THESIS

ROBUST AND INTERPRETABLE STATISTICAL MODELS FOR PREDICTING THE INTENSIFICATION OF TROPICAL CYCLONES

Submitted by

Kyriakos C. Chatzidimitriou

Department of Computer Science

In partial fulfillment of the requirements

for the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Summer 2006

COLORADO STATE UNIVERSITY

June 16, 2006

WE HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER OUR SUPERVISION BY KYRIAKOS C. CHATZIDIMITRIOU ENTITLED ROBUST AND INTERPRETABLE STATISTICAL MODELS FOR PREDICTING THE INTENSIFICATION OF TROPICAL CYCLONES BE ACCEPTED AS FULFILLING IN PART REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE.

Committee on Graduate Work

_____
Committee Member

_____
Committee Member

_____
Adviser

_____
Co-Adviser

_____
Department Head

ABSTRACT OF THESIS

ROBUST AND INTERPRETABLE STATISTICAL MODELS FOR PREDICTING THE INTENSIFICATION OF TROPICAL CYCLONES

Progress on tropical cyclone (TC) intensity forecasting has been much slower than improvements made on predicting the path of a TC. Statistical models such as the Statistical Hurricane Intensity Prediction Scheme (SHIPS) generally outperform numerical models that try to model the dynamics of hurricanes. On the contrary, SHIPS is a simple linear regression model, exhibiting certain drawbacks from a theoretical machine learning perspective. Further, more sophisticated statistical techniques are available, making the major objective of this thesis providing an answer to the question if a linear model the best we can do. To supply an answer, a complete and meticulous procedure will be presented for deriving robust and interpretable statistical models, ranging from learning curves and model assessment methods to feature selection procedures and application of non-linear models. Even though their accuracy is of the same range as that of SHIPS,

knowledge gained from this process conveyed to the identification of new, non-linear features that human experts judge to be significant new findings. Their inclusion led to a more accurate model. Finally, certain limitations are identified and guidelines are given for further improvements.

Kyriakos C. Chatzidimitriou
Department of Computer Science
Colorado State University
Fort Collins, CO 80523
Summer 2006

ACKNOWLEDGEMENTS

First of all, I would like to thank my advisor Dr. Charles Anderson for his directions, suggestions and feedback that led to the completion of this thesis. I would also like to thank Dr. Asa Ben-Hur whose guidance and assistance on various machine learning topics has been invaluable. Additional thanks to Dr. V Chandrasekar for accepting a position on my committee. I am also grateful to Dr. Mark DeMaria for explaining the SHIPS model, providing the data and for his substantial feedback. His work on predicting hurricane intensification through statistical methods was my starting point.

DEDICATION

This thesis is dedicated to whoever helped me, convinced me and supported me to pursue my dream and study in the United States. It has been an amazing experience. Thank you all.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

Tropical cyclones (TCs) have received wide attention from the atmospheric science community over the years, mainly due to their complicated behavior and their after-effects. TCs can develop to be spectacular natural phenomena and an important part of the atmospheric circulation system, but on the other hand they can be quite catastrophic and deadly. At a higher level, TCs are converting heat energy from the ocean to mechanical energy, forming a positive feedback loop, which, along with other factors, leads to their formulation and intensification [Ema03]. TCs originate over tropical or subtropical waters with surface temperatures above 26^oC. They dissipate over land or cold water due to increased friction and/or insufficient energy sources. TCs are categorized based on their intensity, which is measured as 1-minute average sustained surface wind speed. They start as tropical disturbances to later become tropical depressions with maximum winds below 33 knots[1]. The next stage is for a tropical depression to become a tropical storm with maximum winds between 34 to 63 knots. After that TCs are fully developed and are called hurricanes in the western North Atlantic and the eastern North Pacific regions, typhoons in the western North Pacific and severe cyclones or cyclonic storms

[1]Measure of velocity in nautical miles per hour, 1 knot = 1.85 km/hour.

elsewhere [Bow95]. When above 64 knots, they can be further classified as category 1 to category 5 storms based on the Saffir-Simpson Hurricane Scale [BRJL05], with hurricanes larger than category 3 being major hurricanes. The hurricane season is between June and November with its peak in September. Since this study is concerned with the western North Atlantic region, the terms tropical cyclone and hurricane will be used interchangeably. See [Ema03] for a recent overview of the theory of hurricanes.

As mentioned earlier, hurricanes can be quite catastrophic and deadly. See [BRJL05] for facts and numbers justifying this conclusion. For several years researchers have been trying to provide accurate early warnings based on predictions of the behavior of TCs. The overall objective is to heighten awareness in the communities residing along the hurricane's path. Additionally, accurate warnings are of great importance since the costs of an evacuation are huge, ranging from property protection expenses [Man96] to cessation in oil and gas production due to the evacuation of drilling rigs [CJPB04]. Moreover, well established forecast models can help provide a better comprehension of the processes underlying the TC behavior, leading to a better understanding of the atmospheric system and the forecasting of global climate changes [Ema03].

To achieve more precise forecasts, researchers are trying to understand the behavior of TCs throughout their duration in every possible aspect. Mainly, the field is divided into four distinct categories [WRFDM06]:

1. estimating the probability of TC formation at any given time;
2. predicting storm intensity for up to 120 hours into the future;
3. forecasting the hurricane track; and
4. predicting the maximum extent of storm surge flooding.

So far, researchers are capable of forecasting the track with 90% accuracy and up to 72 hours ahead [Abe01], while intensity and formation forecasting accuracy are lagging

far behind. For intensity, even though theoretical bounds exist in the form of maximum potential intensity (MPI) theories, hurricanes never reach them, due to a number of limiting factors (both model and environmental based), making statistically based methods the best for operational forecasts [WW04].

The motivation for the work described here was that since the best predictions are statistically based, recently developed techniques from machine learning and statistical inference can help to create, in many ways, better models. To date, the prediction of intensity change has been sufficiently addressed using multiple linear regression (MLR) in practice, while efficient non-linear regression methods, like neural networks (NNs), can be found in the literature. On the other hand, the procedures for feature selection and for reporting the predictive performance of the derived models have not been investigated to a great extent, in the sense that (1) they widely vary, so comparisons between models are made in an ad-hoc basis; (2) the derived models have an inherent selection bias, i.e. allowance to peek in the test set during feature selection, prohibiting good generalization behavior [AM02]; and (3) they are unstable in terms of performance and understanding [GE03]. For example, it is often the case that at certain seasons the models perform extremely well and in others quite unsatisfactory, while there is a constant update in the set of features used, lowering the interpretability of the models. Moreover, recently developed rule based regression techniques are applied to the dataset in order to identify more elaborate structure behind the intensity predictions that a simple linear model is unable to capture. So based on the above considerations the goal is to further investigate if there is any room for improvement both in accuracy and interpretability.

1.1 Objectives

The goal of this study is twofold: (a) to build robust models; and (b) to build models in formats that experts can interpret easily, delivering additional knowledge about the prob-

lem. Robustness in this context can be defined as a property of a model that is constantly accurate across hurricane seasons or in other words is able to generalize well on unseen data. For deriving such models the Ockham's razor principle was used (this is the same principle as the bias-variance trade-off, feature selection, regularization etc): complexity (in our case extra features) must pay for itself by giving a significant improvement in the error rate during the training procedure and between different seasons [CST00]. This principle is quantified in Chapter 3. Recently developed rule based regression schemes are also a focus of the work presented here. They are applied to the dataset in order to identify more elaborate structure behind the intensity predictions. MLR and NNs fail to provide the human expert with interpretable results regarding possible multiple inter-dependencies of the inputs and the output. In contrast, rule based methods are not only competitive with respect to prediction performance, but also support the capability of discovering multiple correlations in the dataset in an easy to read and validate manner. This could potentially aid the analytical formulation of the problem.

To accomplish the objectives of this thesis the steps taken can be summarized as follows:

- find important properties of the hurricane dataset and its behavior with respect to the different prediction intervals;
- evaluate model assessment methods and provide a framework for future model evaluations;
- apply feature selection with robustness in mind and with different biases of searching the space of features;
- evaluate linear and non-linear models based on the previous results;
- create interpretable rule-based models, measure their accuracy and establish their validity; and

- incorporate extracted knowledge from the rule based methods into the models and assess the efficiency of such approach.

1.2 Organization

This thesis is organized as follows:

The present chapter, Chapter 1, gives a brief description of tropical cyclones and the justification behind this work, along with its objectives.

Chapter 2 provides a background related to this work. First of all, it reviews the literature of intensity prediction models, their development and results, in order for comparisons to be made with the present study. Secondly a detailed description of the dataset used will be given, along with details of the Statistical Hurricane Intensity Prediction System (SHIPS), derived by the same dataset and currently being the state of the art.

Chapter 3 presents the methods and models used in this work in the order applied to the dataset. This chapter is by no means a comprehensive review of the algorithms and the user is referred to the provided bibliography for that. The main goal of this section is to provide justification as to what methods were used and give a summary so that the reader is familiar with the basics of each method.

All the results and the discussions are accumulated in Chapter 4, written in the order derived, since outcomes of one method are inputs for another methods. The results are oriented both in establishing the robustness and the interpretability of the induced models. Comparisons are made to the SHIPS model.

Last but not least, Chapter 5 summarizes the findings of this research, reaches conclusions and points to directions of future work.

Chapter 2

Background

This chapter is concerned with getting the reader acquainted with the literature of TC intensity prediction and especially with the statistical models that have been developed by researchers in the area. Moreover, the derivation of the dataset used for producing the SHIPS model will be explained in detail.

2.1 Related Work

This section presents the research conducted and the most important results derived in the area of TC intensity prediction. The focus is on the statistical and machine learning models that can be found in the literature.

The models for predicting the intensification of TCs can be divided into three categories [Cas04]: (1) Statistical models that use climatological and persistence predictors; (2) statistical-dynamical models that in addition to the above inputs also use predictors from numerical models, known as synoptic predictors; and (3) numerical models that computationally simulate the hurricane through physical principles. One can also add the research conducted to compute the upper bound of the intensity of TCs, also known as maximum potential intensity (MPI) [WW04].

Statistical models were the first to be developed of the three aforementioned ones. They were based on climatology and persistence predictors. Climatology refers to mea-

surements of climatological values such as temperature, pressure and wind, while persistence refers to the current conditions such as the previous intensity change. Statistical Hurricane Intensity Forecast (SHIFOR) was the first operational model for the Atlantic Basin and was based on seven predictors along with second and third order products of them, for a total of ten [JN79]. The predictors were: Julian date, initial storm intensity, intensity change the past 12 hours (h), initial storm latitude and longitude, and eastward and northward components of the storm motion vector. SHIFOR was developed in 1979 and was capable of predicting intensity changes up to 72 h ahead. A more recent statistical model based on climatology and persistence capable of predicting up to 120 h ahead is described in [KDSG03].

The best operational model to date is statistical-dynamical, named Statistical Hurricane Intensity Prediction Scheme (SHIPS) [DK94b, DK99, DMS$^+$05]. Besides climatology and persistence, it includes synoptic predictors from numerical models as well. Synoptic predictors are related to the current and future environmental conditions and sea surface temperature. This approach is called statistical-dynamical [KDS04]. SHIPS, as SHIFOR, uses multiple linear regression to derive the coefficients for the predictors and the second order terms. The predictors are selected from a wide range of variables based on backward elimination. SHIPS was shown to outperform the SHIFOR models by 10 to 15% [DK94b]. Moreover, SHIPS predicts intensity changes up to 120 h into the future. The SHIPS predictors are presented in the next section.

To boost the predicting accuracy of the SHIPS model, two additional improvements were incorporated. It is well known that the sea surface temperature (SST) is one of the most important factors for changing the intensity of TCs and has a one-to-one relationship with the maximum potential intensity of a TC [DK94a]. Thus a more accurate representation of SST through a higher resolution, is shown to reduce errors in predictions [BSD04]. Extending the set of predictors with satellite information has resulted in

improving the SHIPS model forecasts. The extra predictors include the geostationary operational environmental satellites (GOES) infrared imagery (10.7 μm) and ocean heat content (OHC) estimated from satellite altimetry data [DMS$^+$03].

In the third category is the model created in and named after the Geophysical Fluid Dynamic Laboratory (GFDL), making both track and intensity predictions. For a detailed description the reader is referred to [KBTR95]. SHIPS outperformed GFDL in a recent comparison, especially at the early forecast stages [DMS$^+$05]. These three models are included in the suite of the National Hurricane Center (NHC). Their predictions provide guidance to the NHC experts in order to issue the official forecasts.

Besides the use of MLR to build prediction models, non-linear methods have also been popular and showed promising results. Feedforward back propagation neural networks have been applied to both the Atlantic and Pacific basins using the predictors of either statistical and statistical-dynamic models [BH98, BH00, McG04]. The limitation in the literature of the NN models is that they have been built to predict for up to 72 hours ahead, while MLR has been evaluated for up to 120 hours. Additionally, they outperform MLR at only certain time periods, while their black-box nature makes them less interpretable. NNs also have been used in more elaborate settings. In a work described in [Cas04, RBV04], NNs have been applied as self-organizing maps (SOMs) to identify storms with similar behavior. Then the synoptic, persistence and climatological observations of the analog hurricanes, after a feature selection procedure, are used to train a feedforward NN to predict the hurricane intensity.

A recent approach to TC intensity prediction is the use of data mining (DM) methods, such as association rules (AR), to unravel coupling relationships of the physical processes controlling hurricane intensification [TYK05]. Results were promising in finding interesting situations including two, three and more variables that result in the intensification, weakening and little or no intensity change of a TC. One of the limitations of

this work was the fact that the authors approached the task as a classification problem predicting three classes, that is if the hurricane was going to intensify, abate or retain its current wind speed. In the work presented here quantitative output will be provided to the user using like in [TYK05] models comprised of rules. Additionally, only a small subset of the set of predictors presented here was used, with fewer data samples used for the developmental set, while the models built were capable of predictions only 6 hours ahead.

2.2 The Dataset

The main task is to predict by how much the storm intensity, measured by the maximum wind in knots (kt) observed within one minute on the surface of the earth at which the measurement was taken, will change in the next hours. In other words the dependent variable of the models is the intensity deviation in the next hours (DV). Predictions at time $t = 0$ are made for the next 6 h, up to 120 h, in 6 h intervals. This means that each report contains 20 forecasts for the next 5 days (0 to 6, 0 to 12, 0 to 18, ..., 0 to 120 h).

The predictors are the same ones used to derive the SHIPS model. They include climatology, persistence and synoptic parameters. The SHIPS model is based on the perfect prog approach (from the phrase "perfect prognosis"), meaning its predictors are calculated based on the "best-track" data, prepared (post-processed) by the NHC. These data include intensity and track information. The best-track data are not available when the SHIPS model runs in operational mode, but for our purposes training and testing will be made using the "perfect prog" model, since this is the current practice. This is a logical choice, since we are interested in obtaining a robust model, based on less noisy data. Additionally, the models will be more accurate in operational mode once the on-line intensity and track forecasts improve and deviate less from the "best track" data. Finally, since the focus is on using alternative methodologies and on measuring

the accuracy of the derived models, rather than estimating their operational skills, the "perfect prog" approach is well suited.

Our focus will be on TCs formed over the Atlantic basin. When a TC is generated, data are available four times a day at 0000, 0600, 1200 and 1800 UTC, for nearly all named hurricanes. In order to predict from 6 to 120 h ahead and every 6 h, 20 datasets are created and 20 models are trained for predicting intensity changes for each period separately, i.e. from 0 to 6, 0 to 12, 0 to 18 h ahead etc. To create the first dataset, that is from 0 to 6 h, for every hurricane, all measurement at any given time and day are taken as predictors along with the intensity change 6 h later than the given time and day. For example, all the parameter values at time 0000 on 9/1/2006 are the predictors and the intensity change at time 0600 9/1/2006 is the desired output. The same applies to measurements at time 0600 9/1/2006 to 1200 9/1/2006, 1200 9/1/2006 to 1800 9/1/2006, 1800 9/1/2006 to 0000 9/2/2006 etc. For the 0 to 12 h ahead dataset, predictors at, for example, time 0000 9/1/2006 are used to forecast the intensity change at time 1200 9/1/2006, at time 0600 9/1/2006 for time 1800 9/1/2006, at time 1200 9/1/2006 for time 0000 9/2/2006, etc. The same procedure applies to all 20 datasets. It is obvious that the number of samples in each dataset is declining, since the sampling rate is decreasing and because many hurricanes abate before reaching five days of life.

Some of the predictors are denoted as static (S), while others as time dependent (T). Static predictors are evaluated only at time $t = 0$ and the same value is used for all 20 intervals. Time $t = 0$ is relative and corresponds to the time the predictor values were measured. For example, let us say one predictor, the initial maximum winds, is measured at time 0600 9/1/2006. It will be used as an input for the 20 tuples that have their output (DV) calculated at times 1200 9/1/2006, 1800 9/1/2006, 0000 9/2/2006 etc, for the datasets predicting 6, 12 and 18 h ahead, respectively. The time dependent variables are averaged along the track. So, for example, if we want to predict the intensity at

time 1800 based on data at time 0000 of the same day (that will be for the 18 h ahead dataset), time dependent variables are the mean of their values from time 0000 till 1800. This is required to produce more accurate predictions since the position of the storm changes and so is the environment of the storm. For example, the sea surface temperature changes in accordance with the position of the TC. For the models the track of the hurricane is known from the "best track" models. In practice, fairly accurate track forecasting models are used. If the errors are not large a variable is used as a time dependent one, else as static. The full set of predictors is presented in Table 2.1 and the reader is referred to the same table for a categorization of the predictors as static or time-dependent. SHIPS also includes three second-order terms: $VMAX \times INCV$, $VMAX \times SHRD$ and POT^2 for a total of 16 predictors. Data are available from seasons 1982 to 2005. Infrared satellite data, containing information about the storm itself, have shown potential for improving the forecasts, but they are not available for the entire dataset (their availability starts from 1995 and for some cases only), and so they were not included in this study.

Table 2.1: The full set of predictors that are going to be used for our forecasts. The column SHIPS denotes the predictors used in the 2003 SHIPS model [DMS$^+$05].

Predictor	S or T	Abbreviation	SHIPS
1) Initial maximum winds	S	VMAX	$\surd$
2) Max wind change during the past 12h	S	INCV/PER	$\surd$
3) Storm latitude	S	LAT	
4) Storm longitude	S	LON	
5) Climatological sea surface temperature (SST)	S	CSST	
6) Zonal component of storm motion	S	SPDX	$\surd$
7) Meridional component of storm motion	S	SPDY	
8) Pressure level of storm steering	S	PSLV	$\surd$
9) 200-hPa divergence	S	D200	$\surd$
10) 1000-hPa divergence	S	Z000	
11) Gaussian function of (Julian day - peak value)	S	GDATE	$\surd$
12) Max potential intensity - current intensity	T	POT	
13) Distance to land mass	T	DTL	
14) Climatological depth of 20 C isotherm	T	D20C	
15) Same as above for 26 C	T	D26C	
16) Climatological ocean heat content	T	HCON	
17) Reynolds SST	T	RSST	
18) 200-hPa zonal wind	T	U200	$\surd$
19) 1000-hPa relative humidity	T	R000	
20) 200-hPa temperature	T	T200	$\surd$
21) 1000-hPa temperature	T	T000	
22) 1000-hPa θ_e (surface equivalent potential temperature)	T	E000	
23) Average θ_e for positive differences	T	EPOS	$\surd$
24) Average θ_e for negative differences	T	ENEG	
25) Same as 23 but the parcel θ_e is compared with the saturated θ_e of the environment	T	EPSS	
26) Same as 24 but the parcel θ_e is compared with the saturated θ_e of the environment	T	ENSS	
27) 850-700-hPa relative humidity	T	RHLO	
28) 700-500-hPa relative humidity	T	RHMD	
29) 500-300-hPa relative humidity	T	RHHI	$\surd$
30) 850-200-hPa shear magnitude	T	SHRD	$\surd$
31) Heading of the above shear vector	T	SHTD	
32) 850-500-hPa shear magnitude	T	SHRS	
33) Heading of the above shear vector	T	SHTS	
34) Generalized 850-200-hPa shear magnitude	T	SHRG	
35) 850-hPa vorticity	T	Z850	$\surd$
36) Relative eddy momentum flux convergence	T	REFC	
37) Vertical shear times sine of storm latitude	T	LSHR	$\surd$

Chapter 3

Methods

This chapter contains descriptions of the methods and the models used for achieving the objectives of this thesis. Where applicable, pointers will be given for the reader to probe further. The outline of the chapter is as follows: it begins with a presentation of linear (multiple linear regression and derivatives) and non-linear models (neural networks, support vector machines) and then focuses on the rule-based regression methods of RuleFit (RF) and regression based on association rules (RBA). Rule-based methods have higher interpretation capabilities and have shown good potential with respect to predictive performance [FP05, OTK04]. After a brief presentation of learning curves, the chapter concludes with the feature selection algorithms applied to the dataset.

3.1 Multiple Linear Regression and Derivatives

Multiple linear regression (MLR) using a least squares fit is one of the most basic prediction models in the literature. It is also the model SHIPS uses to predict the intensity change of TCs. The linear regression model has the form:

$$\hat{y} = f(x) = \beta_0 + \sum_{j=1}^{p} x_j \beta_j,$$

where $\hat{y}$ is the output of the model, x is the vector of inputs, p is the number of input variables and βs are the parameters (coefficients) of the linear model. The goal when

deriving a MLR model is to estimate the optimal β coefficients using a criterion, based on the training data available. The most common is to minimize the residual sum of squares:

$$RSS(\beta) = \sum_{i=1}^{N}(y_i - \beta_0 - \sum_{j=1}^{p} X_{ij}\beta_j)^2,$$

giving the unique solution for β values:

$$\hat{\beta} = (X^T X)^{-1} X^T y,$$

where X is an $N \times p + 1$ data matrix, which includes N samples with p features each. The vector y contains the N desired outputs.

If the inputs are normalized (mean subtracted and divided by their standard deviation), comparison of the β coefficients can be made to see which inputs provide the largest contribution to the output and thus are more important for the forecast. Additionally, the sign of a β coefficient can give the analogous or opposite relationship between that particular input and the output. The above observations are important when one later tries to interpret the estimated model parameters.

It is often the case that the least squares estimate based on all available inputs is not adequate for two reasons: (a) prediction accuracy and (b) interpretation. The main goal of a linear model is to forecast the intensity change on future, previously unseen data. It is common practice when building the set of inputs to make educated guesses of what features are potentially useful of predicting the output. Unfortunately, wrong decisions and noisy data could lead to poor generalization performance, i.e. accuracy for unseen data. This is not evident when computing the model and reporting the training error, since due to the finite number of training samples, all the variables can be seen to contribute in the predictions and have a certain correlation with the output. One of the main objectives of statistical inference and machine learning is to select the best subset of these input that will eventually provide better future prediction accuracy. Moreover,

this subset is more interesting for interpretation purposes since it is the one capturing the big picture of the process one is trying to model.

Based on the MLR model, the goals described above can be achieved in mainly two ways, (a) by eliminating and (b) by diminishing the contributions of certain variables, translated to setting to zero or penalizing the β coefficients. This reduces the variance of our model (expected squared deviation of the estimate around its mean) and increases the bias of the model (average of the difference of the estimate from the true mean). This is also known as the bias-variance trade-off. To make things more clear, high variance and low bias result in more complex models, while low variance and high bias constitute simpler models. The first has the danger of overfitting the data and the latter the danger of underfitting the data. The best model resides somewhere in the middle. Figure 3.1 displays the bias-variance trade off through a simple example. The methods presented next work towards identifying a "golden section" between bias and variance.

3.1.1 Ridge Regression

Ridge regression tries to both minimize the residual sum of squares and penalize the coefficients, in order to reduce the variance of the model and achieve better generalization behavior. This penalization is also known as shrinkage. The β values are estimated from the solution to the minimization problem

$$\hat{\beta}^{ridge} = \underset{\beta}{argmin} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} X_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\},$$

which has the solution

$$\hat{\beta} = (X^T X + \lambda I)^{-1} X^T y.$$

When $\lambda = 0$ there is no regularization. λ controls the complexity of the model and it is usually reported as the quantity, *effective degrees of freedom*

$$df(\lambda) = tr[X(X^T X + \lambda I)^{-1} X^T] = \sum_{j=1}^{p} \frac{d_j^2}{d_j^2 + \lambda},$$

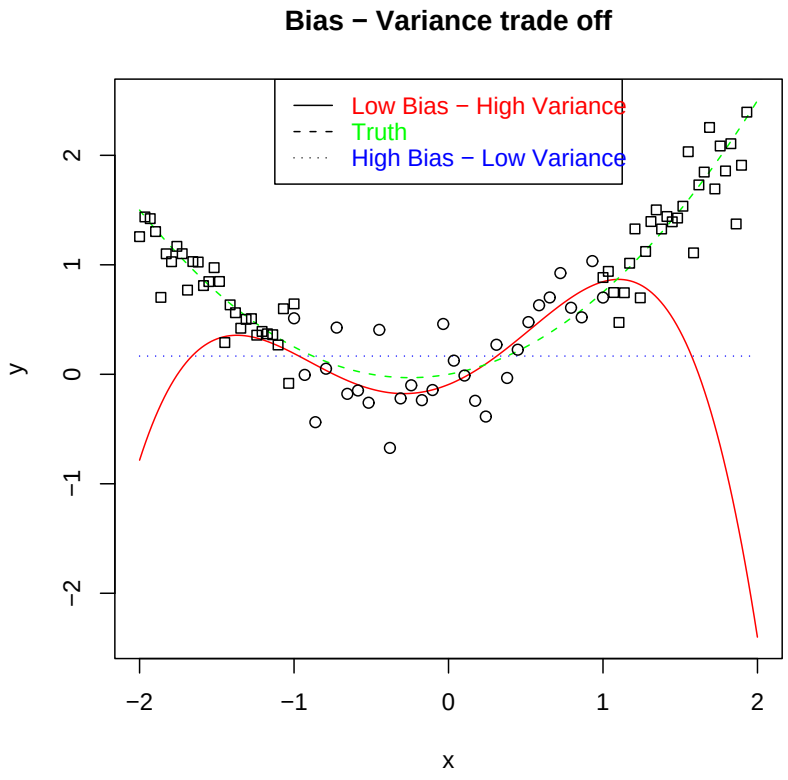


Figure 3.1: The bias-variance trade off through a simple example. The true model is given by the polynomial $0.5x^2 + 0.25x$. Gaussian noise is added with 0 mean and 0.25 standard deviation. The low bias-high variance model is a 4^{th} degree polynomial, while the high bias-low variance model is the mean of the desired output. The circles in the graph represent the training points, while the squares represent not previously know test points.

where d are the singular values of matrix X. When $\lambda = 0$, $df(\lambda) = p$ and on $\lambda \to \infty$, $df(\lambda) \to 0$.

3.1.2 Lasso

Lasso is another shrinkage method like ridge regression. The β values are estimated from the solution to the minimization problem

$$\hat{\beta}^{lasso} = \underset{\beta}{argmin} \sum_{i=1}^{N} \left(Y_i - \beta_0 - \sum_{j=1}^{p} X_{ij}\beta_j \right)^2,$$
$$\text{subject to } \sum_{j=1}^{p} |\beta_j| \leq t.$$

However, unlike ridge regression, choices of *t* can cause coefficients to be set to zero. Thus lasso regression can be used as a subset selection method as well. The tuning parameter *t* is often translated to the standardized parameter $s = t/\sum_1^p |\hat{\beta}_j|$. When $s = 1.0$ the model gives the least squares estimates, while as $s \to 0$ the β coefficients go to zero.

3.1.3 Principal Components Analysis

Each data sample consisting of p predictors can be seen as a point in a p-dimensional space, $\mathbb{R}^p$. This space is spanned by the standard basis

$$\mathbf{e}^{(1)} = (1, 0, \ldots, 0)^T$$
$$\mathbf{e}^{(2)} = (0, 1, \ldots, 0)^T$$
$$\mathbf{e}^{(p)} = (0, 0, \ldots, 1)^T$$

thus any sample $x \in \mathbb{R}^p$ is an *n*-tuple $(\alpha_1, \alpha_2, \ldots, \alpha_p)$ that can be written as:

$$\mathbf{x} = \alpha_1\mathbf{e}^{(1)} + \alpha_2\mathbf{e}^{(2)} + \cdots + \alpha_p\mathbf{e}^{(p)}.$$

This tuple is essentially one of the data samples in the dataset.

Principal Components Analysis (PCA) is a method of finding an new orthonormal basis $\mathfrak{B} = \{\mathbf{b}^{(1)}, \ldots, \mathbf{b}^{(p)}\}$ for the data. Now any given tuple can be also written as:

$$\mathbf{x} = a_1\mathbf{b}^{(1)} + a_2\mathbf{b}^{(2)} + \cdots + a_p\mathbf{b}^{(p)}.$$

This new orthogonal basis produced by PCA is optimal in the sense that it minimizes the mean squared truncation error of a D-term expansion of **x**

$$\hat{\mathbf{x}} = a_1\mathbf{b}^{(1)} + a_2\mathbf{b}^{(2)} + \cdots + a_D\mathbf{b}^{(D)},$$

where the error (mean squared error) is defined as

$$\epsilon_{mse} = \sum_{j=1}^{D}(\hat{x_j} - x_j)^2/D.$$

One method of producing the new orthonormal basis is by calculating the singular value decomposition (SVD) of the data matrix X, $X = UDV^T$, after mean centering the data. The new orthonormal basis are the right singular vectors.

PCA provides an optimal subspace for every truncation level $1 \leq D < p$. If $D = p$ then the error is zero. Principal component directions are ordered with respect to the variance captured when the original data are projected to that direction. Thus projection of the data onto fewer than p dimensions is possible, losing the least amount of information for that specific number of dimensions.

Like $df(\lambda)$ in ridge regression and s in lasso regression, D is a parameter that needs to be estimated in order to get good generalization behavior for the derived model. For further information on PCA the reader is referred to [Kir01]. The contents of this section are based on [HTF01].

3.2 Neural Networks

For a complex prediction problem such as forecasting intensity change of TCs, it is usually the case that there exist non-linear relationships between the inputs and the output.

A model often used to capture these non-linear dependencies is a standard feed-forward multilayer neural network (NN), trained using the back-propagation procedure. The NN consists of a layer of input units, a layer of hidden units and a layer of output units. The units of each layer are fully connected with the units of the next section by weighted connections. First the hidden and then the output units calculate the weighted linear combination of their inputs and then apply a threshold or activation function to derive their output. In this particular study, two continuous functions were chosen, the sigmoid function for the units of the hidden layer and the linear function for the output units.

The weights of the hidden and the output layer can be defined as matrices α and β, with dimensions $(p+1) \times M$ and $M \times 1$, where p is the number of input variables and M the number of hidden units. The single output provides the intensity change. It is calculated as follows

$$\hat{y} = g(\beta^T \sigma(\alpha^T X)),$$

where σ is the vector sigmoid function and g is the linear function.

The goal of the back propagation algorithm is to adjust the connection weights in order to minimize the error between the actual output of the network and the desired output of the training data. The error it tries to minimize is the sum of squared error over all N samples:

$$E = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

To minimize the error a gradient descent procedure is constructed that adjusts the weights in matrix form, also known as the batch mode algorithm. The update equations

are as follows:

$$
\begin{aligned}
z &= \sigma(\alpha^T X) \\
\delta &= (y - o) \\
\alpha^{new} &= \alpha^{old} + \rho_h X(\beta^{new}\delta z(1-z))^T \\
\beta^{new} &= \beta^{old} + \rho_o z \delta^T,
\end{aligned}
$$

where z is the output of the hidden layer, δ is the difference (error) of the desired output versus the output of the network, ρ_h is the learning rate of the hidden layer and ρ_o is the learning rate of the output layer.

The procedure stops after a prespecified number of iterations or epochs. It is known that every bounded continuous function can be approximated by a neural network with two layers if no restrictions are placed on the number of hidden units [Mit97].

As in the linear case, there is again a bias-variance trade off. The main factor that prevents the network to achieve a good generalization behavior are the weights, mainly, and the number of hidden units. For this study the number of the hidden units was set to a high number (50), since fewer hidden units were found to be inefficient based on the accuracy of the network. Furthermore, the following techniques were employed to prevent overfitting

- *Early stopping*, where the training set is split between a training and a validation set, and the final weights are the ones that achieve the best error performance (root mean square error in this case) in the validation set.

- *Weight decay*, where there is a penalty term for the magnitude of the weight. This way the back-propagation searches for weights with smaller magnitudes, preventing overfitting of the network by setting the weights too high and in consequence saturating the outputs of the sigmoid layer.

3.3 Support Vector Machines

Support vector machines (SVMs) are an increasingly popular methodology for developing classification models. With respect to the problem at hand, SVMs can be applied to the case of regression as well. The algorithm considered here is know as ϵ-SV regression, its goal being to find a function $\hat{y} = f(x) = \langle w, x \rangle + b$ that has at most ϵ deviation from the targets y, for all the training data, and at the same time minimizing the Euclidean norm of the weights, i.e., $\|w\|^2$. One may form the above as an optimization problem, but in many cases it will be infeasible. Thus a "soft margin" loss function can be embedded giving the following minimization problem

$$\begin{aligned} &\text{minimize } \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}(\xi_i + \hat{\xi}_i) \\ &\text{subject to } \begin{cases} y_i - \langle w, x_i \rangle - b \leq \epsilon + \xi_i \\ \langle w, x_i \rangle + b - y_i \leq \epsilon + \hat{\xi}_i \\ \xi_i, \hat{\xi}_i \geq 0 \end{cases} \end{aligned} \tag{3.1}$$

The C constant determines the trade off between minimizing $\|w\|^2$ and the deviations from ϵ tolerated, while ξ and $\hat{\xi}$ are slack variables used to cope with otherwise an infeasible optimization problem. The above formulation corresponds to the ϵ-insensitive loss function defined as

$$|\xi|_\epsilon := \begin{cases} 0 & \text{if } |\xi| \leq \epsilon \\ |\xi| - \epsilon & \text{otherwise.} \end{cases}$$

From a computational point of view, instead of solving the primal problem (3.1), it is more convenient to consider its dual Lagrangian formulation:

$$\begin{aligned} &\text{maximize } \begin{cases} -\frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i - \hat{\alpha}_i)(\alpha_j - \hat{\alpha}_j)\langle x_i, x_j \rangle \\ -\epsilon\sum_{i=1}^{N}(\alpha_i + \hat{\alpha}_i) + \sum_{i=1}^{N} y_i(\alpha_i - \hat{\alpha}_i) \end{cases} \\ &\text{subject to } \begin{cases} \sum_{i=1}^{N}(\alpha_i - \hat{\alpha}_i) & = 0 \\ \alpha_i, \hat{\alpha}_i & \in [0, C] \end{cases} \end{aligned} \tag{3.2}$$

where α and $\hat{\alpha}$ are the dual variables, an alternative representation of the weights in dual coordinates. This leads to a model of the form

$$f(x) = \sum_{i=1}^{N} (\alpha_i - \hat{\alpha}_i)\langle x_i, x\rangle + b$$

that is a linear combination of training patterns.

This form is called support vector expansion. To advance to non-linearity one can transform the training samples from $\mathbb{R}^p$ into a feature space $\mathfrak{F}$, and then apply the above algorithm. Since only the dot product between the input samples is needed, it suffices to know the kernel function $k(x, x') = \langle \Phi(x), \Phi(x')\rangle$, where Φ the mapping from $\mathbb{R}^p$ to $\mathfrak{F}$. Thus the algorithm becomes:

$$\text{maximize} \begin{cases} -\frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i - \hat{\alpha}_i)(\alpha_j - \hat{\alpha}_j)k(x_i, x_j) \\ -\epsilon\sum_{i=1}^{N}(\alpha_i + \hat{\alpha}_i) + \sum_{i=1}^{N} y_i(\alpha_i - \hat{\alpha}_i) \end{cases}$$

$$\text{subject to} \begin{cases} \sum_{i=1}^{N}(\alpha_i - \hat{\alpha}_i) & = 0 \\ \alpha_i, \hat{\alpha}_i & \in [0, C] \end{cases}$$

and the function:

$$f(x) = \sum_{i=1}^{N} (\alpha_i - \hat{\alpha}_i)k(x_i, x) + b$$

For the purposes of the TC prediction problem, two non-linear kernels were considered: (a) the polynomial kernel

$$k(x, x') = (\gamma\langle x, x'\rangle + c)^m$$

and (b) the radial basis function kernel

$$k(x, x') = e^{-\gamma\|x - x'\|^2}.$$

This section was based in [SS98]. For an introduction in SVMs the reader is referred to [CST00]. The reader is referred to section 3.8 for implementation details.

3.4 RuleFit

In the next two sections, the rule-based procedures are described. Rules are implications and come in the form of if-then statements. A rule is "fired" if the antecedents of the rule are satisfied. If this is the case, the consequent is applied according to respective rule framework. For example, in the case of TC intensification problem a rule can be found in the following format

IF $(-0.5 < INCV)$ AND $(40.5 < POT)$, THEN

Intensity change in the next 6 hours will be 1.5 knots

meaning that if the increase in wind speed the previous 12 hours is above -0.5 knots and the maximum potential intensity is above 40.5 knots the TC's wind speed will increase by 1.5 knots in the next 6 hours.

Finding the optimal set of rules is a combinatorial optimization problem and difficult to solve algorithmically. One approach to approximately solve this combinatorial optimization problem is with genetic algorithms [Fre02]. Alternatively the techniques presented here use an algorithm to derive a set of rules in a greedy fashion, such as from decision trees, and then use another algorithm to build the final regression model that orders the rules and/or determines the contribution of each rule to the output once it is "fired". This way the exponential explosion of the search procedures is avoided in the expense of a more complete search in the space of rules.

The RuleFit algorithm [FP05] is based on the notion of ensemble learning

$$F(\mathbf{x}) = a_0 + \sum_{m=1}^{M} a_m f_m(\mathbf{x})$$

where a set of base learners, learn M models $f_m(x)$, and where their outputs are linearly combined with a_m being their contributions to the output. The contributions of each rule are usually derived using a regularization method like the lasso procedure, given a loss

function. In the case of RuleFit the base learners are rules derived from decision trees constructed from the data at the beginning of the algorithm. The decision trees can be constructed using for example the classification and regression trees (CART) methodology [BFOS83] or the methodology used to build model trees (M5 system) [Qui92]. This approach is similar to building random forests [Bre01]. The different trees are constructed using a different bootstrap sample drawn from the training data. Every tree constructs as many rules as the nodes of the tree, since every path from each node to the root is a rule of the form:

$$r_m(\mathbf{x}) = \prod_{j=1}^{n} I(x_j \in s_{jm})$$

where s_{jm} is an interval $(t_{jm}, u_{jm}]$ for continuous variables and I is the indicator function, which outputs 1 if its input argument is true and 0 otherwise. For categorical variables s_{jm} can be enumerated completely. The previous example can be written as

$$r_m(\mathbf{x}) = I(-0.5 < INCV) \cdot I(40.5 < POT)$$

while a, the contribution of the rule would be equal to 1.5. Rules of this form can be thought as defining a subspace in the feature space, inside which they are true. As an example the area of support of the example rule is depicted in Figure 3.2.

Linear functions are very difficult to approximate with rules, thus RuleFit explicitly incorporates them to its model that becomes:

$$F(\mathbf{x}) = \hat{a}_0 + \sum_{m=1}^{M} \hat{a}_m r_m(\mathbf{x}) + \sum_{j=1}^{p} \hat{b}_j l_j(x_j)$$

where $l_j(x_j)$ are the "Winsorized" versions of the variables, $l_j(x_j) = min(\delta_j^+, max(\delta_j^-, x_j))$, making the model more robust to outliers [FP05]. δ_j^+ and δ_j^- are respectively the β and $(1-\beta)$ quantiles of the data distribution of each

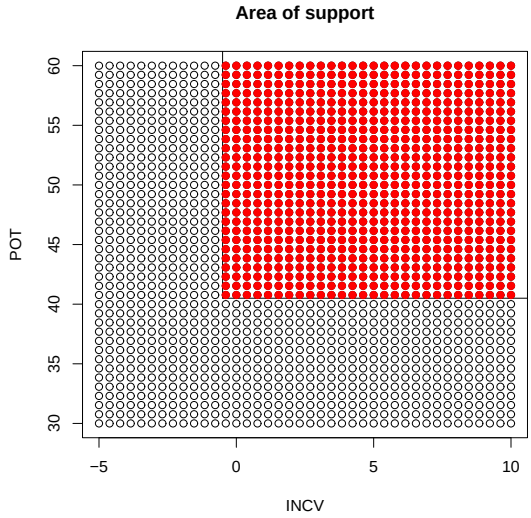


Figure 3.2: Area of support for the example rule. The solid circles denote the area of values that extends to infinity of the variables INCV and POT where the rule is fired.

variable x_j. The coefficients are estimated from the following optimization problem

$$(\{\hat{a}_k\}_0^K, \{\hat{b}_j\}_1^p) = \underset{\{\hat{a}_m\}_0^M, \{\hat{b}_j\}_1^p}{argmin} \sum_{i=1}^{N} \left(y_i - a_0 - \sum_{m=1}^{M} a_m r_m(\mathbf{x}_i) - \sum_{j=1}^{p} b_j l_j(x_{ij}) \right)^2$$
$$+\lambda \cdot \left(\sum_{m=1}^{M} |a_m| + \sum_{j=1}^{p} |b_j| \right)$$

Parameters such as the λ penalty and the tree size (number of antecedents in the rule) can be estimated from the data based on cross-validation procedures.

3.4.1 Variable Importance and Interaction Effects

Two of the most interesting features of the RuleFit framework is the measure of input variable importance and the identification of interaction effects between variables. For the first one, RuleFit tries to identify the most influential predictors based on their frequency of appearance and their influence. The concept of variable importance is

captured by the following equation:

$$J_l = I_l + \sum_{k \in \{m | x_l \in r_m\}} I'_k / m_k$$

where x_l is the input variable whose importance is calculated, m_k is the total number of variables in the rule, I_l the importance of x_l as a linear predictor and I'_k the importance of x_l with respect to the rules it is involved in. I_l is equal to $|\hat{b}_l| std(l_l(x_l))$, with std, being the standard deviation over the data and $I'_k = |\hat{a}_k| \sqrt{s_k(1-s_k)}$, with $s_k = \frac{1}{N} \sum_{i=1}^{N} r_k(\mathbf{x}_i)$, the support of the rule. Thus, one is interested in the values of each J_l as a measure of the importance of each variable.

RuleFit is also capable of capturing interaction effects between variables. A function $F(\mathbf{x})$ is said to exhibit interaction effects between two of its variables x_j and x_k if the difference in the value of $F(\mathbf{x})$ for different values of x_j, depends on the value of x_k. If there is no interaction, one can express $F(\mathbf{x})$ as a sum of two functions, one independent of x_k and the other independent of x_j. A statistical test can be used to measure the fraction of $F(\mathbf{x})$ not explained by the decomposition in order to expose interaction effects. Then one is able to also measure the extent of these interactions between x_j and x_k. For example, to test for the presence of an interaction between two variables x_j and x_k the following statistic can be used:

$$H^2_{jk} = \sum_{i=1}^{N} [\hat{F}_{jk}(x_{ij}, x_{ik}) - \hat{F}_j(x_{ij}) - \hat{F}_k(x_{ik})]^2 / \sum_{i=1}^{N} \hat{F}_{jk}(x_{ij}, x_{ik})$$

For the function $F(x_1, x_2) = x_1 + x_2$, $H^2_{12} = 0$, while for the function $F(x_1, x_2) = x_1 + x_2 + x_1 x_2$, $H^2_{12} \neq 0$, alerting for an interaction between x_1 and x_2. The same intuition holds for three variables and the reader is referred to [Fri01] for a complete discussion.

3.5 Regression Based on Association Rules

Instead of deriving rules from decision trees, regression based on association rules (RBA) [OTK04] derives rules based on the Apriori algorithm [AIS94], used to find association rules in large dataset. RBA consists of two major steps: (a) mining the association rules and (b) constructing the regression model. The RBA rules are of the form:

$$\text{IF } (-0.5 < INCV) \text{ AND } (40.5 < POT), \text{ THEN}$$

$$\text{Intensity Deviation is } \{\hat{\mu} = 1.5, \hat{\sigma} = 12, s = 15\%\}$$

where $\hat{\mu}$ and $\hat{\sigma}$ is the mean and standard deviation of the target variable of the training samples that satisfy the antecedents of the rule. This association rule format is based on [AL99]. Other statistics that can be used are the median in place of the mean and the mean absolute deviation instead of the standard deviation. The standard deviation and the mean absolute deviation denote the dispersion of the output and are used when constructing the final model. The third variable s denotes the support of the rule, i.e., the number of samples that the rule's conditions are true, divided by the total number of samples N.

RBA follows the procedure below:

1. Discretize the target variable into bins.

2. Discretize all the numerical inputs using an information based discretization procedure described in [FI93]. Step 1 and 2 are necessary since association rules are derived using only discretized inputs and outputs. In the case of RuleFit this does not hold since there are procedure of deriving decision trees using numerical inputs.

3. Generate frequent itemsets using the Apriori algorithm [AIS94] and then derive the rules by computing the necessary statistics (following the format

above [AL99]). Frequent itemsets are sets of binary attributes that satisfy a prespecified support threshold. Since everything is discretized at this stage (either as categorical variables or as intervals for numerical variables), for every training sample, one can say whether an item (binary attribute) given its value, exists or not in that specific sample. For example in the rule above the items $(-0.5 < INCV)$ and $(40.51 < POT)$ make a frequent itemset if the prespecified by the user support is above 10%, that is if 10% of the training samples have that this combination of attributes and values. One can also see that these tuples are binary. They are either true or false given a sample.

4. Prune the derived rule set from redundant rules, i.e., prune rules that have a more accurate generalization or prune rules that have more accurate complementary rules.

5. Build the regression model by selecting rules using a greedy strategy. The rules are ordered based on their accuracy (large support, small dispersion) and then selected based on a sequential covering algorithm for the training data. There exist two rule schemes, 1-RBA and weighted k-RBA. For 1-RBA, the rules are searched in the order that they appear in the set and the mean value of the first rule that satisfies a given sample is returned as a prediction. If the weighted k-RBA is used then the best k-rules are selected and the final value returned is:

$$f(z) = \frac{\sum_{i=1}^{k} w_{ri}\mu_{ri}}{\sum_{i=1}^{k} w_{ri}}.$$

Weights can be the same (average-k), equal to the support of each rule (probabilistic-k) or the inverse of the variance (precision-k). If k-RBA is used, k rules per sample are needed. A default rule is also constructed in case the rules do not satisfy all the training data (just a mean value of the target variable on the remaining unsatisfied samples).

3.6 Learning Curves

A learning curve is a plot graphing the performance measure of a learning algorithm (y-axis) versus the number of training examples (x-axis). For producing these curves it is required to divide the dataset into two disjoint sets, the training set, and the test set. By gradually increasing the number of the samples from the training set that participate in the derivation of the model and by measuring the performance of that model for the test set we get the quality of the prediction accuracy as a function of the size of the training set [RN03]. Learning curves can be used for both classification and regression tasks.

Learning curves are a useful tool for many different reasons. First of all, if the error decreases as the size of the training set grows, this is an indication that there is some pattern behind the data that the algorithm is learning. This particular observation should also be accompanied by the fact that a random model (same inputs with randomly permuted output) will fail to find the same patterns as the normal one. Additionally, a "saturated" graph, that is a graph where the error stops to decrease as the number of sample increases, is a sign that there are sufficient data to build robust models. Additionally, learning curves are used to identify reasonable partitions of data for the methods of assessing the quality of the models. One example would be to identify the number of folds in a k-fold cross-validation (CV) procedure [HTF01]. In a CV procedure the model is built using $k - 1$ folds and testing is performed on the remaining fold. Each time a different fold is left out for testing. In the end, the mean test error is reported. Finally, in the case presented here, since 20 learning curves will be built, containing the same variables and using the same model, the difference between the learning curves can identify properties of the datasets as predictions move further into the future.

The number of folds, k, in a CV procedure balances between high variance and computational burden when k=N and bias towards overestimating the prediction error with small values of k, like k=3, 5 or 10. In order to estimate a good value of k, a

heuristic approach was used involving learning curves. A nice estimate of how many training samples are necessary is to pick the size when the curve begins to saturate.

3.7 Feature Selection

Five feature selection methods were used and are described below. For picking a particular set of features, the following two rules were applied.

- The final number, k, of features is the mean number of features selected for each one of the 20 datasets. The k most often picked features are incorporated into the final set for each method. Even though different features could be selected for different hours this rule was used, since the goal is to build only one set of features.

- Also as mentioned in the introductory chapter Ockham's razor principle is applied. Thus for each dataset the smallest set of features is selected that has a mean error within 1 standard error above the minimum mean error [HTF01]. The intuition behind the 1 standard error threshold is that the models should not be more complicated unless their performance decreases (in our case the RMSE increases) by at least 1 standard error from the minimum.

3.7.1 Forward Selection and Backward Elimination

The Forward Selection (FS) and Backward Elimination (BE) procedures are among the most often used methods regarding feature selection techniques [HTF01]. For this work the following versions of FS and BE were used. FS starts with zero features (just the intercept term) and constructs a linear model for each input variable separately. The variable giving the higher F-statistic is included in the model and the procedure is reap-

plied. The F-statistic is defined as

$$F = \frac{(RSS_0 - RSS_1)/(p_1 - p_0)}{RSS_1/(N - p_1 - 1)},$$

where RSS_1 is the residual sum of squares of the bigger model with p_1 variables and RSS_0 the previous model with p_0 variables. BE starts with the full set of variables and removes the variable giving the lowest F-statistic and the procedure is re-applied until no variable is left. Selection of the best set is based on the two rules discussed above.

3.7.2 Genetic Algorithms

Forward and backward stepwise procedures are greedy methods, performing a greedy search in the variable space and thus may not be optimal. Additionally, it can be shown empirically that [GE03] a variable that is completely useless by itself can provide significant performance when used with others, and two variables useless by themselves can be useful together. Based on the above considerations, stepwise addition or elimination may miss some of the interdependencies between variables.

Based on that, the use of genetic algorithms (GAs) was considered as a variable subset selection method. A solution in the context of GAs is a string (chromosome), which in this case defines the variables that belong or not to the final subset. The first step for the implementation of a GA involves the specification of an appropriate coding for each possible solution. The GA implemented in this study uses binary chromosomes. The bits involve the variable that should be included in the construction of the model at hand (1 designates a possibly relevant attribute and 0 a possibly irrelevant one). Each chromosome is assessed through a cost function (fitness function). The algorithm manipulates iteratively a finite set of chromosomes (population), based on the mechanism of evolution. At each iteration (generation), chromosomes are subjected to certain operators, such as crossover and mutation, which are analogous to processes which occur in natural reproduction. During this reproduction process, only the best chromosomes are

allowed to survive to the next generation, following the "survival of the fittest" principle. This evolutionary process is repeated until the population "converges" according to a termination criterion.

The fitness of each chromosome is based on the equation:

$$F = RMSE + \rho\frac{p-n}{p},$$

where n is the total number of variables and p is the number of variables selected [VOM99]. The factor ρ determines a penalty term towards sets of features with large cardinality. The factor ρ was chosen to be 1 standard error (se), matching the previous discussion, even though 0.5 se and 2 se were also considered with less satisfactory results.

Three selection functions were implemented: tournament, roulette wheel and a custom one. The last gave the best results. The procedure is as follows:

Repeat the following steps for a fixed number of generations:

1. Select the two policies with the highest fitness in the population.

2. With the given crossover probability (0.8), randomly select a crossover point as an integer between 1 and u. Generate one new string by combining the first part of the first string with the second part of the second string, and combine the two remaining parts to form a second new string. Otherwise, if crossover is not performed, just take the two selected strings as the two new strings.

3. With the given mutation probability (0.1), randomly choose a new value for a bit to insert in each position in the new strings.

4. Evaluate the two new strings based on the fitness function.

5. For each new string, if its fitness is better than the worst one currently in the population, replace the worst string with the new one.

For an introduction to GAs the reader is referred to [Whi94].

3.7.3 Neural Networks

One approach for feature selection with NNs is based on a heuristic that exploits the weights and the structure of the NN [LG99]. It is based on the following criterion:

$$S_i = \sum_{j \in H} \left(\frac{|a_{ij}|}{\sum_{i' \in I} |a_{i'j}|} \sum_{k \in O} \frac{|b_{kj}|}{\sum_{j' \in H} |b_{kj'}|} \right),$$

where I, H and O denote the input, hidden and output layers, respectively. The intuition behind this metric is that the importance S of input variable i is the sum of absolute values of all the normalized products of weights from the input to the output(s). Inputs should be in the same range (normalized) in order to avoid biasing variables with large values against the ones with smaller values. In this setting, the values of the weights are interpreted in the same way like the β values in MLR. Backward elimination can be applied by deleting each time the variable with the smallest S value and then retraining the network and reapplying the procedure.

3.8 Implementation Details

The majority of the code was written in R, a language and environment for statistical computing and graphics (`http://www.r-project.org`). For applying SVMs to the dataset a wrapper to LIBSVM for the R statistical language was used, named e1071. The RBA framework was developed using the Java programming language (`http://java.sun.org`) and extends the Waikato Environment for Knowledge Analysis (WEKA) [WF00] (`http://www.cs.waikato.ac.nz/~ml/`). For using the RuleFit framework with this dataset an interface to R was used that can be found in `http://www-stat.stanford.edu/~jhf/R-RuleFit.html`. Also for performing lasso regression the lars package was used which can be found in

(`http://www.r-project.org`). For all the other methods the code was written in R by the author.

Chapter 4

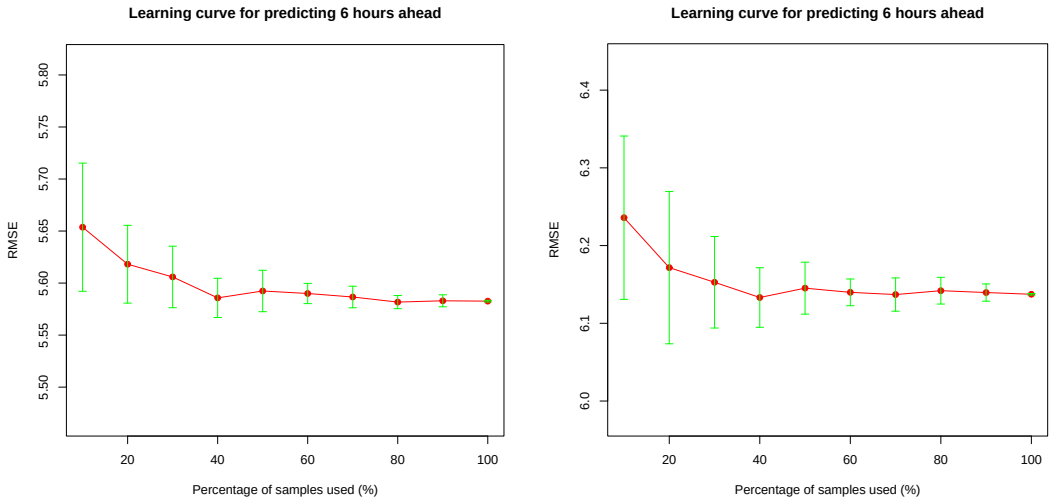
Results

Through this chapter the reader will be stepped through the complete procedure for deriving robust and interpretable models for forecasting the intensification of TCs. The results begin with the derivation of learning curves and findings pertaining to them. A discussion on the model assessment method of choice follows. Afterwords, the two main results will be presented: feature selection and model evaluation. The chapter will continue with a discussion of the interpretation skills of the models, which will lead to the development and assessment of a hand-picked model.

Seasons 1982 to 2003 were used as the training (developmental) set, while seasons 2004 and 2005 were kept as test sets. Also the notion of incremental training and testing will be used in order to check for stability between seasons. This is an iterative procedure where the model is developed from season 1982 to season x and is tested for season $x + 1$, then developed for seasons 1982 to $x + 1$ and tested for season $x + 2$ etc. All the variables are normalized by subtracting their mean and then dividing by their standard deviation unless otherwise stated. The means and standard deviations are determined from the training data.

4.1 Learning Curves

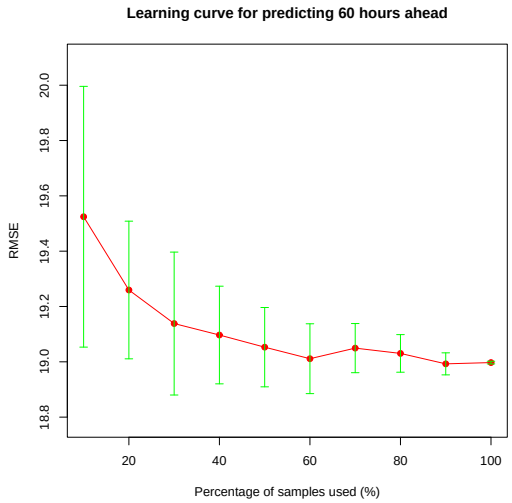
The learning curves were created to measure the RMSE of MLR as the number of exemplars increased. The set of SHIPS 2003 predictors was used, as a measure of how an efficient model will behave given the data available. Figures 4.1, 4.2 and 4.3, show the learning curves for the models forecasting 6, 60 and 120 hours ahead, for both the SHIPS model without and with the randomly permuted output. The averages and the standard errors were found by reapplying the algorithm 20 times for a particular size, each time using a different randomly drawn subset. It was observed that this value (20) reduced the variability and produced smoother curves.



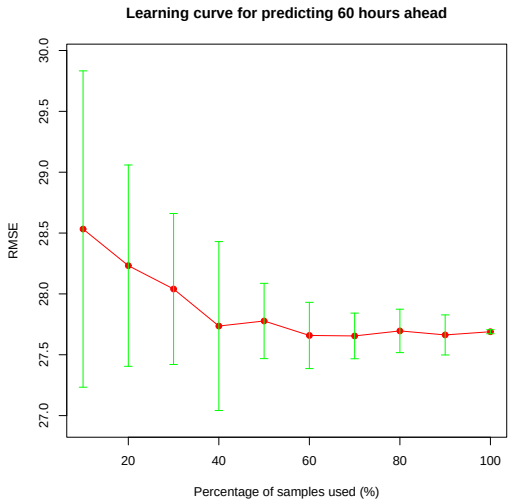
(a) Normal output (b) Permuted output

Figure 4.1: Learning curve for the SHIPS 2003 model 6 hours ahead.

First of all, based on the learning curves, it is evident that there is a pattern to be learned, especially since in the later hours (120 hours ahead), the difference between the model with the randomly permuted output and the SHIPS model reaches 13 knots. The error earlier, i.e., 6 hours ahead, is much smaller since most of the desired outputs are 0 (no intensification observed) and thus the random permutation of the output plays

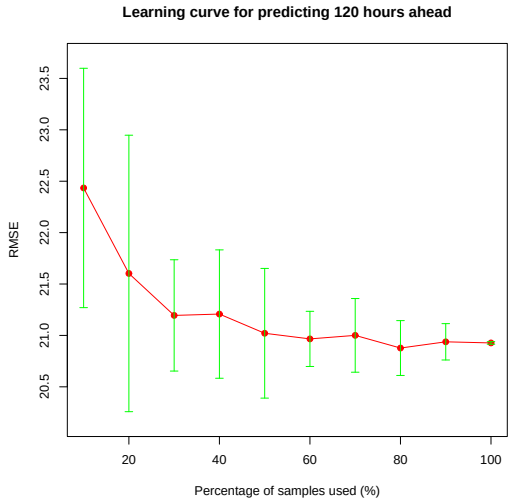


(a) Normal output

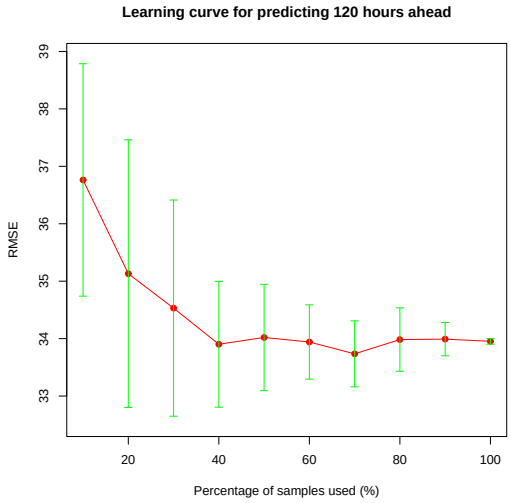


(b) Permuted output

Figure 4.2: Learning curve for the SHIPS 2003 model 60 hours ahead.



(a) Normal output



(b) Permuted output

Figure 4.3: Learning curve for the SHIPS 2003 model 120 hours ahead.

no significant role. Additionally, a comparison of the learning curves of just the SHIPS models (for 6, 60 and 120 hours) can lead to the conclusion that more "noise" is being added when moving to later hours. This is something identified by the higher mean RMSE and the higher variance in the RMSE, when not all of the samples are included during training. Moreover, the learning curves show that the data are adequate to obtain robust models, since the saturation of the RMSE in the curves is evident in all cases. Finally, from the curves a good value for *k*, the number of folds for performing cross-validation, can be set to 3, which corresponds to 66% of the developmental set will be used as training and the rest 33% used for validation. The value of 66% was found by inspecting all the graphs and finding the percentage of the training test where the curve begins to saturate.

4.2 Model Assessment

Knowing a good number of folds (k=3), the model development process is continued by evaluating model assessment methods. Through them one would like to get a prediction of the test error that is as accurate as possible. Based on that measure of accuracy one can select the model assessment method of choice, giving more accurate predicted error test rates, without peeking at the separate test set and biasing the feature selection and error reporting results. Several model assessment methods were evaluated: training error, 5 3-fold cross-validation (CV) procedures, each one with a new random selection, their average, leave-one-season-out (LOSO) jackknife procedure, leave-one-hurricane-out (LOHO) jackknife procedure and .632+ bootstrap method (B.632+) repeated 5 times [HTF01]. The methods were applied to the training set to provide the predicted (estimated) test error on unseen data. Their performance was evaluated based on their accuracy to predict the actual test error as measured by the relative error (RE):

$$RE = \frac{|estimatedTestError - realTestError|}{realTestError}\%$$

All of the above techniques were evaluated using MLR and a NN (50 hidden sigmoid neurons with weight decay and linear output) on the whole set of features and on the 2003 SHIPS features. Their performance as an average in all the 20 datasets is reported in Table 4.1. LOSO outperforms any other assessment method for all the four combinations of variables and models, predicting more accurately the test error. Since the test set is a new season, it is intuitive for the LOSO procedure to give the best estimate, but the comparison was made as a sanity check. The LOSO procedure will be used here on for producing the rest of the results.

Table 4.1: The relative error, RE, of model assessment algorithms. LOSO (in boldface characters) outperforms any other given method.

		All variables		SHIPS 2003 variables	
		MLR (%)	NN (%)	MLR (%)	NN (%)
Methods	Train	22.83	52.46	8.08	33.72
	CV1	21.5	30.63	7.38	16.69
	CV2	21.58	30.22	7.37	16.17
	CV3	21.62	29.81	7.42	16.19
	CV4	21.59	29.83	7.39	15.53
	CV5	21.54	29.66	7.47	16.64
	CV avg	21.57	30.03	7.39	16.25
	LOSO	**14.63**	**10.98**	**6.84**	**8.34**
	LOHO	21.43	14.68	12.89	12.64
	B.632+	22.24	38.25	7.69	22.38

4.3 Feature Selection

The next step was to select the features giving the best predictions for unseen data, using the procedures presented in Chapter 3. Table 4.2 summarizes the findings. For ridge regression, PCA regression and lasso regression different degrees of complexity (λ, D and s, respectively) were used per dataset, giving more freedom to the models. For the GAs 100 genes were randomly constructed for the initial population and the

reproduction occurred for 200 generations. The factor ρ was chosen to be 1 standard error, following the discussion of Ockham's razor principle.

Since the best performing feature selection method was that of GAs (Table 4.2), this procedure was chosen to search for non-linear features as well. The initial set of features were the ones found by the initial GA (12 features) and three different approaches were considered for adding non-linear terms to the initial set of variables into consideration:

1. **GA N1**: the transformations x^2 and $x \times y$ for the 3 most frequently selected variables of the first GA procedure were added.

2. **GA N2**: for each variable form the transformations x^2, x^3, $1/x$ and $\sqrt{x}$. Add the one which had the highest correlation with the output.

3. **GA N3**: for a special set of the most interpretable predictors: *VMAX, INCV, D200, POT, SHRD, T200, PSLV, Z850, GDAY, EPOS, LHSR* [DeM06] the complete set of the transformations x^2 and $x \times y$ was considered.

Table 4.3 shows the linear selected predictors for each feature selection procedure, while Table 4.4 provides the non-linear terms selected by the three additional GA procedures.

The main conclusion was that no set of features consistently performed better than the others. This can be attributed to the independence between seasons, meaning that different seasons have different kinds of hurricanes, leading to deviations in the RMSE. Knowing for example that rapidly intensifying hurricanes are one of the drawbacks of statistically modeling the intensification of TCs, a season with more hurricanes of this kind would give a higher RMSE. The other main characteristic was that the GA procedures outperformed more conventional feature selection methods like backward elimination and forward selection with respect to the framework presented here.

Table 4.2: RMSEs of different feature selection techniques and the SHIPS model for the LOSO model assessment method and the 2004 and 2005 test sets. The asterisk denotes the presence of non-linear terms, while the number in the parenthesis display the number of linear terms. Boldface numbers indicate the best performing set of predictors.

Methods	LOSO	2004	2005	Mean 04-05	Features
MLR	18.08	20.44	21.89	21.16	37
Ridge	19.90	24.96	27.47	26.22	37
PCA	20.07	22.38	23.74	23.06	Variable
FS	17.88	19.32	22.43	20.87	13
BE	18.84	20.03	22.91	21.47	4
Lasso	20.54	20.91	21.35	21.13	Variable
NN	19.34	20.56	21.61	21.09	7
GA	17.42	18.75	19.67	**19.21**	12
GA N1*	16.82	18.83	22.76	20.79	10 (8)
GA N2*	16.98	19.20	**19.44**	19.32	19 (13)
GA N3*	**16.27**	17.92	24.13	21.03	20 (3)
SHIPS*	17.35	**17.14**	22.27	19.70	16 (13)

4.4 Performance Comparisons

This section provides a comparison of the best MLR methods with non-linear features and other non-linear methods against the SHIPS 2003 model. These include the two best performing sets of features derived from the GA feature selection techniques with non-linear features, two non-linear methods, NNs and SVMs, and two rule-based methods, rule ensembles derived with the RuleFit framework and rules based on association rules derived with the RBA framework. The set of input variables for NNs and SVMs was the set of linear variables found from the initial GA procedure, since these algorithms should be capable of producing necessary non-linearities by themselves. RuleFit is applied in the total set of variables (37), since it empirically produced better results with that set and rule-based schemes can be considered to do implicit feature selection. On the other hand, the initial set of variables for RBA was that of the variables produced by the first GA procedure because it produced better results with that set. For NNs and SVMs a

Table 4.3: Linear features chosen by different algorithms.

Variable	SHIPS	FS	BE	GA	GA N1	GA N2	GA N3	NN
1) VMAX	√	√	√	√	√	√		√
2) INCV	√	√	√	√	√	√	√	
3) LAT				√				√
4) LON								
5) CSST								
6) SPDX	√			√	√	√		
7) SPDY								
8) PSLV	√							
9) D200	√							
10) Z000								
11) GDATE	√	√						
12) POT		√	√	√	√	√		√
13) DTL								
14) D20C					√	√		
15) D26C								
16) HCON								
17) RSST								√
18) U200	√	√						
19) R000		√						
20) T200	√	√		√		√	√	
21) T000				√				
22) E000				√				
23) EPOS	√						√	
24) ENEG								
25) EPSS								
26) ENSS		√						
27) RHLO								
28) RHMD								
29) RHHI	√							
30) SHRD	√	√	√	√	√	√		√
31) SHTD		√						√
32) SHRS		√						
33) SHTS						√		
34) SHRG				√				√
35) Z850	√	√		√	√	√		
36) REFC								
37) LSHR	√	√		√	√	√		

Table 4.4: Non linear features chosen by the second-pass GAs.

Method	Variables
GA N1	$SHRD^2$, $SHRD \times Z850$
GA N2	$INCV^2$, $E000^3$, $1/SHRD$, $1/SHRG$, $T000^3$, POT^2
GA N3	$T200^2$, $SHRD^2$, $LSHR^2$, $VMAX \times INCV$, $VMAX \times LSHR$, $GDAY \times Z850$, $GDAY \times LSHR$, $INCV \times SHRD$, $INCV \times Z850$, $INCV \times POT$, $T200 \times PSLV$, $T200 \times D200$, $EPOS \times SHRD$, $SHRD \times LSHR$, $D200 \times LSHR$, $POT \times LSHR$
SHIPS	$VMAX \times INCV$, $VMAX \times SHRD$, POT^2

grid search procedure using the LOSO technique was performed for parameter tuning. For NNs, the number of hidden units ($5, 10, 20, 25,$ **50**) and the weight decay factor ($0.1, 0.05, 0.01,$ **0.001**, 0.0001) was searched. Both learning rates were set to a low value ($\rho_h = 0.005$, $\rho_o = 0.0005$), while the number of epochs to a high of 3000 epochs. The value was high since the network converged after approximately 100 epochs. For SVMs the degree of the polynomial kernel ($1, 2, 3$) and the gamma factor ($1, 0.5, 0.1,$ **0.05**) for the radial basis kernel was explored, in addition with the ϵ factor ($3, 2, 1, 0.5,$ **0.1**, 0.05) for both of them. The SVM with a radial basis kernel (SVMR) was more accurate than the SVM with a polynomial kernel. In boldface characters are the selected values.

Tables 4.5 and 4.6 display the RMSEs of all the methods for all datasets and for both 2004 and 2005 seasons, while Table 4.7 summarizes the findings. In every interval (6 to 120 hours ahead) a model wins if it has the lowest RMSE test error. SVMs (11 wins) with the radial basis kernel were more accurate (mean RMSE test error for season 2004 and 2005) with RuleFit (14 wins) following close. SHIPS (12 wins) was not consistent in its performance between 2004 and 2005, a case where SVMs did really well. RBA was not at all competitive with any of the other methods, something that can be attributed to the absence of linear terms, whose behavior cannot be captured by a model consisting only of rules [FP05]. On the other hand, RuleFit with a combination of rules and linear

terms performed adequately well. Last but not least, NN (3 wins) did overfit certain hours, ranking as the fourth performing model. Again, there was not any consistently high performing model, whereas the largest difference in the mean RMSE for both 2004 and 2005 seasons between the best models was around 1.5 knots (except RBA). This particular observation is an indication that with this particular dataset the differences between different hypotheses in the model space are very small where no variation in the modeling is consistently better than another.

Table 4.5: The RMSE performance for hours 6 to 60 of the best models considered in this study. Testing was performed over two seasons (S): 2004 and 2005.

Hour	S	SHIPS	GA N1	GA N2	GA N3	RF	RBA	NN	SVMR
6	04	5.58	5.65	5.63	5.62	**5.54**	6.12	5.55	5.55
	05	5.50	5.57	5.57	5.52	5.47	6.25	**5.39**	5.43
12	04	8.54	8.80	8.76	8.66	8.50	9.88	**8.47**	8.56
	05	8.96	9.30	9.15	9.00	**8.51**	10.48	8.55	8.53
18	04	**10.82**	11.30	11.22	11.10	11.02	13.06	10.86	10.92
	05	11.85	12.41	12.09	11.84	11.07	13.31	11.15	**10.98**
24	04	**12.66**	13.43	13.28	13.08	13.36	15.62	12.93	12.81
	05	14.15	14.79	14.36	13.97	**13.03**	16.07	13.19	13.28
30	04	**14.08**	15.13	14.96	14.58	14.78	18.47	14.86	14.72
	05	16.36	16.72	16.35	15.90	**14.07**	19.72	15.19	15.22
36	04	**15.23**	16.38	16.23	15.71	16.32	21.45	16.38	16.05
	05	18.25	18.32	17.91	17.47	**16.81**	22.28	16.95	16.92
42	04	**16.23**	17.20	17.20	16.58	17.88	23.89	17.82	17.15
	05	20.00	19.65	19.08	18.82	**17.59**	25.49	17.92	18.25
48	04	17.29	18.15	18.33	17.54	**16.68**	24.16	19.05	18.31
	05	21.63	20.82	20.09	20.03	**18.84**	25.87	19.15	19.26
54	04	18.35	19.21	19.48	18.59	**17.44**	25.44	20.39	19.42
	05	23.12	21.79	20.73	21.09	20.70	28.87	22.81	**20.23**
60	04	19.08	20.20	20.51	19.49	**19.03**	25.65	21.45	19.61
	05	24.57	22.81	21.31	22.14	21.81	28.84	24.37	**20.86**

Additionally, incremental training and testing was applied for the periods 2001 to 2005 (5 seasons) for the most accurate models. In incremental training and testing the models are trained for seasons 1982 to x (for example 1982 to 2000) and tested on season

Table 4.6: The RMSE performance for hours 66 to 120 of the best models considered in this study. Testing was performed over two seasons (S): 2004 and 2005.

Hour	S	SHIPS	GA N1	GA N2	GA N3	RF	RBA	NN	SVMR
66	04	19.63	21.16	21.45	20.18	**18.46**	29.05	21.84	19.94
	05	26.10	24.06	21.89	23.34	22.59	32.50	**21.42**	21.46
72	04	20.20	21.96	22.28	20.94	**19.17**	28.43	23.27	20.53
	05	27.14	25.47	22.56	24.51	**22.28**	31.18	26.67	22.38
78	04	20.63	22.55	22.92	21.61	**19.37**	28.15	23.99	20.86
	05	27.81	26.92	23.00	25.82	25.76	31.82	22.83	**22.33**
84	04	**20.83**	22.94	23.45	22.31	20.92	29.05	23.75	21.41
	05	28.00	28.03	22.98	27.03	24.51	32.13	26.76	**22.37**
90	04	**20.88**	23.21	23.83	22.67	21.86	28.23	23.65	22.14
	05	28.30	29.17	22.92	28.13	25.03	31.63	26.62	**22.21**
96	04	**20.82**	23.42	24.17	22.59	23.25	31.66	23.68	22.15
	05	28.44	30.15	22.81	29.10	24.83	35.39	26.07	**22.09**
102	04	**20.82**	23.71	24.55	22.69	22.69	30.59	26.56	22.59
	05	28.51	31.06	22.98	29.93	24.72	31.00	22.38	**22.15**
108	04	**20.62**	23.93	24.89	22.46	24.24	30.79	25.37	22.51
	05	28.81	31.97	23.58	30.91	23.78	31.54	22.56	**21.93**
114	04	**20.31**	24.07	25.22	22.2	25.58	30.95	25.21	22.88
	05	28.95	32.85	24.28	32.21	22.56	33.42	21.91	**21.64**
120	04	**20.15**	24.17	25.64	22.05	22.28	33.39	25.88	22.93
	05	28.93	33.33	25.19	33.05	22.66	32.21	22.03	**21.44**

Table 4.7: The mean test RMSE performance of the models in all hours for the season 2004 and 2005.

Season	SHIPS	GA N1	GA N2	GA N3	RuleFit	RBA	NN	SVMR
04	**17.14**	18.83	19.20	18.03	17.92	24.20	19.55	18.05
05	22.27	22.76	19.44	21.99	19.33	26.00	19.70	**18.45**
04-05	19.70	20.79	19.32	20.01	18.62	25.1	19.62	**18.25**

x+1 (for example 2001). This could help in the identification of the stability of each model over several seasons. Tables 4.8 and 4.9 present the results for all hours, while Table 4.10 summarizes the findings. RuleFit was the best method, especially due to its stability between seasons, with all other models following closely. It should be noted that SVMs overfitted the 2001 season without which their mean RMSE performance was 17.00 ± 1.04 knots between seasons.

Table 4.8: The mean RMSE performance for hours 6 to 60 based on incremental training and testing, along with the standard deviation of the RMSE for the seasons 2001 through 2005.

Hour	SHIPS	GA N1	GA N2	RuleFit	SVMR
6	4.93±0.63	4.98±0.62	4.99±0.62	4.91±0.63	4.89±0.57
12	7.89±0.98	8.11±0.97	8.13±1.00	7.75±0.82	7.78±0.80
18	10.10±1.40	10.47±1.42	10.48±1.48	9.85±1.26	9.93±1.10
24	11.79±1.83	12.34±1.84	12.33±1.87	11.90±1.60	11.73±1.50
30	13.34±2.16	14.07±2.07	13.96±2.07	13.21±1.42	13.45±1.84
36	14.69±2.41	15.54±2.32	15.34±2.21	14.88±1.66	14.88±2.06
42	15.92±2.71	16.83±2.73	16.52±2.41	16.57±1.17	16.22±2.30
48	17.12±2.95	18.00±3.22	17.62±2.62	17.42±1.74	17.42±2.52
54	18.25±3.19	19.07±3.70	18.66±2.87	18.79±2.23	18.56±2.85
60	19.44±3.41	20.13±4.31	19.71±3.23	19.67±0.92	19.47±3.77

4.5 Interpretation

In this section the focus will be on the interpretation capabilities of RuleFit. RuleFit as a framework provides the opportunity to translate the derived rules into easily read diagrams, displaying the importance of each input variable and its interaction with other variables. This comes in addition to studying the rules in their original format. Of course the particular interpretation capabilities described in Chapter 3 can be applied to other models as well, but RuleFit was chosen since it gave the lowest RMSE performance.

The first goal was to estimate the input variable importance. Figure 4.4 has the

Table 4.9: The mean RMSE performance for hours 66 to 120 based on incremental training and testing, along with the standard deviation of the RMSE for the seasons 2001 through 2005.

Hour	SHIPS	GA N1	GA N2	RuleFit	SVMR
66	20.69±3.74	21.17±5.01	20.79±3.59	20.21±3.31	20.47±4.39
72	21.75±4.25	22.01±5.83	21.69±4.22	19.91±3.80	21.5±4.94
78	22.62±4.67	22.74±6.51	22.50±4.80	21.92±4.35	22.29±5.64
84	23.32±4.61	23.36±6.76	23.23±4.95	23.58±3.47	23.11±5.68
90	23.83±4.38	23.72±6.69	23.77±5.03	23.19±3.15	23.57±5.66
96	24.08±4.28	23.93±6.54	24.11±5.00	23.22±1.86	23.83±5.19
102	24.30±4.43	24.24±6.43	24.44±4.94	23.34±3.34	24.03±4.81
108	24.56±4.87	24.72±6.11	24.79±4.94	23.40±1.58	23.91±4.13
114	24.74±5.40	25.22±5.76	25.05±4.52	24.20±2.81	24.08±3.77
120	24.97±5.96	25.71±5.21	25.29±4.33	23.04±3.28	23.85±3.43

Table 4.10: The mean and standard deviations of the mean RMSEs for the 5 seasons (2001 through 2005) based on incremental training and testing.

Model	SHIPS	GA N1	GA N2	RF	SVMR
Mean	18.42	18.82	18.67	**18.05**	18.25
Std	2.74	3.87	3.14	**1.55**	2.94

average importance over the 20 datasets for all 37 predictors. It is interesting that the 6 most selected predictors (*VMAX, INCV, POT, SHRD, Z850* and *LSHR*) from Figure 4.3 along with SHRG have the highest importance rate. The figure additionally indicates that the variables are potential contributors (none seems to be completely irrelevant), but as mentioned earlier as one moves further ahead, predictions are more erroneous, making models with less important variables more susceptible to noise and overfitting.

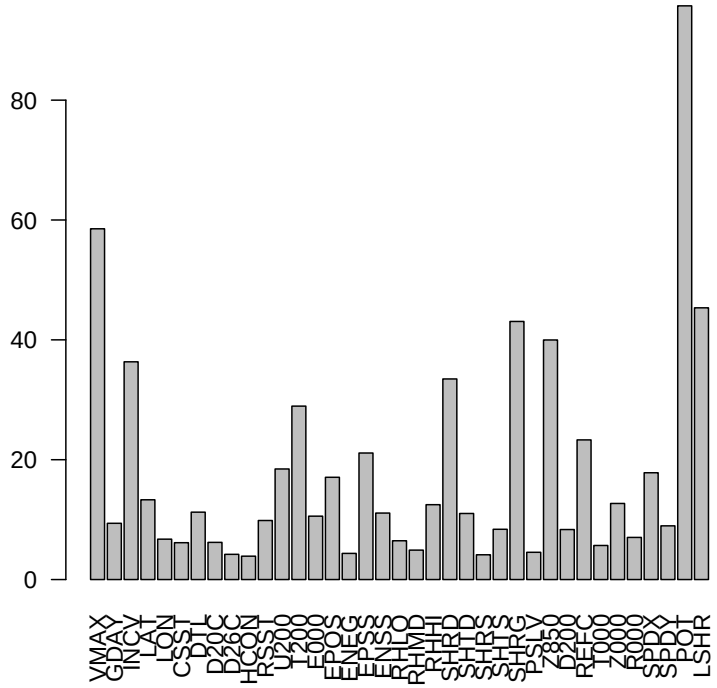


Figure 4.4: Variable importance based on the rules derived by RuleFit for all 20 datasets.

RuleFit rule interpretation is also easy for the user. Below some interesting rules between the five most important rules of their dataset are displayed, derived using the set of the most interpretable predictors (*VMAX, INCV, D200, POT, SHRD, T200, PSLV, Z850, GDAY, EPOS, LHSR* [DeM06]):

```
IF (-3.5 < INCV) AND (50.5 < POT), THEN
TC speed will increase by 0.75 knots 6 h ahead
```

This rule suggests that if the maximum potential intensity of the TC is high and there was an increase in the wind speed the last 12 hours that will lead to an increase in the wind speed six hours from now.

```
IF (42.5 < VMAX) AND (17 < SHRD), THEN
TC speed will decrease by 0.95 knots 12 h ahead
```

Knowing that shear (SHRD) has a negative contribution to the TC intensification plus the fact that VMAX is already above a certain magnitude will cause the hurricane to abate. On the other hand:

```
IF (SHRD < 22) AND (61.5 < POT), THEN
TC speed will increase by 3 knots 42 h ahead
```

A low SHRD value along with high potential intensity will lead the hurricane to intensify at t=42 hours.

```
IF (POT < 72) AND (3 < LSHR), THEN
TC speed will decrease by 1.5 knots 24 h ahead


IF (POT < 71) AND (3.5 < LSHR)
TC speed will decrease by 3 knots 42 h ahead
```

Both rules suggest that if the shear (even though partially canceled for higher latitude storms - LSHR) is big enough and the potential intensity is low the hurricane intensity will decrease.

```
IF (27.5 < VMAX < 107.5) AND (-4 < INCV) AND
(62.4 < Z850), THEN
```

```
TC speed will increase by 7.3 knots 54 h ahead


IF (VMAX < 107.5) AND (-0.5 < INCV) AND
(66.4 < Z850), THEN
TC speed will increase by 6.4 knots 60 h ahead
```

These rules display the fact that if the winds are not extremely high and there was an increase in the last 12 hours along with a synoptic environment more cyclonic than average at 850mb (Z850) the TC will intensify (the mean of Z850 is around 23 in the datasets).

```
IF (4.5 < INCV) AND (SHRD < 21) AND
(36 < POT), THEN
TC speed will increase by 5.2 knots 60 h ahead
```

For this rule, if the hurricane has intensified, the shear is low and the potential intensity is high, the hurricane's wind speed will increase.

```
IF (Z850 < 1.3) AND (44 < POT < 99), THEN
TC speed will decrease by 3.5 knots 66 h ahead
```

A below average Z850 with an average potential intensity will weaken the hurricane.

```
IF (0.8 < GDAY) AND (14.7 < Z850) AND
(73 < POT), THEN
TC speed will increase by 12 knots 90 h ahead
```

Finally, a hurricane at the peak of its season, with large Z850 and large POT will intensify after 90 hours.

The most important aspect of the RuleFit framework with respect to MLR, NNs or SVMs is providing, both through rules and interaction diagrams, interdependencies

between one or two variables with other predictors in the set. To explore them the following procedure was performed: First, the two variables that interact the most with the others were identified and then interaction diagrams for those two together and for the two separately against all others were derived for each dataset. Some examples of the most often seen two variable interactions can be found in Figures 4.5, 4.6 and 4.7. No three variable interaction was found to be relatively important based on the procedure above. Perhaps a more exhaustive search can identify three variable interactions.

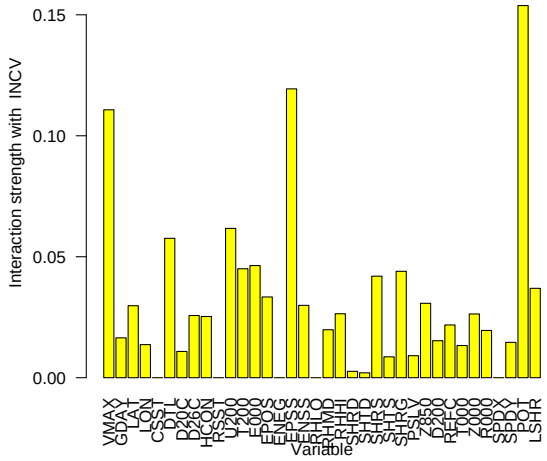


Figure 4.5: Interaction of INCV all variables. Notable interaction seen with VMAX, POT and EPSS (12 hours ahead).

4.5.1 Hand-picked Model

Different interactions were found for the different datasets. The most common and major interactions of variables X and Y were picked to create features in the form $X \times Y$ and a hand-picked model was built using MRL, in an effort to incorporate knowledge to aid the accuracy of the predictions. The 7 most important predictors, as mentioned earlier were selected, along with LAT that showed interaction effects with VMAX. Also

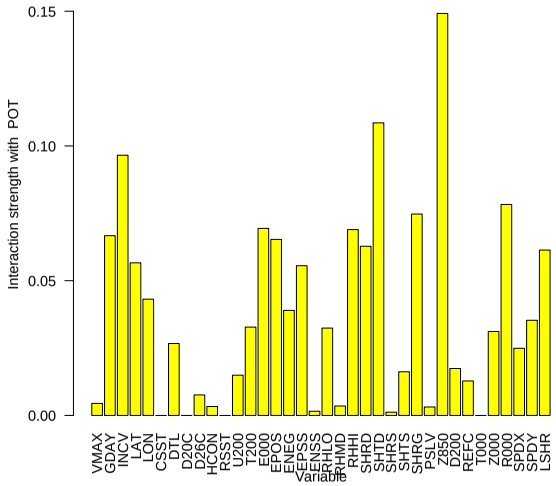


Figure 4.6: Notable Interaction of POT with Z850 (54 hours ahead).

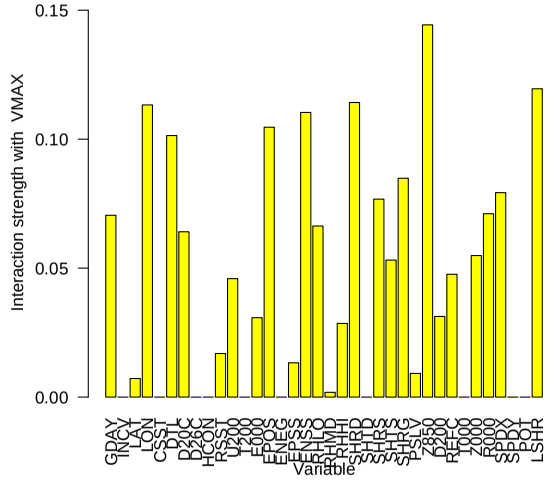


Figure 4.7: Interaction of VMAX with SHRD, LON, Z850 and LSHR, among others (120 hours ahead).

POT squared (POT2) was included because it was selected by both SHIPS and GA N2. The other features were: $POT \times INCV$ (PINCV), $VMAX \times LAT$ (VLAT), $VMAX \times SHRD$ (VSHRD), $VMAX \times LSHR$ (VLSHR), $VMAX \times SHRG$ (VSHRG), $POT \times Z850$ (PZ850), $POT \times LSHR$ (PLSHR), $POT \times SHRD$ (PSHRD) for a total of 17 features, 8 of them linear. Among them $VMAX \times SHRD$, an important interaction variable used in the SHIPS model [DK99].

The test 2004 RMSE was 17.00 and the test 2005 was 18.59, making it the best model. Furthermore, incremental training and testing was performed, giving a mean of 16.80 and a standard deviation of ± 2.08 making it also one of the most stable models. It should be noted that for 2003 the mean RMSE was only 14.34, while in none of the seasons did the RMSE surpass the 19 knot threshold. In Tables 4.11 and 4.12 the β coefficients for all the datasets are presented.

Table 4.11: The β coefficients of the hand-picked model for hours 6 to 60.

	Forecast (h)									
Pred.	6	12	18	24	30	36	42	48	54	60
VMAX	2.1	5.5	9.5	12.7	15.5	17.6	19.6	20.9	21.8	22.7
INCV	1.3	1.2	0.7	-0.1	-0.8	-1.2	-1.5	-1.3	-0.5	-0.5
LAT	0.5	1.1	2.0	2.5	3.0	3.1	3.4	3.2	2.8	2.4
SHRD	1.7	4.2	7.8	12.7	18.6	24.8	31.4	36.0	38.4	39.8
SHRG	-0.6	-2.3	-4.2	-6.3	-8.9	-11.9	-14.5	-16.4	-17.1	-16.9
Z850	0.6	1.4	2.1	2.7	3.0	3.1	2.8	2.7	2.5	2.3
POT	2.7	6.3	9.	12.2	14.3	16.1	17.7	19.2	21.4	23.4
LSHR	-0.7	-1.2	-2.9	-5.3	-8.7	-11.6	-14.9	-16.1	-15.7	-15.8
POT2	-1.8	-4.1	-5.9	-6.8	-7.3	-7.6	-7.6	-7.2	-7.3	-7.7
PINCV	0.7	2.0	3.2	4.6	5.5	6.1	6.5	6.2	5.4	5.2
VLAT	-2.0	-5.0	-8.8	-11.8	-14.9	-17.7	-20.4	-22.0	-22.8	-23.9
VSHRD	-3.5	-9.8	-18.5	-27.7	-37.6	-46.7	-55.3	-60.8	-63.1	-64.2
VLSHR	2.1	5.8	11.4	16.6	22.4	27.2	31.8	33.7	33.8	34.3
VSHRG	0.3	1.8	4.1	7.3	10.9	14.9	18.2	20.5	21.2	20.5
PZ850	-0.1	-0.2	-0.3	-0.3	-0.1	0.3	1.2	1.9	2.6	3.1
PLSHR	0.6	0.8	1.4	2.7	4.5	6.6	8.6	10.0	10.6	11.7
PSHRD	-0.6	-0.9	-1.5	-3.2	-5.7	-8.9	-12.8	-16.4	-19.4	-22.6

Table 4.12: The β coefficients of the hand-picked model for hours 66 to 120.

	Forecast (h)									
Pred.	66	72	78	84	90	96	102	108	114	120
VMAX	23.8	24.1	25.1	26.1	28.3	30.3	32.2	33.2	32.5	30.3
INCV	-1.2	-1.6	-1.8	-1.9	-2.5	-2.9	-3.1	-3.4	-4.1	-4.5
LAT	1.8	1.0	0.6	0.2	0.8	1.7	2.4	2.4	1.4	-0.4
SHRD	41.5	41.1	40.4	40.5	42.9	45.3	47.2	48.0	49.4	49.7
SHRG	-17.1	-17.3	-16.8	-16.3	-15.6	-15.9	-16.3	-16.8	-17.3	-17.8
Z850	2.1	1.8	1.2	0.6	-0.1	-1.1	-2.2	-2.7	-3.0	-3.3
POT	25.4	27.1	28.7	30.4	31.6	32.8	34.2	35.9	37.7	39.3
LSHR	-15.7	-12.9	-10.1	-7.8	-8.4	-8.3	-7.6	-6.0	-4.4	-1.4
POT2	-8.2	-8.3	-8.0	-7.7	-6.7	-5.8	-5.1	-5.2	-5.4	-5.5
PINCV	5.8	5.9	5.7	5.3	5.5	5.4	5.0	4.7	4.8	4.9
VLAT	-25.2	-25.8	-27.0	-27.8	-29.9	-32.6	-35.1	-36.7	-35.9	-33.1
VSHRD	-66.1	-66.5	-65.7	-64.4	-65.0	-65.8	-66.6	-66.7	-66.5	-64.7
VLSHR	34.4	32.6	30.6	28.4	29.1	29.9	30.4	30.0	28.1	23.9
VSHRG	20.8	21.8	21.6	20.7	19.2	18.5	18.1	17.9	18.0	18.5
PZ850	3.7	4.0	4.8	5.5	6.5	7.6	8.7	9.0	9.1	9.0
PLSHR	12.7	12.5	11.9	11.5	11.7	10.8	9.7	8.3	7.3	6.1
PSHRD	-26.0	-27.7	-29.5	-32.9	-35.5	-38.0	-39.9	-40.9	-42.7	-44.1

The seasons for incremental training and testing were too few in order to test for statistical significance of the improvement of this model over SHIPS, if there is any. More testing seasons are needed. The main conclusion that can be drawn is that more potential features may be available in order to improve the SHIPS model. On the other hand, these additional features will not eliminate some intrinsic difficulties like predicting rapidly intensifying hurricanes or other special kinds of hurricanes. Still for some seasons the model will be great and for others will be somewhat inaccurate. It is the author's belief that by adding or subtracting features the behavior between seasons will change but on average it will be the same. Thus some major changes need to be introduced and some ideas for that are discussed in the next chapter.

4.5.2 Rapid Intensification: Hurricane Wilma

One of the most difficult kinds of predictions is to forecast a rapidly intensifying hurricane. As an extra comparison the hand-picked model was tested against the SHIPS 2003 model for this kind of prediction. Figures 4.8 and 4.9 display four forecasts at time 1200 UTC given by the SHIPS and the hand-picked model, respectively.

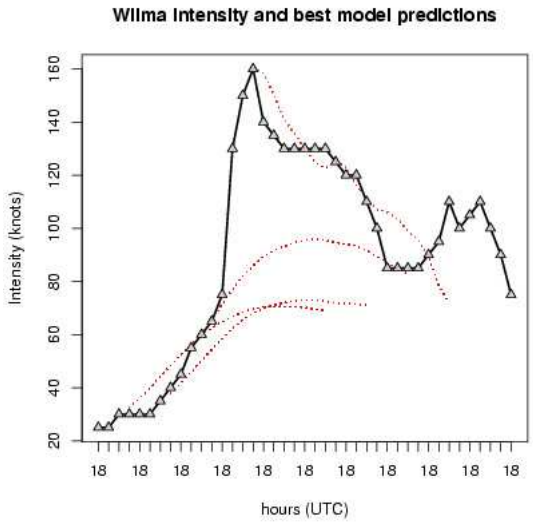


Figure 4.8: Predicting the intensification of hurricane Wilma up to 120 hours ahead using the predictors of the SHIPS 2003 model. The solid line indicates the wind speed of the hurricane at different times and days, while the dotted lines are the predictions of the wind speed by the SHIPS model.

It is evident that none of the models is able to capture the rapid intensification of hurricane Wilma in its entirety. It is still under investigation from the atmospheric science community to develop a solid understanding for interpreting the rapid intensification of hurricanes. Both models have their advantages, though. The hand-picked model gives a smaller error regarding the maximum intensity of the hurricane, while the SHIPS model makes more accurate prediction for the phase where the hurricane abates. The hand picked model follows the curve during that period, but provides an overestimate of wind speed. If one argues that predicting the intensification phase is more important than

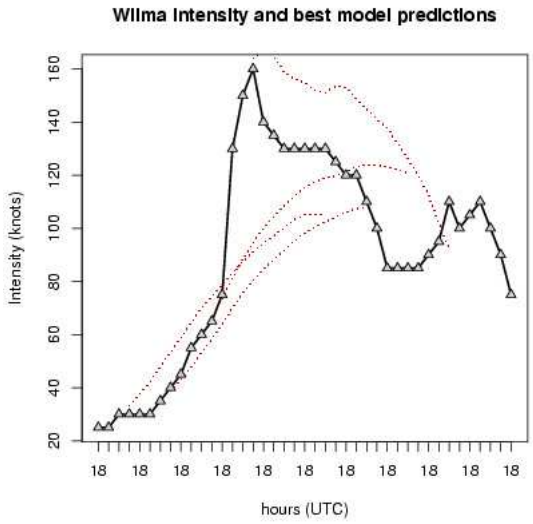


Figure 4.9: Predicting the intensification of hurricane Wilma up to 120 hours ahead using the predictors of the hand-picked model. The solid line indicates the wind speed of the hurricane at different times and days, while the dotted lines are the predictions of the wind speed by the hand-picked model.

the phase where the hurricane weakens, the hand-picked model can be thought as more competent.

Chapter 5

Conclusions and Future Work

The goal of this thesis was to determine if recent methods from the statistical, machine learning field could produce better models than simple linear regression with backward elimination. A rigorous procedure was followed for computing learning curves, evaluating model assessment methods, applying an ensemble of different feature selection techniques, deriving complex linear and non-linear models and studying interaction effects between variables, with good generalization performance the objective in mind. It should also be noted that many of the techniques studied here were not previously found in the literature of TC intensity forecasting (feature selection using GAs, SVMs, regression rule-based schemes).

The main conclusions are summarized in the following paragraphs. "Noise" increases as predictions are made further ahead into the future. This can be attributed to the averaging procedure for computing the variables along the track of the hurricane. On the other hand when this is not the case (6 hours ahead) the error of the models is below or at the ± 5 knot error that is attributed to the error in the measurement of the wind speed of a hurricane.

Genetic algorithms for feature selection outperformed other methods such as PCA, backward elimination and forward selection in the framework considered in this study, since it looks for combination of variables instead of considering one variable at a time

in a greedy fashion.

Non-linear methods (SVMs, NNs) did not provide more accurate predictions and they were found to overfit certain time intervals (especially after 72 hours) and/or seasons (case of 2001 and SVMs). On the other hand, linear models with non-linear aspects (MLR with non-linear terms, RuleFit with linear terms and rules) were empirically proved to be more robust and effective.

No model was able to significantly outperform the 2003 SHIPS model. A hand picked model, based on the knowledge acquired from the results presented in this study (feature selection, variable importance, rules, interaction diagrams) was capable of outperforming the SHIPS by 1.5 knots in the incremental training and testing giving also a lower standard deviation of 1 knot between the seasons into consideration.

The results were also found to be very sensitive to the particular season. For example, in all the models the error increased when predicting the 2005 season versus the 2004 season. One factor that plays a role in this behavior is that that certain seasons have more rapidly intensifying hurricanes than others.

So far statistical methods outperform more general dynamical models. In order to further boost their accuracy certain steps can be considered, and are presented here as future work that can be done in this field. First of all, infrared satellite data, containing information about the storm itself, have shown potential for improving the forecasts, but they are not available for the entire data sample, and so were not included in this study. Secondly, instead of just calculating the average along the track, one can use the minimum and the maximum of the particular variables to account for big or small oscillations of the values along the track of the TC. Going further, all the values along the path may be used. Since more inputs will be available the feature selection procedure must be reapplied. Finally, since interactions seemed to be helping the model perhaps a more detailed study of them with the input of experts would result in more accurate

predictions. It is the author's personal belief that the models without the improvements above will not be able to perform better than what is presented here.

REFERENCES

[Abe01] Sim D. Aberson. The ensemble of tropical cyclone track forecasting models in the north atlantic basin (1976-2000). *Bulletin of the American Meteorological Society*, 82(9):1895–1904, 2001.

[AIS94] R. Agrawal, T. Imielinski, and A. Swami. Fast algorithms for mining association rules. In *Proceedings of the 20th International Conference on Very Large Databases*, 1994.

[AL99] Yonatan Aumann and Yehuda Lindell. A statistical theory for quantitative association rules. In *Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Diego, CA, 1999.

[AM02] Cristophe Ambroise and Geoffrey J. McLachlan. Selection bias in gene extraction on the basis of microarray gene-expression data. In *PNAS*, volume 99, pages 6562–6566, May 2002.

[BFOS83] L. Breiman, J. H. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees.* Wadsworth, 1983.

[BH98] Jong-Jin Baik and Hong-Sub Hwang. Tropical cyclone intensity prediction using regression method and neural network. *Journal of the Meteorological Society of Japan*, 76(5):711–717, October 1998.

[BH00] Jong-Jin Baik and Hong-Sub Hwang. A neural network model for predicting typhoon intensity. *Journal of the Meteorological Society of Japan*, 78(6):857–869, December 2000.

[Bow95] Nathaniel Bowditch. *The American Practical Navigator: An Epitome of Navigation*, chapter 36 Tropical Cyclones, pages 505–520. National Imagery and Mapping Agency, 1995.

[Bre01] L. Breiman. Random forests, random features. Technical report, University of California, Berkeley., 2001.

[BRJL05] Eric S. Blake, Edward N. Rappaport, Jerry D. Jarrell, and Christopher W. Landsea. The deadliest, costliest, and most intense united states tropical

cyclones from 1851 to 2004 (and other frequently requested hurricane facts). Technical Memorandum NWS TPC-4, NOAA, Tropical Prediction Center, National Hurricane Center, Miami, Florida, August 2005.

[BSD04] Robbie Berg, Chris Sisko, and Mark DeMaria. High resolution SST in the SHIPS model: Improving Operational Guidance of Tropical Cyclone Intensity Forecasts. In *26th Conference on Hurricanes and Tropical Meteorology*, 2004.

[Cas04] Anthony Veneros Castro. A neural network approach to predict hurricane intensity in the north atlantic basin. Master's thesis, University of Puerto Rico, 2004.

[CJPB04] Timothy J. Considine, Christopher Jablonowski, Barry Posner, and Craig H. Bishop. The value of hurricane forecasts to oil and gas producers in the gulf of mexico. *Journal of Applied Meteorology*, 43(9):1270–1281, 2004.

[CST00] Nello Cristianini and John Swave-Taylor. *An Introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press, 2000.

[DeM06] Mark DeMaria. Personal communication, 2006.

[DK94a] Mark DeMaria and John Kaplan. Sea surface temperature and the maximum intensity of atlantic tropical cyclones. *Journal of Climate*, 7(9):1324–1334, 1994.

[DK94b] Mark DeMaria and John Kaplan. A statistical hurricane intensity prediction scheme (ships) for the atlantic basin. *Weather and Forecasting*, 9(2):209–220, June 1994.

[DK99] Mark DeMaria and John Kaplan. An updated statistical hurricane intensity prediction scheme (ships) for the atlantic and eastern north pacific basins. *Weather and Forecasting*, 14(3):326337, June 1999.

[DMS+03] Mark DeMaria, Michelle Mainelli, Lynn K. Shay, John A. Knaff, and James P. Kossin. Improvements in Real-Time Statistical Tropical Cyclone Intensity Forecasts Using Statellite Data. In *Joint 12th Conference on Satellite Meteorology and Oceanography and 3rd Conference on Artificial Intelligence Applications to Environmental Science*, 2003.

[DMS+05] Mark DeMaria, Michelle Mainelli, Lynn K. Shay, John A. Knaff, and John Kaplan. Further improvements to the statistical hurricane intensity prediction scheme (ships). *Weather and Forecasting*, 20(4):531–543, August 2005.

[Ema03] Kerry Emanuel. Tropical cyclones. *Annual Review of Earth and Planetary Sciene*, 31:74–104, 2003.

[FI93] U. M. Fayad and K. B. Irani. Multi-interval discretization of continuous-valued attributes for classification learning. In *Proceedings of IJCAI-93*, 1993.

[FP05] Jerome H. Friedman and Bogdan E. Popescu. Predictive Learning via Rule Ensembles. Preprint, October 2005.

[Fre02] Alex A. Freitas. *Data Mining and Knowledge Discovery with Evolutionary Algorithms*. Springer, 2002.

[Fri01] Jerome H. Friedman. Greedy functions approximation: a gradient boosting machine. *Annals of Statistics*, 29:1189–1232, 2001.

[GE03] Isabelle Guyon and Andre Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

[HTF01] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2001.

[JN79] B. R. Jarvinen and C. J. Neumann. Statistical forecasts of tropical cyclone intensity. Tech. Memo 10, 22, NOAA, 1979.

[KBTR95] Yoshio Kurihara, Morris A. Bender, Robert E. Tuleya, and Rebbecca J. Ross. Improvements in the gfdl hurricane prediction scheme. *Monthly Weather Review*, 123:2791–2801, 1995.

[KDS04] John A. Knaff, Mark DeMaria, and Charles R. Sampson. An Introduction to the statistical Typhoon Intensity Prediction Scheme (STIPS). In *26th Conference on Hurricanes and Tropical Meteorology*, 2004.

[KDSG03] John A. Knaff, Mark DeMaria, Charles R. Sampson, and James M. Gross. Statistical, 5-Day Tropical Cyclone Intensity Forecasts Derived from Climatology and Persistence. *Weather and Forecasting*, 18:80–92, 2003.

[Kir01] Michael Kirby. *Geometric Data Analysis: An Empirical Approach to Dimensionality Reduction and the Study of Patterns*. Wiley, 2001.

[LG99] P. Leray and P. Gallinari. Feature selection with neural networks. *Behaviormetrika*, 26, January 1999.

[Man96] Mylene Mangalindan. How much does it cost to prepare for a hurricane? $670,000 per mile . . . at least. *The Virginian-Pilot*, 1996.

[McG04] Michael G. McGauley. Hurricane intensity forecasting with neural networks. In *26th Conference on Hurricanes and Tropical Meteorology*, 2004.

[Mit97] Tom M. Mitchell. *Machine Learning*. McGraw-Hill, 1997.

[OTK04] Aysel Ozgur, Pang-Ning Tan, and Vipin Kumar. Rba: An integrated framework for regression based on association rules. In *SIAM International Conference on Data Mining (SDM'04)*, Florida, USA, 2004.

[Qui92] J. Ross Quinlan. Learning with continuous classes. In *AI*, 1992.

[RBV04] Nazario D. Ramirez-Beltran and Anthony Veneros. Upper air information and neural networks to estimate hurricane intensity. In *26th Conference on Hurricanes and Tropical Meteorology*, 2004.

[RN03] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice-Hall, Englewood Cliffs, NJ, 2nd edition edition, 2003.

[SS98] A. J. Smola and B. Schölkopf. A tutorial on support vector regression. NeuroCOLT Technical Report NC-TR-98-030, Royal Holloway College, University of London, UK, 1998.

[TYK05] Jiang Tang, R. Yang, and M. Kafatos. Data mining for tropical cyclone intensity prediction. In *Sixth Conference on Coastal Atmospheric and Oceanic Prediction and Processes*, 2005.

[VOM99] S. Vinterbo and L. Ohno-Machado. A genetic algorithm to select variables in logistic regression: example in the domain of myocardial infarction. *Journal of the American Medical Informatics Association*, 6(Suppl.):984–988, 1999.

[WF00] Ian H. Witten and Eibe Frank. *Data Mining: Practical machine learning tools with Java implementations*. Morgan Kaufmann, 2000.

[Whi94] Darrell Whitley. A genetic algorithm tutorial. *Statistics and Computing*, 4:65–85, 1994.

[WRFDM06] Hugh E. Willoughby, E. Rappaport, and Jr F. D. Marks. Hurricane forecasting: The state of the art. In *First Symposium on Policy Research*, 2006.

[WW04] Y. Wang and C.-C Wu. Current understanding of tropical cyclone structure and intensity changes - a review. *Meteorology and Atmospheric Physics*, 87:257–278, 2004.