an additional argument to any of the implication operator methods that are being called. For instance, the AND operator method with two arguments is shown below:

```
and(argument1, argument2);
```

An additional layer can be added to a rule which uses this method by simply inserting another argument as follows:

```
and(argument1, argument2, argument3);
```

This additional argument may identify the rule as being specific to some particular crop. In this way, the expert system could provide frost warnings which are specific to multiple crops. Several methods were also built into the expert system that provide a means of troubleshooting while developing new fuzzy rules. These include methods for viewing aggregate membership function outputs as well as outputting the firing strengths of individual fuzzy rules.

3.3 Web-Based Interface

The usefulness of the GENIE tool is dependent upon the ease with which potential users can access its information. A web-based tool offers convenient access for many users, however, it must be user-friendly in order to gain acceptance. Most producers do not have time to search through vast amounts of information in order to find relevant warnings. The web interface of GENIE was built with this constraint in mind. It was designed to offer a quick graphical warning scheme that is easily accessible yet offers detailed information if the user is interested.

The interface was designed with four main panes that include a banner, a color map, a control panel, and a detail pane, as shown in Figure 3.6. The banner advertises the most severe warning level that is being predicted at any of the statewide locations for the next twelve hours. Producers can quickly identify the most extreme predictions by viewing banner messages (Figure 3.6a). The background color of the banner changes depending on the level of risk associated with the banner message. The color map also offers producers a quick assessment of statewide conditions as well as those in local areas (Figure 3.6b). Warning levels are actually only produced for areas where AEMN weather stations are located. These stations are denoted by small icons on the map which will display their location names when moused over. Warning levels from these locations are used to interpolate the colors across the rest of the map. Producers also have the ability to see how the map's frost landscape will change over the next twelve hours. Using the controls panel, users can either manually skip to a particular prediction horizon or click on the "Play Slide Show" button to see the map scroll through each hourly prediction (Figure 3.6c). At any time a producer can click on one of over 75 AEMN station icons to obtain more detailed information from the detail pane (Figure 3.6d). The location of the AEMN station as well as the current wind speed and maximum warning level are shown in the header field of this pane. Below this header is a graph, plotting the observed air temperatures and dew point temperatures for the past twelve hours as well as the values of these variables for the next twelve hours, as predicted by GENIE's ANN models (Smith et al., 2008; Shank et al., 2008). The information in the detail pane helps to explain the mathematical and scientific basis for the origin of the expert system's warnings by providing access to the complete set of variables used to produce the warnings.

The web interface was developed as a Java Applet. Applets allow for an interactive user interface even with large graphics files. This is a result of the applet being run directly on a user's computer as opposed to being run from a web-server. Additionally, Java Applets can

be developed as independent applications and then easily added to any existing website with a few lines of HTML code.

3.4 Results

The strength of the expert system's fuzzy rules is in their ability to produce granular warnings for any given value of air temperature, dew point temperature, and wind speed. After refining the rules, using the scenarios provided by the experts, GENIE was evaluated on an expanded set of scenarios which had not been previously used for model development. Table 3.4 shows the output for various air and dew point temperatures with a wind speed of 4 km/h. Output values have been rounded to the nearest whole number for clarity. Higher output values correspond with decreases in either air temperature or dew point temperature and there is a large range of values within each broad warning level. For instance, within the *no warning* level, values range from 52 to 172. These varying output levels provide GENIE's warnings with their granularity. For those scenarios that are present in Table 3.1, the GENIE's outputs are all in the appropriate ranges to match the experts' warnings.

A surface plot in Figure 3.7 shows the expert system's output landscape for wind speeds of 4 km/h and 20 km/h, using various combinations of air and dew point temperature. Values drop to zero for the unrealistic scenarios in which air temperature is lower than dew point temperature. Within each broad warning level, output values vary depending on their proximity to adjacent warning levels. Wind speed does not have a significant impact on the expert system's output values until temperatures drop below -3.0°C. This is consistent with the warnings provided by local agrometeorologists in Tables 3.1 and 3.3. At wind speeds of 4 km/h there were also no *hard freeze* warnings issued. As expected, these warnings do appear with higher wind speeds.

3.5 Summary and Conclusions

A fuzzy expert system was developed to produce frost threat levels for use by producers across the state of Georgia. The specific numeric warning levels produced by the system offer more granularity than the textual warnings currently made available by the NWS. Additionally, the underlying ANN models were trained with AEMN data collected from a wide range of locations in Georgia, making the system specifically tailored to Georgia's agricultural production. A web interface was developed to provide producers with convenient access to the frost warnings. The interface was consolidated into one interactive page to allow for quick access to all of the relevant information. Additionally, it offers transparency to the users by providing the predicted variables that were used in the expert system's decision making process. This allows users to develop confidence and trust in the warnings that are being provided by the system.

Although the warnings produced by the GENIE tool were developed specifically for blueberry producers, the fuzzy rules could easily be extended to accommodate any number of other crops. Future work may offer producers the ability to tailor the warnings to a specific crop. Also, as previously mentioned, the expert system does not make use of predicted wind speeds. In the future, a predictive wind speed model will be developed and incorporated in order to increase the reliability of the warnings. An opportunity also exists to extend the expert system so that it may provide heat stress warnings. Like frost, heat stress is dependent upon air temperature, dew point temperature, and wind speed. Heat stress is a concern for agricultural producers as it relates to both workers and livestock. For instance, heat stress can cause dairy cows to produce 20 to 30% less milk and in some cases even result in death (Jones and Stallings, 1999). Incorporating heat stress warnings into the GENIE tool could allow it to serve as a year-round resource for agricultural producers throughout the state of Georgia.

Bibliography

Boatright, S. R. and J. C. McKissick (2008). 2007 Georgia farm gate value report. Technical Report AR-08-01, The University of Georgia.

Engelbrecht, A. P. (2002). *Computational Intelligence An Introduction*. West Sussex: John Wiley and Sons.

Fonsah, E. G., K. C. Taylor, and F. Funderburk (2007). Enterprise cost analysis for middle Georgia peach production. Technical Report AGECON-06-118, The University of Georgia Cooperative Extension.

Hadjmichael, M., A. P. Kuciauskus, P. M. Tag, R. L. Bankert, and J. E. Peak (2002). A meteorological fuzzy expert system incorporating subjective user input. *Knowledge and Information Systems 4*(3), 350–369.

Hansen, B. (1997). Sigmar: A fuzzy expert system for critiquing marine forecasts. *AI Applications 11*(1), 59–68.

Hoogenboom, G. (1993). The Georgia automated environmental monitoring network. In K. J. Hatcher (Ed.), *1993 Georgia Water Resources Conference*, Orlando, pp. 398–402. The University of Georgia.

Jain, A., R. W. McClendon, and G. Hoogenboom (2006). Freeze prediction for specific locations using artificial neural networks. *Transactions of the ASABE 49*(6), 1955–1962.

Jones, G. M. and C. C. Stallings (1999). Reducing heat stress for dairy cattle. Technical Report 404-200, Virginia Cooperative Extension.

MathWorks (1999). *Fuzzy Logic Toolbox for Use with MATLAB*. Natick, MA: MathWorks.

Perry, K. B. (1994). Frost/freeze protection for horticultural crops. Leaflet 705-A, North Carolina Cooperative Extension Service.

Shank, D. B., R. W. McClendon, J. O. Paz, and G. Hoogenboom (2008). Ensemble artificial neural networks for prediction of dew point temperature. Accepted for publication by *Applied Artificial Intelligence*.

Smith, B. A., G. Hoogenboom, and R. McClendon (2008). Artificial neural networks for automated year-round temperature prediction. Submitted for publication to *Computers and Electronics in Agriculture*.

Smith, B. A., R. W. McClendon, and G. Hoogenboom (2006). Improving air temperature prediction with artificial neural networks. *International Journal of Computational Intelligence 3*(3), 179–186.

Snyder, R. L. and J. P. de Melo-Abreu (2005). *Frost Protection: fundamentals, practice and economics Volume 1.* Rome: Food and Agriculture Organization of the United Nations.

Takle, E. S. (1990). Bridge and roadway frost: Occurrence and prediction by use of an expert system. *Journal of Applied Meteorology 29*, 727–734.

Temeyer, B. R., W. A. G. Jr., K. A. Jungbluth, D. Burkheimer, and D. McCauley (2003). Using an artificial neural network to predict parameters for frost deposition on Iowa bridgeways. In *2003 Mid-Continent Transportation Research Symposium*, Ames.

Zadeh, L. (1965). Fuzzy sets. *Information and Control*, 338–353.

Table 3.1: Frost warnings supplied by domain experts for wind speeds < 8 km/h.

	Dew Point Temperature (°C)									
Air Temperature (°C)	4.0	3.0	2.0	1.0	0.0	-1.0	-2.0	-3.0	-4.0	-5.0
4.0	N	N	P	P	M	M	S	S	S	S
3.0	*	N	P	P	M	M	S	S	S	s
2.0	*	*	P	P	M	M	S	S	S	S
1.0	*	*	*	P	M	S	S	S	S	S
0.0	*	*	*	*	M	S	S	S	S	S
-1.0	*	*	*	*	*	S	S	S	S	S
-2.0	*	*	*	*	*	*	S	S	S	S
-3.0	*	*	*	*	*	*	*	S	S	S
-4.0	*	*	*	*	*	*	*	*	S	S
-5.0	*	*	*	*	*	*	*	*	*	S

*Air temperature is never lower than dew point temperature.

N	No Warning	M	Mild Frost
P	Possible Frost	S	Severe Frost

Table 3.2: Frost warnings supplied by domain experts for wind speeds = 12 km/h.

	Dew Point Temperature (°C)									
Air Temperature (°C)	4.0	3.0	2.0	1.0	0.0	-1.0	-2.0	-3.0	-4.0	-5.0
4.0	N	N	P	P	M	M	S	H	H	H
3.0	*	N	P	P	M	M	S	H	H	H
2.0	*	*	P	P	M	S	S	H	H	H
1.0	*	*	*	P	S	S	S	H	H	H
0.0	*	*	*	*	S	S	S	H	H	H
-1.0	*	*	*	*	*	S	S	H	H	H
-2.0	*	*	*	*	*	*	S	H	H	H
-3.0	*	*	*	*	*	*	*	H	H	H
-4.0	*	*	*	*	*	*	*	*	H	H
-5.0	*	*	*	*	*	*	*	*	*	H

*Air temperature is never lower than dew point temperature.

N	No Warning	S	Severe Frost
P	Possible Frost	H	Hard Freeze
M	Mild Frost		

Table 3.3: Frost warnings supplied by domain experts for wind speeds > 16 km/h.

Air Temperature (°C)	Dew Point Temperature (°C)									
	4.0	3.0	2.0	1.0	0.0	-1.0	-2.0	-3.0	-4.0	-5.0
4.0	N	N	P	P	M	M	S	H	H	H
3.0	*	N	P	P	M	M	S	H	H	H
2.0	*	*	P	P	M	M	S	H	H	H
1.0	*	*	*	M	M	M	S	H	H	H
0.0	*	*	*	*	M	M	S	H	H	H
-1.0	*	*	*	*	*	M	S	H	H	H
-2.0	*	*	*	*	*	*	S	H	H	H
-3.0	*	*	*	*	*	*	*	H	H	H
-4.0	*	*	*	*	*	*	*	*	H	H
-5.0	*	*	*	*	*	*	*	*	*	H

*Air temperature is never lower than dew point temperature.

N	No Warning	S	Severe Frost
P	Possible Frost	H	Hard Freeze
M	Mild Frost		

Table 3.4: GENIE's numeric outputs for various scenarios of air temperature and dew point temperature. Wind speed is 4 km/h.

Air Temperature (°C)	Dew Point Temperature (°C)																		
	4.0	3.5	3.0	2.5	2.0	1.5	1.0	0.5	0.0	-0.5	-1.0	-1.5	-2.0	-2.5	-3.0	-3.5	-4.0	-4.5	-5.0
4.0	52	87	129	172	240	268	298	362	433	461	489	555	631	631	631	680	694	694	703
3.5	*	88	130	172	240	302	330	362	433	500	527	555	631	680	680	682	697	703	712
3.0	*	*	131	172	240	302	365	398	433	500	555	585	631	680	694	697	701	708	721
2.5	*	*	*	172	240	302	365	434	468	500	555	604	655	680	694	703	708	710	723
2.0	*	*	*	*	240	302	365	434	500	532	555	604	668	693	694	703	711	713	724
1.5	*	*	*	*	*	302	365	434	500	566	591	604	668	700	700	703	711	715	726
1.0	*	*	*	*	*	*	365	434	500	566	625	639	668	700	703	706	711	715	727
0.5	*	*	*	*	*	*	*	434	500	566	625	665	674	703	712	713	713	715	727
0.0	*	*	*	*	*	*	*	*	500	566	625	665	690	704	714	717	717	717	727
-0.5	*	*	*	*	*	*	*	*	*	566	625	665	690	704	714	717	717	718	728
-1.0	*	*	*	*	*	*	*	*	*	*	625	665	690	704	714	716	717	718	728
-1.5	*	*	*	*	*	*	*	*	*	*	*	665	690	704	714	716	717	718	728
-2.0	*	*	*	*	*	*	*	*	*	*	*	*	690	704	714	716	717	718	728
-2.5	*	*	*	*	*	*	*	*	*	*	*	*	*	704	714	716	717	718	728
-3.0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	714	716	717	718	728
-3.5	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	716	717	718	728
-4.0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	717	718	728
-4.5	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	718	728
-5.0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	728

*Air temperature is never lower than dew point temperature.

No Warning Possible Frost Mild Frost Severe Frost

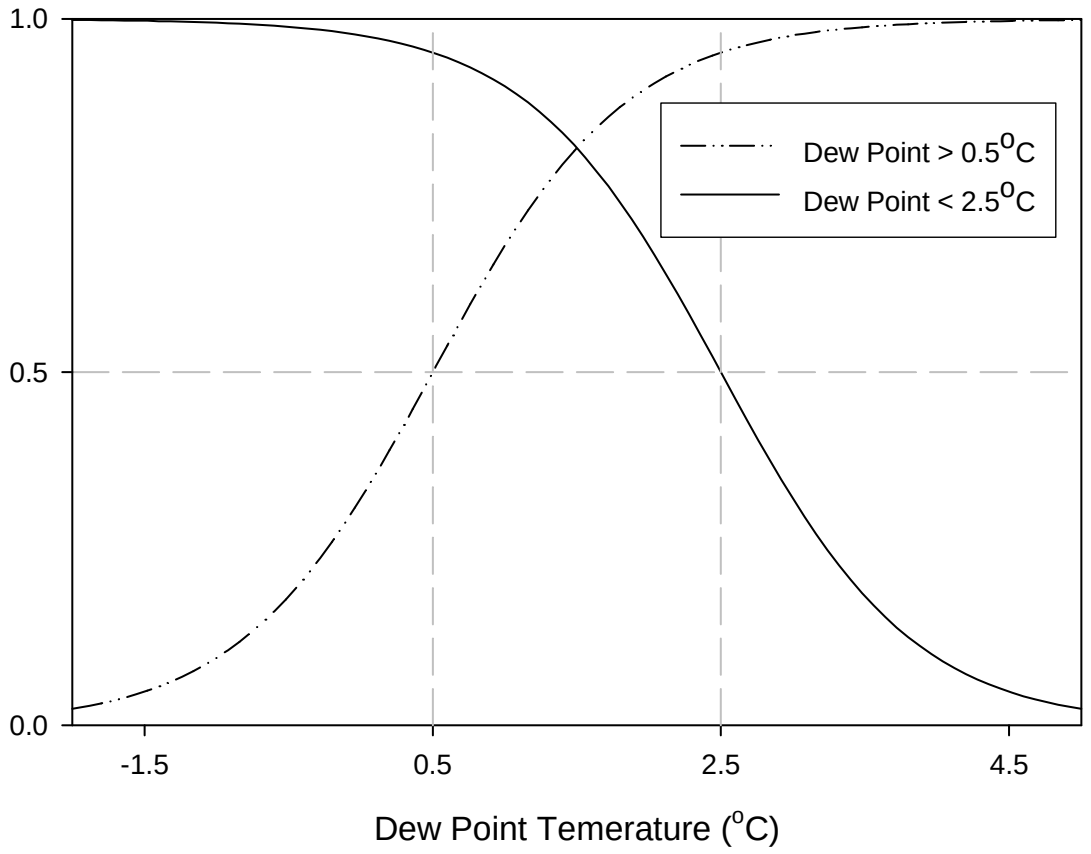

Figure 3.1: Sigmoid membership functions for dew point temperature greater than 0.5°C and less than 2.5°C. Midpoints of each function are at the intersection of the dashed grid lines.

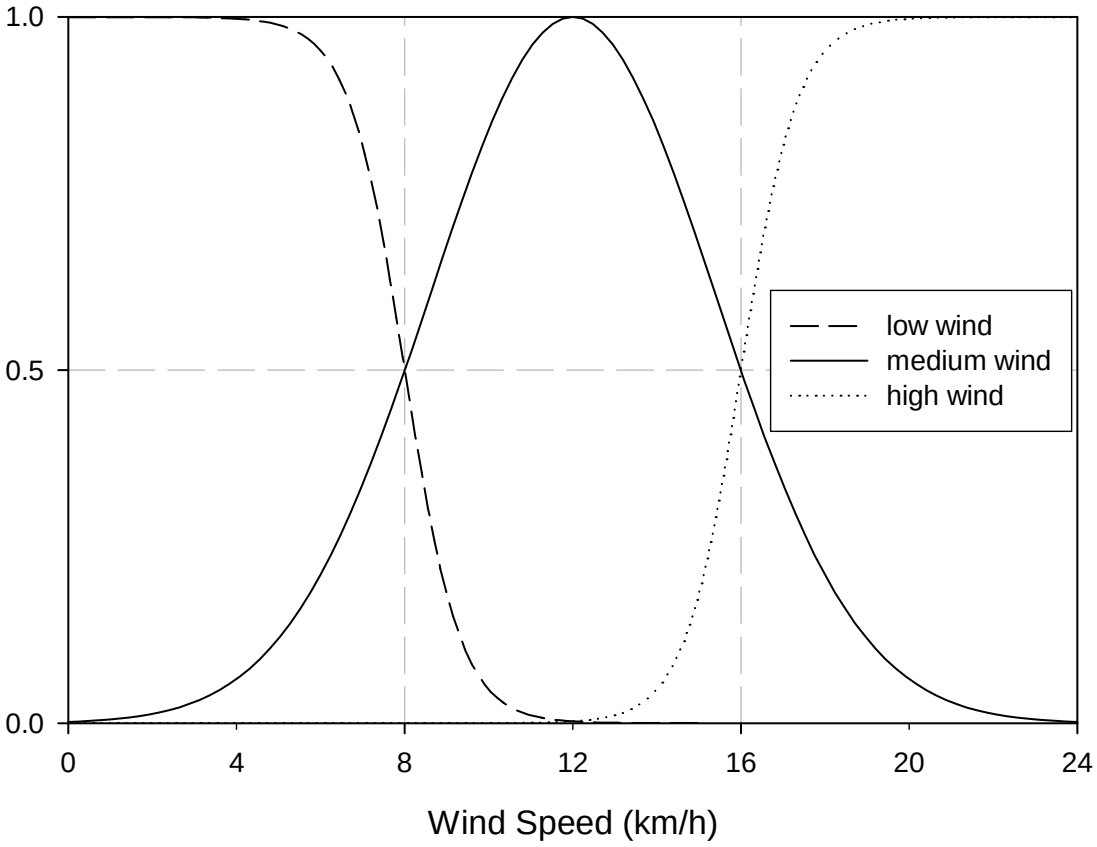

Figure 3.2: Sigmoid membership functions for low and for high wind speed and Gaussian membership function for medium wind speed.

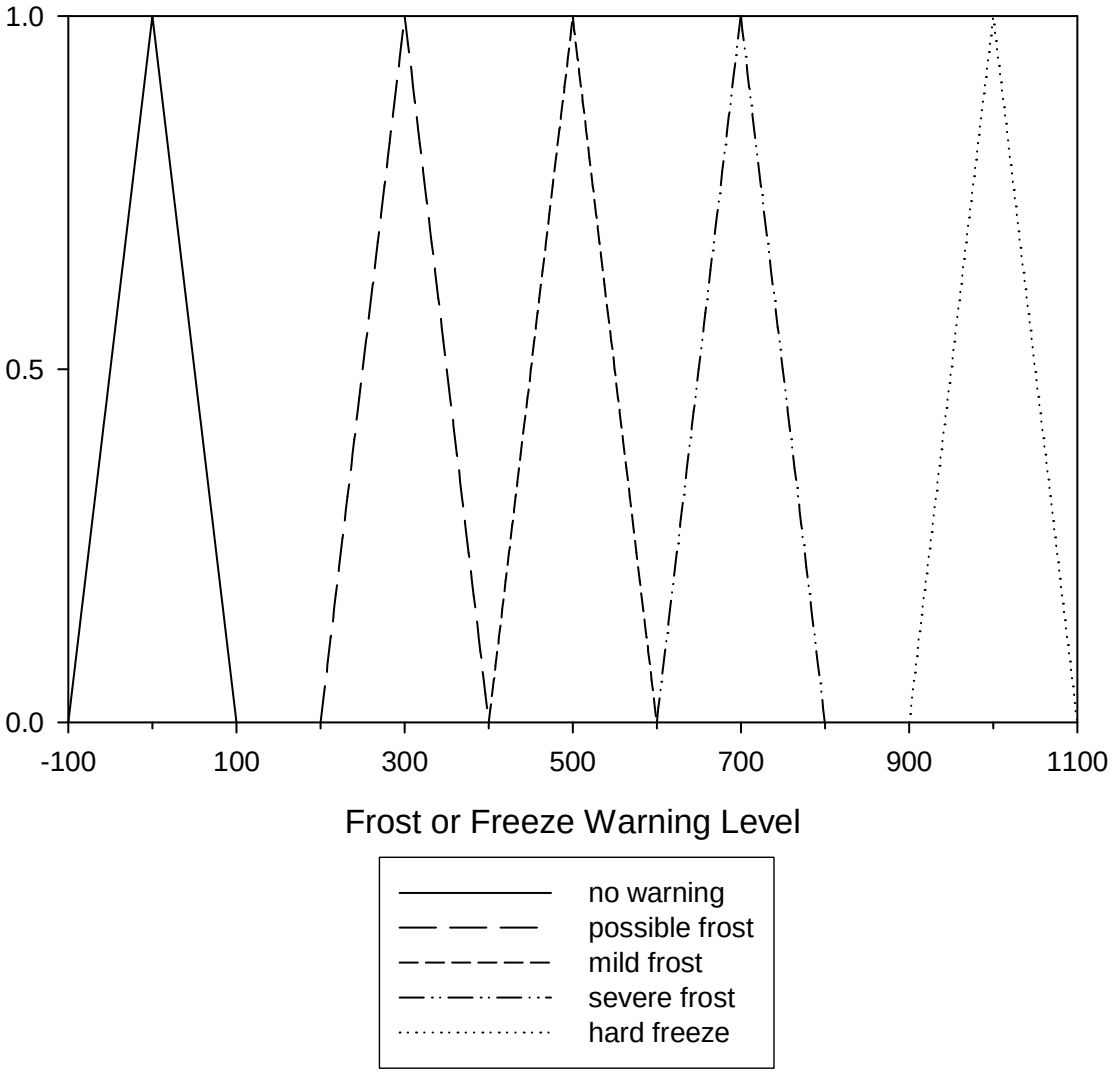

Figure 3.3: Triangular output membership functions.

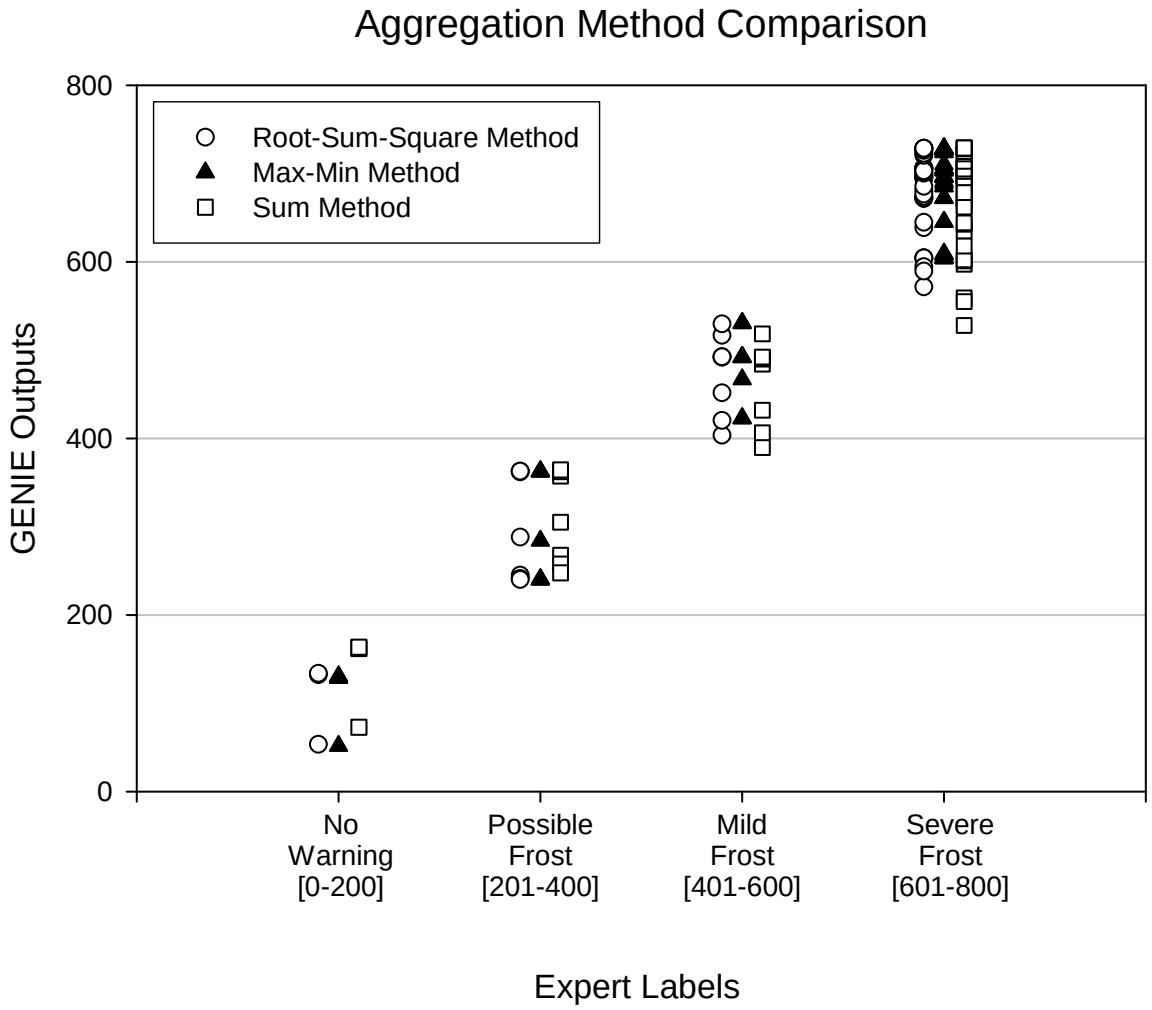

Figure 3.4: Crisp outputs obtained using different aggregation methods for every combination of air temperature (°C) and dew point temperature (°C) in the integer set [4,-5] with a wind speed of 0 km/h. Labels provided by the experts for these scenarios are listed along the bottom axis.

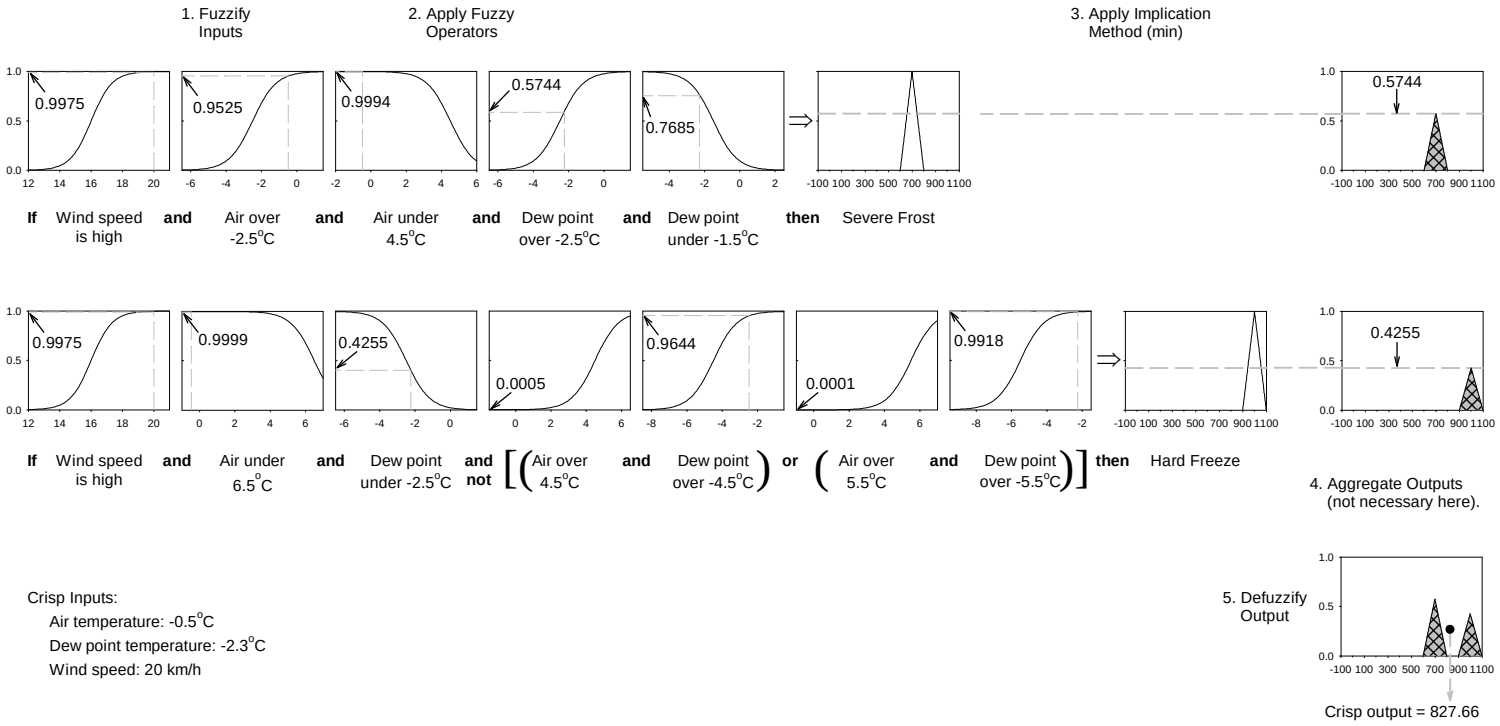

Figure 3.5: A simple example of GENIE's fuzzy inference process using two of its rules. The fuzzy operators have the following mathematical equivalences: AND = min, OR = max, NOT = complement.

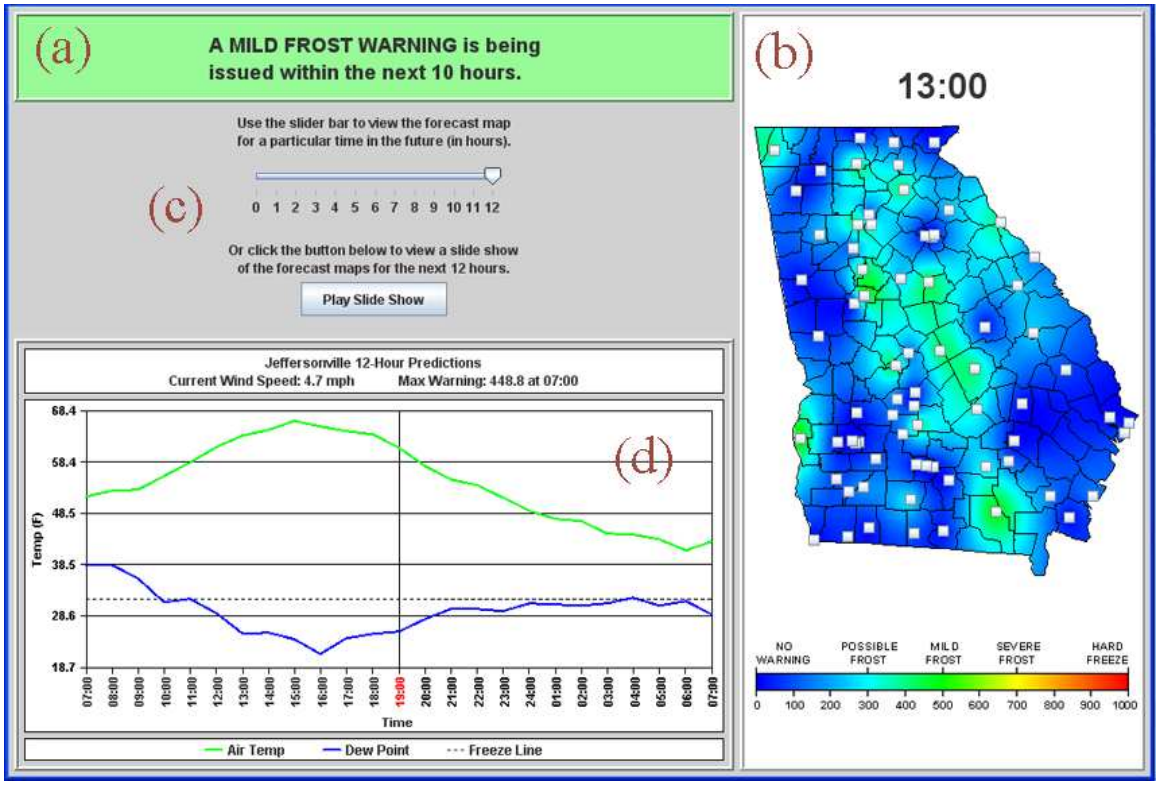

Figure 3.6: A screenshot of GENIE's web interface. (a) The most severe threat level that is being issued statewide, (b) the hourly color maps and station locations, (c) the control panel, and (d) detailed station information, including station specific maximum warning levels.

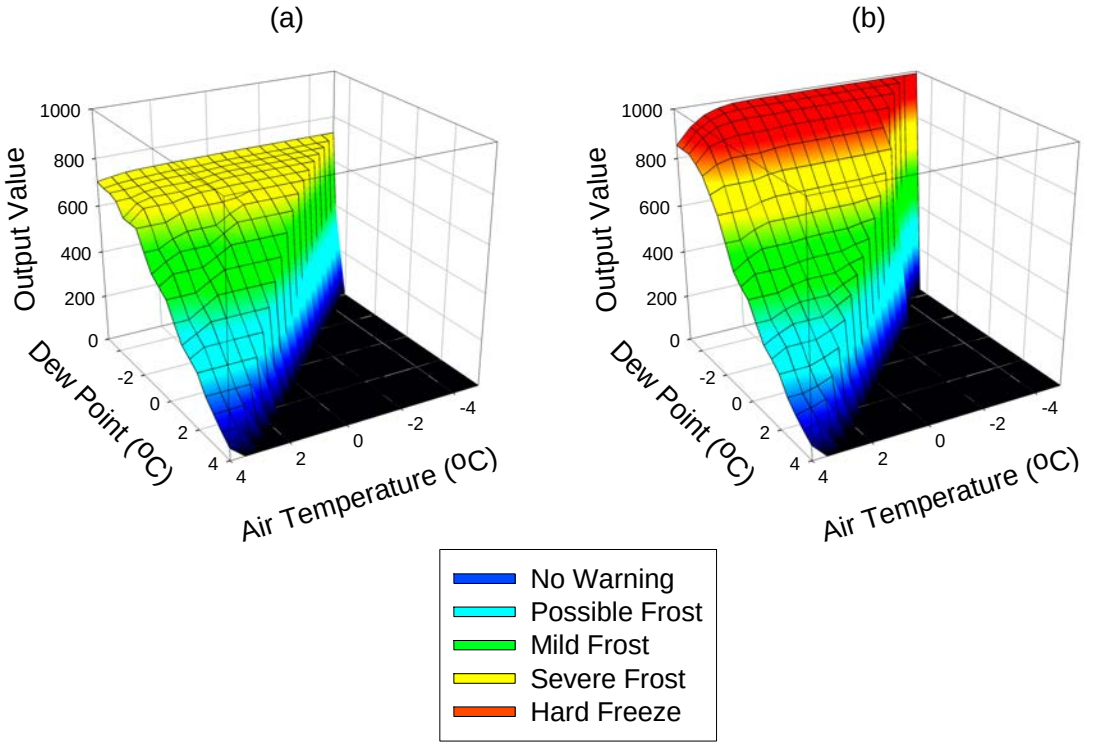

Figure 3.7: GENIE output values for various combinations of air temperature and dew point temperature. (a) Wind speed = 4 km/h, (b) Wind speed = 20 km/h.

Chapter 4

Summary and Conclusions

The two goals of this research were to (1) evaluate the capability of SVR in predicting air temperature for horizons from one to twelve hours and (2) to develop a fuzzy logic-based expert system which can be used to assist Georgia growers in predicting the risk of frost up to 12 hours in advance. In Chapter 2, a methodology was proposed which allows reduced SVR training sets to be created without a need to pre-process the data. Using this methodology, the SVR algorithm was applied to the problem of air temperature prediction, a domain in which massive amounts of data have been collected by Georgia's AEMN weather stations. Resulting SVR models were shown to be more accurate than ANN models which had been trained on data sets of over 300,000 and competitive with ANN models which had been trained on over 1.25 million patterns. In Chapter 3, the Georgia Extreme-weather Neural-network Informed Expert (GENIE) was introduced. The GENIE tool is a neural network-based fuzzy expert system that was developed to assist Georgia blueberry producers in assessing the risk of frost. The fuzzy logic rules of this system incorporated the knowledge of local agrometeorologists while considering the effects of air temperature, dew point temperature and wind speed on the severity of frost occurrences. Warnings from one to twelve hours ahead were made possible by supplying the expert system with the predictions of previously developed ANN models for air temperature and dew point temperature. Finally, a web-based interface was created so that this tool's warnings could be disseminated to Georgia farmers across the state of Georgia.

The SVR models described in this research were developed using the radial-basis function kernel. Evaluating other kernels was beyond the scope of this work, however, alternate kernels

may prove more effective. Additionally, parameter tuning was performed using only the four-hour prediction horizon data. It is likely that model accuracy would be increased if parameters were tuned for each individual prediction horizon. Future research involving SVR and air temperature prediction could also benefit from more systematic methods of creating reduced training sets. Heuristic methods attempt to exploit characteristics of support vectors in order to identify them prior to applying the SVR algorithm. This pre-processing step would be rather lengthy when dealing with the large AEMN data set, however, the benefits may prove worthwhile. The accuracy of the GENIE tool would be increased by incorporating predictive models of wind speed. Future work may involve the development of such models using ANNs or SVR. The warnings currently produced by the GENIE tool are specific to blueberries. Since each crop has a different tolerance to frost, enabling the tool to produce warnings for multiple crops would involve the addition of rules specific to each individual crop. This extension to the fuzzy rule set would be fairly straight forward and would greatly increase the GENIE tool's utility to Georgia farmers. Another useful extension to the expert system would be a more transparent explanation of the decision making process. Producers wishing to see why a particular warning was issued could benefit from a list of the rules with the highest firing strengths. Transparency will increase the producers' level of trust with the expert system and ultimately make GENIE a more effective tool.

Bibliography

Almeida, M. B., A. P. Braga, and J. P. Braga (2000). Speeding svms learning with a prior cluster selection and k-means. In *Proceedings of the 6th Brazillian Symposium on Neural Networks*, Rio de Janeiro, pp. 162–167.

Attaway, J. A. (1997). *A History of Florida Citrus Freezes*. Lake Alfred, FL: Florida Science Source, Inc.

Boatright, S. R. and J. C. McKissick (2008). 2007 Georgia farm gate value report. Technical Report AR-08-01, The University of Georgia.

Chen, K.-Y. and C.-H. Wang (2007). Support vector regression with genetic algorithms in forecasting tourism demand. *Tourism Management 28*(1), 215–226.

Cooper, W. C., R. H. Young, and F. M. Turrell (1964). Microclimate and physiology of citrus: Their relation to cold protection. *Agricultural Science Review*, 38–50.

Cortes, C. and V. Vapnik (1995). Support vector networks. *Machine Learning 20*(3), 273–297.

Cristianini, N. and J. Shawe-Taylor (2000). *An Introduction to Support Vector Machines*. Cambridge: Cambridge University Press.

Dong, B., C. Cao, and S. E. Lee (2005). Applying support vector machines to predict building energy consumption in tropical region. *Energy and Buildings 37*, 545–553.

Engelbrecht, A. P. (2002). *Computational Intelligence An Introduction*. West Sussex: John Wiley and Sons.

Fonsah, E. G., K. C. Taylor, and F. Funderburk (2007). Enterprise cost analysis for middle Georgia peach production. Technical Report AGECON-06-118, The University of Georgia Cooperative Extension.

Friess, T.-T., N. Cristianini, and C. Campbell (1998). The kernel adatron algorithm: A fast and simple learning procedure for support vector machines. In *15th International Conference on Machine Learning*, Madison, pp. 188–196. Morgan Kaufman Publishers.

Guo, G. and J.-S. Zhang (2007). Reducing examples to accelerate support vector regression. *Pattern Recognition Letters 28*, 2173–2183.

Hadjmichael, M., A. P. Kuciauskus, P. M. Tag, R. L. Bankert, and J. E. Peak (2002). A meteorological fuzzy expert system incorporating subjective user input. *Knowledge and Information Systems 4*(3), 350–369.

Hansen, B. (1997). Sigmar: A fuzzy expert system for critiquing marine forecasts. *AI Applications 11*(1), 59–68.

Hoogenboom, G. (1993). The Georgia automated environmental monitoring network. In K. J. Hatcher (Ed.), *1993 Georgia Water Resources Conference*, Orlando, pp. 398–402. The University of Georgia.

Hsu, C.-W., C.-C. Chang, and C.-J. Lin (2000). A practical guide to support vector classification. Technical report, National Taiwan University.

Jain, A., R. W. McClendon, and G. Hoogenboom (2006). Freeze prediction for specific locations using artificial neural networks. *Transactions of the ASABE 49*(6), 1955–1962.

Joachims, T. (1999). Making large-scale SVM learning practical. In B. Schölkopf, C. Burges, and A. Smola (Eds.), *Advances in Kernel Methods - Support Vector Learning*, Chapter 11, pp. 169–184. Cambridge, MA: MIT Press.

Jones, G. M. and C. C. Stallings (1999). Reducing heat stress for dairy cattle. Technical Report 404-200, Virginia Cooperative Extension.

Khan, M. S. and P. Coulibaly (2006). Application of support vector machine in lake water level prediction. *Journal of Hydrologic Engineering 11*(3), 199–205.

Martsolf, J. D., J. F. Gerber, E. Y. Chen, H. L. Jackson, and A. J. Rose (1984). What do satellite and other data suggest about past and future florida freezes? In *Proceedings of the Florida State Horticultural Society*, pp. 17–21.

MathWorks (1999). *Fuzzy Logic Toolbox for Use with MATLAB*. Natick, MA: MathWorks.

Mori, H. and D. Kanaoka (2007). Application of support vector regression to temperature forecasting for short-term load forecasting. In *IEEE Joint Conference on Neural Networks*, Orlando, pp. 1085–1090. IEEE.

Müller, K.-R., A. J. Smola, G. Rätsch, B. Schölkopf, J. Kohlmorgen, and V. Vapnik (1997). Predicting time series with support vector machines. In *ICANN '97: Proceedings of the 7th International Conference on Artificial Neural Networks*, London, pp. 999–1004. Springer-Verlag.

Osuna, E., R. Freund, and F. Girosi (1997). An improved training algorithm for support vector machines. In *Neural Networks for Signal Processing VII. Proceedings of the 1997 IEEE Workshop*, New York, pp. 276–285. IEEE.

Perry, K. B. (1994). Frost/freeze protection for horticultural crops. Leaflet 705-A, North Carolina Cooperative Extension Service.

Platt, J. C. (1999). Fast training of svms using sequential minimal optimization. In B. Schölkopf, C. Burges, and A. Smola (Eds.), *Advances in Kernel Methods - Support Vector Learning*, pp. 185–208. Cambridge, MA: MIT Press.

Qin, Z., Q. Yu, J. Li, Z. yi Wu, and B. min Hu (2005). Application of least squares vector machines in modelling water vapor and carbon dioxide fluxes over a cropland. *Journal of Zhejiang University 6*(6), 491–495.

Rychetsky, M., S. Ortmann, M. Ullmann, and M. Glesner (1999). Accelerated training of support vector machines. In *IJCNN '99 International Joint Conference on Neural Networks*, Volume 2, pp. 998–1003.

Schölkopf, B., C. Burges, and V. Vapnik (1996). Incorporating invariances in support vector learning machines. In C. von der Malsburg, W. V. Seelen, J. Vorbruggen, and B. Sendhoff (Eds.), *ICANN '96 International Conference on Artificial Neural Networks*, Volume 1112, Berlin, pp. 47–52. Springer.

Schölkopf, B. and A. J. Smola (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge: MIT Press.

Shank, D. B., R. W. McClendon, J. O. Paz, and G. Hoogenboom (2008). Ensemble artificial neural networks for prediction of dew point temperature. Accepted for publication by *Applied Artificial Intelligence*.

Smith, B. A., G. Hoogenboom, and R. McClendon (2008). Artificial neural networks for automated year-round temperature prediction. Submitted for publication to *Computers and Electronics in Agriculture*.

Smith, B. A., R. W. McClendon, and G. Hoogenboom (2006). Improving air temperature prediction with artificial neural networks. *International Journal of Computational Intelligence 3*(3), 179–186.

Smola, A. J. and B. Schölkopf (2004). A tutorial on support vector regression. *Statistics and Computing 14*, 199–222.

Snyder, R. L. and J. P. de Melo-Abreu (2005). *Frost Protection: fundamentals, practice and economics Volume 1.* Rome: Food and Agriculture Organization of the United Nations.

Takle, E. S. (1990). Bridge and roadway frost: Occurrence and prediction by use of an expert system. *Journal of Applied Meteorology 29*, 727–734.

Temeyer, B. R., W. A. G. Jr., K. A. Jungbluth, D. Burkheimer, and D. McCauley (2003). Using an artificial neural network to predict parameters for frost deposition on Iowa bridgeways. In *2003 Mid-Continent Transportation Research Symposium*, Ames.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. New York: Springer-Verlag.

Ward Systems Group (1993). *Manual of NeuroShell 2*. Ward Systems Group.

White, G. F. and J. E. Haas (1975). *Assessment of Research on Natural Hazards*. Cambridge: MIT Press.

Yang, M.-H. and N. Ahuja (2000). A geometric approach to train support vector regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Hilton Head, SC, pp. 430–437.

Zadeh, L. (1965). Fuzzy sets. *Information and Control*, 338–353.

Zhang, Z., X. Zhang, and P. de Harrington (2007). Support vector regression and radial basis function neural networks applied to semi-quantitative prediction of rhubarbs. In *Third International Conference on Natrual Computation ICNC 2007*, Volume 1, Haikou, pp. 661–664.

Appendix

Support Vector Regression

Support Vector Regression (SVR) is a machine learning tool which uses a nonlinear mapping to transform input data into a higher-dimensional feature space where linear regression can be performed (Vapnik, 1995). SVR uses input-output pairs to develop a function which not only fits the training data, but which also maintains the highest possible degree of generality so as to be applicable to future, unseen data.

Typically, SVR is implemented as ε-SVR, which makes use of an ε-insensitive loss function (Vapnik, 1995). This type of loss function allows for deviation from the true targets by at most ε. The ε-SVR algorithm (referred to simply as SVR from this point forward) does not recognize errors as long as they fall within $\pm\varepsilon$ of the learned function.

Training data is usually of the form:

$$\{(\bar{x}_1, y_1), (\bar{x}_2, y_2), \cdots, (\bar{x}_l, y_l)\} \tag{1}$$

$$\bar{x}_i \in \mathbb{R}^d,\ y_i \in \mathbb{R},\ \text{and } l = \text{ number of examples}$$

Each pattern, therefore, contains a vector, $\bar{x}_i$ of real-valued inputs as well as a scalar, y_i representing the target value. In the case of a linear function, f, the solution would be given in the form:

$$f(\bar{x}) = \langle \bar{w}, \bar{x} \rangle + b \tag{2}$$

$$\bar{w} \in \mathbb{R}^d,\ \ \in \mathbb{R}$$

The term $\bar{w}$ represents a vector of weights which have been learned by the SVR algorithm. The notation $\langle \cdot, \cdot \rangle$ denotes the inner product of two vectors. As noted earlier, it is desirable

for f to have a high degree of generality. In other words, it should be loosely fit to the training data as to avoid overfitting. SVR accomplishes this by forcing the function to be as *flat* as possible. Equation (2) is considered *flat* if $\bar{w}$ is small, which can be achieved by minimizing the norm[1] of $\bar{w}$ (Smola and Schölkopf, 2004). The entire problem is therefore, formulated as the following convex optimization problem:

$$\begin{aligned} &\text{minimize} && \frac{1}{2}\|\bar{w}\|^2 \\ &\text{subject to} && \begin{cases} y_i - \langle \bar{w}, \bar{x}_i \rangle - b & \leq \quad \varepsilon \\ \langle \bar{w}, \bar{x}_i \rangle + b - y_i & \leq \quad \varepsilon \end{cases} \end{aligned} \tag{3}$$

This optimization problem assumes that a function exists which approximates every pair $(\bar{x}_i, y_i)$ in the training data with ε precision. To deal with instances where this is not the case, and to account for errors in the training data, Vapnik introduced two slack variables, ξ_i and ${\xi_i}^*$. These slack variables compute the error for underestimating and overestimating the true value. With the addition of these two variables, the optimization problem in Equation (3) becomes:

$$\begin{aligned} &\text{minimize} && \frac{1}{2}\|\bar{w}\|^2 + C\sum_{i=1}^{l}(\xi_i + {\xi_i}^*) \\ &\text{subject to} && \begin{cases} y_i - \langle \bar{w}, \bar{x}_i \rangle - b & \leq \quad \varepsilon + \xi_i \\ \langle \bar{w}, \bar{x}_i \rangle + b - y_i & \leq \quad \varepsilon + {\xi_i}^* \\ \xi_i, {\xi_i}^* \geq 0 \end{cases} \end{aligned} \tag{4}$$

This is the primal form of the objective function for linear SVR. Here, C is a constant, known as the penalty factor, which controls the trade-off between the complexity of the function and the frequency with which errors are allowed when estimating the true value of the training patterns. The higher the value of C, the greater the influence of the empirical error component on the optimization problem. This component $(\xi_i + {\xi_i}^*)$ amounts to any difference, greater than ε between the estimated values and the target values.

[1]The norm of a vector is a characterization of its length. The most common form is the Euclidean norm, $\|\bar{x}\| = \sqrt{{x_i}^2 + {x_2}^2 + \cdots + {x_n}^2}$. Note that this is equivalent to the 2-norm, $\|\bar{x}\|_2 = \left(\sum_{i=1}^{n} |x_i|^2\right)^{\frac{1}{2}}$

The primal form of the optimization problem involves handling inequality constraints directly, which can be difficult. It is often easier to solve an alternate, dual form of the problem (Smola and Schölkopf, 2004). A dual description of the problem is obtained by constructing a Lagrange function, which introduces dual variables. The Lagrange function, L, constructed from the primal objective function in Equation (4) is shown below.

$$\begin{aligned} L = \ & \frac{1}{2}\|\bar{w}\|^2 + C\sum_{i=1}^{l}(\xi_i + {\xi_i}^*) - \sum_{i=1}^{l}(\eta_i\xi_i + {\eta_i}^*{\xi_1}^*) \\ & - \sum_{i=1}^{l}\alpha_i(\varepsilon + \xi_i - y_i + \langle\bar{w}, \bar{x}_i\rangle + b) \\ & - \sum_{i=1}^{l}{\alpha_i}^*(\varepsilon + {\xi_i}^* + y_i - \langle\bar{w}, \bar{x}_i\rangle - b) \end{aligned} \tag{5}$$

Here, $\eta_i, {\eta_i}^*, \alpha_i, {\alpha_i}^*$ are the dual variables, or Lagrange multipliers. It is known that the partial derivatives of L with respect to the primal variables $(\bar{w}, b, \xi_i, {\xi_i}^*)$ must be equal to zero (Cristianini and Shawe-Taylor, 2000). We can, therefore, eliminate these partial derivatives from Equation (5) resulting in the following dual optimization problem:

$$\begin{aligned} \text{maximize} \quad & \begin{cases} -\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i - {\alpha_i}^*)(\alpha_j - {\alpha_j}^*)\langle\bar{x}_i, \bar{x}_j\rangle \\ -\varepsilon\sum_{i=1}^{l}(\alpha_i + {\alpha_i}^*) + \sum_{i=1}^{l} y_i(\alpha_i - {\alpha_i}^*) \end{cases} \\ \text{subject to} \quad & \sum_{i=1}^{l}(\alpha_i - {\alpha_i}^*) = 0 \text{ and } \alpha_i, {\alpha_i}^* \in [0, C] \end{aligned} \tag{6}$$

Another interesting consequence of deriving the partial derivatives of L with respect to the primal variables is that we can obtain the following equation:

$$\bar{w} = \sum_{i=1}^{l}(\alpha_i - {\alpha_i}^*)\bar{x}_i \tag{7}$$

This equation allows $\bar{w}$ to be described as a linear combination of the training patterns. Substituting this value into Equation (2) the solution function now appears in the following form:

$$f(\bar{x}) = \sum_{i=1}^{l}(\alpha_i - {\alpha_i}^*)\langle\bar{x}_i, \bar{x}\rangle + b \tag{8}$$

It is important to note that the complete algorithm can be expressed in terms of inner products between the data. Additionally, the α values will be zero for many of the training patterns, which provides for a sparse solution. The patterns whose α values are not set to zero are referred to as the support vectors.

The algorithm and equations described so far have been for the simple case when SVR seeks to find a linear function. In order to deal with nonlinear functions, the data must be transformed from the input space to a higher dimensional feature space using a nonlinear transformation function,Φ. A straightforward extension of linear SVR would be to simply substitute $\Phi(\bar{x})$ for every instance of $\bar{x}$ in Equation (6) and Equation (8). This approach quickly becomes infeasible, however, as $\bar{x}$ is mapped into higher and higher dimensions. The solution is to avoid mapping the data altogether through what is known as the "kernel trick".

A kernel takes advantage of Mercer's Theorem, shown in Equation (9), which states that for certain mappings, Φ, and any two mapped points $\bar{u}$ and $\bar{v}$, the dot product of these points can be evaluated using a kernel function, k, without ever explicitly knowing the mapping (Schölkopf and Smola, 2001).

$$\langle \Phi(\bar{u}), \Phi(\bar{v}) \rangle \equiv k(\bar{u}, \bar{v}) \tag{9}$$

The polynomial function of degree 2, for instance, has a correspondence with two points which have been mapped from $\mathbb{R}^2 \rightarrow \mathbb{R}^3$ using the transformation function below:

$$\Phi\begin{pmatrix} s_1 \\ s_2 \end{pmatrix} = \begin{pmatrix} {s_1}^2 \\ \sqrt{2}s_1 s_2 \\ {s_2}^2 \end{pmatrix} \tag{10}$$

In this equation, s_1 and s_2 refer to the components of $\bar{s}$. As an example, consider the points $\bar{u} = \binom{2}{3}$ and $\bar{v} = \binom{4}{5}$. In order to calculate $\langle \Phi(\bar{u}), \Phi(\bar{v}) \rangle$ without a kernel function, each point would first need to be mapped into the higher dimension as follows:

$$\Phi\begin{pmatrix} 2 \\ 3 \end{pmatrix} = \begin{pmatrix} 4 \\ 6\sqrt{2} \\ 9 \end{pmatrix} \tag{11}$$

$$\Phi\begin{pmatrix}4\\5\end{pmatrix} = \begin{pmatrix}16\\20\sqrt{2}\\25\end{pmatrix} \tag{12}$$

Then the inner product would need to be calculated in this higher dimensional space as shown below:

$$\left\langle \begin{pmatrix}4\\6\sqrt{2}\\9\end{pmatrix}, \begin{pmatrix}16\\20\sqrt{2}\\25\end{pmatrix} \right\rangle = (4\times 16) + (6\sqrt{2}\times 20\sqrt{2}) + (9\times 25) \tag{13}$$

$$= 64 + 240 + 255$$

$$= 529$$

Using the polynomial kernel, the solution is simply derived as follows:

$$\left\langle \Phi\begin{pmatrix}2\\3\end{pmatrix}, \Phi\begin{pmatrix}4\\5\end{pmatrix} \right\rangle = \left\langle \begin{pmatrix}2\\3\end{pmatrix}, \begin{pmatrix}4\\5\end{pmatrix} \right\rangle^2 \tag{14}$$

$$= (8+15)^2$$

$$= 529$$

When using a kernel such as this in place of very high dimensional transformations, the computational cost savings are substantial.

Since the SVR algorithm only depends on the inner products between the patterns, it can be easily augmented to deal with nonlinear functions by simply replacing these inner products with the kernel function, producing the following new optimization problem:

$$\begin{aligned} \text{maximize} \quad & \begin{cases} -\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i - \alpha_i{}^*)(\alpha_j - \alpha_j{}^*)k(\bar{x}_i, \bar{x}_j) \\ -\varepsilon\sum_{i=1}^{l}(\alpha_i + \alpha_i{}^*) + \sum_{i=1}^{l} y_i(\alpha_i - \alpha_i{}^*) \end{cases} \\ \text{subject to} \quad & \sum_{i=1}^{l}(\alpha_i - \alpha_i{}^*) = 0 \text{ and } \alpha_i, \alpha_i{}^* \in [0, C] \end{aligned} \tag{15}$$

Likewise, the solution function can be written in the form:

$$f(\bar{x}) = \sum_{i=1}^{l}(\alpha_i - \alpha_i{}^*)k(\bar{x}_i, \bar{x}) + b \tag{16}$$

Although it is intuitive to think of SVR as performing linear regression in the higher-dimensional feature space, in practice, the kernel allows SVR to perform all the necessary computations in the lower-dimensional input space.