2014

Application and Extension of Weighted Quantile Sum Regression for the Development of a Clinical Risk Prediction Tool

Ghalib Bello
Virginia Commonwealth University

Application and Extension of Weighted Quantile Sum Regression for the Development of a Clinical Risk Prediction Tool

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at Virginia Commonwealth University.

By

Ghalib A. Bello

B.A. Whittier College, Whittier, CA 90608

Director: Dr. Chris Gennings, Professor, Biostatistics

Virginia Commonwealth University

Richmond, Virginia

May, 2014

ACKNOWLEDGEMENTS

First and foremost I thank God, without whose will none of this would be possible.

I would like to express my deepest gratitude to my advisor, Dr. Chris Gennings, for her guidance, support, and excellent mentorship. Her insights and constructive feedback have guided and shaped this dissertation and her constant encouragement fueled my enthusiasm. It has been a great privilege and pleasure to work under her outstanding guidance.

My appreciation also goes to my committee members, Dr. Robert Johnson, Dr. Nitai Mukhopadhyay, Dr. Arun Sanyal and Dr. David Wheeler, for their time, their efforts in reading my oral exam and dissertation, and their valuable comments, suggestions and corrections. I owe special thanks to Dr. Johnson for his mentorship, advice and encouragement throughout my doctoral studies. I learned a great deal from him and I feel privileged to have worked with him. I am extremely thankful to Dr. Mukhopadhyay for his advice, assistance, and his steadfast support from my first statistics courses to the completion of my program.

I am also indebted to Dr. Roy Sabo for contributing to my growth as a statistician through his role as a supervisor and collaborator on multiple projects. He has been incredibly supportive of, and patient with me throughout our association. I have benefited greatly from my research assistantship in the VCU Department of Family Medicine and Population Health and I would like to thank all the people I had the pleasure of working with over the years.

Last but not least, I am deeply grateful to my family for their unwavering belief in me and their constant encouragement and patience throughout my long journey in academia. Without their love, sacrifices and prayers, I would never have made it to where I am today.

Table of Contents

List of Figures

Figure **Page**

Chapter 2

Chapter 3

Chapter 4

Chapter 5

Chapter 6

List of Tables

Chapter 5

Chapter 6

Abstract

APPLICATION AND EXTENSION OF WEIGHTED QUANTILE SUM REGRESSION FOR THE DEVELOPMENT OF A CLINICAL RISK PREDICTION TOOL

By Ghalib A. Bello, Ph.D.

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at Virginia Commonwealth University

Virginia Commonwealth University, 2014

Director: Dr. Chris Gennings, Professor, Biostatistics

In clinical settings, the diagnosis of medical conditions is often aided by measurement of various serum biomarkers through the use of laboratory tests. These biomarkers provide information about different aspects of a patient's health and the overall function of different organs. In this dissertation, we develop and validate a weighted composite index that aggregates the information from a variety of health biomarkers covering multiple organ systems. The index can be used for predicting all-cause mortality and could also be used as a holistic measure of overall physiological health status. We refer to it as the Health Status Metric (HSM). Validation analysis shows that the HSM is predictive of long-term mortality risk and exhibits a robust association with concurrent chronic conditions, recent hospital utilization, and self-rated health.

We develop the HSM using Weighted Quantile Sum (WQS) regression (Gennings et al., 2013; Carrico, 2013), a novel penalized regression technique that imposes nonnegativity and unit-sum constraints on the coefficients used to weight index components. In this dissertation, we develop a number of extensions to the WQS regression technique and apply them to the construction of the HSM. We introduce a new guided approach for the standardization of index components which accounts for potential nonlinear relationships with the outcome of interest. An extended version of the WQS that accommodates interaction effects among index components is also developed and implemented. In addition, we demonstrate that ensemble learning methods borrowed from the field of machine learning can be used to improve the predictive power of the WQS index. Specifically, we show that the use of techniques such as weighted bagging, the random subspace method and stacked generalization in conjunction with the WQS model can produce an index with substantially enhanced predictive accuracy. Finally, practical applications of the HSM are explored. A comparative study is performed to evaluate the feasibility and effectiveness of a number of 'real-time' imputation strategies in potential software applications for computing the HSM. In addition, the efficacy of the HSM as a predictor of hospital readmission is assessed in a cohort of emergency department patients.

Chapter 1

Introduction and Prospectus

1.1 Introduction

In the context of clinical/medical applications, a *risk score* is a diagnostic tool for gauging the health of a patient and/or predicting their prognosis for a particular condition. In the last few decades, clinical risk scores have become useful and indispensable tools for clinical diagnosis and medical decision making (Steyerberg, 2009). They generally combine various measures of health risk factors (e.g. cholesterol level, blood glucose, smoking status, demographics) into a composite score that is capable of predicting the risk of a certain endpoint/outcome.

Most clinical risk scores are developed to predict outcomes for a specific condition or population and therefore have limited usefulness outside the scope of their intended target. Some, for example, focus on prediction of risk for particular conditions, e.g. cardiovascular disease (Framingham Risk Score (Wilson *et al.*, 1998)), kidney disease (QKidney® (Hippisley-Cox *et al.*, 2010)), diabetes (ADA Diabetes Questionnaires (Heikes *et al.*, 2008)), etc. Others focus on prediction of health outcomes for specific cohorts, e.g. pediatric patients (PRISM) and intensive care patients (APACHE (Knaus *et al.*, 1991)). Recent work (Horne *et al.*, 2009; Gennings *et al.*, 2012) has led to the development of risk scores with a more general scope of applicability. These are primarily intended to predict all-cause mortality for the general population, as opposed to specific cohorts. One such instrument is the Intermountain Risk Score (Horne *et al.*, 2009), henceforth referred to as the IMRS. The IMRS includes test results from the Complete Blood Count (CBC) and the Basic Metabolic Profile (BMP), a panel of tests for

assessing metabolic health. It also includes age in its risk model, which is perhaps the strongest predictor of mortality. Therefore it is possible that the predictive power of this risk-assessment tool may, in large part, be due to the use of age as a component of the risk model.

In the following chapters, we detail the development, validation and extension of a new risk score for producing a holistic measure of overall health. This Health Status Metric (HSM) covers a wider range of tests than the IMRS. It includes results from the Complete Blood Count, the Lipid Panel, and the Comprehensive Metabolic Panel (CMP). The latter is an expanded version of the Basic Metabolic Panel which includes tests of liver function and provides a broader and more extensive assessment of the body's chemical balance and metabolism. The Lipid Panel provides, among other things, assessment of cardiac risk, which is one of the most prevalent causes of mortality in the United States (Hoyert & Xu, 2012). In addition, the HSM also includes serum biomarkers like Hemoglobin A1c (a measure of blood glucose concentration), Phosphorus, and C-reactive protein which are known to be prognostic indicators of multiple health conditions (Black *et al.*, 2004; Goldman & Schafer, 2011; Luk *et al.*, 2013; Matsushita *et al.*, 2010). The HSM also does not use demographic risk predictors (instead adjusting for them) and, with the exception of blood pressure, is composed entirely of biomarkers from common clinical laboratory tests. This index, with just clinical biomarkers, demonstrates strong predictive ability for all-cause mortality, and multiple endpoint-specific causes of mortality (liver disease, kidney disease, diabetes).

Therefore the HSM could potentially be a useful tool in clinical settings for accurate quantification of mortality risk (life expectancy) in individuals with known health issues. The HSM also correlates strongly with current health status as assessed by self-rated health, concurrent chronic conditions and recent hospital utilization. It could therefore be used also as a

holistic measure of current health status. Because of the HSM's use of a wide range of biomarkers spanning multiple organ systems, it may serve as a particularly effective clinical tool for early identification of at-risk individuals that are asymptomatic at the time of measurement.

The HSM is computed as a weighted sum of the standardized values of each biomarker measurement, where each biomarker is weighted according to its (empirically-determined) relative strength of association with mortality. In other words, higher weights are assigned to biomarkers demonstrating a strong association with mortality, although this trend does not generally hold when high correlations exist among biomarkers. However the biomarkers we used in this study do not exhibit high intercorrelation or multicollinearity therefore the magnitudes of the weights tend to reflect the associative strength of the corresponding biomarkers with mortality. The weights are computed using Weighted Quantile Sum (WQS) regression. WQS regression was introduced in Gennings *et al.* (2013) and characterized by Carrico (2013); it is a penalized regression technique that is particularly useful in variable selection problems where complex correlation patterns exist among the explanatory variables. WQS regression imposes a unit-interval and unit-sum penalty on the weights associated with the explanatory variables. This unique type of constraint has been shown to produce greater variable selection accuracy than more traditional penalized regression techniques (e.g. ridge regression) in scenarios where variables are highly collinear (Breiman, 1996a; Carrico, 2013). Prior to this point, WQS has primarily been applied to variable selection problems involving environmental chemical mixtures (Gennings *et al.*, 2013; Christensen *et al.*, 2013) but our application of the technique in this study is geared towards prediction (of mortality risk), as opposed to variable selection. The WQS technique is particularly ideal for risk score development because it produces an easily

interpretable index with a fixed, standardized range that allows for uniform comparison within a general population.

In constructing the HSM, we encountered unique challenges that required the development of a number of extensions to the Weighted Quantile Sum regression methodology. For example, in the original form of the WQS characterized in Carrico (2013), standardization of each component (e.g. environmental chemical exposure levels, biomarker levels) is carried out by dividing its range of measured values into quartiles. This is done primarily to dampen the disproportionate effect of extreme outliers on the results. However the use of quartiles (or other quantiles) to solve this problem is an *ad hoc* strategy that assumes the relationship between the components and the outcome/response is monotonic. In this thesis, we develop a more guided approach to standardizing biomarker measurements which does not involve the use of quantiles. We instead utilize spline-based Cox regression models that model possible non-monotonic relationships between each biomarker and the outcome (mortality).

A second extension to the WQS regression methodology involves the modification of the WQS model to accommodate interaction effects. The original form of the WQS model is based on an additive assumption that the effects of the components on the outcome variable can be adequately modeled without accounting for interactions among them. Therefore the HSM constructed using this model is a simple weighted sum of standardized components; however we reasoned that interaction effects may exist among components and in order to test this, we developed an extended version of the WQS model which allows pairwise interactions among components and could possibly also include interactions between components and adjustment covariates (e.g. age, gender, etc.).

Another extension to WQS methodology involves the use of advanced ensemble methods (borrowed from statistical/machine learning) for reducing variability in the estimated weights. In its original form, the WQS method uses bootstrap aggregation to produce stable weights (Carrico, 2013). However, we found that the use of this method to construct the HSM produces low predictive accuracy compared to more advanced ensemble methods. We developed weighted bootstrap aggregation and tree-based stacked generalization methods that significantly improve the predictive accuracy of the HSM.

1.2 Prospectus

Chapter 2 of this thesis is written in manuscript form. It details the development and validation of the Health Status Metric. The data sources used for developing and testing the HSM are covered, followed by a description of the guided approach we developed to standardize biomarker measurements (presented with examples). The details of the WQS regression method are outlined, particularly the procedure by which the HSM component weights are estimated. Next, the validation analyses performed to test the HSM's performance on independent datasets are covered.

Chapter 3 introduces the extension to the WQS model that allows for inclusion of pairwise interactions among components. Since all possible pairwise interactions among the 24 biomarkers (276 in total) could not be included in the HSM, we had to choose only a small subset of strong interactions. We identified a small set of potentially strong pairwise interactions using Random Survival Forests, a tree-based technique capable of modeling complex interaction effects in high-dimensional datasets. We explore 2 different thresholding techniques for selecting

'significant' interactions and present results for each. This is followed by an evaluation of the HSM's predictive accuracy when interactions are included in the index.

Chapter 4 addresses a significant concern regarding the practical use of the HSM as a clinical/diagnostic risk prediction tool. This is the issue of how to compute the HSM for individuals missing one or more biomarker values. In this chapter, we explore and compare a number of imputation techniques that can be used to tackle this problem. Using complete datasets with simulated missing values, the imputation techniques are compared on such metrics as general accuracy and impact of imputation on the predictive power of the HSM.

In **Chapter 5**, the focus is on the HSM's performance as a predictor, specifically, its predictive power for mortality. The WQS procedure used to construct the HSM involves the use of a form of bootstrap aggregation which entails fitting the WQS model (and estimating HSM weights) for a large number of bootstrap samples and then averaging estimated weights across the samples. In this chapter, we borrow techniques from the area of Machine/Statistical Learning, particularly ensemble methods. We begin by showing that the WQS procedure used to construct the HSM belongs to a common class of ensemble methods known as *bagging*, which makes the HSM a type of bagged predictor. We then experiment with more advanced ensemble methods (weighted bagging, stacked generalization) that could produce a new version of the HSM with higher predictive accuracy.

In **Chapter 6**, we test the HSM on real-world data. We use data from the VCU Medical Center Emergency Department. Patients who visited the Emergency Department (ED) within a fixed time frame (first 2 months in 2011) were followed for a period of 2 years after the initial

visit (labeled the ‘baseline visit’). Subsequent hospital visits (and details on these visits) occurring during the 2 year period were recorded. The goal in this chapter is to use the data on this longitudinal cohort to demonstrate a specific application of the HSM: as a predictor of hospital readmission/utilization after emergency department visits.

Chapter 7 concludes the thesis. This chapter contains a summary and discussion of the studies and results presented in this dissertation. Limitations of the current work are discussed, and future projects extending the ideas presented herein are proposed.

Chapter 2

Development and Validation of a Clinical Risk-Assessment Tool Predictive of All-cause Mortality

2.1 INTRODUCTION

In this chapter, we will detail the components of the Health Status Metric (HSM), the procedure we developed for standardizing biomarker measurements, and the estimation of HSM component weights using Weighted Quantile Sum regression. This will be followed by a description of the various analyses performed to validate the HSM and to demonstrate its versatility as a general-purpose risk score.

2.2 METHODS

2.2.1 Data Source & Risk Score Components

The HSM was developed using the NHANES 1999-2002 (CDC 2002) dataset (n=3406) and validated using the NHANES 2003-2008 (CDC 2008) dataset (n=4670) and the NHANES III:1988-1994 (CDC 1994) dataset (n=10592). The endpoint/outcome of interest was survival data which was obtained from NDI/NHANES Linked Mortality Files. These files are the result of efforts by the NCHS to conduct a probabilistic linkage of NHANES data to death certificate data found in the National Death Index (NDI). The files provide information about the death status and survival times (up to December 31, 2006) of NHANES 1999-2002 and NHANES III participants. In addition, information about underlying cause of death is available in the Linked Mortality Files. Questionnaire data from the continuous NHANES 2003-2008 data was used to examine the relationship between the HSM and a number of self-reported variables: health status, hospital utilization and diagnoses of diabetes, heart, kidney and liver disease.

A total of 24 biomarkers were used to develop the HSM. With the exception of blood pressure, all the biomarkers used are blood count/serum measurements most of which come from the Comprehensive Metabolic Panel, the Lipid Panel, and the Complete Blood Count (CBC), batteries of blood tests that are commonly performed in clinical settings for diagnostic purposes. Below is a list of the biomarkers classified by panel:

- Blood Pressure
- **Comprehensive Metabolic Panel**
  - Waste Products (Blood Urea Nitrogen [BUN], Creatinine)
  - Electrolytes (Sodium, Potassium, Chloride, Bicarbonate, Calcium)
  - Proteins (Albumin, Globulin)
  - Enzymes (Bilirubin, Alkaline Phosphatase [ALP], Aspartate Aminotransferase [AST], Alanine Aminotransferase [ALT])
- **Lipid Panel**
  - Triglycerides, HDL:Total cholesterol ratio
- **Complete Blood Count**
  - White Blood Cell, Red Blood Cell, & Platelet count
  - Hemoglobin, Hematocrit
- **Miscellaneous**
  - Hemoglobin A1c
  - Phosphorus
  - C-reactive protein

2.2.2 Weighted Quantile Sum (WQS) Regression

To construct the Health Status Metric (HSM), we use the Weighted Quantile Sum (WQS) methodology outlined in Carrico (2013) and Gennings *et al.* (2013). The WQS method is a penalized regression technique for multicollinear data. It was originally developed to handle environmental chemical mixture data where the variables (environmental chemical exposure

levels) exhibit complex intercorrelation patterns and can be logically grouped into a composite index. Traditional regression techniques typically fail in the presence of severe multicollinearity, therefore WQS regression presents a viable alternative. Briefly, it involves creating a weighted sum of all variables of interest (standardized onto the same scale) and using the resulting composite as a single variable in a regression model. The weights are unknown model parameters that are constrained to be between 0 and 1 and to sum to 1.

We will now outline the setup for the WQS model. Consider a set of *p* variables which can be logically combined into an index (e.g. a set of environmental chemical exposures, biomarkers, or gene expression levels). These variables might have different units of measurement so we 'standardize' them so that they are all on the same scale. This is typically done by scoring them into quantiles (e.g. tertiles, quartiles) however we will introduce a new standardization approach later in the chapter. Further, consider a separate set of *k* potentially confounding variables that we would like to adjust for (e.g. age, sex). Suppose the distribution of the outcome/response of interest belongs to the exponential family (although other outcomes [e.g. time-to-event] can be used). Then the general WQS model is given by:

$$g(\mu) = \beta_0 + \sum_{j=1}^{k} \alpha_j z_j + \beta_1 \left(\underbrace{\sum_{i=1}^{p} w_i q_i}_{\text{Index}} \right) \quad ----(2.1)$$

$$\text{constraints}: 0 \le w_i \le 1; \quad \sum_i w_i = 1$$

In Equation (2.1), g(·) is the familiar link function for generalized linear models. The quantity in the parentheses is the ***index***, the weighted sum of the *p* variables (each denoted by q_i) referred to earlier. The weights $\{w_i\}$ are unknown parameters that are estimated by fitting the model under

the defined constraints. The other unknown parameters are β_0 (intercept term), β_1 (coefficient of the index), and the $\{\alpha_j\}$ (coefficients of the demographic variables z_j).

Studies (e.g. Carrico, 2013; Gennings *et al.*, 2013) have shown that if a set of variables are highly inter-correlated, then the WQS approach of combining them into an index resolves many of the problems associated with multicollinearity. While the WQS model in Equation (2.1) might resemble the typical regression model, the keen observer will notice a key difference: since the weights $\{w_i\}$ and the β_1 coefficient are unknown parameters, this model is non-linear in its parameters, therefore it is a non-linear model. The parameters are estimated using non-linear estimation techniques which will be described in more detail in a subsequent section.

To guarantee stable estimates of the weight parameters $\{w_i\}$, a bootstrap aggregation technique introduced in Carrico (2013) is used, and the procedure is as follows: Let N_{tr} be the number of subjects in the training set. A large number B of bootstrap samples each of size N_{tr} are drawn (with replacement) from the training dataset. For each bootstrap sample, the model in Equation (2.1) is fit to obtain estimates of the weights. Therefore for each of the b=1 to B bootstrap samples, we obtain a set of p estimated weights $\{\hat{w}_{i(b)}\}_{i=1}^{p}$ and use them to compute an estimate of the index (i.e. a weighted quantile score) specific for that bootstrap sample:

$$\hat{WQS}_{(b)} = \sum_{i=1}^{p} \hat{w}_{i(b)} q_i \quad ----(2.2)$$

These WQS estimates are then averaged over the B bootstrap samples to derive the overall weighted quantile score:

$$\overline{WQS} = \frac{1}{B} \sum_{b=1}^{B} \hat{WQS}_{(b)} \quad ---- \ (2.3)$$

Notice that plugging Equation (2.2) into Equation (2.3) allows us to obtain an alternate expression for the overall WQS that is in the form of a weighted sum:

$$\overline{WQS} = \sum_{i=1}^{p} \bar{w}_i q_i, \quad \text{where } \bar{w}_i = \frac{1}{B}\sum_{b=1}^{B} \hat{w}_{i(b)} \quad ----(2.4)$$

This shows that the overall WQS index is just a weighted sum of the *p* variables, where the weights are estimates averaged over a large number of bootstrap samples.

Note that in practice, for each bootstrap sample *b*, the estimated weights $\{\hat{w}_{i(b)}\}_{i=1}^{p}$ are tested to determine if the index they produce ($W\hat{Q}S_{(b)}$) is significantly associated with the outcome/response. Using the data in each bootstrap sample *b*, the significance of $W\hat{Q}S_{(b)}$ is tested with the following model:

$$g(\mu) = \beta_0 + \sum_{j=1}^{g} \alpha_j z_j + \beta_1 W\hat{Q}S_{(b)}\Big|_b$$

If $W\hat{Q}S_{(b)}$ is not statistically significant then this is an indicator of the low 'signal strength' of the weight estimates derived from this particular bootstrap sample, therefore it is not included in the computation of the overall WQS estimate given in Equation (2.3). Note that the above model is fitted to the data in bootstrap sample *b*. However, the subset of the training dataset not selected to be in bootstrap sample *b* (i.e. the 'out-of-bag' data [see Chapter 5]) could be used instead.

2.2.3 Health Status Metric (HSM) Construction

The HSM is constructed using the WQS methodology outlined above. In Equation (2.1) the WQS model is in the form of a generalized linear model but as mentioned, other response types can be used. The HSM is constructed using NHANES 1999-2002 linked mortality files as the training data. The outcome is therefore survival time so a Weibull Accelerated Failure Time

model was used a basis model for the WQS technique. The Weibull parameterization was chosen due to its demonstrated superior tradeoff between parsimony and goodness-of-fit in the training dataset. The model is given below:

$$\log T_k = \mu + \sum_j \alpha_j z_j + \beta \underbrace{\sum_i w_i r_i}_{\text{HSM}} + \sigma \Psi_k \quad ----(2.5)$$

Notice that the index in this model is the HSM. The r_i denote the standardized values for the biomarkers. Standardization of explanatory variables is an integral part of the WQS methodology and will be discussed in careful detail in a subsequent section. T_k denotes the random variable associated with the survival time of the k^{th} individual, β is the unknown coefficient of the HSM index, and μ and σ are parameters of the Weibull distribution, the $\{z_j\}$ represent the demographic covariates (age, gender, race, etc.) and the $\{\alpha_j\}$ are their unknown coefficients.

As mentioned earlier, the WQS has a non-linear form so fitting the model in Equation (2.5) requires non-linear optimization techniques. In Equation (2.5), $\mathbf{\Psi}_k$ is a random variable used to model the random deviation of $\log T_k$ from its expected value according to the model (Collett, 2003). The equation can be solved to produce an expression for a realization ψ_k of this random variable:

$$\psi_k = \left(\log t_k - \mu - \sum_j \alpha_j z_j - \beta \sum_i w_i r_i \right) \Big/ \sigma$$

The log-likelihood for the Weibull AFT model can then be expressed in terms of ψ_k:

$$\log L = \sum_k \delta_k (\psi_k - \log \sigma) - e^{\psi_k},$$

where δ_k is a censoring indicator for the k^{th} individual (0 = assumed alive, 1 = deceased). This log-likelihood function is maximized to obtain estimates for the parameters (particularly the

weights). Let us denote the vector of unknown parameters as $\boldsymbol{\vartheta}$ with dimensions (c × 1), where c is the total number of parameters in the model:

$$\underset{(c\times 1)}{\boldsymbol{\vartheta}} = \left[\sigma \;\; \mu \;\; \{\alpha_j\} \;\; \beta \;\; \{w_i\}\right]^T$$

Then the log-likelihood is maximized and estimates of the parameters are obtained. In Lagrangian formulation, we have:

$$\hat{\boldsymbol{\vartheta}} = \arg\max_{\boldsymbol{\vartheta}} \left[\log L(\boldsymbol{\vartheta}; \boldsymbol{r}, \boldsymbol{z}) - \lambda\left(\sum_i w_i - 1\right)\right]$$

The maximization of this log-likelihood function (subject to the specified constraints) is essentially a linearly-constrained nonlinear optimization problem which was solved numerically using the Non-Linear Programming procedure (PROC NLP) in SAS 9.3 (SAS Institute, Cary, NC). The Trust Region algorithm (Moré & Sorensen, 1983; Celis, Dennis & Tapia, 1984) was used with initial values for the weights corresponding to a uniform distribution across all 24 biomarkers used in the analysis (i.e. w_i = 1/24, for all i). We also tried other sets of starting values in order to assess convergence stability. Using a Dirichlet distribution, we generated different sets of initial values for the weights: $\boldsymbol{w} \sim Dirichlet(\alpha_1, ..., \alpha_p), \;\; \alpha_1 = \alpha_2 = \cdots = \alpha_p = 1.$ For each set, the optimization was carried out and the model converged to the same final set of estimates, indicating a lack of sensitivity to initial conditions.

As discussed in the previous section, a large number of bootstrap samples are generated from the training set and the model fitting process described above is repeated on each sample to obtain estimates of the weights; these estimates are then averaged across all bootstrap samples to obtain the HSM:

$$HSM = \sum_{i=1}^{24} \bar{w}_i r_i, \quad \text{where } \bar{w}_i = \frac{1}{B}\sum_{b=1}^{B} \hat{w}_{i(b)} \quad ----(2.6)$$

2.2.4 Standardization of biomarker measurements

As discussed earlier, in the WQS technique, biomarker measurements (or any other variable type, e.g. environmental chemical exposures) are first converted into a common ordinal scale before fitting the model. We have made references to this procedure in earlier sections using the term 'standardization'. This term, as used in this thesis, is not to be confused with the familiar process of standardizing normally-distributed variables to produce a variable with a mean of 0 and a standard deviation of 1. Instead, we use the term in a more general sense, to refer to a procedure for converting a number of variables with different units of measurement onto the same scale. This process is particularly relevant to our study because the biomarkers being combined to form the HSM have a variety of units and, as such, exist on different scales. In order to combine biomarkers of varying units into one unidimensional composite index, it is useful to first convert them all to the same scale.

Another reason for using a common standardized unit for all biomarkers is that the distribution of most of the biomarkers measured in the NHANES dataset is skewed, with some extreme outliers present. These extreme outliers exert a disproportionate effect on the results and therefore to dampen their effects, standardization is used. The WQS methodology, in its original form, requires standardizing biomarker measurements (or measurements of any other variable type) by 'scoring' the range of measurements into quantiles (e.g. quartiles, quintiles). If, for example, quartiles are used, each biomarker measurement (regardless of its original unit) is assigned a standardized value of 0, 1, 2 or 3 depending on which quartile it falls into, e.g. if it falls into the 1^{st} quartile, it is assigned a value of 0, and so on. It is straightforward to see that this simple, ad hoc approach solves the two problems mentioned above, i.e. all biomarkers

(regardless of original unit of measurements) will end up on the same ordinal quantile-based scale and extreme outliers will naturally fall into one of the quantiles.

However, for our particular application, there are a number of disadvantages to using quantiles for standardization. The HSM is designed to be a risk score for which higher values indicate higher mortality risk (or poorer health status) and lower values indicate low risk. Thus the HSM in its final form should have a positive, monotonic relationship with mortality. Since the HSM is a weighted sum (with nonnegative weights) of the standardized biomarker levels, the latter have to demonstrate the same relationship with mortality, i.e. a positive monotonic relationship. Therefore for constructing the HSM, a standardization procedure is needed that converts raw biomarker levels into a standardized scale that has a positive, monotonic association with mortality. However, certain biomarkers in their original (unstandardized) scale show a different relationship with mortality, e.g. some show a negative monotonic relationship while others may have a non-monotonic, convex (U-shaped) association (see Figures 2.1a-c below). Therefore a standardization procedure is needed which can convert biomarkers that have a variety of functional relationships to mortality onto a uniform, standardized scale with a positive monotonic association with mortality. The default quantile-based standardization used in WQS regression is not appropriate for this purpose since it tends to preserve the shape of the relationship between each biomarker and mortality. To account for possible non-linear associations between certain biomarkers and mortality, we used a more guided approach to standardize the biomarker measurements.

First, each biomarker's range of measurements is transformed onto a relative hazard scale in the following way: Cox proportional hazards regression models with smoothing splines (Therneau & Grambsch, 2000) are used to plot the relationship between each biomarker's levels

and mortality (quantified as relative hazard) after adjusting for age, gender and race (see examples of these plots in Figures 2.1a-c). The equation below shows the model used for any particular biomarker x_i :

$$\ln h(t) = \ln h_0(t) + \beta_s\, sex + \beta_r\, race + \beta_a\, age + f_{x_i}(x_i) \qquad ---(2.7)$$

In the equation above, the function f_{x_i} is the smoothing spline function for biomarker x_i. This spline function has internal parameters that determine its shape, and these parameters are estimated when the model is fit. Once all parameters in the model above are estimated, one can compute the estimate of the *relative hazard,* which we are defining here as the hazard relative to the training dataset sample average. The portion of this quantity attributable to the biomarker effect can then be plotted against biomarker levels (see Figures 2.1a-c). We will refer to these plots as *partial prediction plots*, on account of their similarity to plots of the same name used in generalized additive models (e.g. see Christensen & White, 2011). The use of smoothing splines to generate these plots allows any non-linear or non-monotonic relationships that may exist between a biomarker and mortality to be modeled. These plots therefore allow the range of raw measurements (in original units) for each biomarker to be mapped onto the relative hazard scale. We divide this scale into 10 equal-sized intervals (strata), representing discrete levels of risk. The lowest stratum is assigned a value of 0 (indicating the lowest risk level) and the highest a value of 9 (highest risk level). This standardization procedure facilitates the transformation of a set of raw biomarker measurements (with a variety of units) into a uniform, ordinal scale of 0-9, with each transformation allowed to be monotonic or non-monotonic depending on the nature of the association between the corresponding biomarker and mortality.

The resulting standardized ordinal values have an intuitive appeal because, for all biomarkers, higher standardized values will indicate less desirable biomarker levels and lower values will indicate healthier biomarker levels. For example, in Figure 2.1a below, a Bicarbonate level of, say, 10 mmol/L falls into the highest relative hazard stratum and thus gets assigned a standardized value of 9, representing the highest level of risk relative to the population baseline. A higher Bicarbonate level of, say, 25 mmol/L falls into the lowest stratum of risk and thus gets assigned a standardized value of 0. But an even higher Bicarbonate level of, say, 37 mmol/L falls into the highest risk stratum and thus gets assigned a standardized value of 9. This clearly demonstrates the non-monotonic relationship between Bicarbonate level and mortality, with Bicarbonate levels that are too high or too low being 'unhealthy', and the healthy range (lowest stratum of risk) being somewhere between 21 and 30 mmol/L, according to the plot in Figure 2.1a. It is worthwhile noting that this range matches with the clinical/laboratory reference range for 'Normal' levels of serum bicarbonate, which is 23-29 mmol/L (Goldman & Schafer, 2011). Thus the lowest stratum of risk observed in our plot corresponds closely to what is clinically considered to be the 'Normal' range. We noticed this trend for some of the other biomarkers we used. We also observed that many biomarkers demonstrate the same type of 'U-shaped' relationship with mortality that Bicarbonate does, i.e. where 'medium' levels of the biomarker are associated with lower mortality risk and extremely low or high levels of the biomarker are associated with higher mortality risk.

The benefit of this new, guided approach to standardization is in allowing these non-monotonic associations to be implicitly accounted for in constructing the HSM. Note that the training dataset (NHANES 1999-2002) was used to fit the smoothing spline-based Cox proportional hazards models described above. The population in this training dataset is large and

diverse enough that the relative hazard partial prediction plots generated may be considered robust approximations of the true relationship between each biomarker's levels and mortality in the general population.

Figure 2.1a

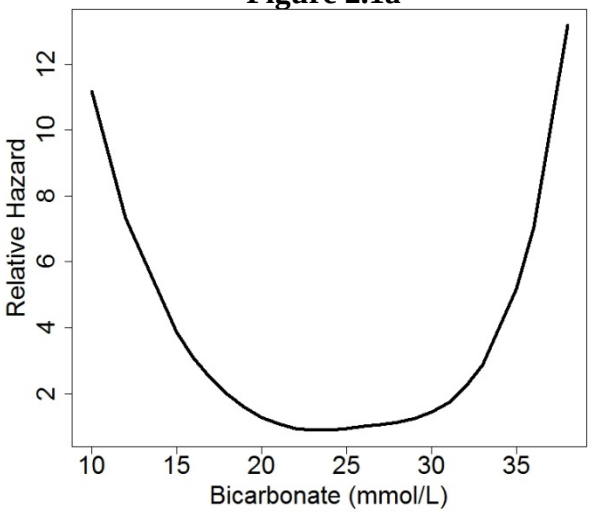


Figure 2.1b

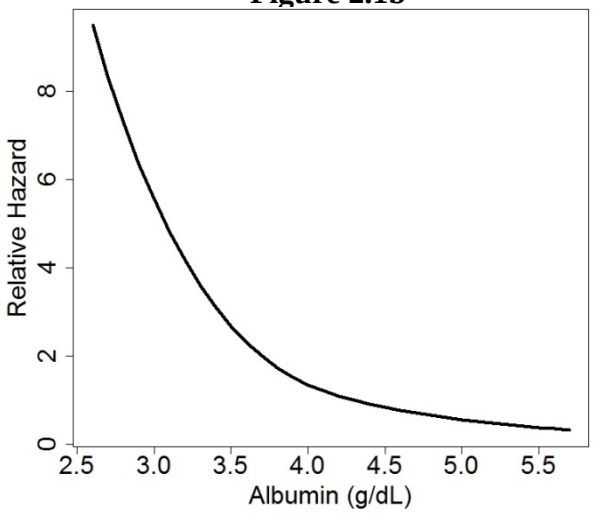


Figure 2.1c

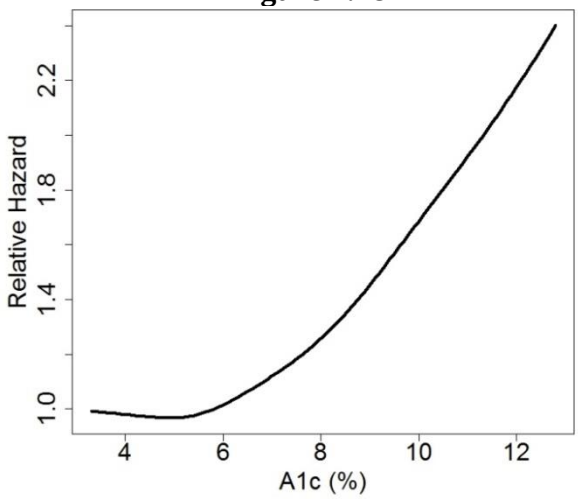


Figure 2.1a-c: Examples of relative hazard partial prediction plots used for transformation of raw biomarker measurements onto the relative hazard scale. Each plot represents the multivariate adjusted spline-smoothed partial relative hazard estimates as a function of biomarker level.

After standardization using the new approach described above, each biomarker ends up on an ordinal scale ranging from 0-9. Because the standardized biomarker values range from 0-9 and the weights add up to 1 and are constrained between 0 and 1, the HSM ends up having a range of 0-9. Thus an individual with an HSM score equal to 0 is one who falls into the lowest risk ('healthiest') stratum on all biomarker measurements, and an individual with an HSM score equal to 9 is one who falls within the highest risk stratum on all biomarkers measured. HSM scores between 0 and 9 indicate health risk levels falling between these two extremes, with higher scores indicating greater mortality risk.

2.2.5 Validation

The NHANES III dataset was used as a test/validation set to assess the predictive accuracy of the HSM composite. The population in this dataset shares no overlap with the NHANES 1999-2002 population (training set) used to generate the weights for the HSM. These weights were used to compute HSM scores for individuals in the NHANES III dataset. The standardization of the biomarker measurements for NHANES III individuals was carried out using the relative hazard partial prediction plots computed for the NHANES 1999-2002 population (examples of which are plotted in Figures 2.1a-c), rather than recomputing new relative hazard functions specifically for the NHANES III dataset. The rationale behind reusing the NHANES 1999-2002 relative hazard functions is that the eventual goal of this project is to be able to compute the HSM for individual patients without requiring any information about the distribution of biomarker measurements in the populations they belong to. As discussed earlier, due to the large sample size and diversity of the NHANES 1999-2002 dataset, the relative hazard functions computed using this population are robust estimates of the true underlying biomarker-

mortality relationships, and are thus suitable for use in the standardization of biomarker measurements of individuals in other datasets.

2.2.5.1 Assessing the HSM's predictive accuracy for all-cause mortality

To test the predictive power of the HSM on survival time in the NHANES III dataset, we will use two approaches:

I. A Weibull AFT (Accelerated Failure Time) model will be used to model the association between HSM and mortality, adjusting for the potentially confounding variables: age, gender, race, BMI, and Poverty Income Ratio. In this validation model, the statistical significance and sign of the HSM coefficient would be used as indicators of the strength and accuracy of the HSM variable as a predictor for survival time. In particular, since the HSM is constructed in such a way that higher values signify worse survival outcomes, a negative and statistically significant HSM coefficient in the validation model would imply that the HSM is a strong predictor of mortality in the test/validation dataset.

II. C-statistics: These are commonly used in clinical risk score development to test predictive discrimination. In fact, they are considered the standard tool for quantifying the predictive accuracy of risk scores designed to predict binary outcomes (Steyerberg, 2009). To test a binary classifier, Receiver Operating Characteristic (ROC) curves are typically generated to provide a graphical depiction of the variation in sensitivity and specificity of a binary classifier as the threshold settings are varied (Hosmer & Lemeshow, 2000). The area under the ROC curve (termed AUC) can be seen as a measure of the ability of a binary classifier to discriminate between 'positive' and 'negative' cases. A more interesting interpretation of the AUC is as a type of concordance index. Let *T* be a binary classifier that, for an individual *i*, assigns a score $\hat{S}_i$

based on the values of individual *i*'s explanatory variables ($\boldsymbol{x}_i$). Further, let this score be such that a higher value signifies a higher probability of being 'positive' on the binary outcome of interest, e.g. having a disease or health condition. We will henceforth refer to 'positive' individuals as cases and the 'negative' ones as controls. For any arbitrary pairing of case and control, the probability that *T* assigns a higher score to the case can be interpreted as a measure of concordance/agreement between prediction and observed response. For a particular case-control pair (denoted by *i* and *j*), define the occurrence of concordance as a random variable U_{ij} where:

$$U_{ij} = \begin{cases} 1, & \text{if } \hat{S}_i > \hat{S}_j \\ 0.5, & \text{if } \hat{S}_i = \hat{S}_j \\ 0, & \text{if } \hat{S}_i < \hat{S}_j \end{cases}$$

The probability of concordance for any arbitrary case-control pair can then be estimated from the sample as

$$\frac{1}{N_p} \sum_{i,j} U_{ij} ,$$

where N_p is the total number of all unique case-control pairs (*i,j*) in the dataset. This is known as the concordance index and it turns out to be equivalent to the AUC (Steyerberg *et al.,* 2010). Thus a useful statistical interpretation of the AUC is as a type of concordance statistic, or 'C-statistic' for short. The relationship to the rank correlation measure, Kendall's Tau, is worth noting here, as is the direct equivalence to the Mann-Whitney U-statistic.

A similar type of concordance index can be defined in situations where the predicted outcome of interest is a right-censored survival outcome. In (Harrell *et al.,* 1982), a concordance index for right-censored survival outcomes was introduced. Referred to as Harrell's C, this statistic is defined in a similar manner to the definition of the AUC given above, but the

computation is of course complicated by the presence of censored outcomes. Let *H* be a predictor for a survival outcome which, for an individual *i*, assigns a score h_i based on individual *i*'s covariates ($\boldsymbol{x}_i$). And let h_i be such that higher values signify a worse outcome/prognosis. Then for each individual or observation *i* let (T_i, δ_i, h_i) represent the observed time, binary censoring indicator (where 0 indicates censorship) and computed risk score. For a pair of individuals (*i*, *j*), define this pair as informative if it is possible to know which individual survived longer. Thus a pair (*i*, *j*) is informative if any one of the following cases is true:

- $\delta_i = \delta_j = 1$
- $\delta_i = 1,\ \delta_j = 0,\ T_i < T_j$
- $\delta_i = 0,\ \delta_j = 1,\ T_i > T_j$

The Harrell's C-statistic is then the proportion of informative pairs (*i*, *j*) exhibiting concordance between the prediction score (h_i, h_j) and the observed survival times (T_i, T_j), i.e. either ($h_i < h_j$) & ($T_i > T_j$), or ($h_i > h_j$) & ($T_i < T_j$). In words, concordance is defined as the case whereby the individual with the higher (worse) score has the shorter survival time and vice versa. Harrell's C thus represents an empirical estimate of the probability of concordance of any randomly selected pair. If we denote this probability by *c*, Harrell's concordance index can be formally expressed as:

$$\hat{c} = \frac{\sum_{i<j} I(T_i < T_j) I(h_i > h_j) I(\delta_i = 1) + I(T_j < T_i) I(h_j > h_i) I(\delta_j = 1)}{\sum_{i<j} \left[I(T_i < T_j) I(\delta_i = 1) + I(T_j < T_i) I(\delta_j = 1) \right]}$$

In the expression above, $I(\cdot)$ denotes an indicator function that evaluates to 1 when its argument is true. Additional minor modifications can be made to account for ties in the scores

and/or survival times; for details on this we refer the interested reader to Harrell *et al.* (1982) and Harrell, Lee & Mark (1996).

Both AUC and Harrell's C have a range of 0 to 1, with 0.5 indicating a predictor with no discriminative power and increasing values indicating better discriminative power. A value of 1 indicates a predictor with perfect performance. To assess the predictive power of the HSM, we will first compute Harrell's C which is a natural measure to use in this study since our outcome is a survival outcome. In order to use the AUC, we would need to dichotomize the survival outcomes to create binary outcomes. We do this by defining some clinically relevant time point for life expectancy (e.g. 5 years or 10 years) and then defining a binary outcome as follows: Let t_{bin} be the clinically-relevant time point measured in years. Then for an individual i with observed survival time T_i, a binary outcome Y_i representing life-expectancy for t_{bin} years can be defined as:

$$Y_i = \begin{cases} 1 & if\ T_i \le t_{bin} \text{ and } \delta_i = 1 \\ 0 & if\ T_i > t_{bin} \end{cases}$$

Note that Y_i is undefined when $T_i \le t_{bin}$ and $\delta_i = 0$.

As an example illustrating this definition, if the time point of interest (t_{bin}) is chosen to be 5 years, then the binary variable representing 5-year life expectancy would be defined so that those who were known to have died 5 or fewer years after their participation in NHANES would be assigned a value of 1 and those who were known to have lived past the 5-year point would be assigned a value of 0. Using this binary outcome, we can compute the AUC for the HSM. In section 2.3.1, we will discuss the special relevance of this life-expectancy binary outcome to interpreting HSM scores.

2.2.5.2 Assessing the HSM's predictive accuracy for cause-specific mortality

As mentioned earlier, the NHANES III Linked Mortality files also contain information about cause of death (stored in variable *UCOD_113*). This information was used to test the ability of the HSM to predict mortality arising from specific chronic illnesses such as cardiovascular disease (codes 053-075), liver disease (codes 093-095), kidney disease (codes 097-101) and diabetes (code 046). Logistic regression (with Firth's bias-correction (Firth, 1993) for low-prevalence outcomes) was used to test the predictive power of HSM for mortality due to each of these conditions. Age, gender, race, Poverty Income Ratio (PIR) and BMI were adjusted for.

2.2.5.3 Assessing association of HSM with indicators of concurrent health

Questionnaire data from participants in NHANES between 2003 and 2008 was used to test the relationship between HSM score and the following self-reported variables: health status, hospital utilization and diagnoses of Diabetes, heart, kidney and liver disease. Table 2.1 below summarizes the questionnaire items used. Note that the items corresponding to self-reported diagnosis of various heart conditions (Items MCQ160B-MCQ160F) were condensed into one variable indicating whether or not a respondent had been notified by their doctor of at least one of these conditions. For the questionnaire variables with binary (Yes/No) responses, logistic regression was used to model each variable's relationship with HSM whilst adjusting for age, gender, race, Poverty Income Ratio (PIR) and BMI. Analysis of the relationship between HSM and questionnaire variables with more than 2 response categories was carried out using either linear or Poisson regression (see Table 2.1 for summary) depending on which provided a better fit to the model (as determined by the Akaike Information Criterion).

Table 2.1: NHANES 2003-2008 selected questionnaire items and regression techniques used to model their relationship with HSM

Variable Name	Questionnaire Item	# of response categories	Analysis Technique
HUQ010	Self-rated health	5	Linear Regression
HUQ050	# of times healthcare received over past year	6	Poisson Regression
HUQ080	# of times over past year respondent was overnight hospital patient	6	Poisson Regression
DIQ010	Doctor ever told respondent they have Diabetes?	2 (Yes/No)	Logistic Regression
MCQ160L	Doctor ever told respondent they have liver condition?	2 (Yes/No)	Logistic Regression
KIQ020	Doctor ever told respondent they have weak/failing kidneys?	2 (Yes/No)	Logistic Regression
MCQ160B-MCQ160F	Doctor ever told respondent they have Congestive Heart Failure, Coronary Heart Disease, Angina, Heart Attack, Stroke	2 (Yes/No)	Logistic Regression

2.3 RESULTS

To estimate the weights for the HSM, we used B=1000 bootstrap samples and the computed bootstrap-averaged weight estimates $\overline{w}_i$ corresponding to each biomarker are plotted below in Figure 2.2. The plot indicates that Phosphorus has, by a significant margin, the largest weight of any biomarker while Red Blood Cell count and Sodium levels appear to be down-weighted to zero or near-zero values.

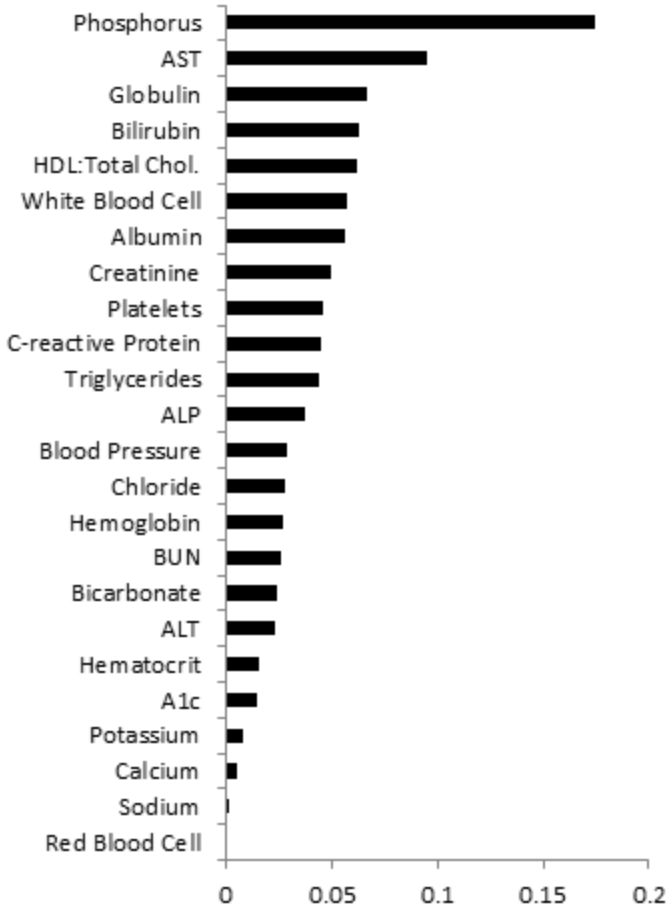


Fig. 2.2: Bootstrap-averaged weights used to construct the HSM

Figure 2.3 below shows the distribution of HSM scores in the NHANES III test/validation population. The HSM demonstrated strong predictive ability ($p < .0001$, $\hat{\beta}_{HSM}$ negative) for all-cause mortality in this validation set. Figures 2.4a-f (see **Appendix II**)

show a series of Kaplan-Meier curves (adjusted for age and gender) plotted for different HSM ranges. We have included one of the plots below (Figure 2.4e) to illustrate the effect of HSM on survival for a particular age group and gender combination. A logrank test indicates significant difference (p<.0001) in survival trends among the strata.

Fig. 2.3: Distribution of HSM in NHANES III population

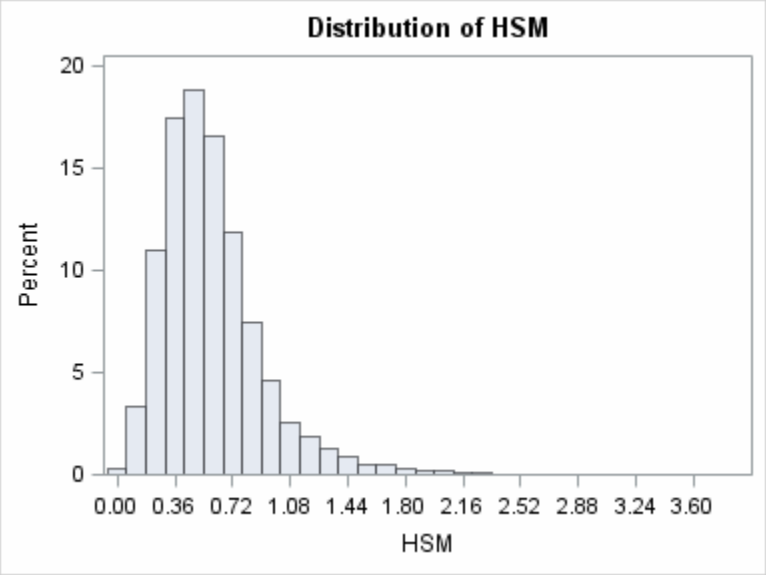
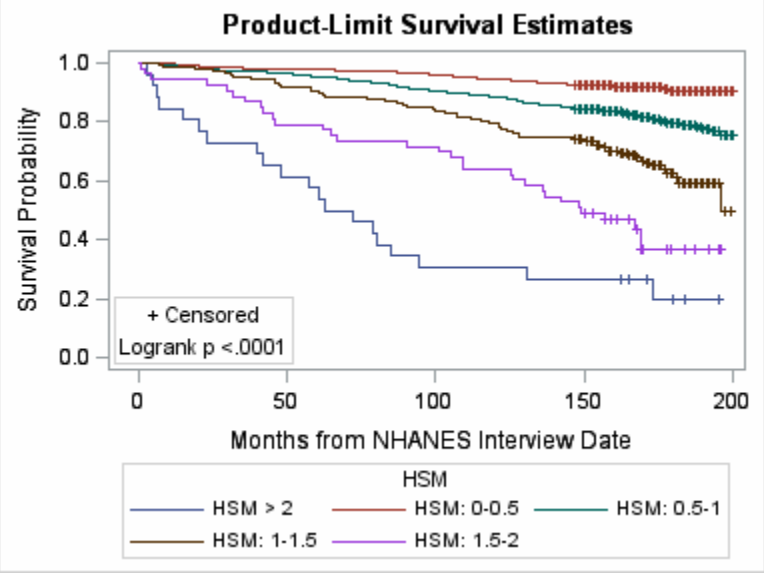


Figure 2.4e: Kaplan-Meier curves plotted for various ranges of HSM for Males in 40-64 age group



For cause-specific mortality, the HSM demonstrated high predictive validity (see Table 2.2 below). The cause-of-death analyses indicated that a 1-unit increase in HSM increases risk of death from liver disease by a factor of ~4, kidney disease by a factor of 2.3, and diabetes by a factor of 2.2.

The analysis of items in the NHANES 2003-2008 questionnaire data reveals a robust association between an individual's HSM score and their current health status as assessed by self-rated health, self-reported hospital utilization (in the months prior to NHANES participation), and the following self-reported physician-diagnosed health conditions: heart disease, liver disease, kidney disease, and diabetes (see Table 2.3 below). The results indicate that higher HSM scores are associated with lower self-rated health and more frequent hospital visits. The odds ratio estimates suggest that a 1-unit increase in an individual's HSM score is associated with a 3-fold increase in the odds of having been diagnosed with diabetes, a 2.1-fold increase in the odds of having been diagnosed with a liver condition, a 4.7-fold increase in the odds of having been diagnosed with weak/failing kidneys, and 2.2-fold increase in the odds of having been diagnosed with one or more of the following cardiovascular diseases: congestive heart failure, coronary heart disease, angina, heart attack and stroke.

These results should be interpreted with caution. It is tempting to interpret them to mean that increased HSM in any individual is indicative of elevated risk of diabetes, liver, kidney, and cardiovascular disease. However this would be an incorrect interpretation of the results since they are simply statistical associations observed at the population level. In other words, a particular individual with a relatively high HSM score may not necessarily be at elevated risk for all the aforementioned conditions. The specific conditions (if any) that an individual is at risk of due to relatively high HSM score would depend on their particular biomarker profile. The HSM

score should be seen as a predictor of general mortality, not as a predictor of particular illnesses and health conditions.

Table 2.2: Predictive Validity of HSM (as measured by p-value & Odds Ratios [covariate-adjusted]) for death caused by a variety of chronic ailments.

Cause of Death	p-value	Odds Ratio (95% CI)
Cardiovascular Disease	0.5	0.9 (0.8-1.1)
Liver Disease	<.0001	3.7 (2.3-6.0)
Kidney Disease	0.004	2.2 (1.3-3.7)
Diabetes	<.0001	2.3 (1.6-3.4)

ɔnship with self-reported hospital utiliz-ation and physician-diagnosed health conditions

Questionnaire Item	p-value	Odds Ratio (95% CI)
Self-rated health	<.0001	N/A
# of times healthcare received over past year	<.0001	N/A
# of times over past year respondent was overnight hospital patient	0.003	N/A
Doctor ever told respondent they have Diabetes?	<.0001	3.0 (2.3-4)
Doctor ever told respondent they have liver condition?	<.0001	2.1(1.5-3)
Doctor ever told respondent they have weak/failing kidneys?	<.0001	4.7(3.2-7)
Doctor ever told respondent they have congestive heart failure, coronary heart disease, Angina, heart attack, or stroke	<.0001	2.2(1.6-3)

For all-cause mortality, we assessed the predictive power of the HSM using the measures (Harrell's C and AUC) described in sub-section 2.2.5.1. The results are summarized in Table 2.4 below.

Table 2.4: Predictive accuracy of HSM

Measure	Estimate
Harrell's C	0.7
$AUC_{1\text{-year}}$	0.78
$AUC_{5\text{-year}}$	0.74

As the results indicate, the HSM exhibits reasonable predictive power for all-cause mortality in the validation dataset.

2.3.1 Interpretation of HSM Score

HSM scores can be directly translated into projected mortality risk at certain time points in the future. The plots in Figures 2.5a-b below illustrate the relationship between HSM score and probability of mortality 5 years and 10 years after HSM score determination. These plots are adjusted for age group and gender, so they can be used to determine an individual's age- and gender-adjusted 5- and 10-year life expectancy based on their present HSM score. Mortality risk at alternate time points can also be easily computed for specific HSM scores.

Figure 2.5a: Age- and Gender-adjusted Relationship between HSM and Predicted 5-year mortality risk

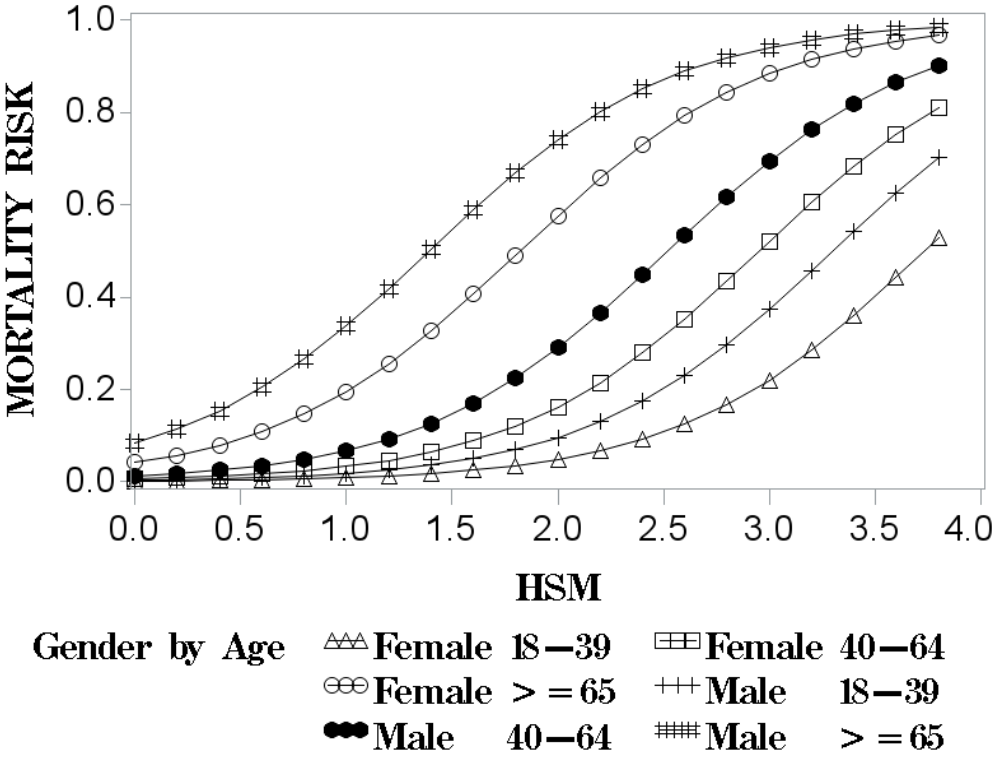
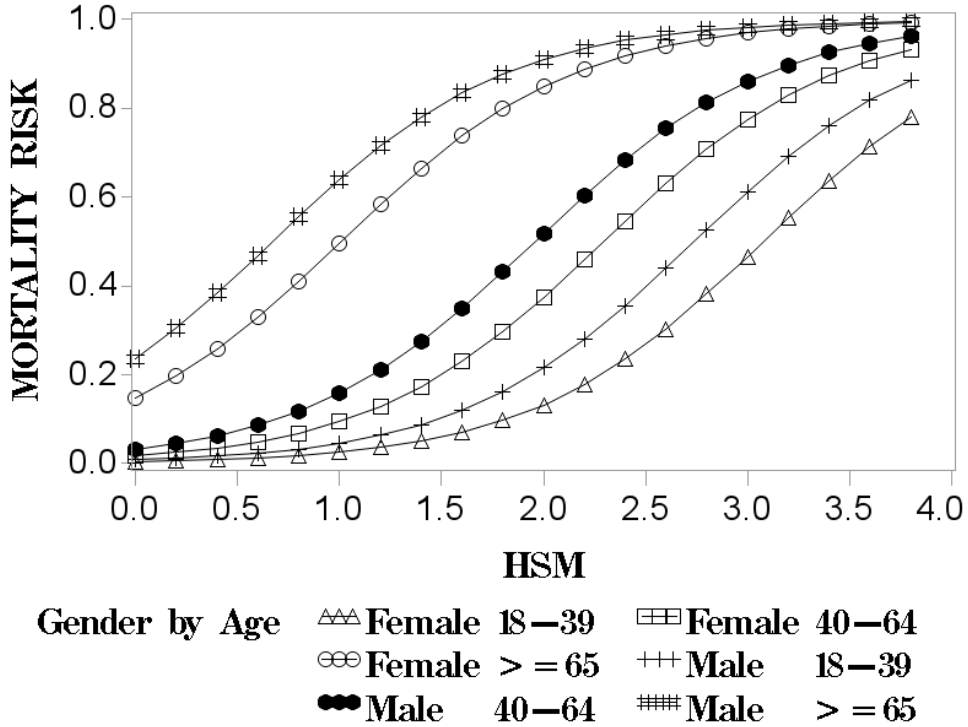


Figure 2.5b: Age- and Gender-adjusted Relationship between HSM and Predicted 10-year mortality risk



2.4 CONCLUSION AND DISCUSSION

Using biomarker and survival data, we have developed and validated a composite score which serves a dual purpose as a fairly comprehensive measure of overall health and a prognostic tool capable of predicting mortality risk for the general population.

To construct the HSM, we used Weighted Quantile Sum (WQS) regression. A key step in this technique is the standardization of variables to be combined into an index. We extended the WQS methodology by introducing a new, guided approach for standardizing variables. The original method of standardization was an ad hoc approach that involved scoring each variable into pre-defined quantiles. This unguided approach does not account for possible non-monotonic or non-linear relationships between biomarker levels and mortality. To address this issue, we introduced a novel standardization technique that allows standardized values to reflect any non-monotonic relationships that may exist between biomarker measurement levels and mortality.

Validation analysis of the HSM demonstrated that it is both a reasonably accurate gauge of current health status and a reliable predictor of life expectancy. Higher HSM scores tend to be linked with lower self-rated health, higher frequency of hospitalization, higher likelihood of chronic health conditions (in the present and in the future) and decreased life expectancy.

Nearly all the biomarkers used in constructing the index can be obtained from common laboratory tests (Comprehensive Metabolic Panel, Lipid Panel, Complete Blood Count) performed on patients as part of the diagnosis process or routine checkups. The HSM provides a straightforward way to combine all these markers of various aspects of health into a single score that serves as a numerical estimate of current overall health and future mortality risk.

This makes it potentially useful as a tool for prediction of general risk (with mortality as an endpoint). The HSM would provide clinicians who use it with an evidence-based, data-driven

assessment of general mortality risk that could aid decision-making. And unlike some risk scores which predict mortality only for individuals with a particular disease, the HSM is a general-purpose risk score that could be used to predict mortality for individuals with a wide range of conditions.

The HSM could also be used as a measure for tracking a patient's general health over time. Certain longitudinal clinical studies that follow overall health status over time may benefit from the use of a validated, general-purpose risk score like the HSM.

In healthcare quality assessment studies, the HSM could be adapted for use as a metric for comparing patient overall health among different health care providers. As an example of such an application, the HSM could be used to obtain estimates of age- and gender-adjusted 5-year life expectancy for patients seen by individual or institutional health care providers. These estimates provide a way to make standardized comparisons of patient health outcomes among multiple healthcare providers.

2.4.1 Study Limitations

A limitation of the current method of computing the HSM is the reliance on a large number of biomarkers (24). While these biomarkers are routinely measured in clinical settings, individual patient health records might be missing one or more components. Techniques for solving this problem will be discussed in Chapter 4.

While the HSM incorporates a wide range of biomarkers spanning multiple organ systems, this range is by no means exhaustive. Certain aspects of physiological health (e.g. reproductive health, gastrointestinal health, endocrine function) are not evaluated in a direct manner by the HSM.

Chapter 3

Extending the HSM to accommodate interaction effects

3.1 Introduction

The HSM is constructed based on an additive assumption that the effect of the biomarkers on mortality can be adequately modeled without accounting for interactions that may exist among biomarkers. In this chapter, the validity of this assumption will be examined. We will develop a version of the HSM that includes interaction effects among biomarkers. This will be accomplished by making a simple modification to the WQS regression technique that allows for inclusion of between-component interactions and computation of the corresponding weights. A small subset of all the possible pairwise interactions among the 24 biomarkers will be chosen for inclusion in the extended HSM. The selection of the candidate interactions will be based on strength of association with mortality, and will be carried out using Random Survival Forests. After developing the extended HSM, we will perform tests to compare it to the original HSM (which does not account for interactions).

3.2 Extending the HSM

The HSM in its original form is a weighted linear index of biomarkers given by: $\sum_i w_i x_i$. We now consider incorporating pairwise interaction effects among the biomarkers. Higher order interactions will not be considered at this time. An extended version of the HSM that allows for inclusion of interaction effects is given below:

$$HSM_{\text{inter}} = \sum_i w_i x_i + \sum_{(j,k) \in H} w_{jk} x'_j x'_k \qquad ---- (3.1)$$

In the above equation, the HSM is divided into an additive component and an interaction component. In the interaction component, *H* is the set of all pairwise interactions under consideration. Each pairwise interaction effect has one corresponding weight (given by w_{jk} above). The x_j' and x_k' are scaled versions of standardized biomarker values in each pairwise interaction. The scaling is done so that the extended HSM (denoted $\text{HSM}_{\text{inter}}$) has the same range as the original (additive) HSM.

To estimate weights for the extended HSM in equation (3.1), we propose the following simple modification to the original form of the WQS technique used to derive the HSM in Chapter 2:

$$\log T = \mu + \sum_l \alpha_l z_l + \beta_1 \left(\sum_i w_i x_i + \sum_{(j,k) \in H} w_{jk} x_j' x_k' \right) + \sigma \Psi \quad ----(3.2)$$

This formulation incorporates the interaction component into the index portion of the model. Constraints on the weight parameters are given by:

$$\sum_i w_i + \sum_{(j,k) \in H} w_{jk} = 1, \quad 0 \le w_i \le 1, \quad 0 \le w_{j,k} \le 1$$

3.3 Selecting Interaction effects

As mentioned earlier, the term *H* in Equations (3.1) & (3.2) above is the set of all pairwise interactions under consideration. Without *a priori* knowledge of what interactions are important to include, all possible pairwise interactions would have to be included in the HSM and the WQS model used to derive its weights. For 24 biomarkers, the number of unique pairwise interactions is $\binom{24}{2}$ = 276. Combined with the demographic variables $\{z_l\}$, the total number of variables in the WQS model defined in equation (3.2) would sum up to over 300. It is natural to ask whether

the WQS technique is capable of handling such a large model. Studies utilizing this technique in the past have focused on small-to-moderate sized models (e.g. on the order of dozens of variables) and the technique has not been applied to 'large *p*' data such as the type encountered in microarray analysis, for example. WQS-based models are fit using nonlinear optimization routines such as Trust Region, Newton-Raphson Ridge Optimization, Conjugate Gradient, etc. These are iterative techniques that use quadratic approximations of nonlinear objective functions and require repeated computation of first-order and (sometimes) second-order partial derivatives. Most cannot handle optimizations with a large number of variables (e.g. on the order of 100s or 1000s). Out of all the optimization techniques available in the SAS-based NLP procedure (SAS Institute, Cary NC) we used in our studies, the Conjugate Gradient technique is the only one capable of handling very large optimization problems. However our attempts to use this optimization technique for large models with hundreds of variables have so far been confounded by problems with unstable convergence. Therefore including all possible pairwise interactions in the WQS model in Equation (3.2) was not a feasible option.

One solution is to use suitable alternative techniques to search the space of all possible pairwise interactions to identify a select few important interactions that can be included in our WQS model. The problem of detecting important interaction effects (especially bivariate interactions) in high-dimensional data is one that has been studied in multiple fields, particular in Genomics. For example, an important aspect of genome-wide association studies (GWAS) is the detection of gene-gene interactions that may manifest as statistically detectable effects on one or more outcomes. In such studies, the simple approach of testing for individual pairwise interactions (i.e. marginal tests for association) is often used, however more recent studies have

proposed advanced techniques for identifying interactions in a more holistic fashion (see Cordell (2009) and Wu *et al.* (2010)).

One particularly promising approach is the use of Random Forests (Breiman, 2001), a popular machine learning technique commonly used for prediction. Random Forests may be seen as an extension to Classification and Regression Trees [CART] (Breiman *et al.*, 1984) in that they combine a large number of 'randomized' decision trees to form an ensemble predictor (i.e. a forest) that typically has greater predictive accuracy than its individual components. Random forests have several desirable properties that have made them a popular tool particularly in areas requiring the analysis of high-dimensional data. Random Forests remain relatively robust for problems with a large number of variables (which in some cases can exceed the number of samples) and their underlying tree-based structure provides a completely nonparametric approach for adequately modeling complex interactions and correlations that may exist among variables in high-dimensional data. Also, the strategic use of sampling/resampling and randomization in tree construction produces forests with low generalization/prediction error (Breiman, 2001).

A particularly appealing feature of Random Forests (RF) is the ability to adaptively discover interaction effects among several variables. This makes it an ideal technique for the problem of selecting the most significant interaction effects of a large set of such potential effects. It has been applied in a number of genomic studies as a tool for exploring and detecting gene-gene and epistatic interactions (Jiang *et al.*, 2009; Liu, Ackerman & Carulli, 2011; Winham *et al.*, 2012; Pan *et al.*, 2013; Staiano, Di Taranto & Bloise, 2013).

In the present study, RF methodology will be used to nonparametrically model the complex interaction networks among the biomarker variables and to identify a select few for inclusion in our WQS model.

Our WQS model (see Equation (3.2)) uses a censored survival response (observed survival time and censoring indicator), however, the original and commonly-used implementations of Random Forests only handle binary and continuous (as for regression) responses. Binary responses are predicted by building a forest from classification trees while continuous responses require regression trees. Since the introduction of the Random Forests algorithm for classification and regression, multiple attempts have been made to adapt this methodology for censored survival outcomes, beginning with Breiman (2002) who outlined instructions for a software to handle survival data; however no formal description of its underlying algorithm was published. Different *ad hoc* techniques have since then been proposed. Most of these are off-the-shelf techniques which avoid dealing directly with the survival times but instead transform them into a variable type that could be handled by conventional Random Forests algorithms. For example, Hothorn *et al.* (2006) introduced a technique in which the survival times are first log-transformed then analyzed using regression trees weighted to account for censoring. However Ishwaran *et al.* (2008) introduced, to our knowledge, the first full extension of Random Forests methodology to time-to-event data. This implementation (termed *Random Surival Forests*) deals directly with the survival outcomes and does not require transforming them. We chose to use this implementation in our study. Random Survival Forests (RSFs) share most of the desirable features of Random Forests and have been applied in multiple clinical and genomics studies utilizing survival data of varying dimensionality (Chen, Wang & Ishwaran, 2010; Hsich *et al.*, 2010; Rice *et al.*, 2010; Rizk *et al.*, 2010; Gorodeski *et al.*, 2011; Chen & Ishwaran, 2013).

3.4 Random Survival Forest methodology

The basic unit (base learner) of an RSF is a random survival tree. These are identical in structure to the randomized classification/regression trees used in Random Forests. Just like classification/regression trees, survival trees are grown by recursively partitioning data into subsets of increasing homogeneity. In fact, the core algorithms used to construct random survival forests and random forests are quite similar (Ishwaran *et al.*, 2008). The RSF algorithm is outlined below:

1. Draw *B* bootstrap samples from the learning set. Each bootstrap sample will almost always contain only a subset of the unique datapoints in the learning set (~63.2% on average), the other unselected ~36.8% is referred to as *out-of-bag* (OOB) data.
2. For each bootstrap sample, beginning with the root node which contains all the sample data, grow a randomized survival tree by using the following procedure: out of the *p* total variables in the data, randomly select a candidate subset of size *m*; split the node using the candidate variable that maximizes survival difference between the two resulting nodes (termed daughter nodes). Survival difference is measured using the log-rank test, therefore the node is split using the variable that maximizes the log-rank statistic (among all *m* candidate variables).
3. Recursively carry out Step 2 on every new node created, thereby growing a tree structure. A node can no longer be split when doing so will create at least one daughter node with less than d_o unique deaths, where d_o (>0) is a preset parameter representing the minimum allowable number of deaths in each node. Continue growing the tree until no nodes exist which can be split. The unsplit nodes are referred to as terminal nodes. Unlike CART, no pruning is performed on trees once they are complete.

4. To assign predictions to each terminal node *h* in a survival tree, the failure times and censoring status of all individuals in the node are used to construct a nonparametric estimate of the cumulative hazard function (CHF) via the Nelson-Aalen estimator. For a terminal node *h*, let *N*(*h*) be the number of distinct event times. Denote these by $t_{1,h} < t_{2,h} < \cdots < t_{N(h),h}$. Let $d_{l,h}$ and $Y_{l,h}$ respectively denote the numbers of deaths and individuals at risk at time $t_{l,h}$. Then the Nelson Aalen-estimator of the CHF for node *h* is given by:

$$\hat{H}_h(t) = \sum_{t_{l,h} \le t} \frac{d_{l,h}}{Y_{l,h}}$$

All individuals in *h* are assigned the same CHF. Note that if a new test case/individual (with known covariates $\boldsymbol{x}_i$) is dropped down a tree T_b (i.e. a tree grown from the b^{th} bootstrap sample) and allowed to propagate from node to node based on the splitting rules of each node, they will end up at a particular terminal node *h'* and the CHF prediction for such an individual would thus be $H(t|\boldsymbol{x}_i) = \hat{H}_{h'}(t)$. This defines the CHF for the particular tree T_b.

5. Average the CHF for each tree to obtain the ensemble CHF
6. For the OOB data, calculate the prediction error for the ensemble CHF. For RSFs, the prediction error is computed as 1-C where C is Harrell's concordance index, an extension of the AUC (area under the ROC curve) concept used for binary classifiers to survival data. As discussed in Chapter 2, Harrell's C provides a measure of the concordance between predicted survival and observed survival time.

3.4.1 Variable Importance Measures

Variable selection using random survival forests relies on the use of Variable Importance (VIMP) measures to rank variables by order of their "importance" in the model. The "importance" in this context is the contribution of each variable to the predictive power of the random forest predictor. An important and influential variable is one whose exclusion from the training data has a relatively large deleterious effect on prediction accuracy.

In formal notation, variable importance is defined as follows (Ishwaran 2007; Ishwaran *et al.*, 2008):

$$VIMP_v = PE_v - PE \quad ----(3.3)$$

The above equation defines the VIMP for a variable *v* as the difference between the prediction error obtained when *v* is "*noised up*" (given by PE_v above) versus the prediction error (*PE*) otherwise. Noising up a variable *v* in a random survival forest involves the use of a randomization procedure (see Ishwaran (2007)) that has the effect of lessening or 'dampening' the contribution of *v* to the forest's prediction. To 'noise up' a variable *v* in the forest, in Step 6 of the RSF algorithm outlined above, for each OOB case dropped through trees, at each node split by variable *v*, rather than follow the splitting rule defined for the node, the left or right daughter node is chosen at random (with equal probability) and this process is continued for every subsequent node encountered downstream. This process is called *random daughter assignment* and when carried out for variable *v* in every tree in the forest, it effectively 'scrambles' any relationship that variable *v* has with the response and the resulting forest's predictions are no longer influenced by *v*'s contribution. Therefore if *v* is actually an important and predictive variable, the new random survival forest produced when *v* is noised up would be expected to have a higher prediction error than the former; thus its VIMP would be a large

positive number. Conversely, 'noising up' a non-influential variable should not adversely affect the prediction accuracy of the forest. In fact in some cases, 'noising up' a variable actually slightly improves the forests predictive accuracy.

The VIMPs for all variables in a large model can be computed in the manner described above and ranked in descending order of importance. The resulting ranked list provides valuable insights as to which variables are the most important and facilitates variable selection.

A measure of association between two variables *v* and *w* can also be created using a similar process. The association measure is defined below (Ishwaran, 2007):

$$A_{v,w} = \Delta_{v,w} - (\Delta_v + \Delta_w) \qquad ----- \ (3.4)$$

As the equation above implies, the association measure (given by $A_{v,w}$ in the above equation) for a pair of variables (*v*,*w*) is the difference between the prediction error obtained when *v* and *w* are *jointly* noised up ($\Delta_{v,w}$ [referred to as the *paired VIMP*]) versus the sum of the prediction errors obtained from separately noising up each of the variables ($\Delta_v + \Delta_w$). An association measure ($A_{v,w}$) that is close to zero indicates that the two variables *v* and *w* have a weak or non-existent interaction effect. In other words the prediction error arising from *jointly* noising up variables *v* and *w* can be closely approximated by summing the prediction errors arising from individually noising up each variable separately. In contrast, an association measure that deviates significantly from zero (in either direction) is evidence of an interaction effect between the variables involved. The association measure can be used to select interaction effects that are important for prediction. The random survival forest technique is able to handle a large number of variables and interactions among those variables are naturally and intuitively modelled by the tree structure. Association measures can be computed for all unique pairs of variables. Ranking

the interactions in descending order of the absolute value of association measures will provide insight as to which interactions are more important than others.

The VIMP (univariate or joint) in its commonly-used form does not have a statistical threshold that can be used as a cutoff for variable selection. This is because the way the VIMP is defined makes it difficult to find a closed form null distribution for it. A few attempts have been made to find a rigorously-defined statistical threshold of significance for VIMP measures (van der Laan, 2006; Molinaro *et al.*, 2011) but the methods proposed have either been too computationally intensive or difficult to implement.

Studies that have used VIMP measures for variable selection usually adopt an *ad hoc* rule to define the selection threshold. For example some select the top *k* ranked variables, where *k* depends on the total number of variables but it usually relatively small in comparison (see Winham *et al.* (2012)). A few studies that have used RSF methodology have defined the threshold of significance as 5% of the maximum VIMP (Rice *et al.*, 2010; Rizk *et al.*, 2010). Therefore if a *standardized VIMP* is defined by dividing all VIMPs by the maximum observed VIMP, the resulting standardized VIMPs will range from 0 to 1 across all variables and any variable with a *standardized VIMP* above 0.05 is chosen as 'important'. We adopted this rule in our studies and defined a variable as important if its univariate standardized VIMP exceeded the 0.05. On the other hand, for the joint (interaction) VIMP, there are no set-in-stone rules and in our studies we used the top 10% of joint/interaction VIMPs as the importance threshold.

3.4.2 Minimal Depth

Recently, a new measure of variable importance called *minimal depth* has been proposed (Ishwaran *et al.*, 2010a). Its main advantage over the traditional VIMP discussed above is that a closed form distribution can be obtained for it. This enables the derivation of a statistically rigorous threshold that can be used for variable selection.

The main idea behind the minimal depth is that important (highly predictive) variables tend to be used to split tree nodes earlier in the tree construction process, i.e. they tend to be closer to, or at the top of the tree (root node) (Strobl *et al.*, 2007). Less predictive variables tend to be used for splitting nodes later in the tree construction process (i.e. they are lower in the tree structure) or never used at all.

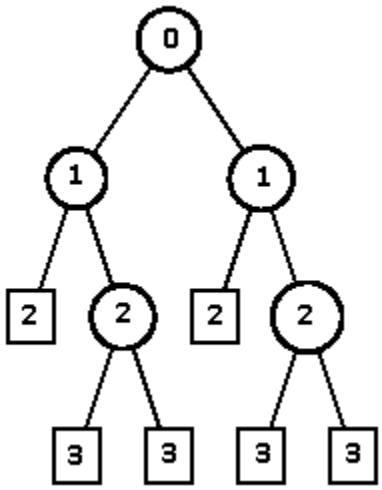


Figure 3.1: *Simple tree structure illustrating the concept of depth. The number within each node represents its depth within the tree.* [Image adapted from Ishwaran (2010a)]

Developing this idea further requires defining an important notion, the notion of 'depth'. The depth at a particular point in a tree is the number of levels/degrees of separation between the root node and that point. The root node itself is designated a depth of 0, each of its two daughters are designated a depth of 1, any daughters of those are designated a depth of 2, and so on. Figure

3.1 above illustrates this concept for a sample tree. The numbers within each node represent the depth of that node in the tree. In the figure, the square-shaped nodes are terminal nodes while non-terminal nodes are circular.

In a tree T, the depth can therefore be an integer ranging from 0 to $D(T)$, where $D(T)$ is the maximum 'distance' between the root node and the most remote terminal node, i.e. the terminal node with the largest number of degrees of separation from the root node. In Figure 3.1 above, the depicted tree has a maximum depth $D(T)$ of 3 since the remotest terminal nodes are all 3 degrees of separation from the root node.

Note that the term 'depth' can apply either to nodes in a tree or the variables used to split those nodes. We can see now that important variables which tend to be higher up in the tree structure will have smaller values for depth. The *minimal depth* for a variable in a tree is therefore defined as the depth of the node in which the variable was first used for splitting in the tree. For example, if a variable is used to split the root node of a tree then its minimal depth in that tree would be 0 since its first use during tree construction is in splitting the root node. Figure 3.2 below provides a visual illustration of this idea using a number of example trees.

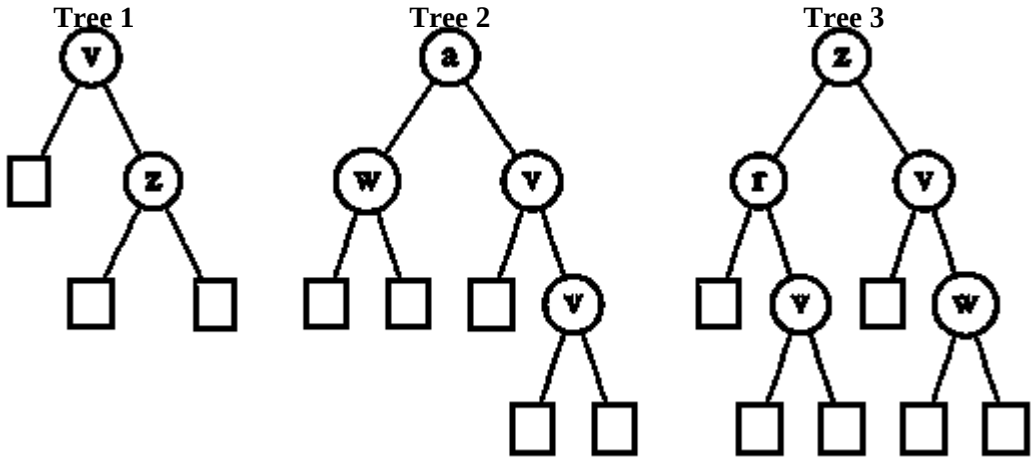


Figure 3.2: *Sample trees illustrating the concept of minimal depth. Each non-terminal node is labelled by the variable used to split it. The minimal depth for each variable is the first node in the tree (counting from the root node) that is split using that variable*
[Images adapted from Ishwaran (2010a)]

In the figure above, the letters within the circular (non-terminal) nodes represent the variables used to split those nodes. The terminal (square) nodes are unlabeled since, by definition, they are not split by any variables. In Tree 1 of Figure 3.2, the minimal depth of variables *v* and *z* would be 0 and 1 respectively. In Tree 2, the minimal depth for variables *a* and *w* would be 0 and 1 respectively, and for variable *v* it will be 1. Notice that even though variable *v* is used **twice** in Tree 2, the first time it is used (i.e. the highest node in which it's used) is the node at depth 1, thus the minimal depth of *v* will be 1. And for Tree 3, the minimal depths of variables *z, r, v* and *w* would be 0, 1, 1 and 2 respectively. Again, notice that in Tree 3, the variable *v* is used twice, however its minimal depth is defined by the depth of the node it was first used to split.

For a tree *T*, the minimal depth D_v for any particular variable *v* would be a random nonnegative integer in the range 0 to $D(T)$. If *v* was not used to split any nodes in tree *T* then, by convention, its minimal depth is set to $D(T)$, the depth of the most remote terminal node, i.e. the

one with the largest number of degrees of separation from the root node. The value of D_v in any particular tree will depend on how high up in the tree structure v was first used to split a node and the expected minimal depth of an important/predictive variable will be smaller than that of a much less predictive one. In general, if $D_v = d$ then it means that the first use of variable v for splitting in the tree was for a node at depth d.

An approximation to the distribution of the random variable D_v was derived in Ishwaran *et al.* (2010a) and is given by:

$$P\{D_v = d \mid v \text{ is a weak variable}\} \approx \left(1-\frac{1}{p}\right)^{l(d)-1}\left[1-\left(1-\frac{1}{p}\right)^{l(d)}\right] \qquad ----(3.5)$$

In Equation (3.5) above, p is the total number of variables in the model and $l(d)$ is the number of nodes existing at depth d. Note that the distribution of D_v is conditioned on v being a weak/non-predictive variable. Therefore the mean of D_v can be used as a threshold for partitioning the set of variables into strongly predictive and weakly predictive, thus it can be used for variable selection. If the minimal depth of each variable for each tree in the forest is averaged across all trees, we get the forest-averaged minimal depth for each variable. Variables with forest-averaged minimal depth falling above the threshold (the mean of the above distribution) are considered weak variables. Variables with forest-averaged minimal depth falling below this threshold would be considered important and the farther below the threshold a variable's minimal depth falls, the more predictive it is considered; recall that when it comes to minimal depth, *lower is better*, the closer a variable's minimal depth is to 0 (the depth of the root node), the more influential it is.

The notion of minimal depth can also be applied to the identification of interactions. To do so, consider a variable *v* with a minimal depth of *d* in a tree *T*. This means that the first node in *T* that was split using *v* was located at a depth *d*. Let us denote this particular node as the *index node* for *v* and denote the subtree formed by splitting this node (and the resulting subsequent daughter nodes) as the *index subtree* of *v*. Therefore, the index node of *v* is the root node of the index subtree of *v*. Figure 3.3 below illustrates this idea more clearly.

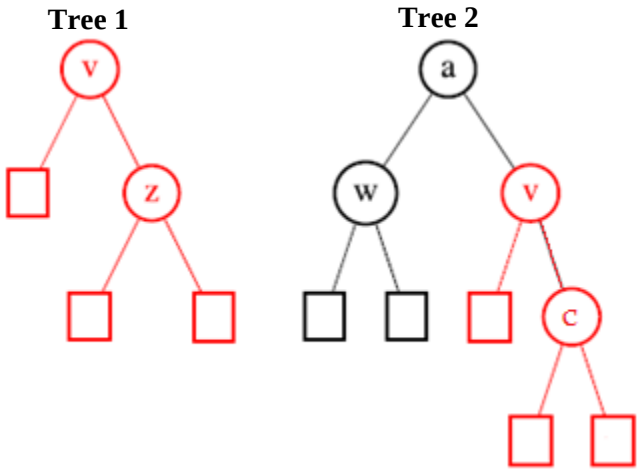


Figure 3.3: *Sample trees illustrating the concepts of 'index node' and 'index subtree'. The portions of each tree colored in red are the index subtrees of variable v, and the root node of each index subtree is defined as the index node of v.*
[Images adapted from Ishwaran (2010a)]

In the two trees depicted in Figure 3.3, the index subtree of variable *v* is depicted in red. In Tree 2, *v* is used for the first time in a node that exists at a depth of 1, therefore this node is the index node of *v* (and the minimal depth of *v* is 1) and the portion of the tree highlighted in red is the index subtree of *v*. In Tree 1, since *v* is used to split the root node, its minimal depth is 0, its index node is the root node, and the ***entire*** tree is an index subtree of *v*.

Within the index subtree of *v*, the variables that are closest to its root node (i.e. index node of *v*) can be regarded as having potential interactions with *v*. The reasoning behind this has been articulated in different forms in Breiman (2003), Bureau *et al.* (2005) and Winham *et al.* (2012).

The fundamental idea is that when variables tend to occur close together in many trees in a forest, they can be thought of as having potential interactions. In particular, commonly co-occurring ***pairs*** of variables can arise when splitting a node using one of the variables makes a subsequent split with the second variable more likely. This idea can be used to identify pairs of variables that may potentially interact. Therefore for a pair of variables (v,w), if w has a low forest-averaged minimal depth within the index subtree of v (or vice versa), the two can be considered as interacting. Unlike the univariate case, there is no threshold for defining exactly how low the minimal depth of one variable (within the index subtree of the other) has to be for both variables to be identified as having an interactive effect. We used an *ad hoc* rule that is defined in the next section.

As mentioned earlier, minimal-depth thresholding has the advantage of being based on a rigorously-defined statistical threshold (at least for the univariate case). Also, unlike the VIMP, defining minimal depth does not require the use of a measure of prediction error. Recall that the VIMP uses 1-C (where C is Harrell's C-index) as the prediction error, however other measures of prediction error can be used (e.g. Brier score (Gerds & Schumacher, 2006)); the choice of prediction error used in computing VIMP has been shown to influence which variables are deemed important, and this is not an ideal property for a variable selection method to have. This problem can be circumvented by using minimal depth thresholding which is not tied to any specific measure of error but is instead based on a simple order statistic defined solely using tree structure.

The main downside of minimal depth-thresholding is that it is not as well-established or well-vetted as the VIMP. It has only been used on experimental/simulation datasets and applied in a small number of studies (see Ishwaran *et al.* (2011), Chen & Ishwaran (2012, 2013)).

However it has been shown in these studies to outperform the VIMP. Therefore we will use both methods to find interactions and develop/test two separate models based on the sets of interaction effects found through both methods.

3.5 Implementation of RSF algorithm

In our studies, we used the package **RF-SRC** [Random Forests for Survival, Regression and Classification] developed by Ishwaran & Kogalur (2013) and implemented in the *R* programming language (R Development Core Team, 2010). To build the RSF, we used the `rfsrc` function of this package. The forests were constructed using 500 trees and the parameter *m* (see Step 2 of the RSF algorithm outlined in Section 3.4) was set to the default value of `ceiling`($\sqrt{p}$), where *p* is the total number of explanatory/input variables in our model. As mentioned earlier, the splitting rule used was the log-rank statistic.

The learning set used in the algorithm was the NHANES 1999-2002 dataset we have discussed in earlier chapters. We used a total of 31 variables: 2 response/outcome variables (survival time, censoring indicator), and 29 input/explanatory variables. The latter consisted of 5 demographic/body measure variables (age, gender, race, PIR and BMI) and 24 biomarker variables. The standardized versions of the biomarkers were used in the RSF algorithm because this is also how they appear in our WQS model. The output of the `rfsrc` function is an RSF 'object', a type of R data structure which stores information about features of the forest and each individual tree used to construct it.

After constructing the survival forests, we computed interactions among all pairs of variables ($\binom{29}{2}$ = 406 in total) using the `find.interaction` function in this package. This

function calculates the univariate VIMP for each variable and also the association measures for all specified pairs of variables. In addition to the VIMP measure, the `find.interaction` function produces a $p \times p$ matrix whose diagonal elements are the normalized minimal depths for each variable, i.e. entry [i][i] corresponds to the normalized minimal depth of variable [i] relative to the root node (normalized with respect to the size of the tree). Entries [i][j] correspond to the normalized minimal depth of a variable [j] with respect to the index subtree of variable [i] (normalized with respect to the size of [i]'s index subtree). The instructions for the function stipulate that the correct way to read and interpret the matrix is by scanning each row (i = 1 to p) for small entries. Small [i][i] entries also having small [i][j] entries are a sign of interaction between variables [i] and [j]. The explanation for this is as follows: recall that small [i][i] entries imply that variable [i] is closer to the root node and thus more important. Small [i][j] entries indicate that variables [i] and [j] tend to occur together more frequently than would be expected by chance, indicating that a potential interaction may exist between them. Therefore in selecting small [i][i] entries also having small [i][j] entries, we are limiting our search for interacting variables to those with strong marginal effects.

3.6 RESULTS

The average OOB prediction error rate of the constructed random survival forest was 17.28%. The figure below shows the convergence of the error rate towards a stable value over the 500 trees used to construct the forest.

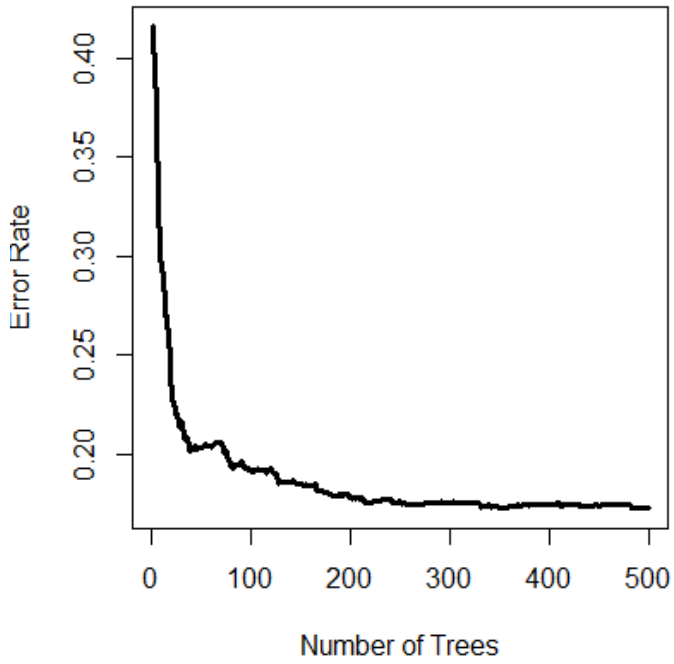


Figure 3.4: *Convergence of the error rate to a stable value over the 500 trees used to construct the Random Survival Forest. Details on the construction of the RSF are given in Section 3.5*

3.6.1 Variable Importance (VIMP)

The plot in Figure 3.5 below shows the Standardized VIMP for each variable used in the RSF. As discussed earlier, the standardized VIMP was computed by dividing the VIMP for each variable by the maximum VIMP (0.0779 [age]). We see that age is the strongest predictor, followed by creatinine, globulin, BUN (Blood Urea Nitrogen), etc. That age is the strongest predictor is not unanticipated especially for survival outcomes. Notice that 7 of the biomarkers and the demographic variable race have values of zero. Some of these values are actually

negative. Negative VIMP is possible and occurs when 'noising up' a variable in a forest actually causes the forest's prediction error to improve (usually only slightly). Zero and negative VIMP values indicate a variable is not predictive. Thus for ease of visualization, we set the negative VIMPs to zero in the plot in Figure 3.5.

The dashed red line in Figure 3.5 below indicates the 5% importance threshold discussed above. The following biomarkers fell above this threshold: C-reactive protein, ALP, Chloride, AST (Aspartate Aminotransferase), A1c, Platelet count, Phosphorus, BUN, Globulin and Creatinine.

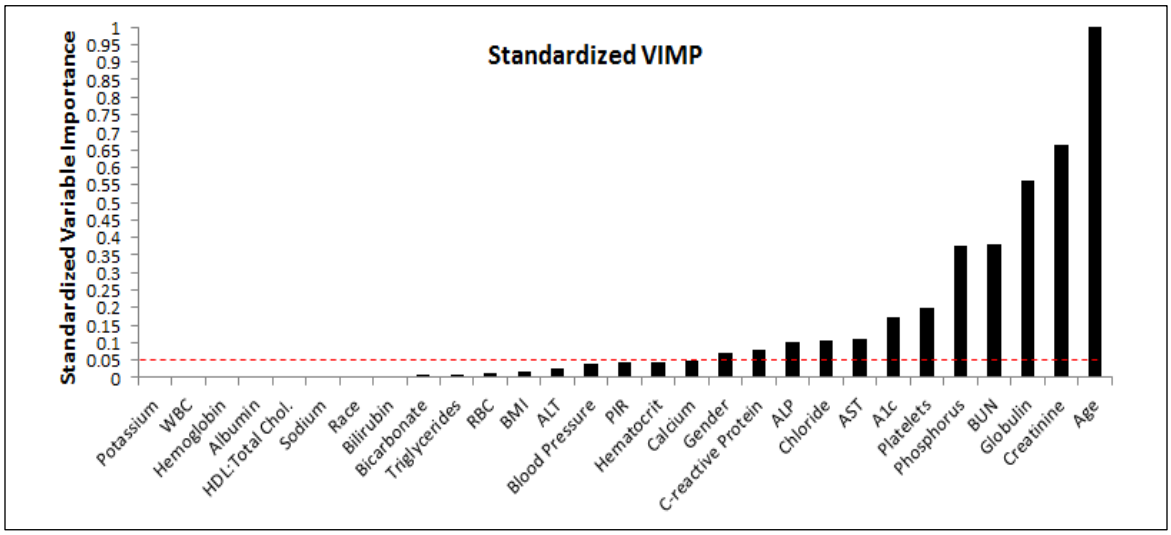


Figure 3.5: *Standardized Variable Importance measures for variables in the Random Survival Forest. The dashed red line indicates the 5% importance threshold used to select variables.*

Identifying interaction effects

As discussed earlier, the association measures for all pairs of variables in our model was computed. An interaction between a pair of variables is considered relatively important if their association measure is high as well as the univariate VIMPs of the individual variables. This rule requires setting thresholds to determine 'high' univariate VIMPs and 'high' association measures. For the univariate VIMP, we use the '5% rule' discussed earlier, designating as important any variable whose standardized univariate VIMP exceeds 0.05. However, applying this rule to association measures results in too many interactions being selected which complicates our WQS model and pushes the limits of the optimization algorithms used to fit the WQS model. We therefore chose to use the top 10^{th} percentile of association measures as the threshold above which an interaction was deemed important. Using this combination of rules we were able to select the following 22 interactions:

Table 3.2: Important Interactions (identified via VIMP)

Creatinine × Globulin	BUN × Phosphorus
Creatinine × BUN	BUN × ALP
Creatinine × Phosphorus	BUN × C-reactive protein
Creatinine × A1c	Phosphorus × Platelets
Creatinine × Platelets	Phosphorus × A1c
Creatinine × C-reactive protein	Phosphorus × C-reactive protein
Creatinine × ALP	ALP × C-reactive protein
Creatinine × AST	
Globulin × BUN	
Globulin × Phosphorus	
Globulin × A1c	
Globulin × C-reactive protein	
Globulin × Platelets	
Globulin × ALP	
Globulin × Chloride	

Table 3.2 above indicates that more than two-thirds of the interactions involve creatinine and globulin, the two biomarkers with the highest VIMPs.

We included these 22 interaction effects in the extended WQS model given in equation (3.2). Added to the 24 univariate biomarker variables, this made 46 biomarker-related (non-demographic variabels). We then fitted the model on the training dataset (NHANES 1999-2002 cohort) using the Trust Region nonlinear optimization algorithm (Dennis, Gay & Welsch, 1981) to obtain estimates of the model parameters, with particular interest in the HSM weights. For 20 of the 46 biomarker-related variables, the estimated HSM weights were zero. The non-zero weights are shown in the plot below.

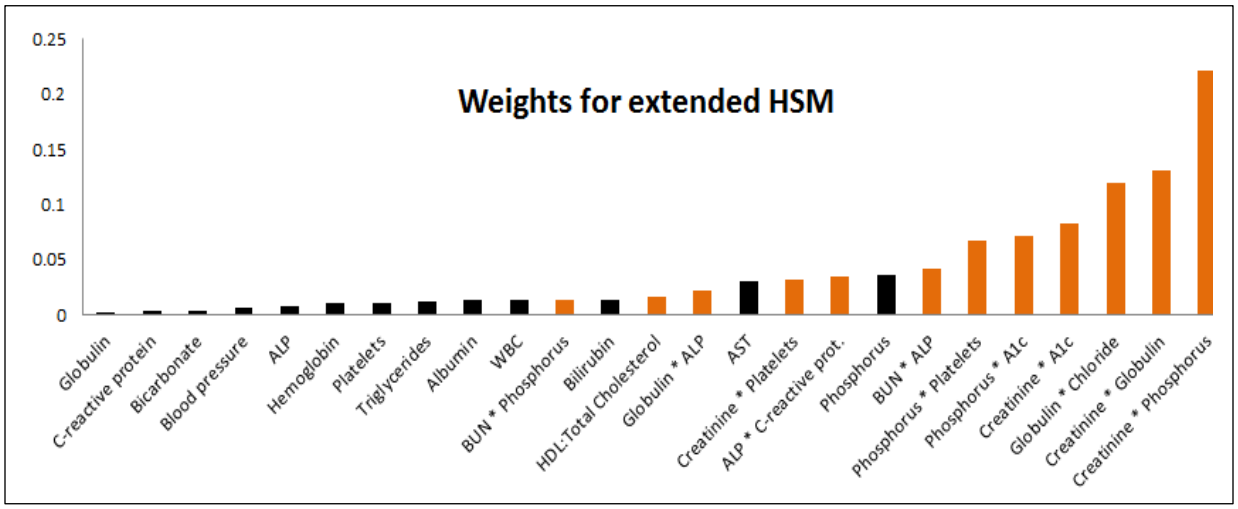


Figure 3.6: *Estimated weights for extended HSM. Interactions selected using VIMP-based thresholding*

The bars highlighted in orange represent weights associated with interaction effects. It can be observed that the weights associated with interactions are the highest and, at the high end, decisively dominate the weights for univariate effects. Out of 22 interaction effects included in the model, only 11 had non-zero estimated weights. Focusing on the univariate effects, we see that phosphorus has the highest weight of any univariate effect, though this weight is relatively low compared with those of some of the interaction effects. Notice as well that nearly half (3 out

of 7) of the interaction effects whose weights exceed that of phosphorus involve phosphorus itself.

To test the predictive accuracy of the extended HSM constructed using these estimated weights, we used an independent validation set (NHANES III). The extended HSM was computed for all individuals with complete data in the NHANES III validation set and Harrell's C-index was used to measure the degree of concordance between HSM predictions and observed survival times. Harrell's C for the extended HSM was compared to that of the regular HSM (with no interactions). The results are summarized in the table below:

Table 3.3: Harrell's C for original & extended HSM

Risk score	**Harrell's C-index**	**95% CI**
HSM	0.7	[0.690, 0.713]
$HSM_{extended}$	0.685	[0.674, 0.697]

The Harrell's C-statistic for the extended HSM is nominally lower than that of the regular HSM with no interactions. Thus including the interactions does not provide any significant improvement to the predictive accuracy of the HSM, and therefore in the interest of parsimony, the simpler HSM is a better option.

3.6.2 Minimal Depth

The plot below shows the ranking of variables based on minimal depth. The dashed red line is the statistical threshold based on the distribution of the minimal depth random variable. The minimal depth threshold was found to be 5.68. Recall that the smaller a variable's minimal depth, the more predictive it is. Figure 3.7 below shows that age*, BMI, PIR (poverty income ratio), BUN*, creatinine*, globulin*, blood pressure, ALP*, platelets*, white blood cell count and triglycerides all fall below the importance threshold and thus can be deemed significant in

the current context. The asterisked variables in the preceding list are those that were also identified as important by the VIMP method (using the *ad hoc* threshold we defined).

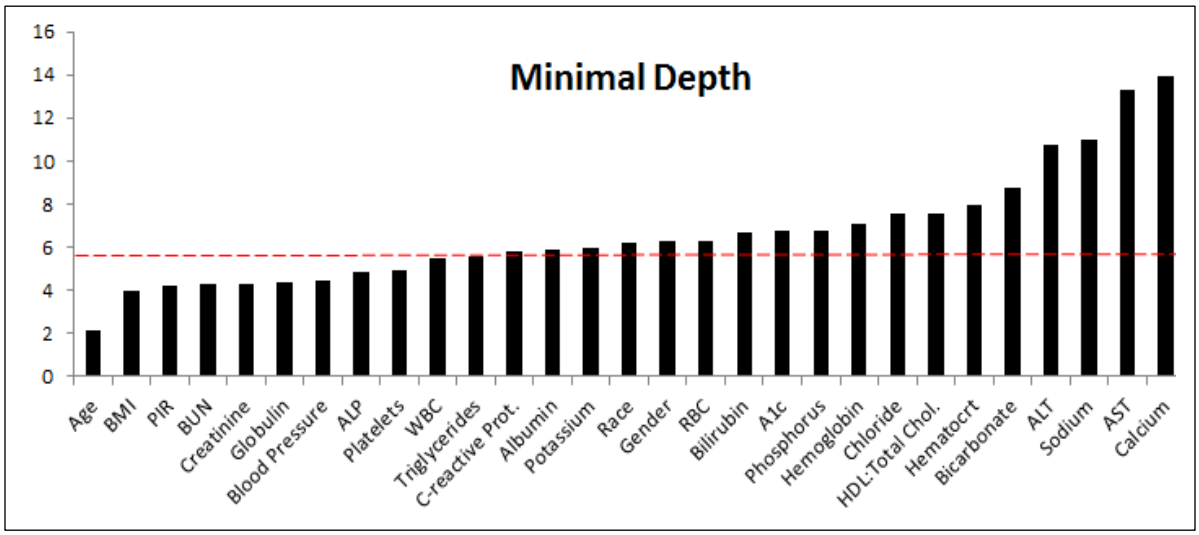


Figure 3.7: *Plot of minimal depths for demographic and biomarker variables used in random survival forest*

The table below shows the univariate and joint minimal depth (just for the biomarker variables).

Table 3.4: Univariate and joint normalized minimal depth for most important biomarkers

	BUN*	Creatinine	Globulin	BP*	ALP*	Platelets	WBC*	Triglyc
BUN*	0.2	0.68	0.54	0.53	0.47	0.53	0.47	0.52
Creatinine	0.56	0.2	0.46	0.47	0.39	0.47	0.37	0.44
Globulin	0.63	0.72	0.2	0.49	0.42	0.49	0.38	0.46
BP*	0.62	0.69	0.51	0.21	0.4	0.44	0.35	0.45
ALP*	0.71	0.81	0.59	0.55	0.23	0.58	0.43	0.5
Platelets	0.69	0.75	0.59	0.58	0.47	0.23	0.42	0.52
WBC*	0.76	0.87	0.63	0.57	0.45	0.53	0.25	0.47
Triglyc	0.72	0.81	0.63	0.58	0.47	0.57	0.41	0.26

*BUN=Blood Urea Nitrogen, BP=Blood Pressure, ALP=Alkaline Phosphatase, WBC=White Blood Cell count

The highlighted diagonal entries represent the normalized minimal depth of the corresponding variable. Notice the table only includes variables whose univariate minimal depth fell below the threshold. As mentioned earlier, we do this because we only consider interactions among

variables with strong marginal effects. The off-diagonal entries represent the normalized minimal depth of variable [*j*] with respect to the index subtree of variable [*i*]. As discussed earlier, 'small' [*i*][*j*] entries indicate a higher likelihood of potential interaction between variables [*i*] and [*j*]. To define the threshold for what a 'small' value is, we used an arbitrary cut-off that seemed reasonable based on observing the relative sizes of entries in the table. For each diagonal entry [*i*][*i*], we defined [*i*][*j*] as 'significant' if it fell below twice the value of [*i*][*i*]. Using this *ad hoc* threshold, we were able to select a number of variable pairs for inclusion in our set of candidate interactions. These are shown in Table 3.4 above as highlighted [*i*][*j*] (off-diagonal) entries. They are listed in the table below:

Table 3.5: Important interactions (identified via minimal depth)

Creatinine × ALP†	Platelets × White Blood Cells
Creatinine × White Blood Cells	White Blood Cells × ALP
Globulin × White Blood Cells	White Blood Cells × Triglycerides
Blood pressure × ALP	Triglycerides × ALP
Blood pressure × White Blood Cells	

† *Interactions also identified as important via the VIMP approach*

Note that out of the 9 interactions selected via minimal depth thresholding, only 1 (Creatinine × ALP) was among the set of 22 interactions selected using the VIMP approach. Also, two-thirds of the selected interactions involve White Blood Cell count. While this particular variable does not have the strongest marginal effect, it appears to have relatively strong interaction effects with all but one (Blood Urea Nitrogen) of the other strong predictors. We included this set of candidate interactions in our extended WQS model given in equation (3.2) and fitted the model using the Trust Region nonlinear optimization algorithm and the training dataset (NHANES 1999-2002 cohort).

The plot below displays the relative magnitudes of the estimated weights. Only non-zero weights are shown.

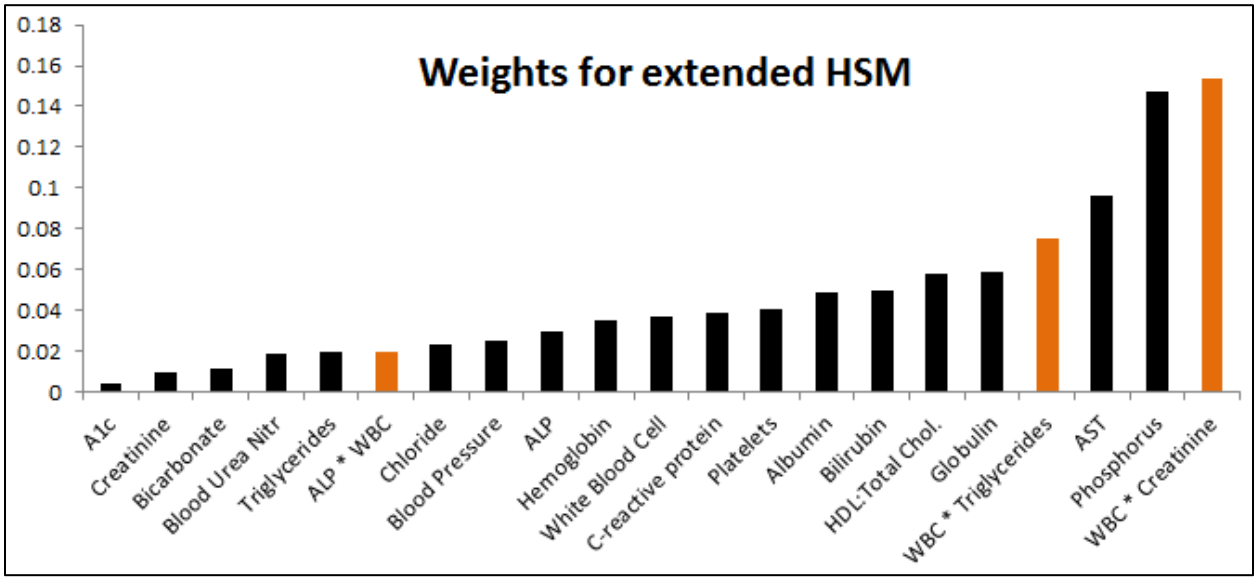


Figure 3.8: *Estimated weights for extended HSM. Interactions selected using minimal depth thresholding*

The bars highlighted in orange represent weights associated with interaction effects. Among all univariate and interaction effects, the one with the largest weight is the interaction White Blood Cells × Creatinine. Out of 9 interaction effects included in the model, only 3 had non-zero estimated weights. Focusing on the univariate effects, we see that Phosphorus has the highest weight of any univariate effect, and this weight is comparable to that of White Blood Cells × Creatinine. To test the predictive accuracy of the extended HSM constructed using these estimated weights, this version of the HSM was computed for all individuals with complete data in the NHANES III validation set and Harrell's C-index was used to measure the degree of concordance between HSM predictions and observed survival times. Harrell's C for the extended HSM was compared to that of the regular HSM (with no interactions). The results are summarized in the table below:

Table 3.6: Harrell's C for original and extended HSM

Risk score	Harrell's C-index	95% CI
HSM	0.7	[0.690, 0.713]
$HSM_{extended}$	0.697	[0.686, 0.708]

Just as in the case of the extended HSM created from including interactions identified through VIMP, we find again that the Harrell's C-index of this extended HSM is actually (nominally) lower than that of the original HSM with no interactions.

3.7 Discusssion

We have developed an extended version of the HSM that accommodates interaction effects. To restrict our focus to just a few important interaction effects, we used random survival forests to identify pairs of variables that potentially interact. We used two different importance measures to select candidate interaction effects and each produced a different set of interactions. There was little overlap between both sets. Each set of interactions was (separately) used to produce an extended HSM. The extended WQS model was used (with the training set [NHANES 1999-2002]) to estimate weights for the univariate and interaction effects for these extended HSMs. However both had lower predictive accuracy than the original HSM (with no interactions) when tested on the NHANES III validation set. A possible cause of this somewhat counterintuitive effect is overfitting. The poor performance of the extended HSM could be because, while the identified interactions were 'important' in the training set, they were not so in the validation set; hence by including interactions, we may have been overfitting the HSM to the training set and consequently reducing its generalization accuracy for other (independent) datasets. To test this conjecture, we ran the random survival forest algorithm on the validation set and found 46 candidate interactions (using minimal depth thresholding). Only 2 of these

overlapped with the set of candidate interactions identified using the training set. This lends support to the idea that the interactions identified in the training set (and used to construct the extended HSM) were highly specific to that dataset and this is one probable reason for the underperformance of the extended HSM when applied to the validation set.

It must be mentioned that several alternatives exist to the particular procedure we used for identifying interactions in this chapter. For example, rather than testing all possible interactions in the same model, a more guided approach such as forward selection could have been used. The variable importance measures used for ranking and selecting interaction effects are based on out-of-bag prediction errors. A possible alternative would be to base the prediction errors on a separate test dataset that shares no overlap with the training set. This particular approach could potentially mitigate the overfitting we observed since the selection of important interactions would be based solely on data that was not used for training the algorithm. Additionally, there may be room for improvement in the overall accuracy of the random survival forest predictions we obtained. Random forest parameters such as the number of trees, the number of variables randomly sampled at each split, terminal node size, and choice of splitting rule could be varied (e.g. the number of trees could be increased) in order to find optimal settings that produce uniformly lower prediction errors.

While this study failed to demonstrate any improvement in HSM prediction accuracy due to inclusion of interactions, it contributed a useful and simple modification to the WQS model that can be used in other applications of the WQS regression technique. Recall that WQS regression is ideal for modeling data with highly correlated variables (components) that can be logically grouped into an index. If interest is centered on accounting for interactions among these index components, then the simple extension introduced in this chapter may be used. The

extension can also be easily generalized to allow for other types of interactions, e.g. interactions between demographic covariates and index components. For example, in order to incorporate interactions between a set of index components $\{q_i\}$ and a discrete/categorical demographic covariate (e.g. race, gender) with K categories, the formulation given below can be used:

$$g(\mu) = \beta_0 + \sum_l \alpha_l z_l + \beta_1 \left(\sum_k^K I_k \sum_i w_{i,k}\, q_i \right) \quad ----(3.6)$$

$$\text{constraints}: 0 \le w_{i,k} \le 1; \ \sum_k^K \sum_i w_{i,k} = 1$$

Note that the formulation given in Equation (3.6) uses a generalized linear form that encompasses many of the commonly used regression models, thus $g(\cdot)$ is the familiar link function. Alternatively, a survival model (parametric or semi-parametric) could be used. In Equation (3.6), the quantity in parentheses is the new form of the index and I_k is a 'dummy' indicator variable representing the k^{th} level of the categorical covariate.

To include interactions between a set of components $\{q_i\}$ and a continuous demographic covariate z_r we propose the following formulation:

$$g(\mu) = \beta_0 + \sum_l \alpha_l z_l + \beta_1 \left(\sum_i w_i q_i + \sum_i \varpi_i z_r' q_i \right) \quad ----(3.7)$$

$$\text{constraints}: 0 \le w_i \le 1; \ \ 0 \le \varpi_i \le 1; \ \ \sum_i w_i + \sum_i \varpi_i = 1$$

In Equation (3.7), the expression in parentheses represents the extended index (extended to include interactions). Also, z'_r is the unit-scaled version of z_r , i.e. standardized so that it has a range of [0,1]. The unit-scaling of z_r is done so that the range of the extended index is the same as that of the original (without interactions).

Chapter 4

Dealing with missing biomarker values in the implementation of tools for computing the HSM

4.1 Introduction: Missing Values

The HSM is constructed from 24 biomarkers. While most of these are routinely measured in clinical settings, the typical patient/individual is generally unlikely to have the full set of biomarkers. This raises the question of how to compute the HSM risk score for individuals with missing biomarkers. This question is particularly relevant to the feasibility of implementing a tool for computing the HSM. Such a tool could conceivably take the form of a standalone software/app for use in clinical settings or a publicly-accessible web interface for individuals to use, similar to those available for some of the popular risk scores (e.g. QRISK, Framingham Risk Score, MELD, Reynolds Risk Score). Below is a preliminary schematic of a possible interface:

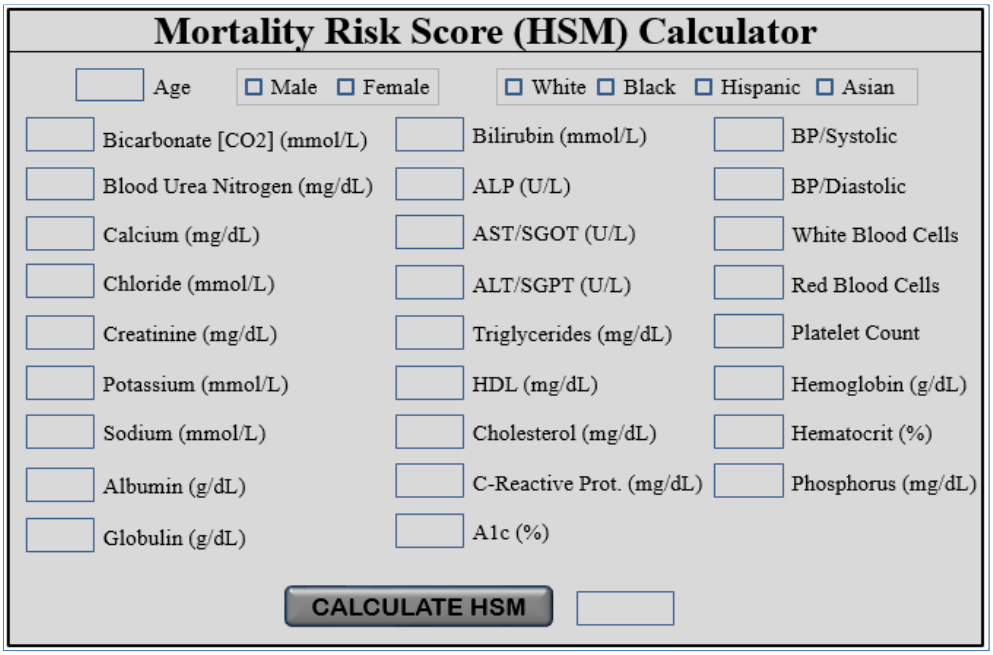


***Figure 4.1**: Schematic depicting a possible web-based or standalone application user-interface for an HSM Risk Calculator*

In the form of a clinical software application, it can be utilized by healthcare providers to compute the HSM for individual patients using information from their medical records. In the form of a web interface, it could be used by the general public in the same way that other popular risk scores are currently used. In either application, the output of the program will be the computed HSM and also 1- and 5-year life expectancy estimates adjusted for the provided age and gender.

Many of the risk scores developed for a variety of conditions have been made available online for members of the general public interested in computing their scores. None of the interfaces we have explored allow or accommodate missing values of risk score components. For example, the Framingham Risk Score web-based calculator hosted by the National Institutes of Health [URL: http://cvdrisk.nhlbi.nih.gov/] will not compute a score for an individual unless they enter all required components (i.e. age, gender, total cholesterol, HDL cholesterol, smoking status and blood pressure).

Therefore a major goal of this project is to explore methods suitable for handling the problem of missing values during routine use of an HSM calculator interface. Ideally, an individual or healthcare provider interested in using the interface to compute HSM but who is unable to fill in all the required components should still (in most cases) be able to obtain an HSM score.

4.2 Methods

To equip the interface with the capability to handle missing values in *real-time*, we will explore different imputation techniques. The problem of carrying out real-time imputations for individual cases (as opposed to a full dataset) is a unique one that has not been extensively studied (Janssen *et al.*, 2009; Kappen *et al.*, 2012). Imputation is typically used to replace missing values in datasets with multiple observations, rather than for replacing missing values in an individual observation/case; therefore some traditional techniques for dealing with missing values (e.g. multiple imputation (Rubin, 1987), EM algorithm (Dempster, Laird & Rubin, 1977)) would be unsuitable for single-case imputation. In addition, a software application with the capability to compute the HSM in the presence of missing values would require an imputation technique that is fast and simple enough to be carried out on-the-fly. In other words, such an imputation technique cannot be computationally intensive and ideally should be non-iterative. The following are a couple of examples of classes of imputation techniques that meet these criteria: measure-of-center-based imputation techniques (e.g. mean imputation, median imputation) and matching-based imputation techniques (e.g. *k*-nearest neighbors imputation (Hastie *et al.*, 1999; Troyanskaya *et al.*, 2001)).

In this chapter, we will use simulated missing data to assess the feasibility and compare the performance of three particular imputation techniques: median imputation, subgroup median imputation, and *k*-nearest neighbors imputation. As implemented in this study, all three techniques use a 'donor' set, i.e. a relatively large external dataset with complete biomarker data (i.e. all 24 biomarkers). We will henceforth refer to this set as the ***donor set***. The donor set we used in this study is the NHANES 2003-2008 biomarker data with a sample size of n=4986. What follows are definitions and descriptions of the imputation techniques we tested:

Median imputation: As discussed in earlier chapters, the standardized measurements for each biomarker range from 0-9. For most biomarkers, the distribution of the standardized values is highly skewed; therefore an appropriate measure of center would be the median. This imputation technique involves simply replacing all missing values of a particular biomarker variable with the computed median of observed values of that biomarker in the donor set. Thus for an individual *i* missing the value of a particular biomarker *v*, we use the median value of *v* in the donor set to replace v_i.

Subgroup median imputation: This method is a more refined version of median imputation. For an individual *i* missing the value of a particular biomarker *v*, rather than replacing this missing value with the overall median value of *v* in the donor set, the median value within the individual's demographic subgroup is used instead. The demographic subgroup can be defined in different ways; in our case we use age group and gender. Six demographic subgroups are defined from a combination of gender and 3 age groups (18-39, 40-64, ≥65).

***k*-nearest neighbors imputation**: This is a more advanced donor-based imputation technique which has been used in a variety of applications and has been shown to demonstrate performance superior or comparable to other imputation techniques (see Troyanskaya *et al.* (2001) and Schwender (2012)). Details on the adaptation of this technique for our specific purposes are described below:

As discussed earlier, the donor set consists of a large number of individuals with a complete set of the 24 biomarkers required to compute the HSM. Define set ***P*** as the full set of 24 biomarkers. Denote an individual using the HSM calculator interface as *i*. Let *K* be the set of biomarkers that individual *i* is able to provide ($K \subseteq \boldsymbol{P}$), e.g. if this subject only provides values

for Albumin, Bilirubin, Bicarbonate and Calcium then these 4 biomarkers would constitute set *K*. The set of biomarkers that the individual is missing values on is therefore given by K^c. The basic idea behind this method is that individual *i*'s values of biomarkers constituting set *K* can be compared to the corresponding values for each case/individual in the donor set. Donor set cases which match closely on these biomarkers are considered 'nearest neighbors' to *i* in the variable subspace defined by *K*. Once these nearest neighbors have been identified, their values of the biomarkers individual *i* is missing (i.e. set K^c) can then be used to impute for *i*.

The next issue is how to define nearest neighbors, i.e. how to quantify the 'nearness' of individual *i*'s observed values to the corresponding values in the donor set. This is done using a suitable distance metric. For every case *j* in the donor set, we define the distance to subject *i* as:

$$d(i,j)^2 = \sum_{k \in K} \left(\frac{x_i^{(k)} - x_j^{(k)}}{r_k} \right)^2 + \sum_{h \in H} \left(\frac{z_i^{(h)} - z_j^{(h)}}{r_h} \right)^2 + \sum_{g \in G} I\left(y_i^{(g)} \neq y_j^{(g)} \right) \quad ----(4.1)$$

x : biomarker variables
z : continuous demographic variables
y : categorical demographic variables
$r_k = range_D(x^{(k)}) = \max_D(x^{(k)}) - \min_D(x^{(k)})$ [D = donor set]
$r_h = range_D(z^{(h)}) = \max_D(z^{(h)}) - \min_D(z^{(h)})$

In Equation 4.1, *K* as defined earlier is the set of observed (non-missing) biomarkers for subject *i*. Set *H* in Equation 4.1 above denotes the set of continuous demographic variables (e.g. age) and set *G* denotes the set of categorical demographic variables (e.g. gender, race). The variables *x*, *z* and *y* thus represent values of the biomarker variables, continuous demographic variables, and categorical demographic variables, respectively. The biomarker variables (as well as the continuous demographic variables) have different scales/units of measurement, so in order

to guarantee that differences in each variable contribute equally to the overall distance, we normalize/standardize the difference for each variable by dividing by the variable's observed range in the donor set. The terms r_k and r_h therefore represent the observed ranges for each biomarker $x^{(k)}$ and continuous demographic variable $z^{(h)}$ in the donor set. These values are used to standardize/normalize the differences so that the variables with larger-valued units of measurement do not dominate the computed distance metric. The reason that demographic variables are used in computing the distance metric $d(i, j)$ will become clear in the next paragraph.

Having defined the components of the expression in Equation (4.1), it is straightforward to see that the total distance $d(i, j)$ between a case j in the donor set and an individual i is defined as the sum of the range-standardized Euclidean distance for the observed biomarker variables, the range-standardized Euclidean distance for observed continuous demographics variables and the 'matching distance' for categorical demographic variables. The matching distance is a simple distance measure that is implemented as an indicator function which evaluates to 0 if the values of two categorical variables match and 1 otherwise. For example, a match in gender between an individual i and a donor set case j would evaluate to 0 under this definition, and a mismatch would evaluate to 1. The choice of which values (0 or 1) are assigned to matches and mismatches can be rationalized as follows: a mismatch (e.g. in gender) indicates dissimilarity between individual i and case j (in the donor set) and is therefore penalized by assigning a higher value of 1 which increases the magnitude of the distance metric $d(i, j)$. On the other hand a match, which evaluates to a value of 0, produces no increase in the distance metric. The levels of several biomarkers of health are known to vary by age group and gender and it is reasonable to assume that individuals in the same gender and/or age group would be similar on at least some

biomarkers. Therefore matching on age and gender (and also BMI) could plausibly improve matching. And since these demographic variables are straightforward for users of the HSM calculator interface to provide, they will rarely be missing and the information contained in these variables can be leveraged to increase matching accuracy.

The distance metric $d(i, j)$ is computed for all cases j in the donor set and the cases with the k smallest distances are selected as the k nearest neighbors to individual i (in the subspace defined by biomarkers in K). Denote the set of these nearest neighbors by C_{KNN}. Then the biomarkers missing for individual i can be imputed by taking a weighted average of the values of the corresponding biomarkers of the nearest neighbors:

$$\hat{\boldsymbol{x}}_{im} = \frac{\sum_{j \in C_{kNN}} d(i, j)^{-1} \boldsymbol{x}_{jm}}{\sum_{j \in C_{kNN}} d(i, j)^{-1}}$$

In the above expression, $\hat{\boldsymbol{x}}_{im}$ is our k-nearest neighbors-based estimate of the missing biomarkers for individual i and $\boldsymbol{x}_{jm}$ are the (known) values of the corresponding biomarkers for the nearest neighbors selected from the donor set. In this weighted average, the weights are the inverse of $d(i, j)$, meaning that more similar donor set cases contribute more to the weighted average. The number of nearest neighbors (k) is determined empirically by testing the performance of a range of k values.

4.3 Comparison of imputation techniques via simulations:

Simulations were used to compare the imputation techniques described above. Starting with a complete dataset with no missing biomarker values (a complete subset [n=10000] of the NHANES III validation dataset), missingness was randomly induced to create an 'artificial'

dataset with missing values. Each imputation technique was then used to 'recover' the missing values in this artificial dataset and different measures were used to judge the quality of the imputations produced by each technique. Two methods were used for randomly inducing missingness in the complete dataset.

The first method allows for the number of missing values per individual to cover an exhaustive range (i.e. as few as 1 missing value to as many as p-1 [p=total # of biomarkers]). It also gives each biomarker an equal probability of being missing. This method was implemented using the following procedure: For each individual i in the dataset, the number of missing biomarkers k_i was randomly selected from a discrete uniform distribution with range [1, p-1]. Next, k_i distinct biomarkers were randomly selected to be missing. Therefore, each individual could have as few as 1 missing biomarker and as many as p-1, with each biomarker having an equal chance of being among the missing set for each individual. Using this method, the expected percentage of missing data would be 50%. This method is useful for testing each imputation technique for any number of missing values; however, the assumptions underlying it may not be realistic. We would expect that in practical situations certain biomarkers would be more likely to be missing than others, given that some (e.g. blood pressure) are measured far more often than others.

The second method addresses this concern by basing the missingness probability of each biomarker on their observed frequency of measurement in a database of patient medical records obtained from the Virginia Commonwealth University Medical Center. Starting with the complete dataset, missingness is randomly induced using the following procedure: for an individual i and biomarker j, the probability that j is missing is given by its overall missingness frequency in the patient dataset.

Using the 2 methods described above (henceforth referred to as the 'equal-probability' and the 'data-guided' approaches), missing values were randomly introduced into the complete dataset. Then each imputation technique was used to impute the artificially induced missing values. So each imputation technique produced one imputed dataset. The accuracy of each imputation technique was assessed in 2 ways:

I. Computing the Root Mean Square Deviation (RMSD), a measure of the total deviation of the imputed values from their corresponding true values. This quantity is computed as follows:

$$RMSD = \sqrt{\frac{\sum_{t=1}^{N_{imp}} \left(q_t^{true} - q_t^{imp} \right)^2}{N_{imp}}}$$

In the expression above, q_t^{true} and q_t^{imp} are (respectively) the true and imputed standardized biomarker measurements for the tth imputed value and N_{imp} is the total number of imputed values in the dataset. As the definition implies, more accurate imputation techniques will have lower RMSDs than less accurate ones and the closer to zero the RMSD is, the better.

II. Secondly, the HSMs calculated for individuals in each imputed dataset were tested for accuracy in predicting mortality. Harrell's C-index was used to quantify the concordance between HSMs predicted using imputed data and observed survival time.

The entire procedure outlined above was repeated 100 times, i.e. we ran 100 independent rounds of simulating missing values in the full dataset (using the 2 methods described above), imputing them using each technique, then computing RMSD and Harrell's C. We did this to reduce bias in the estimates of RMSD and predictive measures obtained from the artificial

datasets. Thus to compare the RMSDs and the Harrell's C-indices among the different techniques, we compare the distributions of their values across the 100 simulations using basic hypothesis tests.

4.4 Results

As mentioned earlier, the base (complete) set in which missing values were simulated was the NHANES III validation dataset with 10000 individuals each with a complete set of the 24 biomarkers we use in this study. The missingness patterns generated by the two approaches we used for randomly inducing missing values are discussed below.

Method 1: Equal-probability approach

Figure 4.2 below shows the distribution of the number of missing biomarkers across individuals (averaged across the 100 simulations). It confirms that each individual in the dataset could have anywhere from 1 to 23 (of a total of 24) biomarkers missing, and each number is equally possible. Also, on average, each biomarker was missing for close to half the individuals in the dataset.

Method 2: Data-guided approach

Figure 4.3 below shows the distribution of the number of missing biomarkers across individuals (averaged across 100 replications). We see that using the observed frequencies as a guide in randomly introducing missing values produces a dataset with roughly 80% of data missing on average. It can also be observed from the figure that the most common number of missing biomarker values for individuals is 20 (roughly 80% of the 24 total biomarkers).

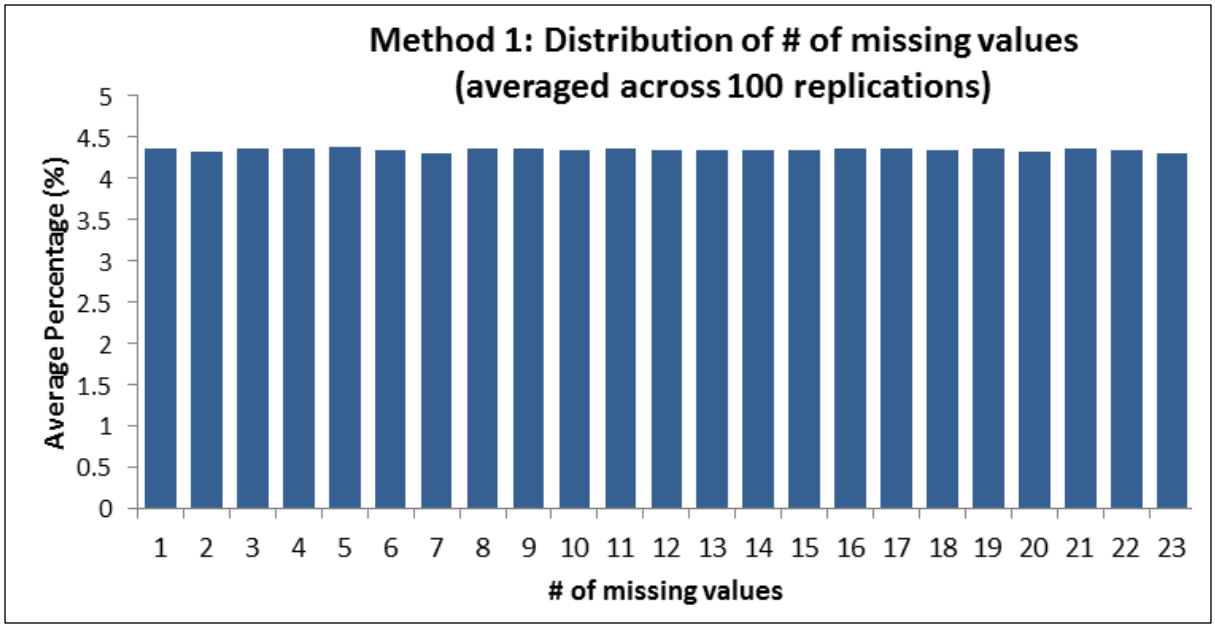


***Figure 4.2**: Distribution of the number of missing values for individuals in dataset (averaged across 100 simulated datasets)*

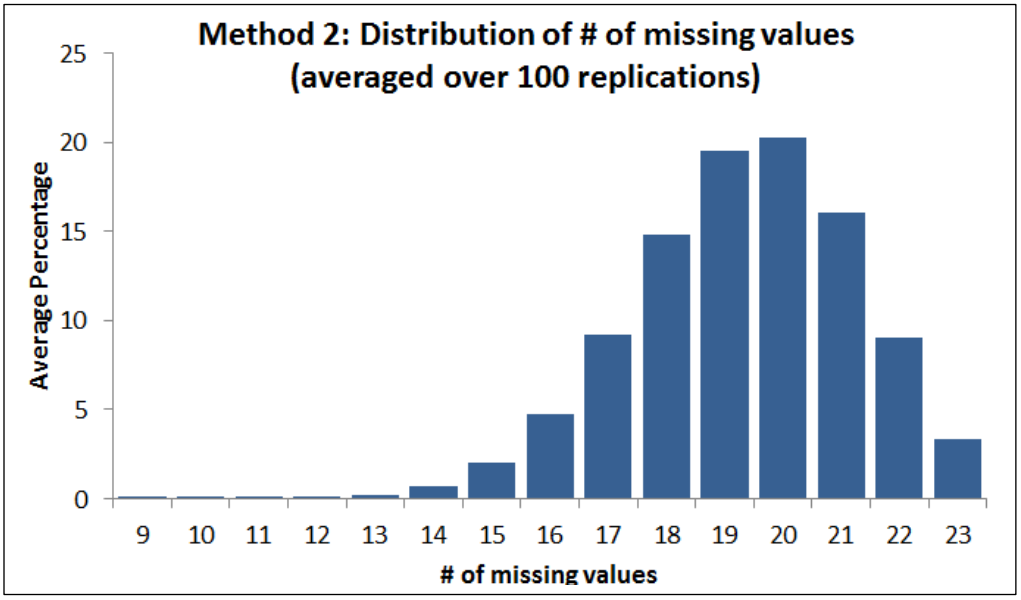


***Figure 4.3**: Distribution of the number of missing values for individuals in dataset (averaged across 100 simulated datasets)*

Performance of Imputation Techniques: As mentioned earlier, the following techniques were used for imputing the missing values simulated in the base set: *k*-nearest neighbors, median imputation and subgroup median imputation. Studies have shown that the performance of the *k*-nearest neighbors technique varies depending on the choice of number of neighbors (*k*). Using the artificial dataset we tested the performance of KNN for a range of values of *k*: 5, 15, 25, 30, 35, 50 and 65. The performance did not vary significantly over this range but *k*=25 had nominally higher performance than the other choices of *k*, therefore this value was chosen. The results are summarized in the figure below:

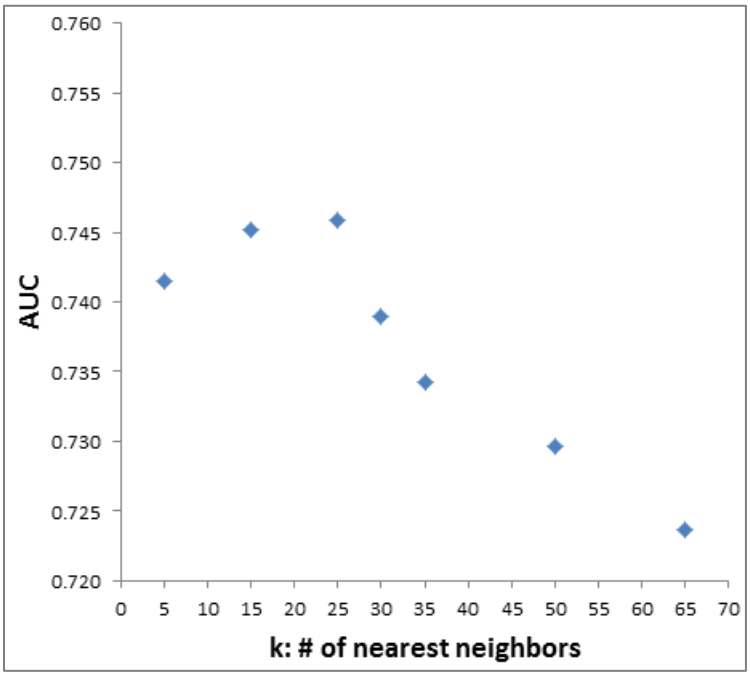


***Figure 4.4**: Plot depicting impact of parameter k (number of nearest neighbors) on predictive performance (as quantified by AUC)*

As mentioned earlier, the entire process of simulating missing values in the complete dataset (using the 2 approaches discussed above), imputing the missing values, and computing performance measures for each imputation technique was repeated 100 times in order to obtain unbiased estimates of the performance measures. Therefore, in the remainder of this chapter, the

results shown for the RMSD and Harrell's C are averaged across the 100 independent rounds of simulations.

The RMSD is an aggregate measure of the overall deviation of imputed values (using each imputation technique) from the known true values in the dataset. A value of 0 indicates an imputation technique with 100% recovery accuracy. To determine whether the RMSD estimates for each imputation technique differ significantly from 0, we used a t-test to compare the distribution of computed RMSD values over the 100 independent rounds to the null value of 0. The figures below show plots of the distributions of RMSD values across the 100 independent rounds for each imputation technique and each approach to simulating missingness.

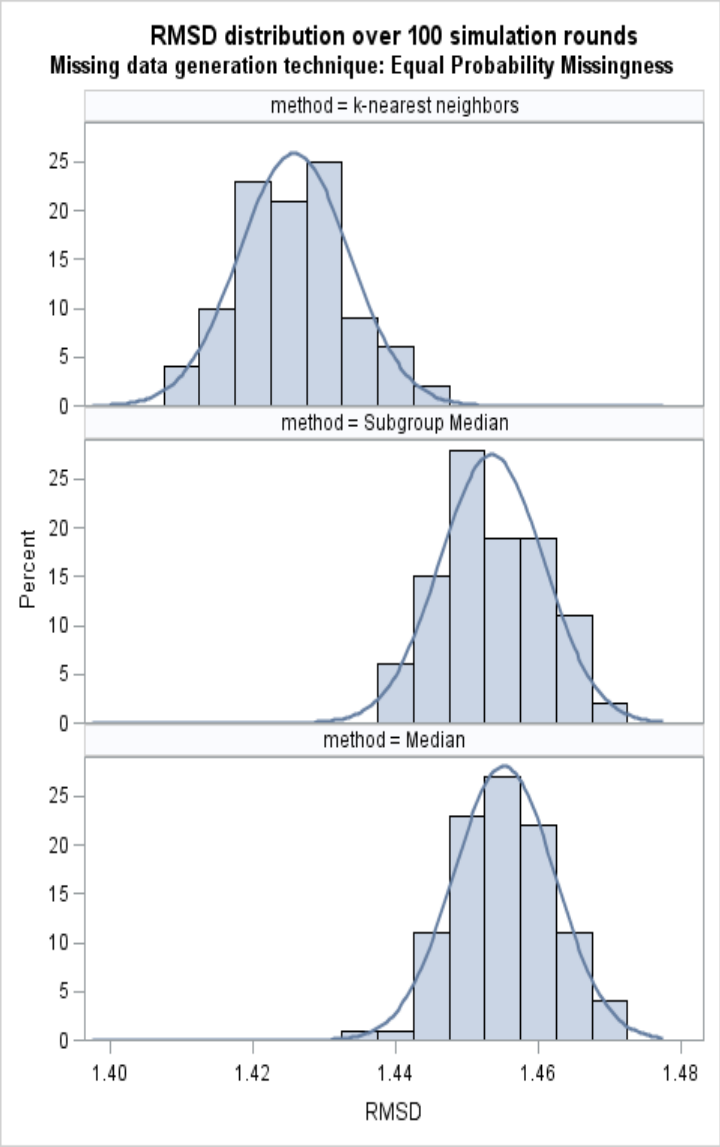


Figure 4.5a*: Distribution of RMSD values across 100 independent simulated datasets. All simulated datasets are generated starting from the same full dataset and randomly inducing missingness using equal-probability' method. RMSD values closer to zero indicate greater imputation accuracy.*

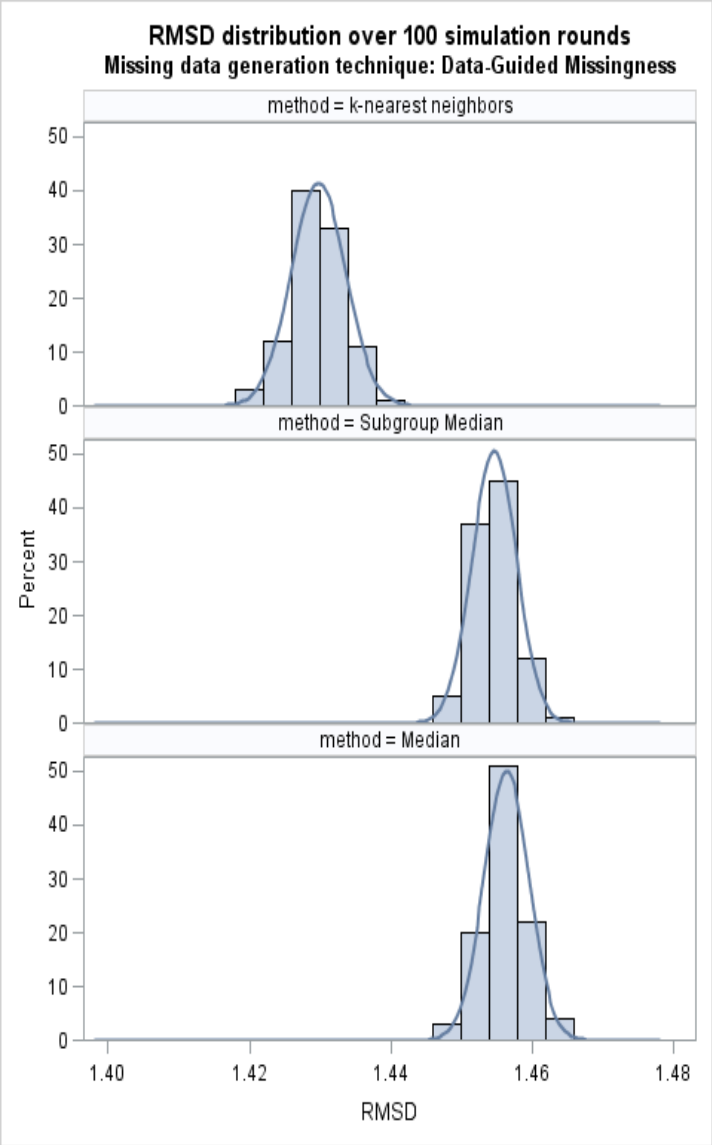


***Figure 4.5b**: Distributions of RMSD values across 100 independent simulated datasets. All simulated datasets are generated starting from the same full dataset and randomly inducing missingness using 'data-guided' approach. RMSD values closer to zero indicate greater imputation accuracy.*

The figures below show plots of the distributions of Harrell's C values across the 100 independent rounds for each imputation technique and each approach for simulating missingness.

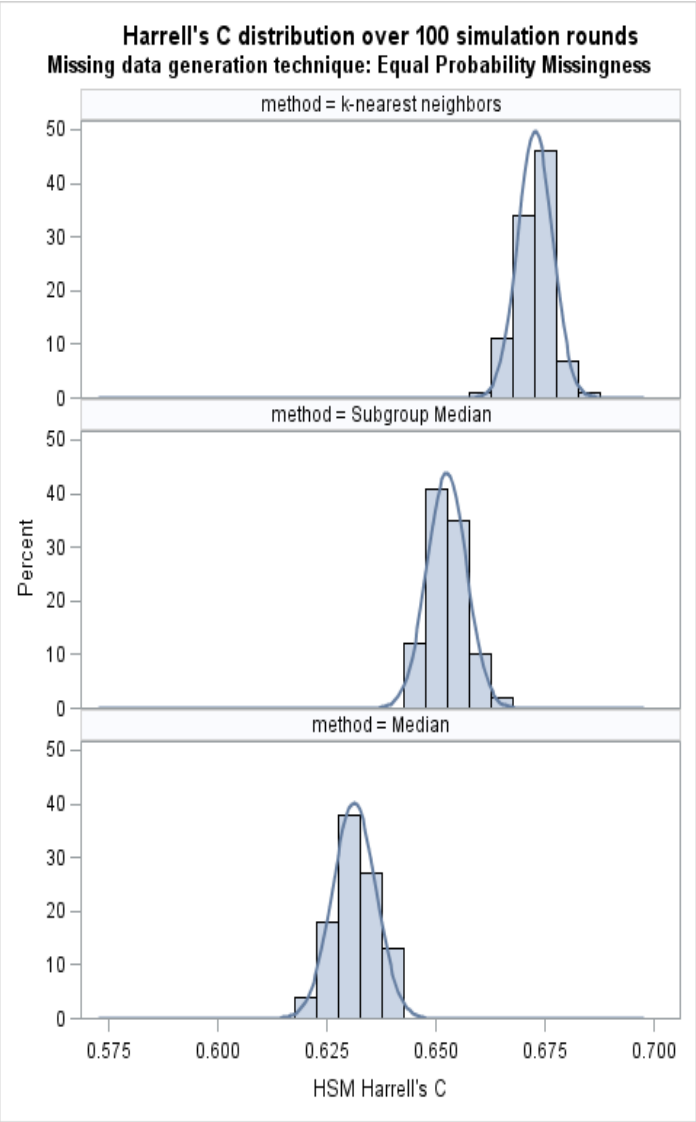


Figure 4.6a*: Distributions of Harrell's C measures across 100 independent simulated datasets. All simulated datasets are generated starting from the same full dataset and randomly inducing missingness using 'equal-probability' approach. Harrell's C values closer to 1 indicate greater predictive accuracy.*

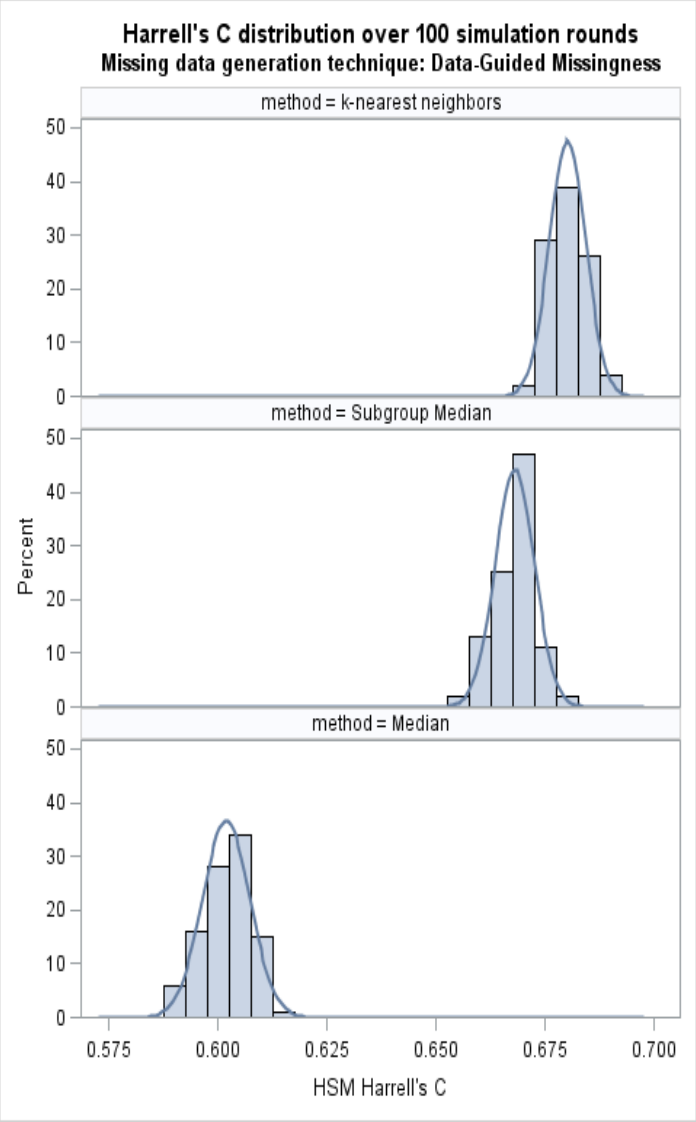


Figure 4.6b*: Distributions of Harrell's C measures across 100 independent simulated datasets. All simulated datasets are generated starting from the same full dataset and randomly inducing missingness using 'data-guided' approach. Harrell's C values closer to 1 indicate greater predictive accuracy.*

For each imputation technique, we compared the Harrell's *C* of the HSM computed from imputed data to that of the HSM computed from the full (complete) data. We used a 1-sample t-test to compare the distribution of imputation-based *C* values (over 100 rounds) to the null value (Harrell's *C* for the HSM computed from the full/complete data). This allowed us to test for a significant difference between the Harrell's *C* measures of HSMs computed from full/complete

data and those computed from imputed data. This process was repeated for each method used to simulate missingness (i.e. 'equal probability' and 'data-guided').

The results summarized in the table below show (for each missingness generation method) the RMSDs and Harrell's C measures (averaged over the 100 independent rounds) for the different imputation techniques we used. The standard deviations over the 100 rounds are also given in parentheses. The asterisked values indicate significant deviation from the null (full/complete data case) based on the statistical tests described in the previous paragraph.

Table 4.1: *RMSD and Harrell's C measures (averaged across 100 simulations) for each imputation technique.*

	Equal Prob. Missingness		**Data-guided missingness**	
	RMSD	Harrell's C	RMSD	Harrell's C
Original	**0**	**0.703**	**0**	**0.703**
KNN (*k*=25)	1.426(.008)*	0.673(.004)*	1.430(.004)*	0.680(.004)*
Median	1.455(.007)*	0.631(.005)*	1.456(.003)*	0.602(.005)*
Subgroup Median	1.453(.007)*	0.652(.005)*	1.455(.003)*	0.668(.005)*

[1]Quantities in parentheses represent standard deviation over 100 rounds

*Asterisks for RMSD values indicate statistically significant deviation from 0. Asterisks for Harrell's C indicate that the estimate of predictive accuracy was (statistically) significantly different (at the 5% level) from that of the original HSM computed with the true values. Statistical significance was assessed by t-tests comparing averaged estimates to null values.

As the results indicate, the RMSDs for all imputation techniques differ significantly from 0. And in all cases, the predictive accuracy (as measured by Harrell's *C*) for HSMs computed from imputed data tends to be significantly worse than that of the HSM computed from full/complete data.

We also compared the imputation techniques to one another based on performance. Specifically, we carried out pairwise comparisons of imputation techniques on RMSD and the Harrell's C measure. To compare RMSD measures for a pair of imputation techniques (denoted

below by *a* and *b*), we computed the following statistic for each of the 100 simulated datasets (each denoted below by *h*):

$$RMSD_{a,h}^2 - RMSD_{b,h}^2$$

We then use a 1-sample t-test to compare the distribution of this statistic to the null value of 0. We use a similar procedure for the Harrell's *C* measure (using the statistic $C_{a,h} - C_{b,h}$).

The results indicate that when we compare the imputation techniques to one another, the *k*-nearest neighbors technique emerges as significantly better on both performance measures. In the artificial datasets produced using the 'equal-probability' and 'data-guided' approaches to inducing missingness, the *k*-nearest neighbors technique demonstrates superior performance on all three performance measures.

Difference	RMSD	Harrell's C
Equal Probability Missingness		
[*kNN*] – [*Subgroup Median*]	-0.0796 (<.0001)	0.0203 (<.0001)
[*kNN*] – [*Median*]	-0.0846 (<.0001)	0.0415 (<.0001)
[*Subgroup Median*] – [*Median*]	-0.00499 (<.0001)	0.0212 (<.0001)
Data-guided Missingness		
[*kNN*] – [*Subgroup Median*]	-0.0717 (<.0001)	0.0121 (<.0001)
[*kNN*] – [*Median*]	-0.0768 (<.0001)	0.0783 (<.0001)
[*Subgroup Median*] – [*Median*]	-0.0051 (<.0001)	0.0662 (<.0001)

Table 4.2*: Mean squared difference (for RMSD) and mean difference (for Harrell's C) between pairs of imputation techniques across simulated datasets. P-values of differences given in parentheses*

Our results agree with those of other studies that have compared *k*-nearest neighbor imputation to other techniques and found its performance either comparable or superior (Troyanskaya *et al.*, 2001; Chipman, Hastie & Tibshirani, 2003).

4.5 Discussion

Note that both methods we used to randomly introduce missing values into the complete dataset assumed that missingness in each biomarker occurs completely at random (MCAR) (Rubin, 1976), i.e. the probability of missing values for each biomarker is independent of both observed and unobserved variables. This is a strong assumption made to simplify the comparison of the imputation techniques. An arguably more common missing data mechanism is referred to as Missingness at Random (MAR), the condition whereby the probability of missing values depends only on observed data. Accounting for the different missingness mechanisms is important in studies assessing or comparing imputation techniques. Future work will focus on repeating the comparisons carried out in this chapter under the assumption of Missingness at Random.

In summary, we explored the use of 3 imputation techniques and the results indicate that regardless of the technique used, there is a significant reduction in general and predictive accuracy. However, comparing the imputation techniques to one another, we found the *k*-nearest neighbors technique showed significantly better performance than the other two (in both artificial datasets), and the subgroup median imputation technique demonstrated better performance than median imputation.

In comparing measures of accuracy for imputation techniques (RMSD, Harrell's C), statistical significance was demonstrated using t-tests comparing the distribution of these measures across 100 simulated datasets. These measures showed very little variation across the 100 simulations (see standard deviation measures in Table 4.1). This is what accounts for the high statistical significance of t-tests for all comparisons we carried out in this study.

All 3 imputation techniques could be described as ‘donor-based’, replacing the missing values of individuals with corresponding values from a large, external donor set (NHANES 2003-2008, n=4986). However each technique differs in the set of donor individuals whose values are used for imputing. The median-imputation technique simply replaces each missing biomarker with the median value of that biomarker across all subjects in the donor set. But the subgroup median uses just a subset of the subjects in the donor set, specifically the subset whose age group and gender match those of the individual for which imputation is being carried out. This produces better results because this subset is expected to be more similar to the individual for which the imputation is being carried out. An even more targeted match is obtained using the *k*-nearest neighbors approach, which imputes using the biomarker values of a smaller but more similar subset of the donor set. The relative performance of these 3 imputation techniques illustrates the importance of matching in donor-based techniques.

The key component of the *k*-nearest neighbors technique is the distance function which quantifies the degree of separation among ‘neighbors’. The distance function used in this study (given in Equation 4.1) is an *ad hoc* formulation designed for our specific purposes. It is a quadratic-form distance that attempts to produce scale-invariance in the computed distances for biomarkers and continuous demographic variables. Other quadratic-forms (e.g. Mahalanobis distance) exist which can produce scale-invariant, unit-less distances; future work will focus on exploring these alternative formulations.

4.5.1 Computational Details

The aim in this chapter was to evaluate computationally fast and relatively accurate imputation methods that could be used in a potential software application (web-based or otherwise) for computing the HSM in the presence of missing values. We found that the *k*-nearest neighbors imputation technique produced the best performance in terms of accuracy. This algorithm involves a nearest-neighbor search which could be computationally intensive and slow if implemented in a naïve fashion, e.g. by looping one by one through all observations/cases in the donor set and computing the distance metric for each. In our studies, we used the more computationally efficient approach of vectorization. We reduced the nearest-neighbor search to a series of simple matrix operations. These operations will now be presented in formal notation but before doing so we will first reproduce Equation (4.1) below for reference purposes:

$$d(i,j)^2 = \underbrace{\sum_{k \in K}\left(\frac{x_i^{(k)} - x_j^{(k)}}{r_k}\right)^2}_{\textbf{Component 1}} + \underbrace{\sum_{h \in H}\left(\frac{z_i^{(h)} - z_j^{(h)}}{r_h}\right)^2}_{\textbf{Component 2}} + \underbrace{\sum_{g \in G} I\left(y_i^{(g)} \neq y_j^{(g)}\right)}_{\textbf{Component 3}} \quad ---- (4.1)$$

x : biomarker variables
z : continuous demographic variables
y : categorical demographic variables

D : donor set
$r_k = range_D(x^{(k)}) = \max_D(x^{(k)}) - \min_D(x^{(k)})$
$r_h = range_D(z^{(h)}) = \max_D(z^{(h)}) - \min_D(z^{(h)})$

Below we show how the distance metric $d(i, j)$ is efficiently and rapidly computed for all cases j in the donor set by using matrix operations. For individual i with a set of p observed biomarkers, denote the vector of values for these biomarkers as $\boldsymbol{x}_i$ and let $\boldsymbol{X}_j$ be the matrix of corresponding biomarker values in the donor set. If the total number of cases in the donor set is N, then **Component 1** in Equation (4.1) can be computed for all donor set cases by executing the following matrix operations:

$$\underset{(N\times p)}{A_1} = \left(\left(\underset{(N\times 1)}{J_N} \times \underset{(1\times p)}{x_i} \right) - \underset{(N\times p)}{X_j} \right) \times \underset{(p\times p)}{R_x^{-1}}$$

$$\underset{(N\times 1)}{A} = \left(\underset{(N\times p)}{A_1} \circ \underset{(N\times p)}{A_1} \right) \times \underset{(p\times 1)}{J_p}$$

N = total number of cases in the donor set

J_λ : $(\lambda \times 1)$ vector of 1's

R_x : diagonal matrix whose diagonal elements are the observed ranges for each biomarker

Matrix operator $\circ$: Hadamard (element - wise) product

Similarly, the vector forms of **Components 2** and **3** (denoted by column vectors *B* and *C* in the derivation below) in Equation (4.1) can be computed as follows:

$$\underset{(N\times v)}{B_1} = \left(\left(\underset{(N\times 1)}{J_N} \times \underset{(1\times v)}{z_i} \right) - \underset{(N\times v)}{Z_j} \right) \times \underset{(v\times v)}{R_z^{-1}}$$

$$\underset{(N\times 1)}{B} = \left(\underset{(N\times v)}{B_1} \circ \underset{(N\times v)}{B_1} \right) \times \underset{(v\times 1)}{J_v}$$

$$\underset{(N\times 1)}{C} = \boldsymbol{I}\left(\left(\underset{(N\times 1)}{J_N} \times \underset{(1\times w)}{y_i} \right) \neq \underset{(N\times w)}{Y_j} \right) \times \underset{(w\times 1)}{J_w}$$

v = number of continuous demographic variables

z_i = row vector representing continuous demographic variables for individual *i*

Z_j = matrix of continuous demographic variables for subjects in the donor set

R_z = diagonal matrix whose elements are the observed ranges for continuous demographic variables

w = number of categorical demographic variables

y_i = row vector representing categorical demographic variables for individual *i*

Y_j = matrix of categorical demographic variables for subjects in donor set

$\boldsymbol{I}(\cdot)$ = vector-based indicator function that operates in elementwise fashion

Keeping in mind that the number of cases in the donor set is N, the above computations can be used to obtain the ($N \times 1$) vector that represents the distances $d(i, j)$ between individual i and every case j (where $j = 1 \ldots N$) in the donor set:

$$\underset{(N\times 1)}{\boldsymbol{d}_{Euc}} = \left(\underset{(N\times 1)}{A} + \underset{(N\times 1)}{B} + \underset{(N\times 1)}{C} \right)^{\circ (1/2)}$$

In the above equation, the operation $(\cdot)^{\circ (1/2)}$ is the Hadamard square root (Reams, 1999) which is essentially the element-wise square root. Once this vector $\boldsymbol{d}_{Euc}$ has been obtained for the entire donor set, it is sorted in ascending order and the donor cases corresponding to the first k elements in the sorted vector are the nearest neighbors.

Chapter 5

Ensemble Methods for improving predictive accuracy of the HSM

5.1 Introduction

In Chapter 2, we demonstrated that the HSM successfully predicts mortality and other outcomes. In particular, the HSM's predictive power for mortality has been assessed using two measures of prognostic accuracy: the *Area under the ROC curve* (AUC) statistic and Harrell's C-statistic.

To get a sense of how the HSM's predictive power compares to that of similar risk scores (i.e. risk scores for predicting all-cause mortality in the general population), we assessed the predictive accuracy of the Intermountain Risk Score (IMRS) (Horne *et al.*, 2009) for individuals in the same NHANES III validation dataset used to test the HSM (see Chapter 2). The IMRS has been discussed in Chapter 1; like the HSM, it was developed as a general-purpose risk score to predict mortality but it uses a limited range of biomarkers and doesn't cover as many facets of physiological health as the HSM does. Because the IMRS and HSM predict the same endpoint, we use the IMRS as a benchmark to which the HSM's predictive accuracy is compared. Table 5.1 summarizes the 2 measures of predictive accuracy for the HSM and the IMRS:

Table 5.1: Predictive Accuracy of HSM compared with that of the Intermountain Risk Score (IMRS)

Measure	HSM	IMRS	HSM (with Age)
Harrell's C	0.7	0.81	0.86
$AUC_{5\text{-year}}$	0.74	0.83	0.87

Focusing on the second and third columns of Table 5.1 we see that, compared to the HSM, the IMRS has a noticeably higher predictive accuracy (as quantified by Harrell's C and AUC). The superior predictive performance of the IMRS is likely due to the explicit inclusion of age as part of the IMRS risk model. Age is, for obvious reasons, an exceedingly strong predictor of mortality and constructing a risk score that explicitly uses this variable generally produces enhanced predictive accuracy. The HSM on the other hand is designed to be a risk score that could also function as a holistic measure of physiological health status and which is comprised of just measurements of health biomarkers. It does not include age or any other demographic variables but, rather, adjusts for these in its risk model. This explains the discrepancy between the predictive accuracies of the two scores. In fact, if we explicitly include age in the HSM (as was done for the IMRS), the resulting HSM exhibits a stronger predictive accuracy than the IMRS. The Harrell's C and AUC for this version of the HSM is shown in Table 5.1 under the heading '***HSM (with age)***'. The expressions below show how this version of the HSM was constructed.

$$\log T = \mu + \boldsymbol{\alpha}'\boldsymbol{y} + \underbrace{\beta_a * age + \beta_1 \sum_i w_i x_i}_{\textbf{HSM (with age)}} + \sigma\psi; \quad \text{constraints}: w_i \in [0,1],\ \sum_i w_i = 1$$

$\boldsymbol{y}$: vector of demographic covariates (excluding age); $\boldsymbol{\alpha}$: corresponding coefficients

$$HSM_{w/age} = \hat{\beta}_a * age + \hat{\beta}_1 \sum_i \hat{w}_i x_i$$

However we will continue to use the original version of the HSM (without age), therefore the goal of the study described in this chapter is to explore various techniques to improve this HSM's predictive accuracy (as quantified by the AUC and Harrell's C). We will borrow methods from statistical/machine learning that have demonstrated success in improving prediction/generalization accuracy in a variety of supervised learning algorithms.

5.1.1 HSM as a predictor

Following Breiman (1996) and Bühlmann & Yu (2002) we will present a formal definition of a predictor. Let $\mathcal{L}$ be a training or learning set given by $\{(Y_i, \boldsymbol{x_i}), i = 1\ldots.N\}$. Here, $\boldsymbol{x_i}$ is a p-dimensional vector of explanatory variables and Y_i is a real-valued response which could be binary, continuous, ordinal, etc. A model (e.g. logistic regression) or learning algorithm (e.g. random forest) can use this learning set data to construct a predictor $H(\boldsymbol{x})$ which predicts the unknown outcome/response for a new observation with explanatory variables $\boldsymbol{x_{new}}$. For example, a classification tree constructed from training/learning data can be thought of as a predictor $H_{CT}(\boldsymbol{x})$ that, for a new observation $\boldsymbol{x_{new}}$, returns a binary value representing the predicted outcome for that observation. The particular way that the information from the learning set $\mathcal{L}$ is used to construct the predictor for the targeted outcome/response is unique to each model/learning algorithm. Some may use a parametric, model-based approach (e.g. linear regression) and others use non-parametric approaches (e.g. random forests, neural networks). At this point, it should be noted that the term 'predictor' as used here (and in the rest of this chapter) is not to be confused with 'predictor variable', an alternative term for 'explanatory variable', 'independent variable' or 'regressor'.

To develop the HSM, the model/learning algorithm we used was the Weighted Quantile Sum (WQS) technique and the learning set $\mathcal{L}$ consisting of data $\{(T_k, \delta_k, \boldsymbol{x}_k, \boldsymbol{z}_k), k = 1,\ldots.N\}$, where T_k, δ_k, $\boldsymbol{x}_k$, and $\boldsymbol{z}_k$ represent the observed time, censorship indicator, biomarker variables, and demographic variables for an individual k in the learning set. For survival outcomes, the WQS may be nested within the framework of an Accelerated Failure Time model; for this example we use a Weibull AFT:

$$\log T = \mu + \boldsymbol{z}'\boldsymbol{\alpha} + \beta_1 \sum_{i=1}^{p} w_i x_i + \sigma\psi \qquad ----(5.1)$$

$$\text{where } w_i \in [0,1],\ \sum_{i=1}^{p} w_i = 1$$

The $\{w_i\}$ in Equation 5.1 are weight parameters estimated by a nonlinear optimization algorithm. The HSM is the weighted sum computed using the estimates of these weights:

$$HSM = \sum_{i=1}^{p} \hat{w}_i x_i,$$

This construct produces a 'score' that is predictive of mortality. Therefore the HSM can be thought of as a predictor $H(\boldsymbol{x})$ that takes an input $\boldsymbol{x}$ (a set of biomarker measurements) and produces a predictive score which is the weighted sum of the standardized biomarker measurements. Note that since the HSM does not directly predict survival time (the response/outcome in Equation 5.1), it does not adhere strictly to the definition of a predictor we outlined earlier in this section. However the predictive score that the HSM produces is directly linked with survival time (i.e. higher scores imply shorter survival times) so it will be considered here, in a loose sense, as a predictor. As discussed in Chapter 2, the HSM score is used to quantify life expectancy and to serve as a holistic measure of physiological health, it is not meant to be a direct estimate of survival time.

In Carrico (2013), the WQS technique on which the HSM is based was characterized and applied to the problem of variable selection in environmental chemical mixtures. The author introduced a bootstrapping step in the construction of the WQS as a means to reduce the variance of the estimated weights. We adopted this technique in the construction of the HSM and results in earlier chapters are based on it. We take a large number (B) of bootstrap samples from the learning set $\mathcal{L}$ and for each sample b, we fit the model defined above to obtain a set of weight

estimates $\{\widehat{w}_{i(b)}\}_{i=1}^{p}$ for that particular sample (p here is the total number of biomarkers). These weights can be used to construct a predictor $H_b(\boldsymbol{x})$ that is specific to the sample b:

$$H_b(\boldsymbol{x}) = \sum_{i=1}^{p} \hat{w}_{i(b)} x_i \qquad ----(5.1a)$$

The final step involves the construction of an 'aggregate' HSM using the weights from all the bootstrap samples as shown below:

$$H_{agg}(\boldsymbol{x}) = \sum_{i=1}^{p} \bar{w}_i x_i, \quad \text{where } \bar{w}_i = \frac{1}{B}\sum_{b=1}^{B} \hat{w}_{i(b)} \qquad ----(5.1b)$$

Comparing Equations (5.1a) and (5.1b), we can see that the aggregate predictor $H_{agg}(\boldsymbol{x})$ is just the average of all the individual predictors $\{ H_b(\boldsymbol{x}) \}_{b=1}^{B}$ obtained from the B bootstrap samples:

$$H_{agg}(\boldsymbol{x}) = \frac{1}{B}\sum_{b=1}^{B} H_b(\boldsymbol{x})$$

Therefore the process we use for constructing the final HSM involves aggregating a large number of bootstrap-generated predictors to produce an aggregate predictor $H_{agg}(\boldsymbol{x})$; in this particular case the aggregation method is the simple average. This process is similar in a number of ways to the procedure used to construct random forests (Breiman, 2001). A large number of 'randomized' classification trees are grown from bootstrap samples and combined to form an aggregated predictor, the random forest. This process is termed *bootstrap aggregation* (or "*bagging"*) and was introduced in Breiman (2001) for improving prediction accuracy of decision trees. The ability of bagging to improve predictive accuracy has also been demonstrated for other learning algorithms. The key to its effectiveness is that averaging a group of predictors produces an aggregate predictor with variance less than or equal to that of any of the individual predictors.

The effect on prediction error can be seen by decomposing the mean-squared prediction error (MSPE) into bias and variance components and noticing that averaging preserves the bias. Thus this variance-reducing procedure gives an aggregate predictor with usually lower MSPE than any of the individual predictors (Bühlmann, 2003). The degree of reduction in MSPE is affected by a number of factors, key among which is the stability of the algorithm/model used to construct the predictor (Breiman, 1996). In the next section we define and briefly discuss stability.

5.1.2 Stability in Learning Algorithms

A stable predictor is defined heuristically as one whose predictions do not change significantly when the learning set $\mathcal{L}$ is slightly perturbed (Breiman, 1996). Perturbation in this context refers to changing the dataset in any number of ways, e.g. deleting/adding records. Studies (e.g. Breiman (1994)) have shown that several commonly-used learning algorithms (including subset selection in linear regression) are unstable to some degree or other. One major influence on the effectiveness of bagging is the instability of the learning algorithm; more unstable algorithms will exhibit greater improvement in prediction accuracy.

We now examine the stability of the learning procedure we use to construct individual HSM predictors by studying the variation in the weights across bootstrap samples. Bootstrap sampling can be seen as a form of learning set 'perturbation' because each bootstrap sample will, in almost all cases, select only a subset of the original dataset and include replicates. Below is a plot of the variation in HSM weights across 1000 bootstrap samples:

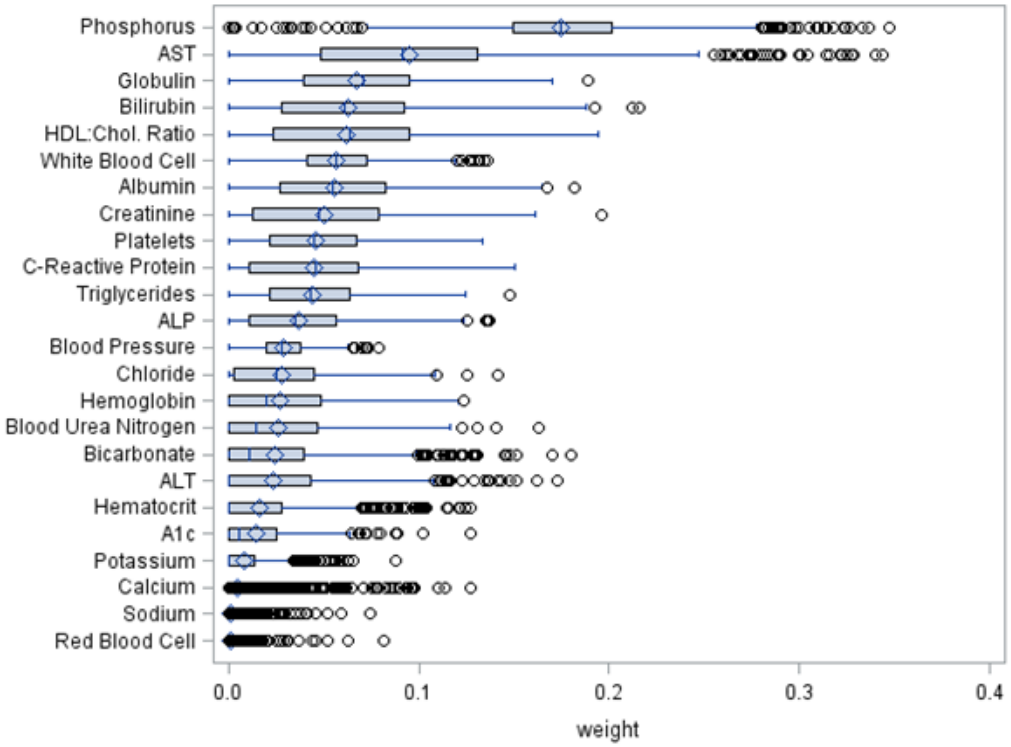


Figure 5.1: Boxplot showing the distribution of biomarker weight estimates across 1000 bootstrap samples.

The moderate variation in weight estimates across bootstrap samples indicates the learning model used to produce the weights is somewhat unstable. This makes it a good candidate for bagging and justifies our use of this method.

5.1.3 Ensemble Learning

While bagging is a powerful technique for model/predictor aggregation, several other aggregation techniques exist, many of which in fact predate the introduction of bagging. The idea of combining models or predictors has been explored in various contexts and statistical applications (Efron & Morris, 1973; Rao & Subrahmaniam, 1973; Berger & Bock, 1976; Green & Strawderman, 1991). In the field of machine learning, aggregation is applied primarily to learning algorithms to improve prediction accuracy and this concept is referred to as *ensemble*

learning. The schematic depicted in Figure 5.2 below illustrates a common type of ensemble learning procedure:

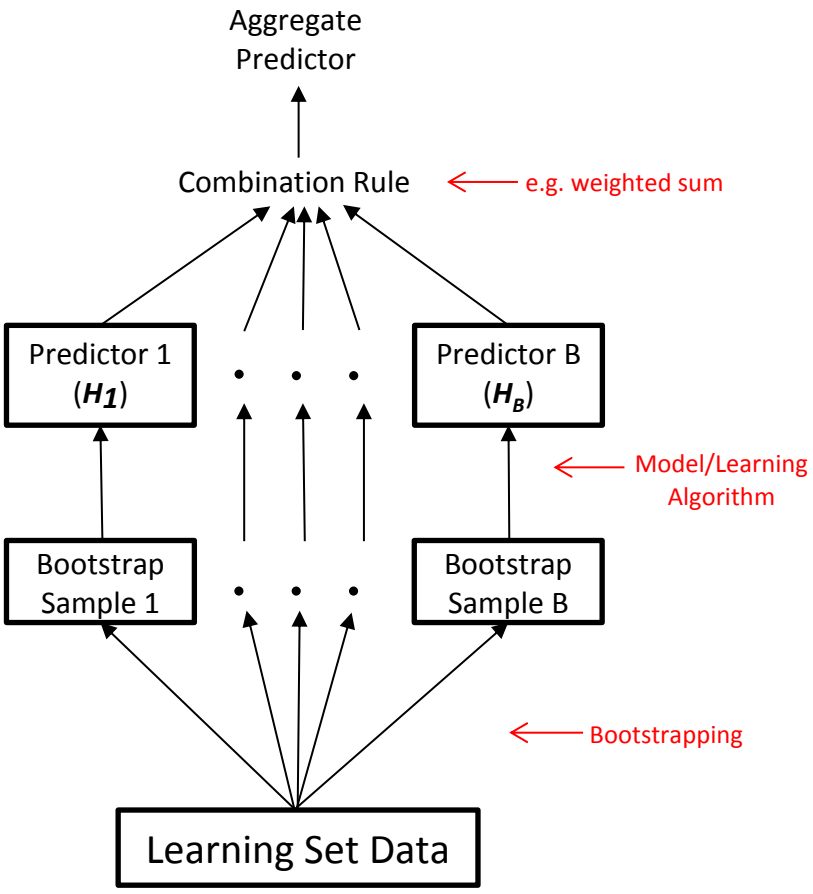


***Figure 5.2**: Schematic illustrating a common procedure for Ensemble Learning. The bootstrapping step could be replaced either by a different resampling technique (e.g. jackknife) or by random partitioning of the learning set into disjoint subsets.*

The procedure begins with generating multiple datasets from the learning/training set. This could be done by resampling (e.g. bootstrapping) or randomly partitioning the learning/training data into a number of disjoint subsets. On each of these generated datasets, the model or learning algorithm is run to construct a predictor $H(\boldsymbol{x})$ which is capable of producing a prediction for any new observation. In Fig. 5.2 the predictors obtained from the generated datasets are labeled H_1 to H_B, where B is the total number of datasets generated from the original learning set. Finally,

these predictors are combined using a combination rule (e.g. averaging, weighted sum, generalized mean) to form an aggregate predictor.

There are a wide variety of ensemble learning procedures but all of them share some common elements (Rokach, 2010) which we list and briefly define below:

Learning set: This is a requisite component of any learning procedure, a dataset (henceforth denoted by $\mathcal{L}$) containing explanatory variables and the outcome/response that we are interested in constructing a predictor for.

Learning algorithm: This is the algorithm that produces predictions. It uses data from $\mathcal{L}$ to 'learn' the relationship(s) between the explanatory variables and the outcome/response variable(s). This model is then used to predict the outcome/response for future cases using just the values of their explanatory variables. For example, in random forests, the learning algorithm is the decision tree algorithm (Breiman *et al.*, 1984; Quinlan, 1986), an algorithm that recursively partitions the learning set into smaller and smaller subsets of increasing homogeneity.

Diversity generation: This is a process that allows the model/learning algorithm to thoroughly explore the input space of the learning set. Different realizations of the data are generated via a variety of techniques. The input space of a typical learning set will have 2 dimensions (i.e. $n \times p$): the observations/cases and the variables. Either (or both) of the dimensions can serve as targets for diversity generation:

Observation Sampling: This involves taking random realizations of the learning set observations. Bootstrapping of the learning set is one form of diversity generation which focuses on resampling of observations/cases in the learning set. This produces a new dataset that is distinct from the original one even though all the *unique* data in the new dataset comes from the

original. An alternative to bootstrapping is randomly partitioning the learning set into a number of disjoint, independent subsets and generating a predictor using each one.

Variable Sampling: Another form of diversity generation involves targeting variables for random selection. In particular, the Random Subspace Method (RSM) introduced in Ho (1998) involves training each predictor on a randomly selected subset of the variables, i.e. if the number of variables is p, then m of these ($m < p$) are randomly selected and used to construct each predictor. This random selection process is repeated for each predictor and the size of the subset m is often fixed for all predictors. Therefore each individual predictor is constructed from some randomly selected m-dimensional subspace of the full variable space. The random forests algorithm uses this technique to generate a diverse set of trees in order to improve the ability of the resulting forest to 'generalize', i.e. to produce accurate predictions for new observations that bear little resemblance or correlation to those in the training set. The use of different subsets of variables to construct predictors decreases the likelihood of overfitting to the learning set and produces an aggregate predictor with higher generalization accuracy. The random subspace method has also been shown to be effective for other modelling/learning techniques, e.g. multiple linear regression (Tan, Li & Qin, 2008; Mielniczuk & Teisseyre, 2014), generalized linear models (Song, Langfelder & Horvath, 2013), multinomial logit models (Prinzie & den Poel, 2008), and Linear Discriminant Analysis (Skurichina & Duin, 2002).

Predictor aggregation: As the term implies, this process involves combing the output of each predictor into a final prediction which ideally would be more accurate than those of any of the individual predictors. The most popular aggregation method is bagging. In the next section, we explore the following two additional methods of aggregating predictors, each of which may provide superior performance to regular bagging:

- Weighted bagging
- Stacked generalization

We will also use the random subspace method alone and in combination with weighted bagging and stacked generalization and examine the effects on the predictive accuracy of the HSM.

5.2 Methods

5.2.1 Predictor aggregation approaches: Beyond bagging

(1) *Weighted Bagging*

Regular (unweighted) bagging involves averaging a number of predictors, giving equal weighting to each. However it is reasonable to expect that certain predictors would have better accuracy than others. Therefore we propose allowing differential weighting for predictors across bootstrap samples. Following Opitz & Shavlik (1996), we base the weighting on prediction accuracy, with better predictors assigned higher weights and thus contributing more to the aggregate predictor. For the HSM, this weighted aggregate predictor would have a general form given by:

$$H_{agg}(\boldsymbol{x}) = \sum_{b=1}^{B} v_b H_b(\boldsymbol{x})$$

Here, v_b is the 'importance' weight assigned to each predictor. In order to preserve the range of the HSM (0 to 9), we would normalize v_b so that it is constrained within [0,1] and sums to 1. We propose the following 2 ways of defining v_b:

a) Using out-of-bag data: Suppose the learning set $\mathcal{L}$ has a sample size of N. Bagging involves selecting (without replacement) a large number of bootstrap samples of size N from $\mathcal{L}$.

In almost all cases, each bootstrap sample D_b will contain only a subset of the unique datapoints in $\mathcal{L}$ and the rest of D_b will be replicates of the datapoints selected from $\mathcal{L}$. In machine learning terminology, the datapoints not selected to be in the bootstrap sample D_b are collectively referred to as *out-of-bag* (*OOB*) data. It is straightforward to show that for uniform random sampling with replacement, the expected proportion of unique datapoints from $\mathcal{L}$ selected to be in a bootstrap sample of size N will be $1\text{-}e^{-1}$ (≈0.632) for large N. Therefore every bootstrap sample will have an OOB counterpart containing (on average) approximately 36.8% of the data. For each bootstrap sample, the accuracy of the predictor constructed on the sample can be tested using the OOB data which essentially functions as an independent test set for the predictor. We propose using the out-of-bag data to compute the prediction accuracy for each predictor $H_b(\boldsymbol{x})$ and using this to weight it relative to the other predictors. This set-up allows the better-performing predictors to have a greater contribution to the aggregate predictor. Therefore the aggregate predictor would be defined as:

$$H_{agg}(\boldsymbol{x}) = \sum_{b=1}^{B} RC_b^{(OOB)} H_b(\boldsymbol{x}) \qquad ----(5.2)$$

$$\text{where}: \; RC_b^{(OOB)} = \frac{C_b^{(OOB)}}{\sum_{g=1}^{B} C_g^{(OOB)}}$$

In Equation 5.2 above, C stands for Harrell's C-statistic. This statistic falls in the range [0, 1], with higher values indicating stronger predictive accuracy. Therefore $1\text{-}C$ is a suitable measure of prediction error for our particular purpose. RC (Rescaled C) is a rescaled version of C, the use of which guarantees that the coefficients of $H_b(\cdot)$ in Equation 5.2 will to sum to 1. This is necessary because the HSM is a score designed to be in the range [0,9], so taking weighted sums

of individual HSM predictors $H_b(\cdot)$ to produce an aggregate HSM requires a set of weights that sum to 1.

b) Using external data: Rather than using OOB data, we propose using an external dataset that is entirely independent of the training set $\mathcal{L}$. Our training set is the NHANES 1999-2002 and the external dataset we use is a randomly selected half of the NHANES III independent cohort. The aggregate predictor here would be given by:

$$H_{agg}(\boldsymbol{x}) = \sum_{b=1}^{B} RC_b^{(Ext)} H_b(\boldsymbol{x}) \qquad ----(5.3)$$

RC_b is defined similarly to (a).

(2) *Stacked Generalization*

The concept of stacked generalization was originally introduced and characterized in Wolpert (1992) and its effectiveness was demonstrated on a neural network. The first documented use of the technique in statistical literature is in Breiman (1993) where it was applied to combining regression trees and ridge regression predictors. Leblanc & Tibshirani (1993) also confirmed the efficacy of this technique.

The previously described techniques (bagging and weighted bagging) combine predictors by unweighted or weighted averaging. In weighted bagging, the weight assigned to each predictor is based on its predictive accuracy. This is an intuitive, simple approach. Stacked generalization is a more sophisticated method which uses a model or learning algorithm to determine the optimal weights to use when combining the predictors. This idea was proposed (albeit in a more general form) by Wolpert (1992).

Note that in this framework, there are now two levels of data and models. At the lower level we have the raw training data and the model(s) used to generate the predictors; these are referred to as *tier-1 data* and *tier-1 model(s)*, respectively. For example, in our particular application, the tier-1 data would be biomarker data from the NHANES and tier-1 model would be the WQS nested in an AFT model (see Equation 5.1) which is used to estimate the weights $\{w_{i(b)}\}$ for each bootstrap predictor $H_b(\cdot)$.

Then the B predictors $\{H_b(\cdot)\}_{b=1}^{B}$ generated by the *tier-1* data and model will be treated as variables in a model/learning algorithm that estimates the optimal weighting parameters to use to combine them. This model is referred to as the *tier-2 model* or *meta-model* and the B predictors $\{H_b(\cdot)\}_{b=1}^{B}$ are considered *tier-2 data* for this model. Figure 5.3 below illustrates this idea in graphic form:

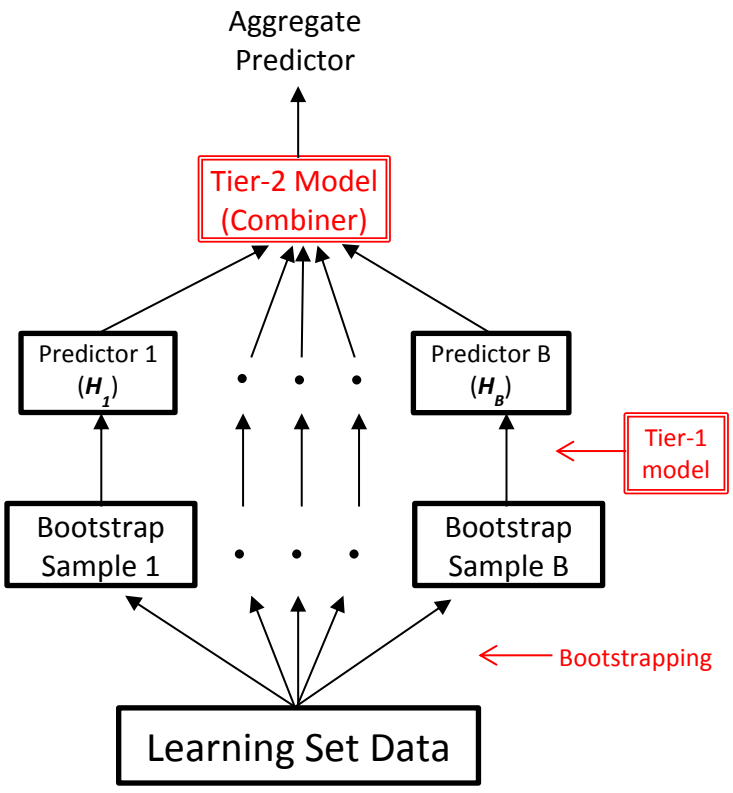


***Figure 5.3**: Schematic illustrating the stacked generalization procedure*

The end-result of stacked generalization would typically be an aggregate predictor that is a simple linear combination of the individual predictors generated from a tier-1 model:

$$H_{st}(\boldsymbol{x}) = \sum_{b=1}^{B} \eta_b H_b(\boldsymbol{x}) \qquad ----(5.4)$$

Here, the $\{\eta_b\}$ are unknown coefficients that are estimated by the tier-2 model; the estimates are the values which best relate the tier-2 variables $\{ H_b(\boldsymbol{x}) \}_{b=1}^{B}$ to the targeted outcome/response. In our particular application, our outcome is survival so we would use the following tier-2 survival model:

$$\log T_k = \sum_{b=1}^{B} \eta_b H_b(\boldsymbol{x}_k) + \varepsilon \qquad ----(5.5)$$

Here, T_k is the censored/uncensored survival time for individual *k*, while $\boldsymbol{x_k}$ is the observed biomarker data for individual *k* in the learning set $\mathcal{L}$. And $H_b(\boldsymbol{x_k})$ is the prediction of predictor $H_b(\cdot)$ for individual *k*. Therefore the model defined in Equation (5.5) implies that predictors whose predictions for individuals have a strong relationship to mortality will have larger η estimates. Therefore they will contribute more to the aggregate predictor defined in Equation (5.4).

Note that the (tier-2) data for the tier-2 model defined in Equation 5.5 is an $N \times B$ matrix where *N* is the sample size (total number of individuals) and *B* is the number of bootstrap samples and hence predictors. Stated formally, the tier-2 data is given by:

$$\left\{ T_k, \delta_k, \left\{ H_b\left(\boldsymbol{x}_k\right) \right\}_{b=1}^{B} \right\}_{1 \le k \le N}$$

Here, δ_k is the censoring indicator for individual *k* and $\{ H_b(\boldsymbol{x}_k) \}_{b=1}^{B}$ is the set of all predictions for individual *k* based on *k*'s vector of explanatory variables $\boldsymbol{x}_k$.

The simple model outlined in Equation 5.5 is merely for instructive purposes and is actually a somewhat naïve formulation of stacked generalization. In practice, the model as defined in its current form will be ineffective for a number of reasons. First of all, each of the predictors $H_b(\cdot)$ was constructed using data from the learning set $\mathcal{L}$. Re-using $\mathcal{L}$ as a learning set for the tier-2 model will lead to overfitting. We tackle this problem by using an independent dataset for fitting the tier-2 model. The next issue is that since the predictors $H_b(\cdot)$ were all constructed from bootstrap samples obtained from $\mathcal{L}$, it is reasonable to expect significant correlation among them. Basic regression models such as the one used in Equation (5.5) generally handle multicollinearity poorly. There are a number of well-known modelling techniques for handling multicollinearity but in our studies we will focus on two that have particularly useful features for the current application:

a) WQS-based model: The Weighted Quantile Sum framework was developed to handle multicollinearity in environmental chemical mixtures and has demonstrated superior performance in variable selection applications (Carrico, 2013; Christensen *et al.*, 2013; Gennings *et al.*, 2013). The key feature that makes WQS attractive for our particular application is the nonnegativity constraint on the weight parameters. Nonnegative weights are desirable for combining learning-based predictors for obvious reasons (see Equation 5.4). Also, regression regularized with the nonnegativity constraint has been shown to handle multicollinearity very effectively. For example this constraint was used in Breiman (1993 & 1996a) to combine regression tree predictors and it was shown to demonstrate superior predictive performance to

ridge regression, another type of regularization. Further, Leblanc & Tibshirani (1993) demonstrated its superior performance over other regularization methods traditionally used to accommodate multicollinearity. The WQS approach also incorporates a unit-sum constraint, i.e. the weights are constrained to sum to 1. Below we define the WQS-based tier-2 model we will use for aggregating the predictors:

$$\log T_k = \beta_1 \sum_{b=1}^{B} \eta_b H_b(\boldsymbol{x}_k) + \varepsilon \qquad ----(5.6)$$

$$\left[\textit{where } \eta_b \in [0,1], \quad \sum_{b=1}^{B} \eta_b = 1 \right]$$

Fitting this WQS model produces estimates of the $\{\eta_b\}$. Because of the nonnegativity and unit-sum constraints the WQS model imposes on the $\{\eta_b\}$, we end up with a ready-to-use set of combination weights that produce an aggregate HSM predictor whose range is the expected [0, 9]:

$$H_{st}^{(WQS)}(\boldsymbol{x}) = \sum_{b=1}^{B} \hat{\eta}_b H_b(\boldsymbol{x}) \qquad ----(5.7)$$

$$\left[\textit{with } \sum_{b=1}^{B} \hat{\eta}_b = 1 \right]$$

As discussed in an earlier chapter, WQS-based models are fit using nonlinear optimization routines most of which cannot handle optimizations with a very large number of variables (e.g. on the order of several 100s or 1000s). Studies utilizing the WQS technique in the past have focused on small-to-moderate sized models (e.g. on the order of dozens of variables) and the technique has not been applied to 'large *p*' data. Recall that our tier-2 data (used in the model in Equation 5.6) will have dimensions *N* x *B* where *B* (=1000) is the number of predictors generated by the tier-1 model and data. Therefore the tier-2 model will be a 1000-variable model which would have to be fit using nonlinear optimization routines. Our initial attempts to use a particular

numerical nonlinear optimization algorithm (the Conjugate Gradient optimization technique) for this 1000-variable model were not successful. So instead, we used bootstrap aggregation in conjunction with the random subspace method wherein only a randomly selected subset of the 1000 tier-2 variables ($\sqrt{1000} \approx 32$) was used at each bootstrap step. An optimization problem with 32 variables is well within the capabilities of most of the nonlinear optimization techniques used for the WQS technique. Therefore we chose this method for implementing WQS-based stacking.

b) Random Forest-based model: The Random Forests learning algorithm is a non-parametric technique that has found extensive use in both machine learning and statistical applications. It is well-known for its ability to effectively handle high-dimensional data with complex correlation patterns among the variables. The *B* variables in our tier 2 data have a high degree of multicollinearity so the random forests algorithm is an ideal choice for the tier-2 model. Because the response in this case is time-to-event data, we use Random Survival Forests introduced in Ishwaran *et al.* (2008). One useful output of most random forests implementations is a set of 'importance' measures for each variable. As the name implies, these variable importance (VIMP) measures quantify the importance of each variable in the overall random forest. Specifically, the VIMP for a variable is a measure of that variable's contribution to the predictive accuracy of the forest. Using Random Survival Forests as our tier-2 model, we will end up with a VIMP for each predictor $H_b(\cdot)$ in the model. We propose using these VIMPs as combination weights for the aggregate predictor. Unlike the WQS weights, the VIMPs do not, by default, sum to 1 so we use a rescaled VIMP (denoted ***RVIMP*** below) that sums to 1 across all the predictors:

$$H_{st}^{RSF}(\boldsymbol{x}) = \sum_{b=1}^{B} RVIMP_b H_b(\boldsymbol{x}) \qquad ----(5.8)$$

To our knowledge, this is the first time this particular approach has been used for stacked generalization.

5.2.2 Random Subspace Method

This method was discussed in an earlier section. In our studies, we will use the random subspace method in conjunction with the various aggregation techniques discussed above. The common step in each of these aggregation techniques is the first step: generating a large number (*B*) of bootstrap samples from the learning set $\mathcal{L}$ and creating a predictor $H_b(\cdot)$ from each sample *b*. The predictor can be constructed using all the 24 biomarker variables or by using a randomly selected subset of the variables (i.e. the random subspace method). We will carry out both approaches for each aggregation technique and compare the results. In the random subspace method, the size of the subset of variables (i.e. the subspace dimensionality *m*) randomly selected at each bootstrap step is fixed for all steps. Choosing the right value for the parameter *m* is important since it influences the method's efficacy, however there are no set-in-stone rules for doing so. In the original paper introducing the random subspace method (Ho, 1998), it was stated that using $m \approx p/2$ yields the best results, however it was suggested in Breiman (2001) that $m \approx \sqrt{p}$ is the optimal setting. This particular setting is a common default for most software implementations of the random forests algorithm. In our studies, we tried different values of *m* between *p*/2 (=12) and $\sqrt{p}$ (≈5). The following values of *m* were used: 5, 6, 9, 12.

5.2.3 Datasets

In our studies, we used 3 datasets. The bootstrap samples used to train each base predictor $H_b(\cdot)$ were obtained from the NHANES 1999-2002 (n=3406) biomarker/survival dataset. Then the base predictors were combined using the 5 aggregation techniques we have discussed. The stacked regression techniques required a separate dataset (an external dataset) for training the tier-2 models for combining the set of predictors $\{ H_b(\cdot) \}_{b=1}^{B}$ generated in the first step. One of the 'weighted bagging' techniques also required an external dataset to produce estimates of the predictive accuracy of the predictors. The external dataset used for these techniques will be referred to as the tier-2 learning set and was obtained by randomly sampling half of the NHANES III biomarker/survival dataset. Therefore the tier-2 learning set had a sample size of n=5792. The other half of the NHANES III biomarker/survival dataset was used as a validation set (n=5801) to compare the final aggregated HSMs produced by the 5 aggregation techniques. In summary, the 3 datasets we used were a tier-1 learning set (n=3406), a tier-2 learning set (n=5792), and a validation set (n=5801).

5.3 Results

We used *B*=1000 bootstrap samples from the tier-1 learning set to create multiple sets of *B* predictors $\{ H_b(\cdot) \}_{b=1}^{B}$ using the WQS model given in Equation (5.1). Each set of predictors was generated either by using all available biomarker variables (24) or by using only a subset (of fixed size *m*) of the variables (i.e. the random subspace method). We tried different subset sizes *m*: 5, 6, 9 and 12. Therefore a separate set of predictors $\{ H_b(\cdot) \}_{b=1}^{B}$ was generated for each value of *m*. For each set of predictors, each of the aggregation techniques was used to combine them to form an aggregate predictor. The following 5 aggregation techniques were used and the predictive accuracies of the resulting aggregate predictors were compared:

- Regular (unweighted) Bagging: This is the technique used for constructing the original HSM whose predictive accuracy we seek to improve on.
- Weighted bagging using:
    - Out-of-bag data
    - External data
- Stacked generalization using:
    - WQS (Weighted Quantile Sum) Regression
    - Random Survival Forests

The results are summarized in Table 5.2 below. As a reminder, the Harrell's C estimates and AUCs for **all** the various aggregation techniques are computed based on the performance of their corresponding aggregate predictors in the validation dataset (n=5801) described in the previous section.

Table 5.2: Harrell's C and AUC for Aggregation Techniques

Aggregation Technique	Variable space dimensionality (*m*)	Harrell's C	$AUC_{5\text{-year}}$
Regular Bagging	Full	0.7094	0.7565
	12	0.7085	0.7571
	9	0.7092	0.7579
	6	0.7131*	0.7619*
	5	0.7106	0.7595
Weighted Bagging (OOB)	Full	0.7101*	0.7571*
	12	0.7115*	0.7592*
	9	0.7132*	0.7607*
	6	0.7184*	0.7656*
	5	0.7167*	0.7639*
Weighted Bagging (External)	Full	0.7102*	0.7571*
	12	0.7117*	0.7596*
	9	0.7135*	0.7611*
	6	0.7190*	0.7664*
	5	0.7171*	0.7645*
Stacking (RSF)	Full	0.7174*	0.7629*
	12	0.7336*	0.7775*
	9	0.7348*	0.7781*
	6	0.7470*	0.7885*
	5	0.7497*	0.7906*
Stacking (WQS)	Full	0.7367*	0.7784*
	12	0.7496*	0.7875*
	9	0.7519*	0.7889*
	6	0.7564*	0.7911*
	5	0.7577*	0.7916*

Highlighted top row indicates results for original HSM

*Predictive measure (Harrell's C or AUC) is statistically significantly better than that of original HSM

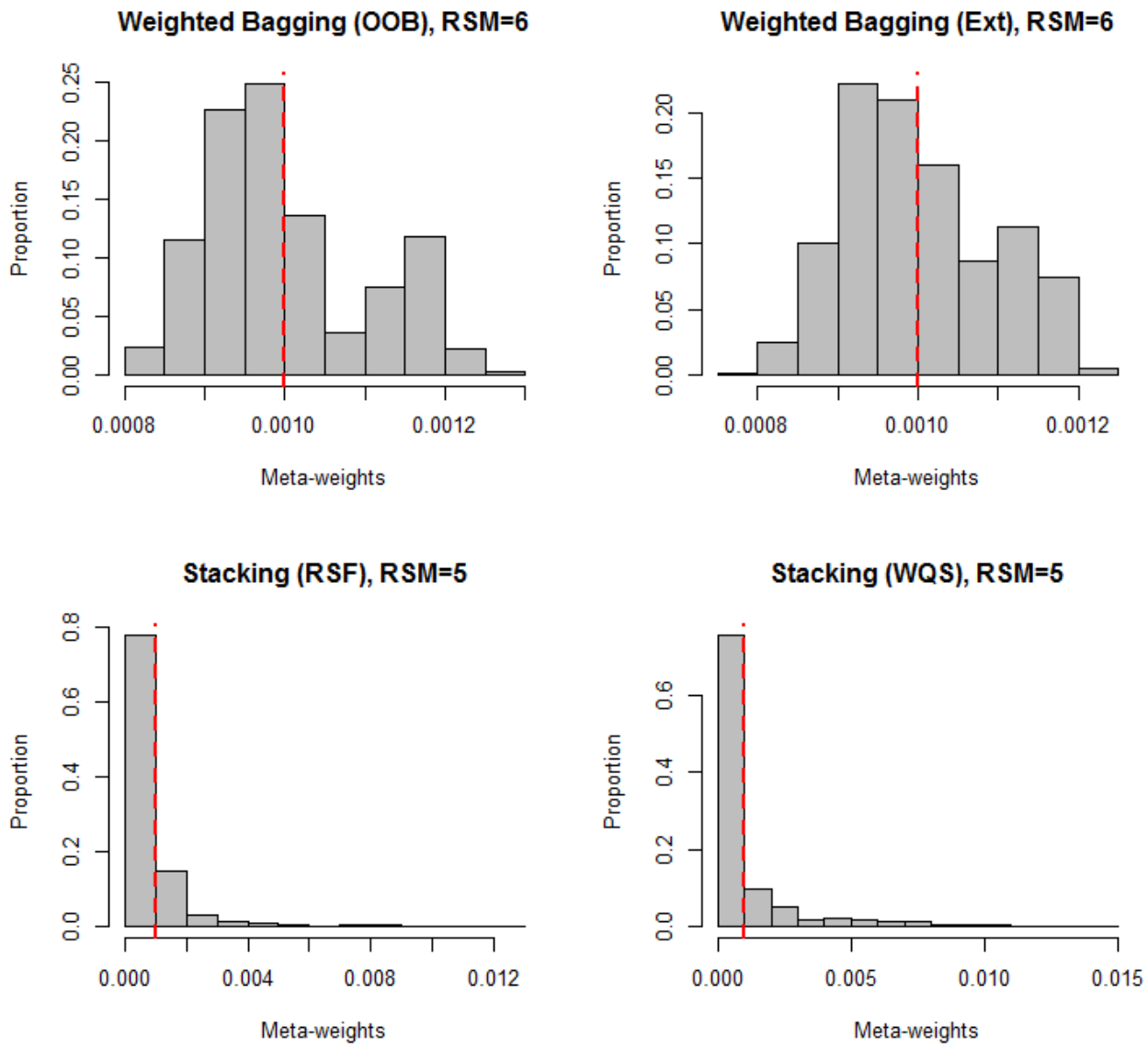


Figure 5.4: Meta-weight distributions for weighted bagging and stacked generalization

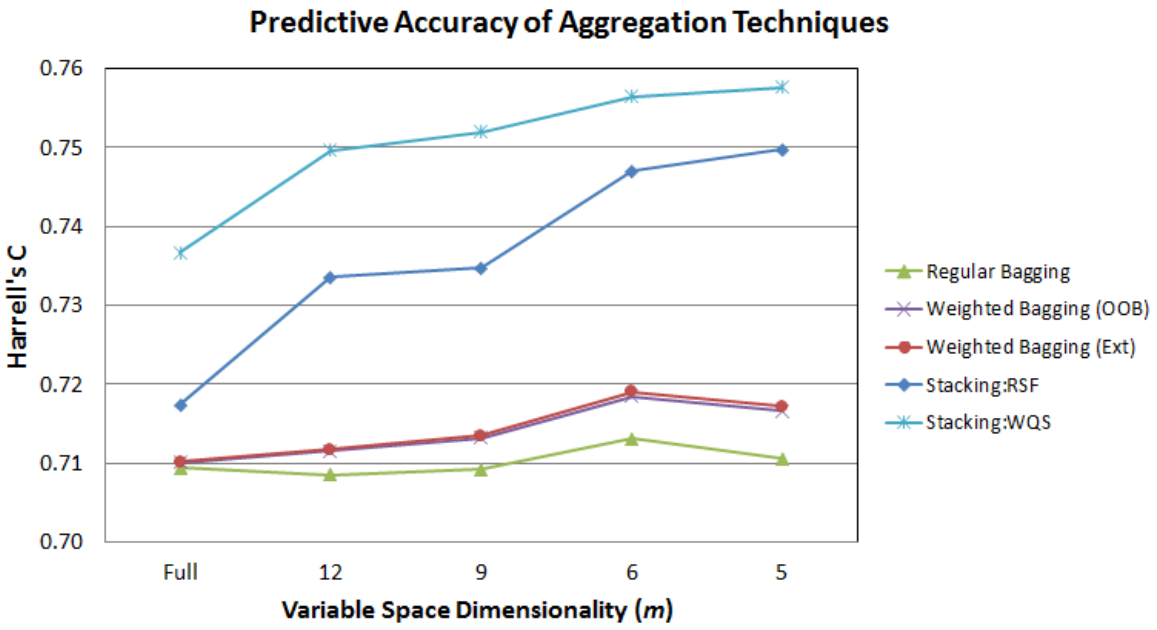


Figure 5.5: Variation in Harrell's C over different variable spaces

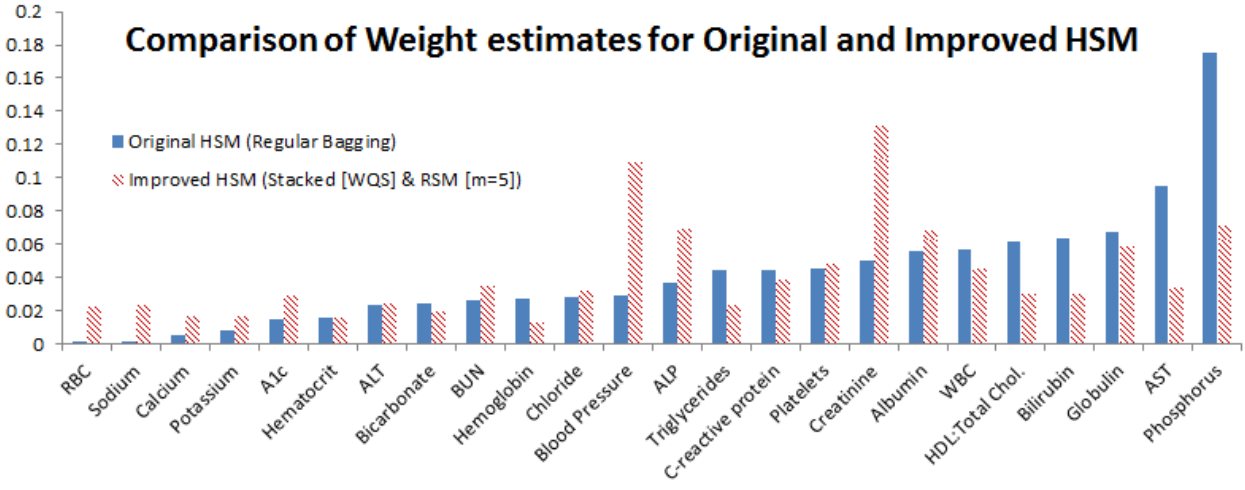


Figure 5.6: Comparison of biomarker weights between original HSM and stacking-enhanced HSM

Table 5.2 summarizes the performance of each aggregation technique. The first entry in the table (highlighted) gives the Harrell's C and AUC for the HSM in its original form. Recall that this form of the HSM is computed through bagging (which in this write-up we refer to as *regular*

bagging) and using all biomarker variables (i.e. the random subspace method is not used). In Table 5.2, the predictive accuracy of this original version of the HSM is used as the benchmark to which the performance of HSMs constructed via other techniques will be compared.

The first group of results in Table 5.2 pertain to the predictive accuracy of HSMs computed using *regular bagging*, i.e. unweighted averaging of predictions across bootstrap samples. The random subspace method was used, however only the m=6 case produced significantly higher predictive accuracy than the original HSM.

Next, we focus on the two weighted bagging techniques. Table 5.2 indicates that both weighted bagging techniques give results which are uniformly superior to those obtained via regular bagging (the asterisks indicate statistically significant superiority). Another interesting thing we observe is the similarity between the Harrell's C and AUC estimates of these 2 weighted bagging techniques. Figure 5.5 shows the variation in Harrell's C over decreasing values of m (variable space dimensionality); we see the plots for both techniques nearly overlap. Examining the effect of different values of m (the variable subspace dimensionality) on the predictive accuracy of the weighted bagging techniques, we see a nearly consistent increasing trend in predictive accuracy as m is decreased from 24 (the full set of biomarkers) to 5. The plot in Figure 5.5 also confirms this.

Next, we go into some detail regarding the *weighted bagging* techniques. The top row of Figure 5.4 shows the distribution of weights assigned to the B predictors by the 2 weighted bagging techniques. To distinguish these weights from the weights assigned to each biomarker in the tier-1 model, we refer to them as *meta-weights*. The red dashed line in each histogram indicates the weight ($1/B$) assigned to each predictor by unweighted bagging (i.e. taking the simple average of all B predictors). Since B=1000, a value of 0.001 would be assigned to all B

predictors under unweighted bagging. The histograms for the two weighted bagging techniques show that the distribution of meta-weights around the mean 0.001 is narrow (~ ±.0002, SD=10^{-4}). Since the meta-weight assigned to each predictor (by each weighted bagging technique) is proportional to its predictive accuracy in the out-of-bag or external dataset, the narrow distribution of meta-weights indicates that there does not exist a wide variation in the predictive accuracies of the predictors $\{ H_b(\cdot) \}_{b=1}^{B}$.

Next, we examine the performance of the stacked generalization-based techniques. Recall that we used RSF and the WQS regression as models to determine the optimal set of meta-weights to use to combine the *B* predictors into an aggregate predictor. The results summarized in Table 5.2 indicate that these techniques perform better than the weighted bagging techniques and produce aggregate predictors whose Harrell's C and AUC estimates are significantly higher than those of the original HSM. The HSM constructed by WQS-based stacking combined with random subspace method (with m=5) produces the best prediction as measured by AUC (AUC =0.792) and also the highest predictive accuracy as measured by Harrell's C (C=0.758). Contrasting these numbers to the corresponding values for the original HSM in Table 5.2 (AUC =0.757, C=0.709), we see significant improvements.

One clue to the superior performance of the two stacked generalization techniques can be found by comparing the meta-weight histograms in Figure 5.4. As discussed earlier, the meta-weights assigned to the bootstrap-based predictors by the two weighted bagging techniques are fairly narrowly distributed about the mean (1/*B*). This results in aggregate predictors that are only slightly better (albeit statistically significantly so) than those obtained from unweighted bagging which gives an equal weight of 1/*B* to each predictor. However the meta-weight distributions for the stacking-based techniques are highly skewed with long tails, with the highest

meta-weights being ~10 orders of magnitude higher than the mean 1/*B*. For both stacking-based techniques, between 75% and 80% of the meta-weights fall below the mean. It is clear that both techniques assign disproportionately high meta-weights to a small percentage of predictors, while the majority are down-weighted. What is unclear is whether this is a reflection of the true predictive power of each predictor or a mere byproduct of the way these techniques handle the high correlation among the predictors.

Examining the variation in Harrell's C and AUC estimates over different values of *m*, we see that there is a consistent increase in prediction accuracy as the variable space dimensionality is decreased from 24 (all available biomarkers) to 5. This corroborates observations from studies in the methodology and application of random forests (which utilize the random subspace method) that indicate that the optimal value of *m* is close to $\sqrt{p}$. Our results suggest that both stacking-based techniques combined with the random subspace method (with $m = \sqrt{24} \approx 5$) give the best improvement on the predictive accuracy of the HSM.

We will henceforth use the HSM derived by WQS-based stacking. From Equation (5.7), the general expression for this aggregate predictor is given by:

$$H_{st}^{(WQS)}(\boldsymbol{x}) = \sum_{b=1}^{B} \hat{\eta}_b H_b(\boldsymbol{x}) \qquad \left[\text{with } \sum_{b=1}^{B} \hat{\eta}_b = 1 \right]$$

Let S_b be the particular subset of biomarker variables (5 in number) selected in bootstrap sample *b* and used to construct predictor $H_b(\cdot)$. Let R_i be the set of bootstrap samples in which biomarker variable x_i was one of the 5 randomly selected variables. Then the following derivation can be carried out:

$$H_{st}^{(WQS)}(\boldsymbol{x}) = \sum_{b=1}^{B} \hat{\eta}_b H_b(\boldsymbol{x}) \qquad \left[with \sum_{b=1}^{B} \hat{\eta}_b = 1 \right]$$

$$= \sum_{b=1}^{B} \hat{\eta}_b \sum_{i \in S_b} \hat{w}_{i(b)} x_i$$

$$= \sum_{b=1}^{B} \sum_{i \in S_b} \hat{\eta}_b \, \hat{w}_{i(b)} x_i$$

$$= \sum_{i=1}^{p} \sum_{b \in R_i} \hat{\eta}_b \, \hat{w}_{i(b)} x_i$$

$$= \sum_{i=1}^{p} \left(\sum_{b \in R_i} \hat{\eta}_b \, \hat{w}_{i(b)} \right) x_i$$

$$\therefore \; \bar{w}_i^{st} = \sum_{b \in R_i} \hat{\eta}_b \, \hat{w}_{i(b)}$$

In the above derivation, the final expression gives the new set of biomarker weights (derived from WQS-based stacking with random subspace method [m=5]) corresponding to each biomarker x_i. These weights have shown superior predictive accuracy to the bootstrap-averaged weights used in the original HSM, therefore we will adopt the former for future use.

As an interesting exercise, we compared the new set of biomarker weights to the original bootstrap-averaged weights (in the old HSM) and the results are summarized in Figure 5.6 above. We can see that in the original HSM, Phosphorus and AST (Aspartate aminotransferase) were assigned the highest weights but in the new/improved HSM, each of these has been down-weighted and Creatinine and Blood pressure are now the highest-weighted biomarkers.

5.4 Discussion

We have explored the effectiveness of a number of heuristic approaches for combining predictors. Each is based on constructing predictors with multiple realizations of the data and then combining them into an aggregate predictor. We started by using a simple modification of the standard bootstrap aggregation (also known as bagging) that involves weighting each predictor based on predictive accuracy in the out-of-bag or external data rather than merely averaging them. Using Harrell's C as a measure of predictive accuracy, we observe that aggregate predictors produced by weighted bagging are significantly better than the original HSM on predictive accuracy as quantified by Harrell's C and AUC. Further investigation showed that the predictive accuracy of each of the individual predictors (for out-of-bag or external data) did not vary greatly; therefore the range of meta-weights attached to each predictor when combining them to form an aggregate predictor was relatively narrow and almost evenly distributed around the mean. Also, the choice of using out-of-bag data or an external and independent dataset did not significantly impact the results.

Our studies also indicate that the random subspace method generally gives better results on predictive accuracy than using all available biomarker variables. For the different techniques we used, the predictive accuracy was generally higher for lower variable space dimensionality *m*, with optimal values of *m* close to the square root of *p* (the total number of biomarker variables). When all available biomarker variables are used to fit the WQS model for each bootstrap sample, it seems that one particular biomarker (Phosphorus) dominates and is assigned a relatively large proportion of the total weight (see Figure 5.1). In fact, for a majority of the *B* bootstrap samples generated from the learning set, Phosphorus tends to have the highest weight in the predictor $H_b(\cdot)$ constructed from each sample. This creates a set $\{ H_b(\cdot) \}_{b=1}^{B}$ of fairly similar predictors.

However, using only a small, randomly-chosen subset of the available biomarker variables at each bootstrap step seems to dampen the dominant effect of Phosphorus by allowing several predictors to be constructed that exclude Phosphorus and use 'weaker' variables. This creates a more diverse set of predictors $\{ H_b(\cdot) \}_{b=1}^{B}$. This particular effect of the random subspace method is the reason it is used in random forests. It is used as a way of injecting more randomness into forest construction by allowing only a subset of variables to be considered when splitting each node. This reduces the effect of highly dominant variables that tend to exert disproportionate influence on the structure of the tree and it results in a more diverse set of trees that are not too highly correlated. The use of trees with a wide variety of structures prevents overfitting and improves the forest's generalization accuracy (Breiman, 2001), which is a measure of the ability of a predictor to correctly predict outcomes for novel cases the kind of which are not encountered in the training set sample. The use of the random subspace method to generate more diverse ensembles is an idea that has been applied to other types of models/learning algorithms, e.g. multiple linear regression, generalized linear models, multinomial logit models and linear discriminant analysis (see Section 5.1.3 for references). These studies demonstrated the benefits of this technique over using the full variable space. Here we have demonstrated that randomized variable selection also enhances predictive accuracy for WQS regression.

While the improvements in HSM predictive accuracy obtained from weighted bagging were statistically significant but small, we found that combining the ensemble of predictors using stacked regression produces better results. Both techniques used for stacked generalization (random survival forests and WQS) yielded aggregate predictors with significantly higher predictive accuracy (as measured by Harrell's C and AUC) than the original HSM. But stacking via WQS produced (nominally) better results on Harrell's C and $AUC_{5\text{-year}}$ than by using random

survival forests. The superior performance of stacked generalization using WQS in fact confirms what has been observed in some of the earliest studies on stacked regression (Leblanc & Tibshirani, 1993; Breiman, 1996a) which showed that stacking predictors using regression-type models with nonnegativity constraints on the meta-weight parameters outperforms other regression techniques (sometimes by a large margin). While the tier-2 models used for stacked generalization in Breiman (1996a) and Leblanc & Tibshirani (1993) were not identical to the WQS model, they share the same key feature: the nonnegativity constraints on the parameters. In addition, WQS imposes a unit-sum constraint however the simulation studies carried out in Breiman (1996a) indicate that applying the unit-sum constraint (in addition to the nonnegativity constraints) does not produce any further reduction in prediction error. To gain some insight into why nonnegativity constraints works so well, we refer the interested reader to Breiman (1996a).

Chapter 6

Application of HSM to External Clinical Dataset

6.1 Introduction

The past chapters have been devoted to the description and extension of the HSM. This score was developed for use as a clinical tool for assessing a significant component of overall physiological health status. The HSM was tested, validated and extended using NHANES data (cohorts 1988-1994 and 1999-2008). It is of interest to test this risk score on clinical data in order to assess its performance and demonstrate its efficacy in practical settings.

To this end, we obtained inpatient and outpatient longitudinal data from the Virginia Commonwealth University Medical Center. Patients visiting the Emergency Department (ED) within a specific time window (January 1 to February 28, 2011) were followed for 2 years subsequent to the ED visit. With this data our goal was to demonstrate a particular application of the HSM: as a clinical tool for predicting the long-term prognosis of patients admitted to the ED based on their biomarker measurements at the time. We used multiple measures of long-term prognosis which will be described in subsequent sections.

Another goal of this effort was to test the feasibility of HSM computation with real-world data. We believe the NHANES data used in previous chapters is of higher quality than most health/clinical biomarker data one would encounter in practical situations. Because the NHANES data were collected as part of a large and comprehensive national health survey, it is extensively curated and as a result has a lower proportion of missing values and data errors than typical clinical data. The VCU Medical Center data contains a significantly higher proportion of missing

data than the NHANES data and in the following sections, we will discuss how this issue was tackled through the use of techniques developed in previous chapters.

6.2 Methods

6.2.1 Data Structure

As discussed earlier, patients with an ED visit in Jan-Feb 2011 were followed for 2 years after that ED visit (which is considered the baseline). For all patients, every subsequent inpatient, outpatient or ED visit in that 2-year window was recorded. For every visit, the following information was recorded: admission/discharge dates, type of visit (e.g. ED, inpatient), discharge status/disposition, age at time of visit, gender, race, BMI, and measurements of the 24 biomarkers which we use to construct the HSM. After cleaning the data, there were n=2189 patients with a total of 26,452 records. Each record corresponds to a database entry of one or more lab test results carried out for a patient during a particular visit.

6.2.2 Missing values

Calculation of the HSM requires the availability of measurements for all 24 biomarkers. The dataset used in this study had a high proportion of biomarkers missing for each patient at the baseline ED visit. To replace missing values, we used *k*-nearest neighbors imputation which is described in detail in Chapter 4. We could only carry out imputations for records missing at most 23 (out of 24) biomarkers. The donor set used was a complete subset of the NHANES III (1988-1994) cohort which consisted of n=10,000 complete records. Age, Gender and BMI (when available) were used as secondary matching variables for the *k*-nearest neighbors algorithm (see Chapter 4).

6.2.3 Updated HSM

In Chapter 5, we developed and compared ensemble methods for improving the predictive accuracy of the HSM. We found that WQS-based stacking (see Chapter 5) provides the most significant improvement in the predictive accuracy of the HSM. Therefore in the present study, we will use the HSM constructed using WQS-based stacking as opposed to the original HSM used and referenced in previous chapters (Chapters 2-4).

6.2.4 Analysis

The primary goal of our analysis was to determine if the HSM of a patient at their baseline ED visit predicts future hospital utilization. We focused on hospital utilization in the 2 years following the baseline ED visit. Hospital utilization was quantified by the number and total duration of hospital visits occurring after the baseline ED visit. We modeled the count of number of subsequent visits and total duration (in days) of visits using HSM, age, gender and race as explanatory variables. We only included patients who had computable HSMs at their baseline ED visit, i.e. patients with at least 1 available biomarker measurement.

6.3 Results

All study subjects had at least one ED visit in the period of January 1 to February 28, 2011. The first ED visit within this period was taken as the baseline ED visit. Below is a table of the discharge dispositions for the 2189 patients at their baseline ED visit:

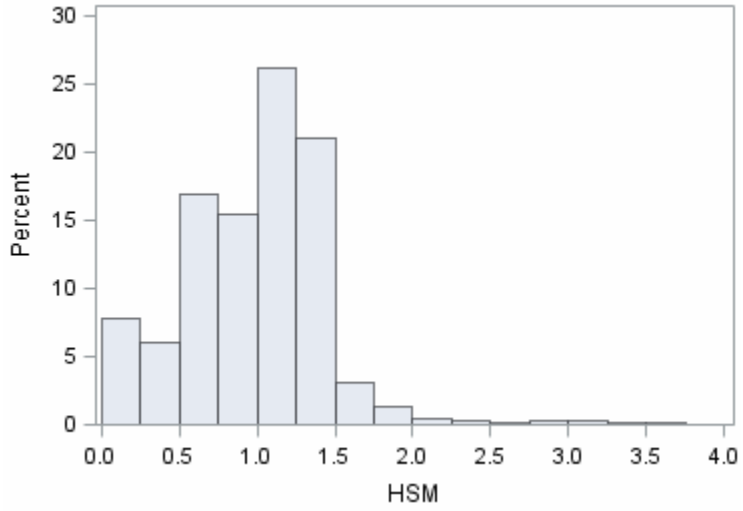
Table 6.1: Discharge Dispositions of patients at baseline Emergency Department visit

Discharge Disposition	Percent
Against medical advice-AMA	10.6
Correctional Facility	0.2
Home/Prior Living Arrangement	89.1
Redirected to L&D-Admin D/C	0.05
Transferred within VCUHS-Admin D/C	0.05

As the table above indicates, the vast majority of patients were discharged home or to a prior living arrangement.

A small percentage of the n=2189 patients had no available biomarkers at the baseline ED visit, so imputation could not be carried out for such patients. Therefore the analytic dataset consisted of just 2119 patients (~97% of original dataset). As discussed earlier, the *k*-nearest neighbor imputation technique was used to replace missing biomarker values at the baseline ED visit for these patients. Figure 6.1 shows the distribution of computed HSM scores in the analytic dataset.

Figure 6.1: Distribution of HSM (at baseline ED visit) in analytic dataset



There is a steep drop-off in frequency beyond HSM ≈ 1.5, indicating that only a small percentage of patients had relatively high HSM scores. Table 6.2 below provides summary statistics for the HSM scores at the baseline ED visit. Table 6.3 summarizes the distribution of age, gender and race in the analytic dataset.

Table 6.2: Summary Statistics for HSM at baseline ED visit

Mean	Q1	Median	Q3	Minimum	Maximum
1.01	0.72	1.11	1.26	0.11	3.97

Table 6.3: Demographic summary for analytical dataset

		Frequency	Percent
Age	18-35	1063	50.17
	36-49	421	19.87
	>=50	635	29.97
Gender	Male	902	42.57
	Female	1217	57.43
Race	White	545	25.79
	Black	1533	72.55
	Other	35	1.66

As mentioned in Section 6.2.4 we modeled the number of post-baseline visits and total duration (in days) of these visits using HSM, age, gender and race as explanatory variables. We used Negative Binomial regression models with log link to model the counts. Figure 6.2 shows the distribution of the numbers of subsequent visits across patients in the analytic dataset.

The results indicate that a patient's HSM at baseline is strongly related ($p = 0.009$) to the number of visits subsequent to the baseline ED visit. Specifically, higher HSM at the baseline ED visit is associated with a higher number of subsequent hospital visits.

We also found that HSM at baseline exhibits a strong positive association ($p < .0001$) with the total duration of time (measured in days) spent in the hospital after the baseline ED visit.

Figure 6.2: Distribution of # of visits subsequent to baseline ED visit

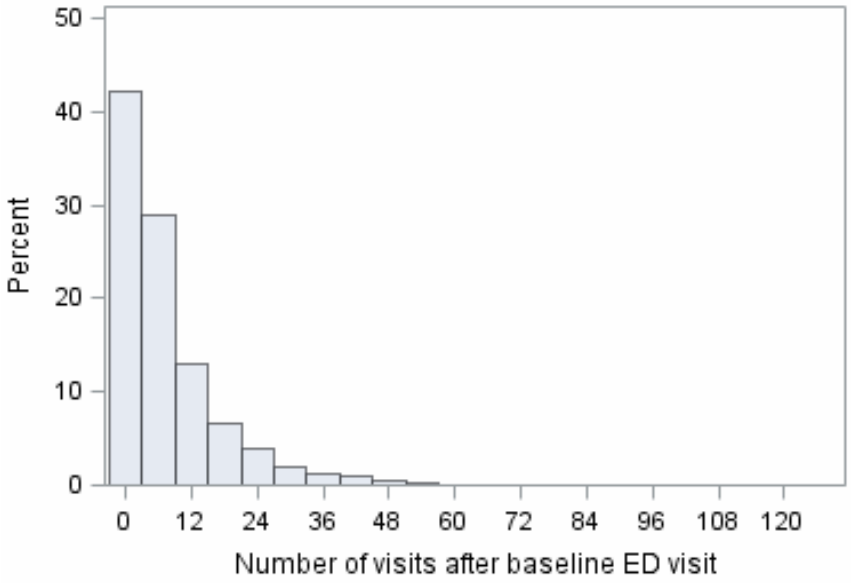


Figure 6.2

6.4 Conclusion

This study demonstrates the use of the HSM in a practical setting. The data we used were culled from the VCU Medical Center Emergency Department. In preparing this dataset for analysis, we encountered challenges specific to real-world data which generally tend not to be as well-curated as the NHANES data we used to develop, extend and test the HSM. As an example, there was a high rate of missing biomarkers in the clinical dataset used in this study, and one of the techniques we tested to handle missing biomarker data (*k*-nearest neighbors) was used to impute the missing values.

The results of the analysis indicate that after adjusting for age, gender and race, the HSM of a patient at their baseline ED visit is strongly related to subsequent hospital utilization. Specifically, higher measured HSM at baseline is associated with greater hospital utilization in

the months subsequent to the ED visit. This agrees with previous findings regarding the relationship between HSM and hospital utilization which are discussed in Chapter 2. From analyzing the NHANES 2003-2008 Questionnaire data (see Table 2.3), we found that significant relationships exist between an individual's current HSM and their self-reported hospital utilization in the past year. Specifically, individuals with higher HSM tended to report receiving more healthcare and having more overnight hospital stays in the months prior to participating in the NHANES. Our results corroborate this finding in a prospective context.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

In this thesis, we have developed a new risk score predictive of all-cause mortality using the Weighted Quantile Sum (WQS) methodology. In the process, we have introduced a few modifications to the WQS methodology that extend its capabilities. In Chapter 2 we introduced a new technique for standardizing variables that remedies a limitation of the standard technique used in the WQS methodology. In Chapter 3, we proposed expanded versions of the WQS model that allow for the inclusion of pairwise interactions among biomarkers and between demographic variables and biomarkers. We discovered that inclusion of interaction effects in the HSM may not in fact be necessary, and that the predictive accuracy is actually reduced as a result of doing so. However while the study did not demonstrate any usefulness to including interactions in the case of the HSM, it introduced simple, useful modifications to the WQS model that extend its applicability. These modifications are general enough that they can be applied to other types of variables (e.g. nutritional values of dietary components, gene expression levels).

In Chapter 5, we introduced a new aggregation approach for the WQS technique that significantly improves the prediction accuracy of the HSM. The approach incorporates the use of the Random Subspace Method with the WQS model in order to generate an ensemble of predictors that are then combined using stacked generalization. We introduced two novel methods for implementing stacked generalization, both of which resulted in aggregate HSM predictors with significantly higher predictive accuracy than originally obtained using the traditional form of the WQS technique.

Many risk scores that have been developed are made available online through web interfaces that can compute these scores for individuals in the general population. These web-based "risk score calculators" are typically simple tools that take values of the required risk score components and run them through a basic algorithm that computes the score. We considered the feasibility of implementing a similar type of web-based tool (or an offline standalone software application for clinicians to use) to compute the HSM for individuals or cohorts of patients. One major issue impacting the feasibility of implementing such a tool is the presence of missing values. In Chapter 4, we addressed this issue by testing and comparing a number of fast, simple and non-iterative imputation techniques that could be used to replace missing values in a software or web application for computing the HSM. The key conclusion in this study was that imputation had a statistically significant negative impact on the predictive accuracy of the HSM. However, among the 3 techniques we tested, we found that the *k*-nearest neighbors imputation technique exhibited superior performance compared to the others. This technique is relatively simple to implement as an algorithm, and does not involve any iterative steps.

In Chapter 6, we applied the HSM to hospital data from the VCU Medical Center Emergency Department (ED). We demonstrated that the HSM could be used as a predictor of hospital utilization after ED encounters, therefore it could serve as a tool for stratifying ED patients by risk.

7.2 Future Work

In Chapter 4, we considered 3 different imputation techniques: median imputation, sub-group median imputation, and *k*-nearest neighbors imputation. These techniques were selected because of their ease of computation; a software application or web interface for computing the HSM could easily carry out any of these imputation techniques in real-time. Other imputation techniques may exist which meet the criteria for ease of computability and future work will focus on exploring such techniques. Their performance will be compared to the original 3 that we tested. These comparisons will be carried out under the less restrictive assumption of *missingness at random* (MAR), which was not done in the present work. Finally, in this chapter, even though we focused on the comparison of imputation techniques that could be used by a software application or web interface to compute the HSM in the presence of missing values, the actual development of such an application does not constitute a part of this thesis and is left to future work.

In Chapter 5, we demonstrated the application of the random subspace method (a machine/statistical learning technique) to WQS regression. We found this improves the predictive accuracy of the HSM, and combining the random subspace method with other ensemble techniques such as bagging or stacking yields further improvements. However the random subspace method can also be of use in more general applications of WQS regression, particularly for modeling very high-dimensional data, i.e. datasets with thousands of variables, such as one would find in genomics studies, for example. As discussed earlier, the WQS technique in its original form is generally not designed to be applied to very high dimensional data due to the limitations of the nonlinear optimization routines used to fit WQS models. In fact, studies utilizing WQS regression have focused primarily on low-dimensional data on the order of

a few dozen variables (typically less than 100). The random subspace method extends the capability of the WQS technique to handle high-dimensional data by allowing focus to be restricted to subsets of the variable space. These subsets are typically small enough that they can be easily handled by the nonlinear optimization algorithms used to fit the WQS model. For example, in Chapter 5, we used WQS regression as a tier-2 model for stacked generalization and the dataset for this model had 1000 variables. The WQS technique would typically not be able to handle a dataset with this many variables, however using the random subspace method, only a small subset of the variables (by convention, `sqrt`[total # of variables]) is chosen for each bootstrap sample. This made the WQS technique feasible for this large dataset and the results showed strong improvement in HSM prediction accuracy. Future work will focus on further applications of this modified WQS technique (which incorporates the random subspace method) to high-dimensional data in different fields. Particular focus will be given to assessing the feasibility and performance of the technique for "large *p* small *n*" problems such as the kind found in genomics studies. This will provide an opportunity to compare this technique to the more well-established techniques for high-dimensional data such as lasso, elastic net, and random forests. In Chapter 3, we introduced an extended version of the WQS that includes pairwise interaction terms among the index components (e.g. biomarkers). With a large number of components, the number of potential interaction terms could be on the order of hundreds or thousands. Integrating the random subspace method into WQS regression would extend its capability to handle such large models. In Chapter 5, the methods explored for improving the predictive accuracy of the HSM were all based on bagging and stacking, two common ensemble techniques. Boosting is another powerful ensemble method that has demonstrated superior predictive performance (Schapire & Freund, 1997). It was originally developed for binary

outcomes and this is its default and most common application. However, recent work (Hothorn *et al.*, 2006; Chen, *et al.*, 2013; Mayr & Schmid, 2014) has focused on extending this algorithm to right-censored survival outcomes. Most of these approaches focus on boosting the concordance index, i.e. using the concordance index (or a smoothed function of it) as an optimization criterion for the boosting algorithm. Future work will focus on applying these implementations of boosting to improving the predictive accuracy of the HSM and comparing the performance to that of the bagging- and stacking-based approaches we used.

The main performance criterion for WQS regression in this thesis has been predictive accuracy (since we are using WQS to construct a risk score). However WQS regression is also used for variable selection problems, especially when complex correlation patterns exist among variables. In such settings, the key performance criterion is variable selection accuracy, i.e. the ability of a model to select the variables with true association with the outcome, and to avoid selecting variables with negligible association. On this criterion, WQS regression has been shown to perform as well as or better than other regularization methods such as ridge regression, lasso and elastic net (Carrico, 2013; Gennings *et. al.*, 2013). In Chapter 5, we demonstrated that incorporating techniques such as stacking and the random subspace method into WQS regression produces significant improvements in prediction accuracy. However the effect of these techniques on variable selection accuracy of WQS regression is currently unknown, therefore this could be a potentially fruitful area to explore in future studies.

The HSM's predictive accuracy for mortality has been evaluated extensively using NHANES data. However it is of interest to assess its performance and robustness in a variety of other datasets. A limitation of this work is that the HSM was only validated using one other data source besides the NHANES: the VCU Medical Center Emergency Department patient data.

This data source did not contain usable mortality data so we instead restricted our focus to testing the HSM's feasibility as a tool for predicting hospital readmission in ED patients. Future work will involve testing the predictive accuracy of the HSM for mortality using other data sources.

Appendix I: Bibliography

Berger, J., & Bock, M. (1976). Combining independent normal mean estimation problems with unknown variances. *Annals of Statistics, 4*, 642-648.

Black, S., Kushner, I., & Samols, D. (2004). C-reactive Protein. *Journal of Biological Chemistry, 279*, 48487-90.

Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). *Classification and Regression Trees*. New York: Chapman & Hall.

Breiman, L. (1993). Stacked regression. Technical report: Statistics Department, UC Berkeley, Berkeley CA.

Breiman, L. (1994). Heuristics of instability in model selection. Technical Report: Statistics Department, UC Berkeley.

Breiman, L. (1996). Bagging Predictors. *Machine Learning, 24*, 123-140.

Breiman, L. (1996a). Stacked regressions. *Machine Learning, 24*, 49-64.

Breiman, L. (2001). Random forests. *Machine Learning, 45*, 5-32.

Breiman, L. (2002). *Software for the masses.* Paper presented at the Meeting of the Institute of Mathematical Statistics, Banff, Canada.

Breiman, L. (2003). Random Forests Manual v4.0. Technical report: Statistics Department, UC Berkeley, Berkeley CA.

Bühlmann, P., & Yu, B. (2000). Explaining Bagging. Research Report No. 92, *Seminar Für Statistik*, ETH, Zurich, Switzerland

Bühlmann, P., & Yu, B. (2002). Analyzing Bagging. *Annals of Statistics, 30*(4), 927-961.

Bühlmann, P. (2003). Bagging, Subagging and Bragging for improving some prediction algorithms. Research Report No. 113, *Seminar Für Statistik*, ETH, Zurich, Switzerland

Bureau, A., Dupuis, J., Falls, K., Lunetta, K., Hayward, B., Keith, T., & Van Eerdewegh, P. (2005). Identifying SNPs predictive of phenotype using random forests. *Genetic Epidemiology, 28,* 171-182.

Carrico, C. (2013). *Characterization of a weighted quantile score approach for highly correlated data in risk analysis scenarios.* Richmond, VA: Department of Biostatistics PhD

Dissertation, Virginia Commonwealth University; 2013.

Celis, M., Dennis, J., & Tapia, R. (1985). A Trust Region Strategy for Nonlinear Equality Constrained Optimization. *SIAM: Numerical Optimization*, 71-82.

Centers for Disease Control and Prevention (CDC). National Center for Health Statistics (NCHS). National Health and Nutrition Examination Survey Data. Hyattsville, MD: U.S. Department of Health and Human Services, Centers for Disease Control and Prevention, 2002, URL: *http://wwwn.cdc.gov/nchs/nhanes/search/nhanes01_02.aspx*

Centers for Disease Control and Prevention (CDC). National Center for Health Statistics (NCHS). National Health and Nutrition Examination Survey Questionnaire. Hyattsville, MD: U.S. Department of Health and Human Services, Centers for Disease Control and Prevention, 2008,URL:*http://wwwn.cdc.gov/nchs/nhanes/search/datapage.aspx?Component=Questionnaire&CycleBeginYear=2007*

Centers for Disease Control and Prevention (CDC). National Center for Health Statistics (NCHS). National Health and Nutrition Examination Survey Data. Hyattsville, MD: U.S. Department of Health and Human Services, Centers for Disease Control and Prevention, 1994, URL: *http://www.cdc.gov/nchs/nhanes/nh3data.htm*

Chen, X., Wang, L., Ishwaran, H. (2010). An integrative pathway-based clinical-genomic model for cancer survival prediction. *Stat. Prob. Lett.*, *80*(17-18), 1313-1319.

Chen, X., & Ishwaran, H. (2012). Random Forests for Genomic Data Analysis. *Genomics*, *99*(6), 323-329.

Chen, X., & Ishwaran, H. (2013). Pathway Hunting by Random Survival Forests. *Bioinformatics*, *29*(1), 99-105.

Chen, Y., Jia, Z., Mercola, D., & Xie, X. (2013). A Gradient Boosting Algorithm for Survival Analysis via Direct Optimization of Concordance Index. *Comput Math Methods Med.*, *e873595, 2013*.

Chipman, H., Hastie, T., & Tibshirani, R. (2003). Clustering Microarray Data. In *Statistical Analysis of Gene Expression Microarray Data*. Boca Raton, Florida: CRC Press.

Christensen, K. L. Y., White, P. (2011). A Methodological Approach to Assessing the Health Impact of Environmental Chemical Mixtures: PCBs and Hypertension in the National Health and Nutrition Examination Survey. *Int J Environ Res Public Health*, *8*(*11*), 4220-37.

Christensen, K. L. Y., Carrico, C. K., Sanyal, A. J., & Gennings, C. (2013). Multiple classes of environmental chemicals are associated with liver disease: NHANES 2003–2004. *International Journal of Hygiene and Environmental Health, 216*(6), 703-709.

Cohn, B., Terry, M., Plumb, M., & Cirillo, P. (2012). Exposure to polychlorinated biphenyl

(PCB) congeners measured shortly after giving birth and subsequent risk of maternal breast cancer before age 50. *Breast Cancer Res Treat, 136*(1), 267-275.

Collett, D. (2003). *Modelling Survival Data in Medical Research*. Boca Raton: CRC Press.

Cordell, H. (2009). Detecting gene-gene interactions that underlie human diseases. *Nature Review Genetics, 10*(6), 392-404.

Cox, D., Wermuth, N. (1990). An approximation to maximum likelihood estimates in reduced models. *Biometrika, 77*(4), 747-761.

Cramer, J. (1986). *Econometric Applications of Maximum Likelihood Methods*. Cambridge, England: Cambridge University Press.

Dempster, A., Laird, N., & Rubin, D. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B, 39*(1), 1-38.

Dennis, J., Gay, D., & Welsch, R. (1981). An Adaptive Nonlinear Least-Squares Algorithm. *ACM Transactions on Mathematical Software, 7*, 348-368.

Der, G., & Everitt, B. (2006). *Statistical Analysis of Medical Data Using SAS*. Boca Raton, FL: CRC Press.

Diaz-Uriarte, R., & Alvarez de Andres, S. (2006). Gene selection and classification of microarray data using random forest. *BMC Bioinformatics, 7(3)*.

Efron, B., & Morris, C. (1973). Combining possibly related estimation problems (with discussion). *Journal of the Royal Statistical Society. Series B, 35*, 379-421.

Firth, B. (1993). Bias Reduction of Maximum Likelihood Estimates. *Biometrika, 80*, 27-38.

Gallant, A. (1987). *Nonlinear Statistical Models*. New York: John Wiley & Sons.

Gaskins, A., Colaci, D., Mendiola, J., Swan, S., & Chavarro, J. (2012). Dietary patterns and semen quality in young men. *Hum Reprod., 27*(10), 2899-2907.

Gennings, C., Heuman, D., Fulton. O., Sanyal, A.J. (2010). Use of desirability functions to evaluate health status in patients with cirrhosis. *Journal of Hepatology, 52*(*5*), 665-671

Gennings C., Ellis R., Ritter J.K. (2012). Linking empirical estimates of body burden of environmental chemicals and wellness using NHANES data. *Environment International, 39(1)*, 56-65

Gennings, C., Carrico, C., Factor-Litvak, P., Krigbaum, N., Cirillo, P., & Cohn, B. (2013). A cohort study evaluation of maternal PCB exposure related to time to pregnancy in daughters. *Environmental Health, 12*(*1*), 66.

Gerds, T., & Schumacher, M. (2006). Consistent Estimation of the Expected Brier Score in General Survival Models with Right-Censored Event Times. *Biometrical Journal, 48*(6), 1029-1040.

Glaser, B. N., I; Chubb, D; Hamshere, M.L.; Sequrado, R; Moskvina, V; Holmans, P. (2007). Analyses of single marker and pairwise effects of candidate loci for rheumatoid arthritis using logistic regression and random forests. *BMC Proceedings, 1(Suppl 1),* S54.

Goldman, L., & Schafer, A. (2011). *Goldman's Cecil Medicine* (24th ed.). Pennsylvania: Saunders Elsevier.

Gorodeski, E., Ishwaran, H., Kogalur, U., Blackstone, E., Hsich, E., Zhang, Z., et. al. (2011). Use of Hundreds of Electrocardiographic Biomarkers for Prediction of Mortality in Postmenopausal Women. The Women's Health Initiative. *Circulation: Cardiovascular Quality and Outcomes, 4*(5), 521-532.

Green, E., & Strawderman, W. (1991). A James-Stein type estimator for combining unbiased and possibly biased estimators. *Journal of the American Statistical Association, 86,* 1001-1006.

Hamming, R. (1950). Error detecting and error correcting codes. *Bell System Technical Journal, 29*(2), 147-160.

Harrell, F., Califf, R., Pryor, D., Lee, K., & Rosati, R. (1982). Evaluating the yield of medical tests. *Journal of the American Medical Association, 247,* 2543-2546.

Harrell, F., Lee, K., & Mark, D. (1996). Multivariable Prognostic Models: Issues in Developing Models, Evaluating Assumptions and Adequacy, and Measuring and Reducing Errors. *Statistics in Medicine, 15,* 361-387.

Hastie, T., Tibshirani, R., Sherlock, G., Eisen, M., Brown, P., & Botstein, D. (1999). Imputing Missing Data for Gene Expression Arrays. Technical Report, Division of Biostatistics, Stanford University, Stanford, CA.

Hastings, C., Mosteller, F., Tukey, J., & Winsor, C. (1947). Low moments for small samples: A comparative study of order statistics. *Ann. Math. Statist., 18*(3), 309-472.

Heikes, K., Eddy, D., Arondekar, B., & Schlessinger, L. (2008). Diabetes Risk Calculator: A Simple Tool for Detecting Undiagnosed Diabetes and Pre-Diabetes. *Diabetes Care, 31,* 1040-1045.

Hippisley-Cox, J., Coupland, C., Vinogradova, Y., Robson, J., Minhas, R., Sheikh, A., Brindle, P. (2008). Predicting cardiovascular risk in England and Wales: prospective derivation and validation of QRISK2. *BMJ, 336*(7659), 1475-1482. doi: 10.1136/bmj.39609.449676.25

Ho, T. (1998). The Random Subspace Method for Constructing Decision Forests. *IEEE*

Transactions on Pattern Analysis and Machine Intelligence, 20(8), 832-844.

Horne, B., May, H., Muhlestein, J., Ronnow, B., Lappe, D., Renlund, D., et. al. (2009). Exceptional mortality prediction by risk scores from common laboratory tests. *The American Journal of Medicine, 122*, 550-558.

Hosmer, D., & Lemeshow, S. (2000). *Applied Logistic Regression*. Hoboken, NJ: John Wiley & Sons.

Hothorn, T., Buhlmann, P., Dudoit, S., Molinaro, A., & van der Laan, M. (2006). Survival Ensembles. *Biostatistics, 7*(3), 355-373.

Hoyert, D., & Xu, J. (2012). Deaths: Preliminary Data for 2011 *National Vital Statistics Reports* (Vol. 61). Hyattsville, MD: National Center for Health Statistics.

Hsich, E., Gorodeski, E., Blackstone, E., Ishwaran, H., & Lauer, M. (2010). Identifying Important Risk Factors for Survival in Patient with Systolic Heart Failure Using Random Survival Forests. *Circulation: Cardiovascular Quality and Outcomes, 4*, 39-45.

Ishwaran, H. (2007). Variable importance in binary regression trees. *Electronic Journal of Statistics, 1*, 519-537.

Ishwaran, H., Kogalur, U., Blackstone, E., & Lauer, M. (2008). Random Survival Forests. *Annals of Applied Statistics, 2*(3), 841-860.

Ishwaran, H., & Kogalur, U. (2010). Random Forests for Survival, Regression and Classification (RF-SRC) (Version 1.3). *R Package*.

Ishwaran, H., Kogalur, U., Chen, X., & Minn, A. (2010). Random Surival Forests for High-Dimensional Data. *Stat. Anal. Data Mining, 4*, 115-132.

Ishwaran, H., Kogalur, U., Gorodeski, E., Minn, A., & Lauer, M. (2010a). High-Dimensional Variable Selection for Survival Data. *J Am Stat Assoc, 105*(489), 205-217.

Ishwaran, H., Kogalur, U., Chen, X., & Minn, A. (2011). Random survival forests for high-dimensional data. *Stat. Anal. Data Mining, 4*(*1*), 115-132.

Janssen, K. J., Vergouwe, Y., Donders, A.R., Harrell, F.E., Chen, Q., Grobbee, D.E., Moons, K.G. (2009). Dealing with Missing Predictor Values When Applying Clinical Prediction Models. *Clinical Chemistry, 55*(5), 994-1001.

Jiang, H., Deng, Y., Chen, H., Tao, L., Sha, Q., Chen, J., et. al. (2004). Joint analysis of two microarray gene-expression data sets to select lung adenocarcinoma marker genes. *BMC Bioinformatics, 5(81)*.

Jiang, R., Tang, W., Wu, X., & Fu, W. (2009). A random forest approach to the detection of

epistatic interactions in case-control studies. *BMC Bioinformatics, 10*(S1), S65.

Knaus, W., Wagner, D., Draper, E., Zimmerman, J., Bergner, M., Bastos, P., et. al. (1991). The APACHE III prognostic system. Risk prediction of hospital mortality for critically ill hospitalized adults. *Chest, 100*(6), 1619-1636.

Kotsiantis, S., Kanellopoulos, D. & Zaharakis, I. (2006). *Bagged Averaging of Regression Models*. New York, NY: Springer US.

Lawless, J. S., K. (1978). Efficient screening of nonnormal regression models. *Biometrics, 34*, 318-327.

Leblanc, M., & Tibshirani, R. (1993). Combining estimates in regression and classification. *Journal of the American Statistical Association, 91*, 1641-1650.

Liu, C., Ackerman, H., & Carulli, J. (2011). A genome-wide screen of gene-gene interactions for rheumatoid arthritis susceptibility. *Human Genetics, 129*(5), 473-485.

Luk, A., Ma, R., Lau, E., Yang, X., Lau, W., Yu, L., et. al. (2013). Risk Association of HbA1c Variability with Chronic Kidney Disease and Cardiovascular Disease in Type 2 Diabetes: Prospective Analysis of the Hong Kong Diabetes Registry. *Diabetes/Metabolism Research and Reviews, 29*, 384-390.

Lunetta, K. H., LB; Segal, J; Van Eerdewegh, P. (2004). Screening large-scale association study data: exploiting interactions using random forests. *BMC Genetics, 5(32)*.

Marshall, G., Warner, B., MaWhinney, S. & Hammermeister, K. (2002). Prospective prediction in the presence of missing data. *Statistics in Medicine, 21*, 561-570.

Matsushita, K., Blecker, S., Pazin-Filho, A., Bertoni, A., Chang, P., Coresh, J., & Selvin, E. (2010). The Association of Hemoglobin A1c with Incident Heart Failure Among People without Diabetes: The Atherosclerosis Risk in Communities Study. *Diabetes, 59*, 2020-2026.

Mayr, A., & Schmid, M. (2014). Boosting the concordance index for survival data – A unified framework to derive and evaluate biomarker combinations. *PLoS ONE, 9(1)*: e84483.

Mielniczuk, J., & Teisseyre, P. (2014). Using random subspace method for prediction and variable importance assessment in linear regression. *Computational Statistics & Data Analysis, 71*, 725-742.

Molinaro, A., Carriero, N., Bjornson, R., Hartge, P., Rothman, N., & Chatterjee, N. (2011). Power of data mining methods to detect genetic associations and interactions. *Human Heredity, 72*, 85-97.

Moons, K. G., Donders, A. R., Steyerberg, E. W., & Harrell, F. E. (2004). Penalized maximum likelihood estimation to directly adjust diagnostic and prognostic prediction models for

overoptimism: a clinical example. *J Clin Epidemiol, 57*(12), 1262-1270. doi: 10.1016/j.jclinepi.2004.01.020

Moré, J., & Sorensen, D. (1983). Computing a Trust-Region Step. *SIAM Journal on Scientific and Statistical Computing, 4*, 553-572.

Ng, V., & Breiman, L. (2005). Bivariate variable selection for classification problem. *Statistics Technical Report, University of California, Berkeley* (692).

Opitz, D., & Shavlik, J. (1996). Actively Searching for an Effective Neural Network Ensemble. *Connection Science, 8*(3-4), 337-354.

Pan, Q., Hu, T., Malley, J., Andrew, A., Karagas, M., & Moore, J. (2013) Supervising random forest using attribute interaction networks. *Lecture Notes in Computer Science: Vol. 7833. Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics* (pp. 104-116). Berlin Heidelberg: Springer-Verlag.

Pedhazur, E. (1997). *Multiple regression in behavioral research* (3rd ed.). Fort Worth, TX: Harcourt Brace.

Prinzie, A., den Poel, D. (2008). Random Forests for multiclass classification: Random MultiNomial Logit. *Expert Syst Appl, 34*(*3*), 1721-1732

Quinlan, J. (1986). Induction of Decision Trees. *Machine Learning, 1*, 81-106.

Rao, J.N.K. & Subrahmaniam, K. (1971). Combining independent estimators and estimation in linear regression with unequal variances. *Biometrics, 27*, 971-990.

Reams, R. (1999). Hadamard inverses, square roots and products of almost semidefinite matrices. *Linear Algebra and its Applications, 288*, 35-43.

Rice, T., Rusch, V., Ishwaran, H., & Blackstone, E. (2010). Cancer of the Esophagus and Esophagogastric Junction: Data-Driven Staging for the 7th Edition of the AJCC/UICC Cancer Staging Manuals. *Cancer, 116*(16), 3763-3773.

Rizk, N., Ishwaran, H., Rice, T., Chen, L., Schipper, P., Kesler, K., et. al. (2010). Optimum Lymphadenectomy for Esophageal Cancer. *Annals of Surgery, 251*(1), 46-50.

Robertson, T., Wright, F., & Dykstra, R. (1988). *Order restricted statistical inference*. New York: Wiley.

Rokach, L. (2010). Ensemble-based classifiers. *Artificial Intelligence Review, 33*, 1-39.

RStudio (2012). RStudio: integrated development environment for R (Version 0.97.248). Boston, MA. URL: *http://www.rstudio.org*

Rubin, D. (1976). Inference and missing data. *Biometrika, 63*, 581-592.

Rubin, D. (1987). *Multiple Imputation for Nonresponse in Surveys*. New York: Wiley & Sons.

SAS Institute Inc. (2011). *SAS 9.3*. Cary, NC: SAS Institute Inc.

SAS Institute Inc. (2011). *SAS/STAT 13.1 User's Guide: The GAM Procedure*. Cary, NC: SAS Institute

Schapire, R.E., & Freund, Y. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *J Comp Sys Sci, 55*, 119-139.

Schell, M., & Singh, B. (1997). The reduced monotonic regression method. *Journal of the American Statistical Association, 92*(437), 128-135.

Schwender, H. (2012). Imputing missing genotypes with weighted k nearest neighbors. *Journal of Toxicology and Environmental Health, Part A: Current Issues, 75*(8-10), 438-446.

Seifter, J. (2011). Acid-base disorders. In L. Goldman & A. Schafer (Eds.), *Cecil Medicine*. Philadelphia, PA: Saunders Elsevier.

Skurichina, M., & Duin, R. (2002). Bagging, Boosting, and the Random Subspace Method for Linear Classifiers. *Pattern Anal. Appl., 5*(2), 121-135.

Song, L., Langfelder, P., & Horvath, S. (2013). Random Generalized Linear Model: A Highly Accurate and Interpretable Ensemble Predictor. *BMC Bioinformatics, 14*(5).

Staiano, A., Di Taranto, M., & Bloise, E. (2013) Investigation of single nucleotide polymorphisms associated to familial combined Hyperlipidemia with random forests. *Vol. 19. Neural Nets and Surroundings* (pp. 169-178). Salerno, Italy: Springer-Verlag.

Steyerberg, E. (2009). *Clinical Prediction Models*. New York: Springer.

Steyerberg, E., Vickers, A., Cook, N., Gerds, T., Gonen, M., Obuchowski, N., *et al.* (2010). Assessing the performance of prediction models: a framework for traditional novel measures. *Epidemiology, 21*(1), 128-138.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society. Series B, 36*(2), 111-147.

Strobl, C., Boulesteix, A., Zeileis, A., & Hothorn, T. (2007). Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC Bioinformatics, 8(25)*.

Tan, C., Li, M., & Qin, X. (2008). Random Subspace Regression Ensemble for Near-Infrared Spectroscopic Calibration of Tobacco Samples. *Analytical Sciences, 24(5),* 647-53.

R Development Core Team (2010). R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing. Retrieved from *http://www.R-project.org/*

Therneau, T., & Grambsch, P. (2000). *Modeling Survival Data: Extending the Cox Model*. New York: Springer-Verlag.

Troyanskaya, O. C., M; Sherlock, G; Brown, P; Hastie, T; Tibshirani, R; Botstein, D; Altman, RB. (2001). Missing value estimation methods for DNA microarray. *Bioinformatics, 17*(6), 520-525.

Tu, Y., Gunnell, D., & Gilthorpe, M. (2008). Simpson's Paradox, Lord's Paradox, and Suppression Effects are the same phenomenon - the reversal paradox. *Emerging Themes in Epidemiology, 5(2).*

van der Laan, M. (2006). Statistical inference for variable importance. *Inter. J. Biostatist., 2*(*1*), 1008

Whyatt, R., Liu, X., Rauh, V., Calafat, A., Just, A., Hoepner, L., *et al.* (2012). Maternal Prenatal Urinary Phthalate Metabolite Concentrations and Child Mental, Psychomotor, and Behavioral Development at 3 Years of Age. *Environ Health Perspect., 120*(2), 290-295.

Wilson, P., D'Agostino, R., Levy, D., Belanger, A., Silbershatz, H., & Kannel, W. (1998). Prediction of coronary heart disease using risk factor categories. *Circulation, 97*(18), 1837-1847.

Winham, S., Colby, C., Freimuth, R., Wang, X., de Andrade, M., Huebner, M., & Biernacka, J. (2012). SNP Interaction Detection with Random Forests in High-Dimensional Genetic Data. *BMC Bioinformatics, 13(164).*

Wolpert, D. (1992). Stacked Generalization. *Neural Networks, 5*(2), 241-259.

Wu, J., Devlin, B., Ringquist, S., Trucco, M., & Roeder, K. (2010). Screen and Clean: A Tool for Identifying Interactions in Genome-Wide Association Studies. *Genetic Epidemiology, 34*(3), 275-285.

Zou, H. (2006). The adaptive lasso and its oracle properties. *J Amer Stat Assoc, 101*, 1418-1429.

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *J Royal Statistical Society B, 67*(2), 301-320.

Appendix II: Figures 2.4a-f: Age- and Gender-adjusted Kaplan-Meier curves for strata defined by HSM range (NHANES III data)

Figure 4a: Age: 18-39, Gender=Female

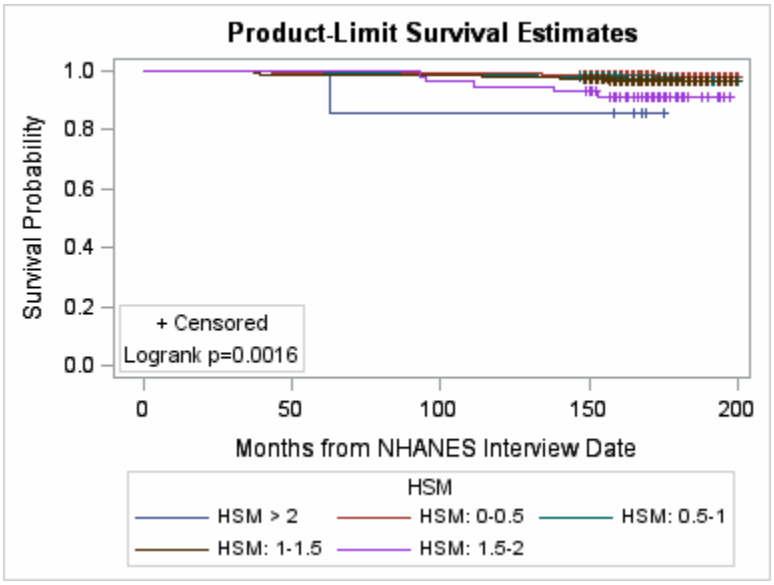


Figure 4b: Age: 40-64, Gender=Female

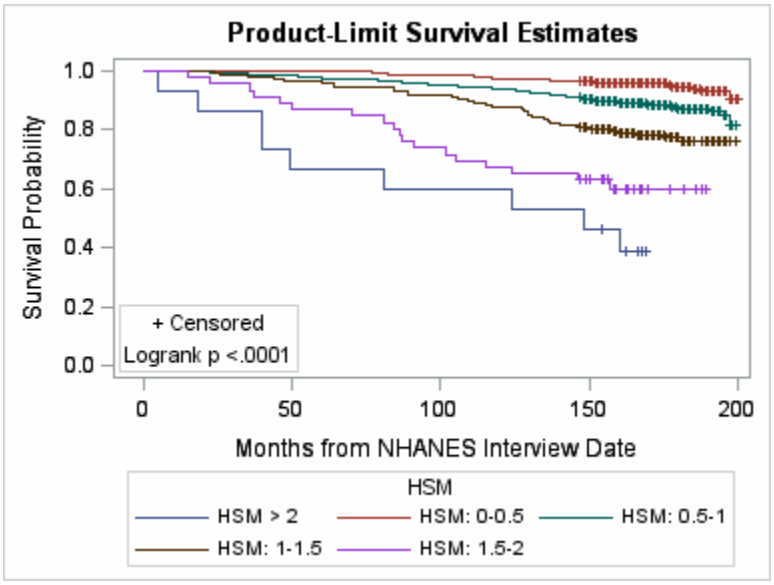


Figure 4c: Age ≥ 65, Gender=Female

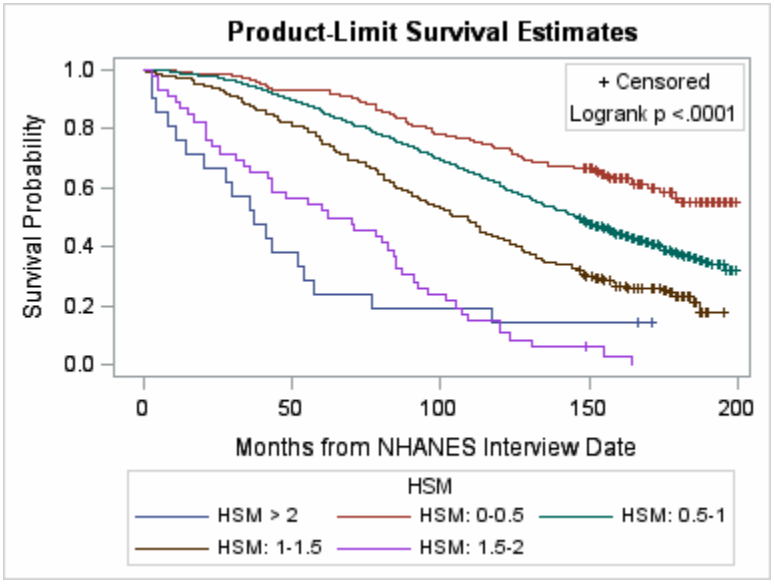


Figure 4d: Age: 18-39, Gender=Male

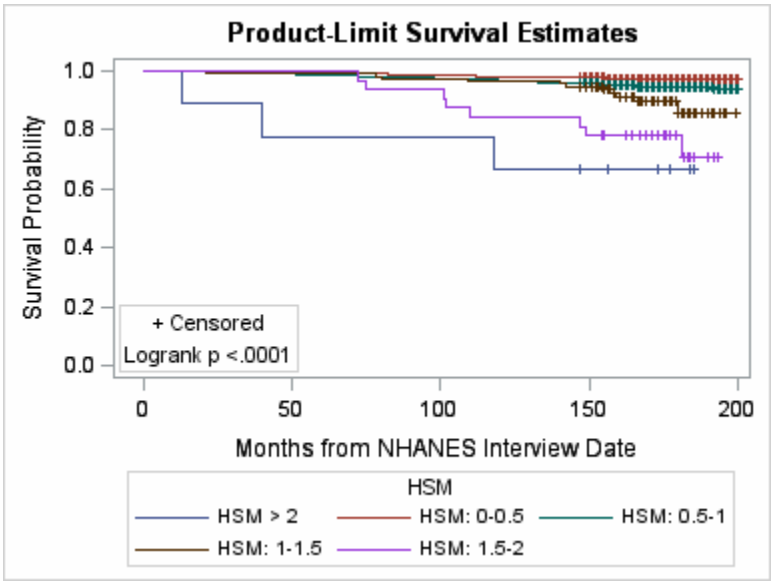


Figure 4e: Age: 40-64, Gender=Male

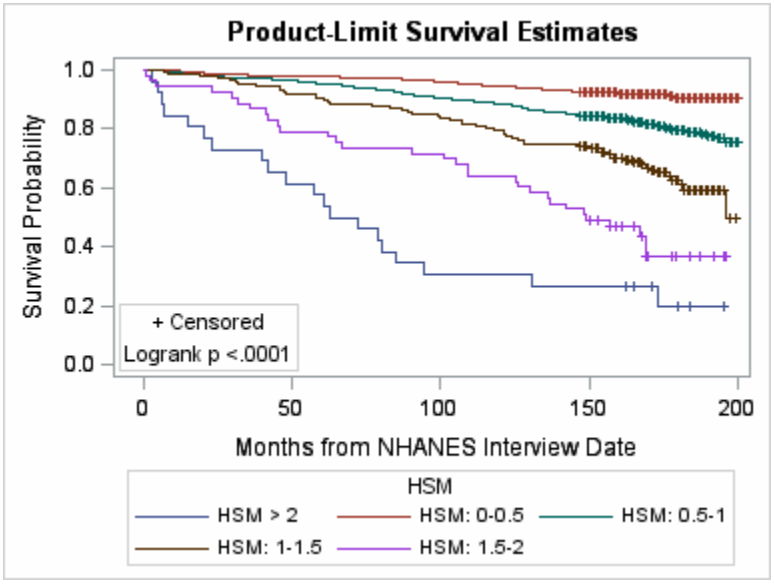


Figure 4f: Age ≥ 65, Gender=Male