ABSTRACT

Title of dissertation: A DATA ANALYTICAL FRAMEWORK FOR IMPROVING REAL-TIME, DECISION SUPPORT SYSTEMS IN HEALTHCARE

Inbal Yahav, PhD Candidate, 2010

Dissertation directed by: Associate Professor Galit Shmueli
Department of Decisions, Operations and Information Technologies

In this dissertation we develop a framework that combines data mining, statistics and operations research methods for improving real-time decision support systems in healthcare. Our approach consists of three main concepts: data gathering and preprocessing, modeling, and deployment. We introduce the notion of offline and semi-offline modeling to differentiate between models that are based on known baseline behavior and those based on a baseline with missing information. We apply and illustrate the framework in the context of two important healthcare contexts: biosurveillance and kidney allocation. In the biosurveillance context, we address the problem of early detection of disease outbreaks. We discuss integer programming-based univariate monitoring and statistical and operations research-based multivariate monitoring approaches. We assess method performance on authentic biosurveillance data. In the kidney allocation context, we present a two-phase model that combines an integer programming-based learning phase and a data-analytical based real-time phase. We examine and evaluate our method on the current Organ Procurement and Transplantation Network (OPTN) waiting list. In both contexts, we show that our framework produces significant improvements over existing methods.

A DATA ANALYTICAL FRAMEWORK FOR IMPROVING REAL-TIME, DECISION SUPPORT SYSTEMS IN HEALTHCARE

by

Inbal Yahav

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Decision, Operations and Information Technologies
2010

Advisory Committee:
Associate Professor Galit Shmueli, Chair/Advisor
Assistant Professor Itir Karaesmen
Professor Louiqa Raschid
Associate Professor Wolfgang Jank
Associate Professor Steve Gabriel

Acknowledgments

Though the following dissertation is an individual work, I could never have reached the heights or explored the depths without the help, support, guidance and efforts of a lot of people.

My deepest gratitude is to my advisor, Prof. Galit Shmueli, for making this dissertation work possible. I have been amazingly fortunate to have an adviser who gave me a close guidelines, yet at the same time, the freedom to explore on my own. Galit taught me how to define and approach research problems and how to ask the right questions.

A special thanks goes to my mentor, Prof. Louiqa Raschid, who continuously supported me in the Ph.D. program. Louiqa has always been there to listen and give advice. Without her encouragement and care, I could not have finished this dissertation.

Besides my adviser and my mentor, I would like to thank and acknowledge the rest of my thesis committee: Prof. Itir Karaesmen, Prof. Wolfgang Jank and Dr. Steve Gabriel, for their insightful comments and constructive criticisms at different stages of my research. I enjoyed working with each one of my committee members and to draw from their great experience.

I am also greatly indebted to my graduate fellows: Gisela Bardossy, Andrew Hall and Daniel Malter, for their care and love. I was lucky to meet them and work with them on various projects and papers.

My final, and most heartfelt, acknowledgment goes to my parents, Avi and Edna, and my immediate family: My husband Ran, and my son Nadav. My family has supported me and encouraged me throughout this endeavor. Their constant love and patience have always constituted a source of inspiration for me. None of this would have been possible without them.

תודה מיוחדת להורי

שהביאוני עד הלום

Table of Contents

List of Tables vii

List of Figures viii

1 **Introduction** 1
1.1 Decision Support Systems in Healthcare . . . 4
1.1.1 Detecting Disease Outbreaks: Univariate and Multivariate Monitoring 5
1.1.2 A Model For Kidney Allocation: Data-Driven Optimization . . . 7
1.2 Research Methods and Goals . . . 8
1.2.1 Framework . . . 9
1.2.1.1 Preprocessing Healthcare Data . . . 9
1.2.1.2 Modeling . . . 12
1.2.1.3 Performance Evaluation . . . 14
1.2.2 Contributions of This Dissertation . . . 15

I Detecting Disease Outbreaks: Univariate and Multivariate Monitoring 18

2 **Background** 19
2.1 Datasets . . . 23
2.1.1 Dataset1: BioALIRT . . . 23
2.1.2 Dataset2: Emergency Department Visits . . . 24
2.1.3 Outbreak Signatures . . . 25

3 **Algorithm combination for improved performance in biosurveillance systems** 28
3.1 Combination Models . . . 30
3.1.1 Control chart combination . . . 30
3.2 Empirical Study and Results . . . 33
3.2.1 Experiment Design . . . 33
3.2.2 Results . . . 33
3.3 Discussion and conclusions . . . 37

4 **Directionally-sensitive multivariate control charts** 38
4.1 Directional monitoring . . . 40
4.2 Multivariate control charts . . . 41
4.2.1 Hotelling's T^2 Control Chart . . . 42
4.2.2 Multivariate EWMA . . . 43
4.3 Directionally-Sensitive Multivariate Control Charts . . . 44
4.3.1 Follmann's Approach . . . 45
4.3.1.1 Extending Follmann's method to MEWMA charts . . . 46
4.3.2 Testik and Runger's Quadratic Programming Approach . . . 47
4.3.2.1 Extending TR's Method to MEWMA Charts . . . 48
4.4 Performance and Robustness Comparison . . . 49
4.4.1 Simulation Setup . . . 49
4.4.2 Impact of Cross-Correlation and Number of Series . . . 50

4.4.3 Robustness to Assumptions . . . 51
4.4.3.1 Unknown Covariance Matrix . . . 51
4.4.3.2 Autocorrelated Series . . . 53
4.4.3.3 Multivariate Poisson data . . . 54
4.4.4 Out of control performance . . . 55
4.4.4.1 Injecting Outbreaks . . . 55
4.4.4.2 Detection of Mean Increases in the Presence of Mean Decreases . . . 58
4.5 Results for Authentic Data . . . 59
4.6 Conclusions and Future Directions . . . 61

5 **Generating Multivariate Poisson Random Variables** 78
5.1 Existing Methods . . . 78
5.1.1 NORTA: NORmal To Anything . . . 81
5.2 Generating Multivariate Poisson Random Variables . . . 84

II Modeling Kidney Allocation: A Data-Driven Optimization Approach 88

6 **Background** 89
6.1 Current Allocation: Priority Points (PP) . . . 91
6.2 KARS' Proposed Allocation: Kidney Allocation Score (KAS) . . . 93
6.3 Literature Survey . . . 96
6.4 Our Approach . . . 100

7 **Proposed Model for Kidney Allocation** 104
7.1 Problem Description and Model Formulation . . . 104
7.1.1 Notation . . . 105
7.1.2 Choosing Objectives . . . 107
7.2 Proposed Real-time Dynamic Allocation Policy . . . 108
7.2.1 Semi-Offline Optimization . . . 108
7.2.2 Knowledge-Based Real-time Allocation Policy . . . 110

8 **Analytical Framework** 112
8.1 Overview . . . 112
8.1.1 Data . . . 113
8.2 Model estimation . . . 115
8.2.1 Estimating Life Years From Transplant (LYFT) . . . 115
8.2.2 Computing Calculated Panel Reactive Antibody (CPRA) . . . 117
8.2.3 Modeling Patient Lifetime . . . 119
8.2.4 Computing Donor Profile Index (DPI) . . . 122
8.3 Model Deployment . . . 124
8.3.1 Semi-Offline Analysis . . . 125
8.3.2 Knowledge-Based Real-Time Policy . . . 128
8.3.2.1 Equity . . . 129
8.3.2.2 Efficiency . . . 130

9 **Discussion and Future Work** 138

A **R codes** 141
A.1 Multivariate monitoring . . . 141
A.2 Multivariate Poisson . . . 154

10 **Glossary of Terms** 157
10.1 Data Mining and Statistical Tools . . . 157
10.1.1 Data Mimicking . . . 157
10.1.2 Linear Regression, Exponential Smoothing, and Differencing . . . 160
10.1.3 Control Charts . . . 162
10.1.4 Survival Analysis . . . 164
10.1.5 The Kruskal-Wallis χ^2 Test . . . 167
10.1.6 Regression Trees . . . 167
10.2 Operations Research Methods . . . 168
10.2.1 Offline and Semi-Offline Optimization . . . 168
10.2.2 Mixed Integer Programming . . . 169

List of Tables

4.1 The relationship between TA and FA rates 57

4.2 Performance of the control charts on authentic data 60

4.3 Summary of performance of multivariate control charts: FA rate as a function of multiple factors 63

4.4 Summary of performance of multivariate control charts: TA rate as a function of pre-set true alert rate and timeliness 64

6.1 The UNOS point system. Source: Zenios (2004) 92

8.1 Patient profile statistics 116

8.2 Donor profile statistics 117

8.3 Transplant failure rate 117

8.4 Predictors used in estimating candidates survival curves 132

8.5 Estimated coefficients of the AFT models (sample size$\approx$ 270k) 133

8.6 Semi-offline performance on training dataset (standard error in parentheses) 133

8.7 Performance of Real-time policies (on training dataset) 134

8.8 Comparison between profile distribution of candidates and recipients (Kruskal-Wallis χ^2-test) 134

8.9 Patient type and organ allocation 134

8.10 Real-time policies performance (on holdout dataset) 136

10.1 Features of three main control charts 165

List of Figures

1.1 Framework building blocks. 9

1.2 Arrivals as an emergency room with complaints of fever. Left: entire dataset. Right: zoom in on the first 40 days. 10

1.3 Distribution of daily patient arrival counts, after adjusting for explainable patterns. 10

2.1 Spanish Flu 1918-1919 (A/H1N1). 20

2.2 Traditional Biosurveillance. (a) Raw series of number of daily military clinic visits with respiratory complaints; (b) Daily military clinic visits series after removing explainable patterns (referred to as residuals); (c) Monitoring the residuals with a control chart. 22

2.3 Daily counts of military clinic visits (top), military filled prescriptions (middle) and civilian clinic visits (bottom), all respiratory- related 24

2.4 Daily counts of military clinic visits (top), military filled prescriptions (middle) and civilian clinic visits (bottom), all gastrointestinal- related 24

2.5 Daily counts of chief complaints by patients arriving at emergency departments in a US city 25

2.6 Injecting a lognormal outbreak signature into raw data 27

3.1 Biosurveillance schematic: univariate monitoring. 29

3.2 Combining control chart outputs. 31

3.3 Combining residuals. 31

3.4 True alert rate distribution for three control charts and their combination, by false alert rate ($\alpha = 0.01, 0.05, 0.10, 0.20$). Means are marked by solid white lines. 34

3.5 Distribution of methods weights for threshold level $\alpha = 0.05$. 35

3.6 True alert rate distribution for select combinations, Panels depict the false alert rate ($\alpha = 0.01, 0.05, 0.10, 0.20$). 36

4.1 Biosurveillance schematic: multivariate monitoring. 40

4.2 True vs. false alert rates for TR's Hotelling chart vs. Follmann's MEWMA chart with restarts 60

4.3 True vs. false alert rates; comparing multivariate control charts with multiple-univariate Shewhart charts 61

4.4 Distribution of false alert rate (FA) in directionally-sensitive Hotelling charts as a function of the number of series p and correlation ρ. The charts are all set to FA=0.05 65

4.5 Distribution of false alert rate (FA) in directionally-sensitive MEWMA as a function of the number of series p and correlation ρ. The charts are all set to FA=0.05 66

4.6 Distribution of false alert rate (FA) in directionally-sensitive charts as a function of training data length (tr) 67

4.7 Distribution of false alert rates (FA) in directionally-sensitive charts as a function of the autocorrelation (θ), when the covariance matrix is known . . 68

4.8 Distribution of false alert rates (FA) in Follmann's directionally-sensitive MEWMA chart with restarts, as a function of the autocorrelation (θ), when the covariance matrix is known 69

4.9 Distribution of false alert rates (FA) in directionally-sensitive charts for Poisson counts, as a function of the Poisson parameter (λ), when the covariance matrix is known 70

4.10 Distribution of true alert (TA) rate in directionally-sensitive Hotelling charts as a function of spike magnitude 71

4.11 Distribution of true alert (TA) rate in directionally-sensitive MEWMA charts as a function of spike magnitude 72

4.12 Distribution of true alert (TA) rate in directionally-sensitive Hotelling charts as a function of spike magnitude when spike is injected into 25% of the series 73

4.13 Distribution of true alert (TA) rate in directionally-sensitive MEWMA charts as a function of spike magnitude when spike is injected into 25% of the series 74

4.14 Distribution of true alert (TA) rate in directionally-sensitive Hotelling charts as a function of outbreak magnitude when the outbreak is injected into 25% of the series 75

4.15 Distribution of true alert (TA) rate in directionally-sensitive MEWMA charts as a function of outbreak magnitude when the outbreak is injected into 25% of the series 76

4.16 Distribution of true alert (TA) rate in directionally-sensitive charts, as a function of spike magnitude in the presence of increasing and decreasing spikes (top), increasing spikes only (middle) and decreasing spikes only (bottom) 77

5.1 Scatter plots for bivariate simulated variables using NORTA, for Normal, Uniform, Poisson($\lambda = 20$) and Poisson ($\lambda = 0.2$) 83

5.2 Comparing the desired correlation to the resulting actual correlation for Poisson bivariate data with low rates 83

5.3 Comparing the desired correlation to the corrected actual correlation 85

5.4 Absolute mean error. Left: Error as a function of the Poisson rates (λ_1, λ_2). Right: Error as a function of the Poisson rate (λ_2) and the desired correlation (ρ). ($\lambda_1 = 0.4$) 86

5.5 Computation time as a function of the data dimension (p) and length (#samples 87

6.1 Increase in waiting list vs. number of donors. 89

6.2 Kidney allocation schematic. 103

7.1 Schematic representation of kidney allocation. 105

7.2 Semi offline representation. 109

8.1 Schematic representation of the analytic study. 114

8.2 Left: Number of candidates (solid line) and donated kidneys (dashed line) added to the waiting list per year. Right: Number of candidates (dark grey) and donated kidneys (light grey) per state. 115

8.3 From left to right: distribution of CPRA, PRA, and non-zero PRA values. . 118

8.4 Residuals (estimated lifetime - observed lifetime) distribution for different survival models. The Weibull distribution (left panel) presents right skewed residuals, which are most appropriate for estimated lifetime. 120

8.5 Effect of diabetes and simultaneous pancreas-kidney transplant on patient lifetime. Solid (black): no diabetes, kidney only. Dashed: diabetes, simultaneous pancreas-kidney transplant. Grey: diabetes, kidney only. 122

8.6 Effect of age on patient lifetime. Solid (black): 20 year old patient with no diabetes. Dashed: 60 year old patient with no diabetes. Grey: 20 year old with diabetes. 122

8.7 Effect of antigens on patient lifetime. Solid: no antigent (ABDR=0). Dashed: a patient with a single antigen A, B and DR. 123

8.8 Relationship between organ survival and recipient lifespan (left) and dialysis time (right). 128

8.9 Relationship between donors' and recipients' age. 128

8.10 Comparison of profile distribution of candidates and recipients. 135

8.11 Regression tree on the allocation outcome that maps organ types to recipients' health profiles. Leaf nodes give average DPI. 136

8.12 Comparing the semi-offline allocation and HKF allocation in terms of DPI (smaller variance is better). The solid line represents the median DPI value, and the whiskers extend to the 5^{th} and 95^{th} percentiles. 137

10.1 Biosurveillance schematic: univariate monitoring. 158

10.2 Biosurveillance schematic: multivariate monitoring. 159

10.3 Kidney allocation schematic. 160

Chapter 1

Introduction

It has been long recognized that the United States healthcare system is in a deep crisis. According to a recent report by the Centers for Medicare and Medicaid Services (CMS), the United States has the most expensive healthcare system worldwide. In 2009, the U.S. spent over $2.5 trillion on health care, accounted for more than 17% of the nations Gross Domestic Product (GDP). At the same time, this budget is still limited compared to the high costs of medical treatments, and the medical supply fails to meet the ever increasing demand.

The implications of this crisis are everywhere. Hospital emergency departments are stretched beyond capacity (yet the number of hospital emergency departments has dropped by 7% in 2009). Hospitals still rely on old-fashioned systems that fail to support innovations in medical technologies. Medical errors cause the unnecessary death of 100,000 Americans annually (according to the Institute of Medicine), and much more.

This U.S. healthcare crisis, which is mainly manifested by gap between high budget, high medical cost and system inefficiency, constitutes a fertile ground for research in many disciplines. One of these disciplines is operations research (OR). The OR literature shows that the field has addressed healthcare problems for more than two decays (e.g., Burton et al., 1978; Brandeau et al., 2004; Sainfort et al., 2005; Romeijn and Zenios, 2008; Royston, 2009). Researchers in OR focus on discovering and implementing better mathe-

matical tools for addressing relevant healthcare delivery issues, for highlighting trade-offs, and for finding better solutions to management problems. Some of the commonly studied applications of OR in healthcare are capacity planning and management of emergency departments (e.g., Green, 2004; Price et al., 2008), location of medical facilities (e.g., Daskin and Dean, 2004; Coskun and Erol, 2010), resource allocation (e.g., Thompson et al., 2009) and patients' choice of treatment models (e.g., Su and Zenios, 2004, 2006). The methodologies used include simulation, queueing theory, linear and non-linear optimization, supply chain management, and others.

With the enormous amounts of health-related data available, healthcare applications have also received a great momentum in the data mining (DM) and statistics disciplines (e.g., Baylis, 1999; Payton, 2003; Liu and Chen, 2009; Porter and Green, 2009). Healthcare information systems collect and store various types of clinical data about patients, hospitals, costs, claims, etc. The focus in data mining and statistics studies has been on extracting patterns and information from such medical data to improve healthcare systems performance, for purposes such as designing and evaluating biosurveillance systems (Shmueli and Burkom, 2010), clinic scheduling (Glowacka et al., 2009), organ allocation (Bardossy and Yahav, 2010), and others or to improve patients' access to data (e.g., Rosow et al., 2003).

Whereas each discipline has its own strengths, integrating methods from different disciplines can offer another step in improving healthcare performance. Statistics and DM can be used to preprocess, visualize and analyze clinical data. which can then be used as input to an OR optimization problem. For example, using DM and statistics to preprocess and analyze data on inpatient arrival and length of stay at emergency department, as

an input to an optimal capacity planning problem. Alternatively, OR simulation-based methods can be used to evaluate the practical performance of a statistical model or data mining algorithm. For example, simulating patients arrivals at emergency departments to examine the performance of a statistical surveillance monitoring system.

In this dissertation we combine techniques from the fields of data mining, statistics and OR for achieving improved performance in two important healthcare domains: bio-surveillance and kidney allocation. The main goal of this work is to create a framework for designing real-time decision support systems for informing health providers and decision makers, which is based on advanced data analytics and key optimization concepts. We emphasize the robustness of our approach and its practical aspects in each of the two healthcare domains.

The novelty of our approach is by incorporating optimizations techniques, which traditionally assume a prior knowledge of the problem input data, and deploying them on noisy data with missing information, after massively preprocessing and completing the data. The optimization methods that we use also shed light on the properties of the desired (yet unachievable under practical, real-time assumptions) outcome. We use these properties to develop and tune decision support algorithms to account for practical data characteristics and to allow deployment in a real-time fashion. We also emphasize time sensitive nature of the solution that is required in health-related decision support systems.

1.1. Decision Support Systems in Healthcare

Healthcare delivery systems have been constantly under pressure to improve performance, in terms of increasing medical productivity, enhancing service levels and guaranteing better access to care. Improving healthcare system performance is important, as it leads to higher quality of life, lower illness levels, and lower death rates.

Recent development in decision support healthcare systems have increased the amount of available medical data, such as data on patients, patients' visits, diseases and disease symptoms. In reaction to these developments, many academic research groups have proposed methods and tools to utilize such data for supporting informed decisions. Yet, in practice, there is still a great gap between the amount of available data and their use in the decision making process. There are two main reasons for this gap. One reason is the complexity of theoretical models in terms of interpretability and deployment. The second reason is that theoretical models commonly rely heavily on impractical assumptions about the data that are violated in practice. In fact, many available models, as we later discuss, are not robust when their underlying assumptions are violated.

In this dissertation we aim at filling the gap between theory and practice, and combining optimization tools and statistical methods with data driven analysis in a unique fashion. We show how our models are tuned to the nature of the available data and how they can be deployed in practice. We apply our techniques to two notable healthcare contexts, namely biosurveillance and kidney allocation.

1.1.1 Detecting Disease Outbreaks: Univariate and Multivariate Monitoring

Biosurveillance is concerned with monitoring information sources to detect, investigate, and respond to an emerging epidemic. One avenue of biosurveillance is the early detection of disease outbreaks (see e.g., Shmueli and Fienberg, 2006; Shmueli and Burkom, 2010). Traditional biosurveillance has focused on the collection and monitoring of medical and public health data that verify the existence of a disease outbreak. Examples are laboratory reports and mortality rates. Although such data are the most direct indicators of a disease, they tend to be collected, delivered, and analyzed days, weeks, and even months after the outbreak. By the time this information reaches decision makers it is often too late to treat the infected population or to react in some other way. Modern biosurveillance has therefore adopted the notion of pre-diagnostic ("syndromic") data in order to achieve early detection. Syndromic data include information such as over-the-counter and pharmacy medication sales, calls to nurse hotlines, school absence records, web-searches on medical websites, and chief complaints by individuals who visit hospital emergency rooms. All these do not directly measure an infection, but it is assumed that they contain an earlier, though weaker, signature of a disease outbreak. These data are also collected and reported on a much more frequent basis (typically daily).

In this work we focus on early detection of disease outbreaks, which is currently implemented in several national-level systems, and that is used to track many health related series over a large number of geographical areas. We concentrate on temporal monitoring, with a goal of improving the performance of existing monitoring algorithms for automated biosurveillance, in terms of alerting properties. In particular, we focus on monitoring univariate and multivariate pre-diagnostic series.

The work is composed of two main parts. In the first part (Section 3) we describe a unique mixed integer programming-based method for combining multiple temporal monitoring techniques that outperforms any of the known single temporal monitoring algorithms. We take a data-mining approach where the choice of combination is set according to the properties of the data. This combined OR and DM approach expands the use of traditional statistical control charts that are widely used in industry and are familiar to healthcare practitioners. We evaluate our method on authentic biosurveillance data and illustrate its robustness to different data properties as well as to different epidemic outbreak magnitudes and shapes.

In the second part (Section 4) we address the practical aspects of multivariate monitoring. Here, multiple series are monitored simultaneously for the purpose of detecting epidemic outbreak signatures in one or more of the series. The main challenge addressed in this context is directionally-sensitive multivariate monitoring, where data are monitored for detecting *increases* in the underlying mean (rather a traditional monitoring that detects shifts in the mean in *any* direction). We describe several multivariate monitoring approaches and evaluate them based on *practical performance aspects* such as robustness to often impractical assumptions, the amount of data required for proper performance, and computational aspects. We perform a large simulation study and examine performance on authentic biosurveillance data.

Motivated by the need for synthetic low count multivariate data in this context, we developed a computationally fast method for generating multivariate Poisson data with a flexible correlation structure (Section 5).

1.1.2 A Model For Kidney Allocation: Data-Driven Optimization

According to the Scientific Registry of Transplant Recipients (SRTR) annual statistics, there are more than 79,000 candidates annually with kidney failure End Stage Renal Disease (ESRD), who are waiting for transplantation in the U.S (OPTN/UNOS, 2008; Zenios et al., 2000). However, because only about 10,000 deceased donor kidneys are available for transplantation each year, more than 20,000 new candidates are added to the waiting list annually.

Under the current kidney Organ Procurement and Transplantation Network (OPTN) allocation system, kidneys are allocated to patients primarily through a combination of tissue matching (also called HLA matching), sensitization level, and waiting time. However, due to recent trends in medicine as well as the shortfall of supply compared to the ever increasing demand, the current system fails to match donors and recipients well. As a result, kidneys with long expected post-transplant survival (often beyond the expected survival of the patient) are commonly allocated to candidates with short expected post-transplant survival.

In 2004 the Kidney Allocation Review Subcommittee (KARS) was formed with the purpose of analyzing ways to improve kidney allocation in the United States. The committee proposed four major considerations in allocating donor diseased kidneys to patients: Life Years From Transplant (LYFT), which determines the estimated additional survival a recipient of a specific kidney may expect to receive, time spent on dialysis, sensitization level, and Donor Profile Index (DPI) that provides a measure of organ quality.

Using these four concepts, we propose a unique two-phase allocation policy that

provides real-time near-optimal allocation under the critical constraint of making an allocation decision in a timely manner. The policy is composed of a learning phase, in which we compute the optimal semi-offline allocation of organs to candidates, based on the entire historical information available at hand, and a knowledge-based deployment phase, in which we derive a knowledge-based allocation policy based on the properties of the optimal allocation and deploy these rules in real-time.

The novelty of our method is that it incorporates the future uncertainty of allocations into the decision process, yet maintains computational feasibility regardless of the challenges that the large dimensionality of the problem presents. We show that our policy outperforms the current allocation system in many respects, such as waiting time to transplantation, match between organs and recipients, lower organ rejection rate, etc.

1.2. Research Methods and Goals

We develop a novel approach to improve healthcare performance by analyzing and modeling healthcare data for deployment in real-time applications. Our approach combines analytic methods from the fields of statistics (ST), data mining (DM), and operations research (OR). The approach consists of three main steps: data preprocessing, modeling, and deployment. The urgent nature of healthcare applications requires that the models be computationally efficient, in order to provide a response within few seconds. We next describe our approach and the main methods that we use in this work.

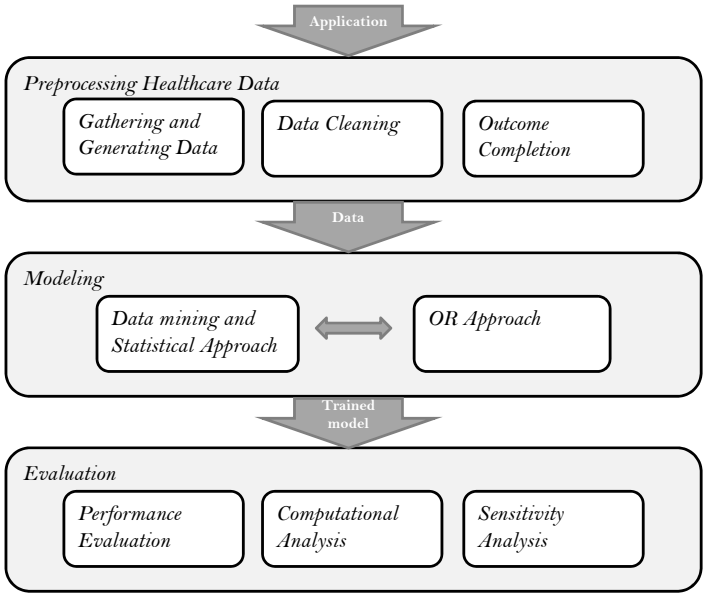

Figure 1.1: Framework building blocks.

1.2.1 Framework

In this section we describe the building blocks in our framework. The float chart in Figure 1.1 depicts the complete process. Each of the building blocks is discussed in detail in the following sections.

1.2.1.1 Preprocessing Healthcare Data

Healthcare data are typically characterized by a mixture of complex data structures and patterns, which makes the analysis of such data nontrivial. Reasons for such observed patterns in the data vary from seasonality, to individual habits, to noise. It is customary to divide the data properties into explainable patterns and unexplainable patterns (or noise). For example, consider data on daily patient arrivals at emergency rooms with complaints of fever, as plotted in Figure 1.2. It is expected that the arrival rate would be

higher in winter compared to summer due to the common cold and seasonal flu. Hence, season is one explained pattern. We can also explain observed day-of-week differences: patients avoid arriving at hospitals on weekends (unless their condition is urgent) and typically postpone their visits to the beginning of the week (Monday-Tuesday). However, there remains a low dispersion of daily patients arrivals counts, after adjusting for the weekly and seasonal explainable effects, which remains unexplained. An example of an arrival distribution is given in Figure 1.3, where it is compared to a Normal distribution (solid line) with the same mean and standard deviation.

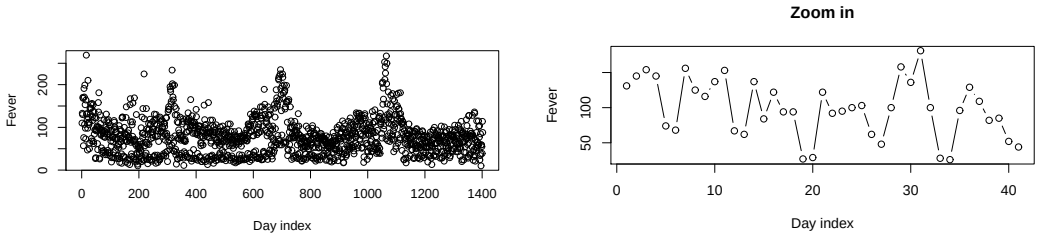

Figure 1.2: Arrivals as an emergency room with complaints of fever. Left: entire dataset. Right: zoom in on the first 40 days.

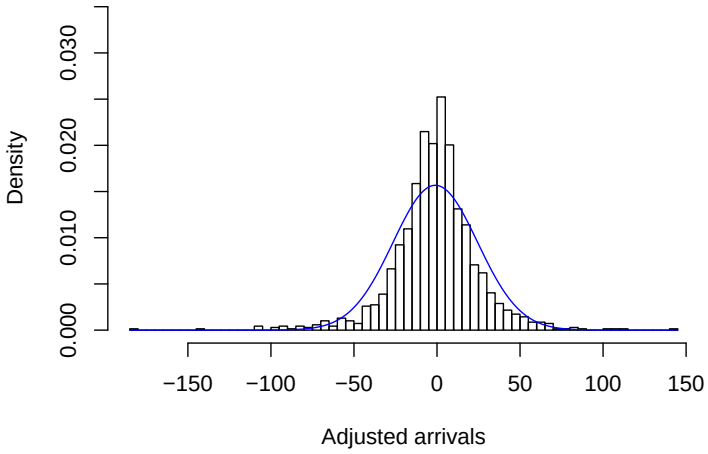

Figure 1.3: Distribution of daily patient arrival counts, after adjusting for explainable patterns.

Cleaning the data of anomalies requires a good understanding of the domain of interest and its context. Theoretical approaches commonly assume a known distribution of the data, such as the Normal or Poisson distribution, and develop models to detect data anomalies under distributional assumptions. However, actual health data usually do not follow classic distributions even after common cleaning and transformation.

The first building block in our approach is gathering sufficient data and cleaning them, so that they can then be used as input to statistical, DM and OR methods. This step should produce data that follow a well behaved and/ or interpretable structure. We therefore must develop robust algorithms that are able to handle such data.

Developing robust algorithms relies heavily on the availability and quality of data. Hence, in the preprocessing step, we generate sufficient amounts of synthetic training data with the same properties as the actual data. The need for data generation arises since the amount of authentic data available for academic purposes is limited (due to privacy issues). We compensate for the lack of available actual data with simulated semi-authentic data. We take two different simulation approaches. The first simulates a 'synthetic' environment, in which we generate all the sources of variation in the data. Whereas this does not mimic a real environment, it enables us to better understand the methods' performance and their robustness to changes in the environment.

The second approach is data-driven simulation. Here, we generate statistical replicas of the original data. The replicas have the same statistical properties as the original data (e.g., mean and variance), but are only statistically equivalent. In other words, the actual values differ across replicas. There are several ways to develop a data-driven simulation. In this work we generate data using a *Data Mimicking* approach, where the statistical

properties of the data are estimated and then used to generate new data with the same features. More information on data mimicking is available in Lotze et al. (2007).

After gathering and cleaning the data we divide them into two sets: training and holdout. The training set is used to build the model and determine its parameters, as is classically done in data mining. The holdout set is later used for performance evaluation. In some cases, we also use a third case, called validation set to tune the model. The validation set cannot be used for evaluating performance, as it is used for model determinatoin. All three datasets are selected to be representative samples of the data that the model will be applied to.

1.2.1.2 Modeling

After preprocessing the data, we move to the important building block of modeling. The goal of this dissertation and its main contribution is to develop a method that is optimal for achieving a given goal with the data at hand, using all the information gathered from the preprocessing step. Ideally, the training data should be complete, and then the modeling part becomes an *offline optimization problem*, where the objective is to maximize some desired outcome, given a set of data (and data replicas), under a set of constraints. Often, however, the training data are incomplete. That is, the recorded outcome and/ or part of the baseline is missing. In this case, the problem becomes a *semi-offline optimization problem*, where the goal is to maximize the desired outcome given a set of *probabilistic* data, under a set of constraints.

Let us consider these optimization problems within the context of biosurveillance and kidney allocation. In the biosurveillance application, where the goal is to produce

early alerts of disease outbreaks, we optimize the expected anomaly detection rate while constraining the false alert rate to be below a required threshold. The false alert constraint corresponds to the available manpower for investigating alerts and the cost that is associated with false alerts. In this application, the daily outcome information, which denotes whether there is an actual outbreak on any given day, is missing. We thus consider a set of probabilistic outbreaks, which we randomly inject into the data.

In the kidney allocation application we face a multi-objective problem that involves the outcome of death rate, matching, equity and more. Here, we optimize a combined measure of the different objectives under individuals' expected health condition and death rate. Because the health condition and death rate of individuals that left the waiting list (after being allocated a kidney) is unknown, a semi-offline optimization is required. In other words, we model a probabilistic future scenario for individuals with missing information and use these scenarios as an input to our optimization problem.

Another important feature of the algorithms that we propose is timeliness. In practice the algorithms will be deployed in real time. Since human life is involved, it is important both to support the decision in a timely manner and to adjust to changes in the data as close as possible to when they occur. For example, a kidney from a deceased donor must be allocated within 48 hours. During this timeframe, an allocation decision should be made, accounting for the possibility of patient rejection (which drives the need for a new allocation decision), the kidney must be shipped, and the transplant done. This short time frame implies that the timeliness of the analytical method makes an allocation decision substantial for the success of a transplant. Another example, from the biosurveillance domain, is an epidemic. When an epidemic outbreak erupts, it is only a matter

of days until it diffuses into large parts of the population. Hence, detecting a disease outbreak must take place very quickly. Every additional day of delay in detection can be crucial in increasing the number of infected individuals and reducing the effectiveness of public health intervention.

1.2.1.3 Performance Evaluation

The last building step in our approach is evaluation, in preparation for deploying the models in real-time systems where future events are unknown. The evaluation block is composed of three steps: (1) performance in terms of the objective, (2) evaluating computational performance and timeliness, and (3) sensitivity analysis. Performance evaluation and timeliness evaluation are studied when the algorithms are applied to test real-time datasets. The performance and timeliness are tested against other known and used methods. Sensitivity analysis is an integral part of the deployment step as it enables us to evaluate the robustness of our methods in terms of their sensitivity to different sources of data variation. In other words, we analyze the performance of the methods when the environment changes, or when applied to new datasets. In reality, data properties change over time and the methods must be able to adjust to such changes.

Another important aspect that influences model deployment is understanding the end user's needs and technical training. Healthcare practitioners tend to prefer deploying methods that are parsimonious and easily interpretable. Hence, we derive the managerial implications of our models and provide a set of guidelines and recommendations to healthcare systems. We also derive, when possible, a set of business rules that can replace the actual models and be used as 'rules of thumb' in the specific contexts.

1.2.2 Contributions of This Dissertation

In this dissertation we develop a framework that combines techniques from the disciplines of data mining, statistics and operations research methods for enhancing performance in real-time decision support systems in healthcare. In particular, we apply optimization algorithms on processed data to improve pattern and information extraction and to tune parameters of data mining algorithms. Our approach consists of three main concepts: data gathering and preprocessing, modeling, and deployment, as shown in Figure 1.1. We apply and illustrate the framework in the context of two important healthcare domains: biosurveillance and kidney allocation. We emphasize the robustness of our approach and its practical aspects in each of the two healthcare domains.

In the biosurveillance context, we focus on the problem of early detection of disease outbreaks. We concentrate on temporal monitoring, with the goal of improving the performance of existing monitoring algorithms for automated biosurveillance in terms of alerting properties. In particular, we focus on monitoring univariate and multivariate pre-diagnostic series.

For monitoring univariate series, we take an approach that *combines* monitoring methods rather than choosing a single one, as commonly done in most published studies and as implemented in several national-level systems. We take a combined OR and DM approach in which we formulate the problem of early detection of disease outbreaks as an integer programming problem with the objective of finding a set of weights for each of the methods considered, such that the weights optimize the combination in terms of true and false alert rates. The weights of the combination are chosen according to the properties of the data, in a DM fashion. This combined OR and DM approach expands the use

of traditional statistical control charts that are widely used in industry and are familiar to healthcare practitioners. We evaluate our method on authentic biosurveillance data and illustrate its robustness to different data properties as well as to different epidemic outbreak magnitudes and shapes.

When multivariate time series are considered, multiple series are monitored simultaneously for the purpose of detecting epidemic outbreaks in one or more of the series. Here, we describe several OR- and statistical-based multivariate monitoring approaches, and evaluate them based on *practical performance aspects* such as robustness to often impractical assumptions, the amount of data required for proper performance, and computational aspects. We perform a large simulation study and examine performance on authentic biosurveillance data.

In the kidney allocation context, we consider the problem of allocating deceased donor kidneys to candidates with kidney failure. Here we combine an optimization technique with ST modeling and DM methods and develop a two-phase allocation policy that involves a learning phase and a real-time, decision-support phase. Our policy accounts for dynamics in the queue, such as patients' expected health condition deterioration, and that of patients joining or leaving due to mortality. In addition, we provide complete model estimation, including candidates' changes in health condition, mortality rate, candidate-organ matching probability, and organ quality. These estimates are then used to tailor our proposed policy to the properties of the kidney waiting list data. In summary, the contribution of this dissertation is in building an overall framework that integrates methods from three prominent data analytic fields, to produce systems that can support decisions in real-time. We also contribute to two important areas in healthcare, by designing practical,

improved systems for early disease detection and for kidney allocation.

Part I

Detecting Disease Outbreaks: Univariate and Multivariate Monitoring

Chapter 2

Background

In 1918, one of the deadliest Influenza pandemics in history erupted, called *The Spanish Flu*. Approximately 20 to 40 percent of the worldwide population fell ill and over 50 million people died worldwide. Outbreaks followed shipping routes from North America through Europe, Asia, Africa, Brazil and the South Pacific. The pandemic reached its peak after 5-6 months (see Figure 2.1[1]). Nearly 40 years later, in February 1957, the *Asian Influenza* pandemic erupted in the Far East. Unlike the *Spanish Flu*, the *Asian Influenza* pandemic virus was quickly identified and vaccines were available 6 months later. Approximately 2 million people died in this outbreak (compared to the 50 million in the *Spanish Flu*). Other known outbreaks in history, such as the *Hong Kong Flu* (1968-69), the *Avian Flu* (1997), *SARS* (2003) and the recent *Avian Flu* (2008-9) also resulted in high death tolls over the years. Unfortunately the threat of new pandemic outbreaks is still looming.

A major goal of public health is to figure out whether and how transmission of diseases can be diminished. Researchers at the *Center for Humanitarian Logistics* at Georgia Tech (`www.tli.gatech.edu/research/humanitarian/projects.php`) have shown that a pandemic outbreak effect can be greatly reduced if quarantine is imposed at the early stages of the disease. The US *Centers for Disease Control & Prevention* (CDC) layout guidelines and strategies for reducing disease transmission, including the use of personal

[1]Figure source: Pandemic Influenza: The Inside Story. Nicholls H, PLoS Biology, available online: http://dx.doi.org/10.1371/journal.pbio.0040050

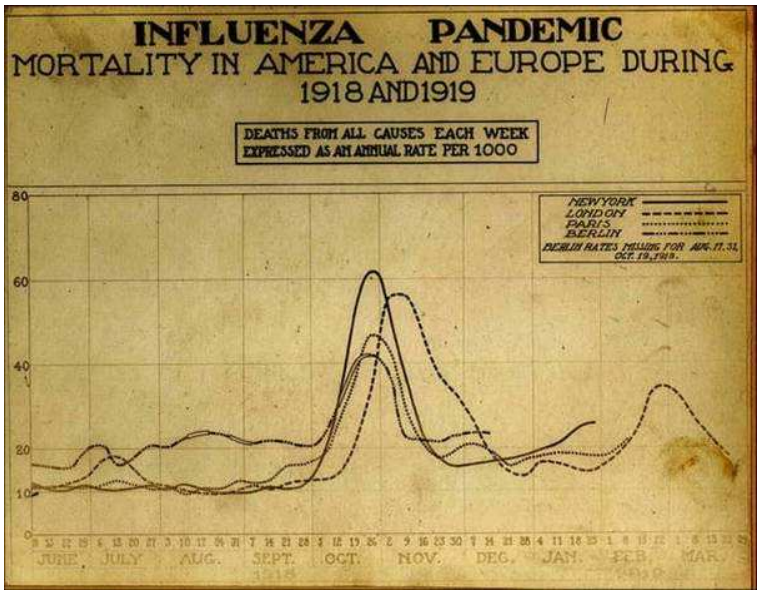

Figure 2.1: Spanish Flu 1918-1919 (A/H1N1).

protective equipment (e.g., masks and gloves), hand hygiene, and safe work practices. The CDC recommendations also outline actions that might be taken during the earliest stage of a pandemic, when the first potential cases or disease clusters are detected. These include individual-level containment measures such as patient isolation and identification, monitoring, and quarantine of contacts (`www.hhs.gov/pandemicflu/plan/appendixf.html`).

The early detection of disease outbreaks therefore plays a major role in preventing disease transmission and reducing the size of the affected population. In modern biosurveillance a wide range of pre-diagnostic and diagnostic daily counts are monitored for the purpose of alerting public health officials when there is early evidence of a disease outbreak. This is in contrast to traditional biosurveillance, where only diagnostic measures (such as mortality and lab reports) are examined, usually locally, and at aggregation levels such as weekly, monthly, or annually. Moreover, in modern biosurveillance the goal is prospective while traditional biosurveillance is more retrospective in nature. Although the tasks and data types and structures differ widely between traditional and modern

biosurveillance, most monitoring algorithms have been migrated from traditional to modern systems. The result is that current modern biosurveillance detection methods suffer from multiple statistical and practical limitations that greatly deteriorate their ability to achieve their intended purpose. For a general overview of the statistical challenges that arise in biosurveillance see Shmueli and Burkom (2010).

A common practice for detecting outbreaks in biosurveillance data is monitoring the data with control charts. Control charts (also referred to as monitoring charts) are used to monitor a process for some quality parameter in order to detect anomalies from desired behavior. In the context of modern biosurveillance, control charts are used to monitor aggregated daily counts of individual health care seeking behavior (such as daily arrivals to emergency departments or medication sales), for the purpose of early detection of shifts from expected baseline behavior. Three control charts are commonly used to monitor such pre-diagnostic daily data, and are implemented (with some variations) in the three main national biosurveillance systems in the Unites States: BioSense (by CDC), ESSENCE (by the Department of Defense), and RODS. The three charts are Shewhart, Cumulative Sum (CuSum) and Exponential Weighted Moving Average (EWMA) charts. These control charts are described in detail in Section 10.1.

One of the main challenges of using control charts to monitor biosurveillance data is that control charts assume that the monitored statistics follow an independent and identically-distributed (iid) normal (or other known) distribution with constant mean and variance. Daily pre-diagnostic counts usually fail to meet this assumption. In reality, time series of such daily counts often contain seasonal patterns, day-of-week effects, and holiday effects. Monitoring such data therefore requires an initial processing step where

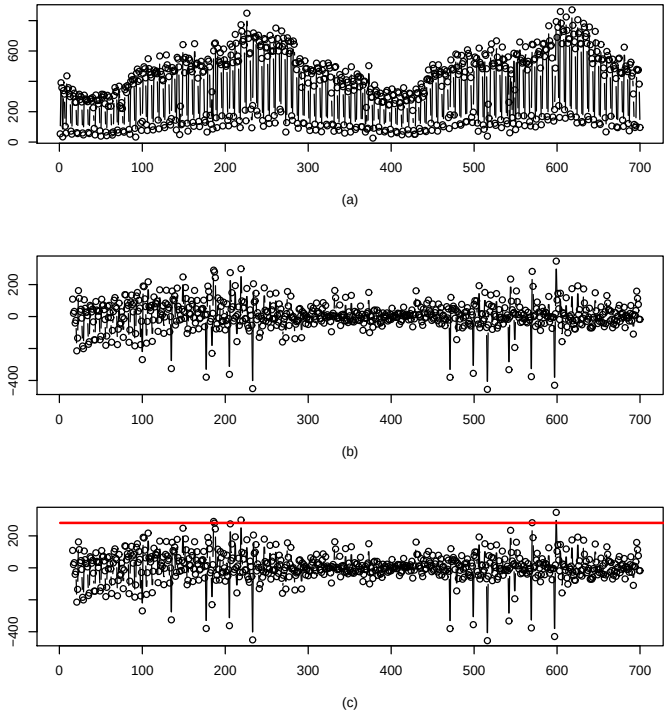

Figure 2.2: Traditional Biosurveillance. (a) Raw series of number of daily military clinic visits with respiratory complaints; (b) Daily military clinic visits series after removing explainable patterns (referred to as residuals); (c) Monitoring the residuals with a control chart.

such explainable patterns are removed. Such methods are described in Section 10.1. For illustration, compare Figures 2.2a and 2.2b that show a series of daily military clinic visits before and after pre-processing. One explainable pattern that is removed is the day-of-week effect, which is clearly visible in Figure 2.2a, but absent from Figure 2.2b. It is customary to refer to the preprocessed series as *residuals*. In modern biosurveillance, control charts are used to monitor the residuals for the purpose of detecting unexplainable patterns, or simply *outbreaks*. Figure 2.2c exemplifies monitoring of residual series.

Another challenge of applying control charts in the modern biosurveillance context is that each type of chart is most efficient at capturing a specific outbreak signature (Box et al., 2009). Yet, in the context of biosurveillance the outbreak signature is unknown, and

in fact the goal is to detect a wide range of signatures for a variety of disease outbreaks, contagious and non-contagious, both natural and bio-terror related. It is therefore unclear which method should be used to detect such a wide range of unspecified anomalies. We address this problem in Chapter 3 in which we describe a novel, data-driven method for combining multiple preprocessing techniques and/or a set of monitoring algorithms that outperform any of the single methods in terms of true and false alarms.

Current temporal monitoring in biosurveillance systems is done univariately, by applying univariate control charts to each series separately. However, a major feature of biosurveillance data is multiplicity in several dimensions: multiplicity of data sources (e.g., over-the-counter medication sales, nurse hotlines, and emergency department visits); multiple locations (e.g., multiple hospitals is a certain region), a variety of diseases of interest, multiple time series from a single source (e.g., medications for treating different symptoms), etc. In Chapter 4 we address the question of feasibility and usefulness of multivariate monitoring techniques for application in biosurveillance.

2.1. Datasets

We describe two authentic biosurveillance datasets. We later use these datasets to illustrate the performance of our monitoring methods.

2.1.1 Dataset1: BioALIRT

This dataset is a subset of the dataset used in the BioALIRT program conducted by the U.S. Defense Advanced Research Projects Agency (DARPA) (Siegrist and Pavlin,

2004). The data include six series from a single city, where three of the series are indicators of respiratory symptoms and the other three are indicators of gastrointestinal symptoms. The series come from three different data sources: military clinic visits, filled military prescriptions, and civilian physician office visits. Figures 2.3 and 2.4 display the six series of daily counts over a period of nearly two years.

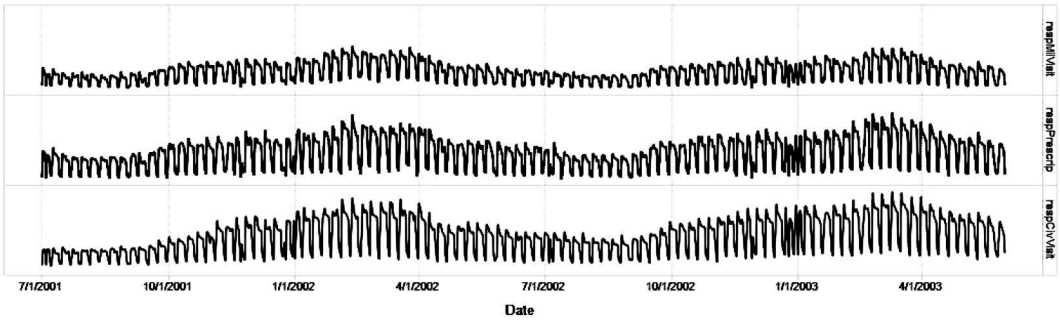

Figure 2.3: Daily counts of military clinic visits (top), military filled prescriptions (middle) and civilian clinic visits (bottom), all respiratory- related

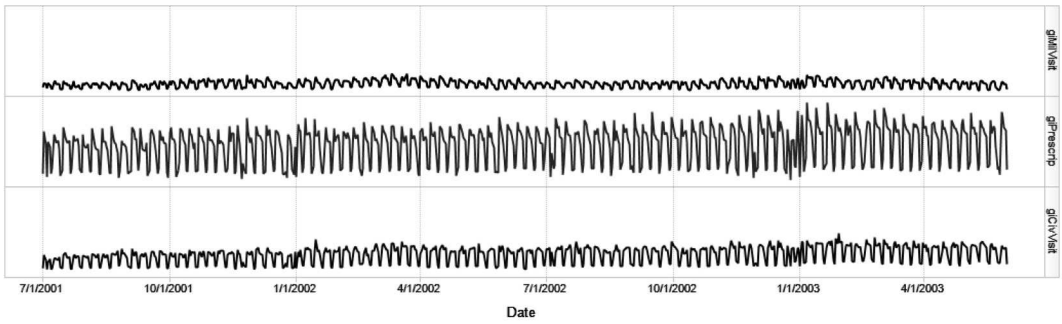

Figure 2.4: Daily counts of military clinic visits (top), military filled prescriptions (middle) and civilian clinic visits (bottom), all gastrointestinal- related

2.1.2 Dataset2: Emergency Department Visits

The data include series of daily counts of patients arriving at emergency departments in a certain US city, between Feb-28-1994 and Dec-30-1997, broken down by the type of "chief complaint". The counts are grouped into 13 categories using the CDC's syndrome

groupings. The data are shown in Figure 2.5. [2]

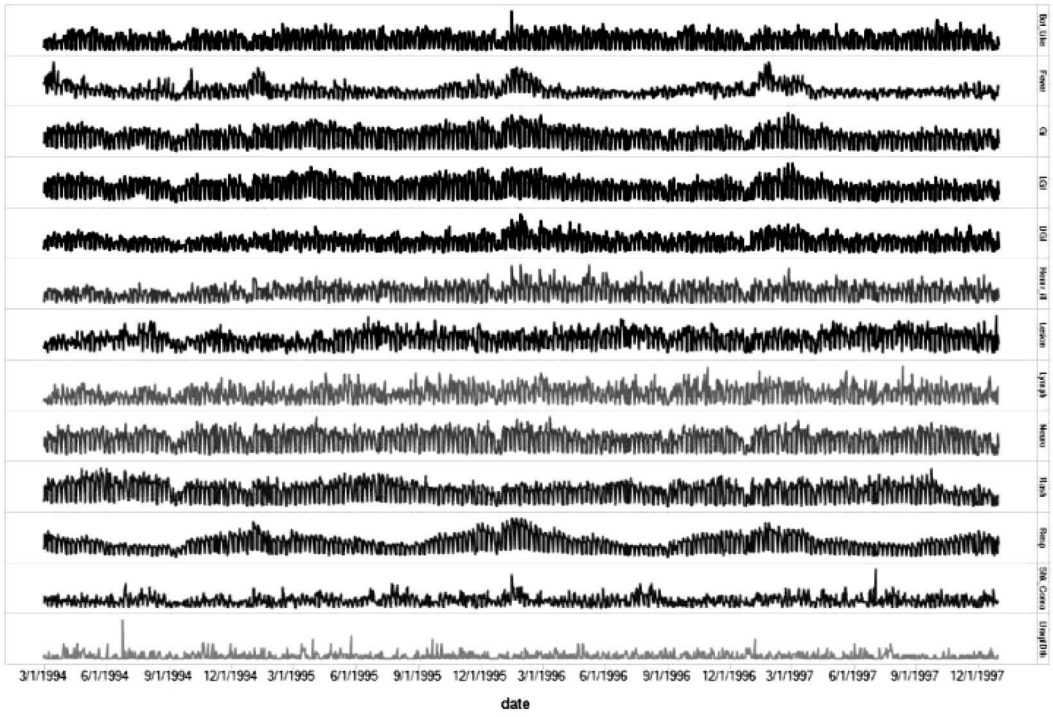

Figure 2.5: Daily counts of chief complaints by patients arriving at emergency departments in a US city

2.1.3 Outbreak Signatures

One of the main challenges with authentic pre-diagnostic data is that they are unlabeled, such that outbreak periods are usually unknown. For purposes of evaluation we therefore assume that the authentic data *do not contain any signatures of unusual diseases,* and the only signatures are those that we inject artificially. This assumption is reasonable when the goal is to detect disease outbreaks that we know are not present in the data (such as an outbreak following a bioterrorist attack or a pandemic such as avian

[2]We thank Dr. Howard Burkom of the Johns Hopkins University's Applied Physics Laboratory, for making this aggregated dataset, previously authorized by ESSENCE data providers for public use at the 2005 Syndromic Surveillance Conference Workshop, available to us.

flu or SARS). The assumption is not reasonable if the goal is to detect an event such as the onset of Influenza which recurs annually. In our case we are indeed interested in detecting an unknown disease outbreak and hence the assumption is reasonable.

When coming to study the performance of our monitoring algorithms, we would like to evaluate the behavior in both the absence and presence of outbreaks. To examine performance in the presence of outbreaks, we inject into the raw data (before preprocessing) outbreak signatures. The insertion into the raw data means that we assume that effects such as day-of-week and holidays will also impact the additional counts due to an outbreak. We simulate two different outbreak signature shapes (For details see Lotze et al., 2007):

Single day spike: We consider small to medium spike sizes, because biosurveillance systems are designed to detect early, more subtle indications of a disease outbreak. We generate a single-day spike outbreak with n number of cases, where n is stochastic and proportional to the standard deviation of the original data:

$$n = a \times sd + \epsilon$$

$$\epsilon \sim N(0, \sigma)$$

Lognormal progression: We consider a (trimmed) lognormal progression signature, because incubation periods have been shown to be well approximated by a lognormal distribution with parameters dependent on the disease agent and route of infection (Burkom, 2003). We generate log-normal shaped outbreaks with shape parameters μ and σ. The peak of the outbreak is approximately on day $\exp(\mu - \sigma^2)$. Similar

to the spike outbreak, the number of cases (i.e., the area of the log-normal shape) is proportional to the standard deviation. We then trim the last t% of the cases to avoid long tails.

Figure 2.6 illustrates the process of injecting a lognormal outbreak into the raw data.

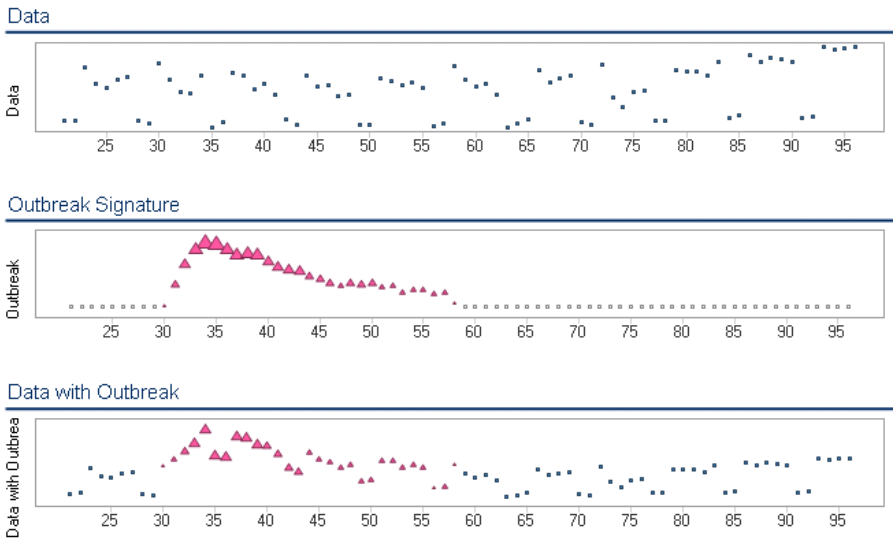

Figure 2.6: Injecting a lognormal outbreak signature into raw data

Chapter 3

Algorithm combination for improved performance in biosurveillance systems

This chapter proposes an enhancement to currently used algorithms for monitoring daily counts of pre-diagnostic data. Rather than use a single algorithm or apply multiple algorithms simultaneously, our approach is based on ensembles of algorithms. The ensembles lead to better performance in terms of higher true alert rates for a given false alert rate threshold.

Most research studies have focused on using a combination of a particular preprocessing procedure with a particular monitoring algorithm. For example, Brillman et al. (2005) present a combination of square regression with a Shewhart monitoring chart. Reis and Mandl (2003) used an autoregressive method integrated with moving average (MA) charts. A literature survey of existing outbreak detection algorithms is outlined by Buckeridge et al. (2005) and Shmueli and Fienberg (2006). In practice, several of the leading national and regional biosurveillance systems use either no preprocessing or a single simple preprocessing method together with a few different monitoring algorithms (typically Shewhart, MA, CuSum, and EWMA charts). The multiplicity in monitoring algorithms creates multiple testing which results in excessive false alarm rates.

Unlike 'well-behaved' series, where an adequate preprocessing technique can be selected based on the data characteristics, pre-diagnostic data are usually non-stationary,

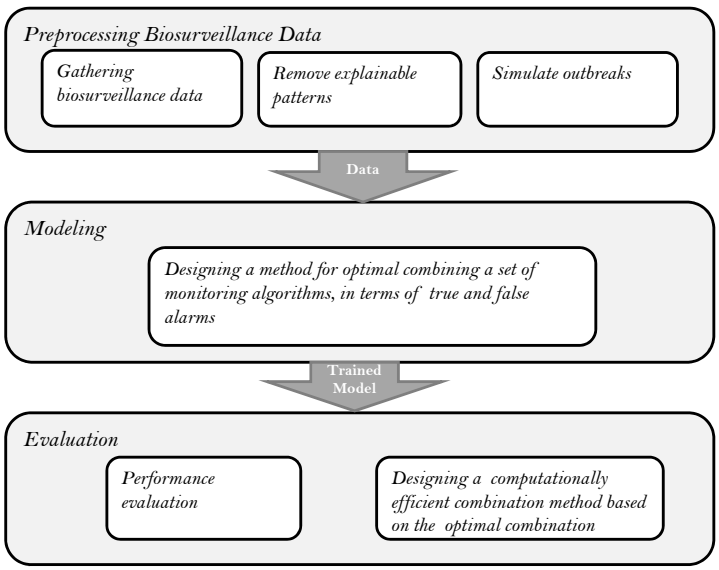

Figure 3.1: Biosurveillance schematic: univariate monitoring.

noisy, and extremely heterogeneous across geographies, data sources, and time. In this context it is hard to determine which preprocessing technique is most adequate for each data stream. Furthermore, when the outbreak pattern is known priori (a signature recognition task), we can design the most efficient monitoring algorithm. However, in the biosurveillance context the nature of a disease outbreak in pre-diagnostic data is typically unknown, and therefore the choice of monitoring algorithm is not straightforward.

Here, we describe a method for combining a set of monitoring algorithms and/or a set of multiple preprocessing techniques that outperform any of the single methods in terms of true and false alarms. We focus on combining monitoring algorithms, and show that by combining results from multiple algorithms in a way that controls the overall false alert rate, we can actually improve overall performance. The float chart in Figure 3.1 depicts the outline of our framework applied to univariate monitoring.

The remainder of the chapter is organized as follows: Section 3.1 introduces the

notion of model combinations in terms of combining residuals and combining control chart output. Section 3.2 applies a control chart combination method to our data, and we display results showing the improvement in detection performance due to method combination. Section 3.3 summarizes the main points and results and describes potential enhancements.

3.1. Combination Models

We consider the problem of linearly combining control charts and/or preprocessing techniques for improving the performance of automated biosurveillance algorithms. We focus on control chart combinations, with the use of a single preprocessing technique (see Figure 3.2). Our combination approach can be similarly applied on residuals from different preprocessing techniques, with the use a single control chart (see Figure 3.3). A comprehensive study of residual combination and optimization of the complete process (combining both residuals and control charts) is available in Yahav et al. (2010).

We assume that the data are associated with a label vector O_t, which denotes whether there is an actual outbreak at day t. We further assume a sufficient amount of training data. The labeled vector and sufficient training data are essential when seeking an optimal combination that increases the true alert rate while maintaining a manageable false alert rate.

3.1.1 Control chart combination

In this section, we assume that the raw data have undergone a preprocessing step for removing explainable patterns. Thus, the input into the control charts is a series

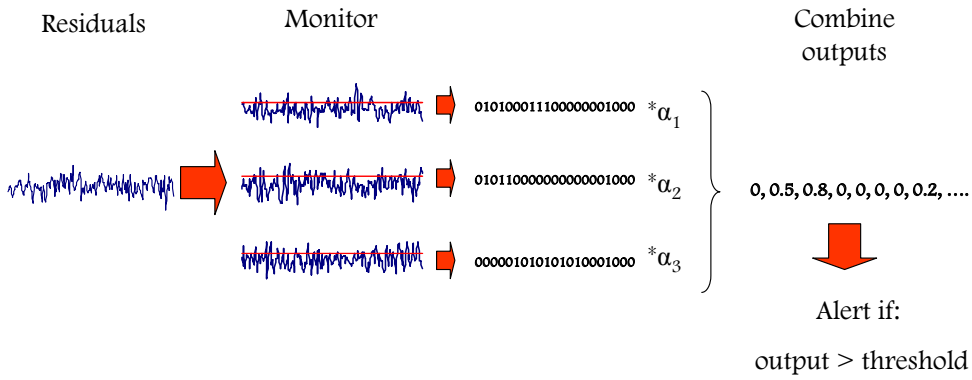

Figure 3.2: Combining control chart outputs.

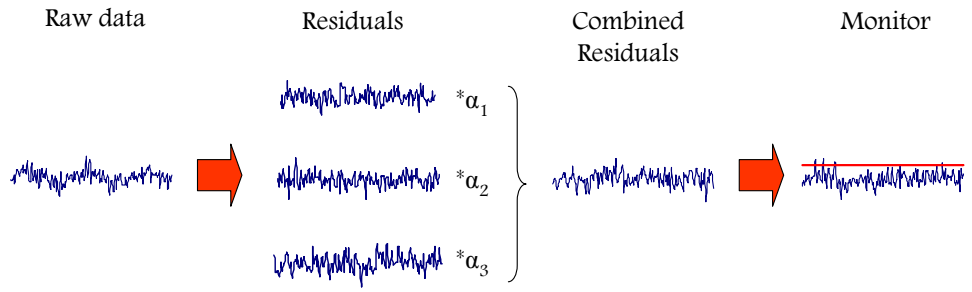

Figure 3.3: Combining residuals.

of residuals. We consider the following three monitoring charts: Shewhart, EWMA and CuSum. We construct a linear combination of the monitoring binary output for the purpose of maximizing the true alert rate, while constraining the false alert rate to be below a specific threshold. The monitoring charts are combined in a semi-offline fashion, that is, the series of residuals is priori known, and the outbreak information (and consecutively, the true and false alert vectors) is completed in the preprocessing step. This formulation yields the following mixed integer programming optimization problem:

$$
\begin{aligned}
&\max \sum_{t=1}^{n} TA_t \\
&\text{s.t.} \\
&(Bin:) && FA_t,\ TA_t \in \{0,1\} \\
&(FA:) && (w_S \times S_t + w_E \times E_t + w_C \times C_t) \times (1 - O_t) - T < FA_t \\
&(TA1:) && [(w_S \times S_t + w_E \times E_t + w_C \times C_t) - T] \times O_t \leq TA_t \times O_t \\
&(TA2:) && (TA_t \times O_t \leq (w_S \times S_t + w_E \times E_t + w_C \times C_t) \times O_t \\
&(FA\_sum:) && \textstyle\sum_{t=1}^{n} FA_t \leq \alpha \times n,
\end{aligned} \tag{3.1}
$$

where FA_t (TA_t) is an indicator for a false (true) alert on day t and the variables w_i are the weight of the control charts (w_i for control chart i). The constraints can be interpreted as follows:

Bin: restricts the false alert (FA) and true alert (TA) indicators on day t to be binary.

FA: is a set of n (training horizon) constraints that determine whether the combined output ($w_S \times S_t + w_E \times E_t + w_C \times C_t$) yields a false alert on day t:

- If there is an outbreak on day t, then $1 - O_t = 0$ and the constraint is satisfied.
- Otherwise ($1 - O_t = 1$), we compare the combined output with the threshold $T = 1$. If the combined output is greater than the threshold, we set FA_t to 1.

TA1, TA2: is a set of $2n$ constraints that determine whether the combined output ($w_S \times S_t + w_E \times E_t + w_C \times C_t$) yields a true alert on day t.

FA_sum: sets the false alert rate to be less than α.

3.2. Empirical Study and Results

In this Section we describe the results obtained from applying the combination methods to authentic pre-diagnostic data with simulated outbreaks. We start by describing the experimental design and then evaluate the methods' performance.

3.2.1 Experiment Design

We consider the data described in Chapter 2, Section 2.1.1. We inject into the raw series 20 outbreak signatures, in random locations (every 10 weeks on average). Each outbreak signature can be either a spike of size $0.5 \times \sigma$ (≈ 60 cases), with probability 0.6, or a trimmed lognormal curve of height $5 \times \sigma$ (≈ 450 cases) with probability 0.4. The peak of the lognormal curve is typically on the 5th or 6th day. We inject a mixture of the two outbreak signatures to illustrate the robustness of the algorithm combination. We repeat this test setting 100 times.

When combining control charts, the desired false alert rate (α) is varied in the range $\alpha = 0.01, 0.05, 0.1, 0.2$. We set the threshold of the monitoring charts to meet the desired overall false alert rate α, using one year of training data (referred to as the *experimental threshold*).

3.2.2 Results

We start by preprocessing the raw series using Holt Winters exponential smoothing. Control charts (Shewhart, EWMA, CuSum, and the combined output) are then used to monitor the series of residuals. Finally, we calculate the false and true alert rates produced

by each method. For the lognormal outbreak signature, we consider a true alert only if the alert took place before the peak, because timely alerting plays an important role in diminishing the spread of a disease.

In the first experiment we optimize the control chart combination separately for each of the 100 tests. Note that the weights are computed based on a *training dataset of 150 days only* and not based on the entire dataset. Figure 3.4 depicts the results of this experiment. The different panels correspond to different levels of experimental threshold α. Each panel shows the true alert rate distribution for each of the 4 thresholds. Medians are marked as solid white lines, boxes correspond to the inter-quartile range, and the lines extend between the 5^{th} and 95^{th} percentiles. The results clearly show the advantage of the combined method in terms of both increasing true alert rate, subject to a given false alert rate, and in reducing the variance of the true alert rate.

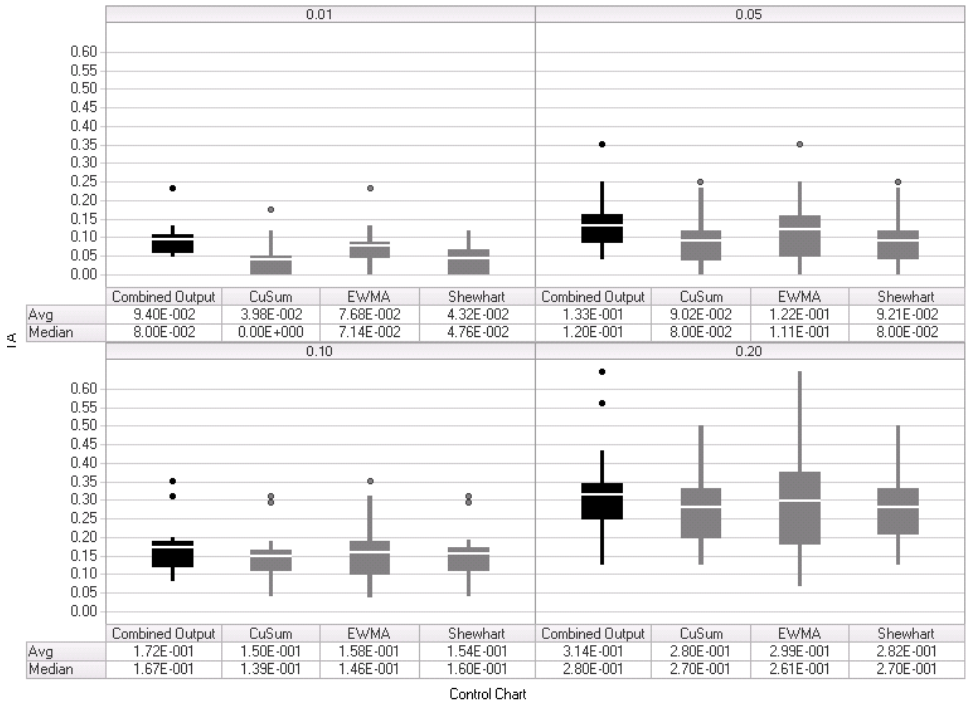

Figure 3.4: True alert rate distribution for three control charts and their combination, by false alert rate ($\alpha = 0.01, 0.05, 0.10, 0.20$). Means are marked by solid white lines.

The main drawback of the first experiment is that the computation is very time consuming. Since achieving the optimal weights for the control charts is an NP complete problem, computation time increases exponentially in the length of the training data. Moreover, examining the actual weights shows that EWMA and Shewhart charts dominate the combination such that alerts are mostly determined by one of them (e.g., Shewhart) combined with an alert by one other method (e.g., either EWMA or CuSum). Figure 3.5 depicts the distribution of the methods' weights for threshold level $\alpha = 0.05$. In an effort to reduce computation time, yet seek for good combinations, we take a hybrid approach: We choose among a small set of pre-determined combinations that appear to work well. This approach greatly reduces computation time and allows for real-time computation in actual settings.

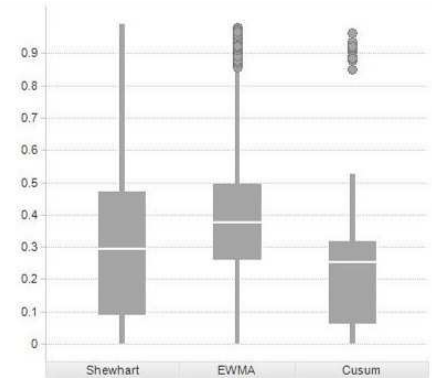

Figure 3.5: Distribution of methods weights for threshold level $\alpha = 0.05$.

Based on the general results found in the first experiment for the optimal weights, in the next experiment we chose two settings of pre-set weights:

Shewhart+: The algorithm signals an alert at time t if the Shewhart statistic signals an alert, and at least one other chart signals an alert.

EWMA+: The algorithm signals an alert at time t if the EWMA statistic signals an

alert, and at least one other chart signals an alert.

The resulting true alert rates are shown in Figure 3.6. We observe that for a very low experimental false alert rate threshold ($\alpha = 0.01$) the two new combination charts (Shewhart+ and EWMA+) do not perform as well as the individual Shewhart and EWMA charts. However, when the experimental false alert rate threshold is higher ($\alpha = 0.05$) the new charts perform at least as well as the ordinary charts, and even outperform the optimal combination (based on training data) when $\alpha > 0.05$. None of the methods violated the experimental false alert rate threshold by more than 10% when $\alpha = 0.01$, and 3% when $\alpha > 0.05$.

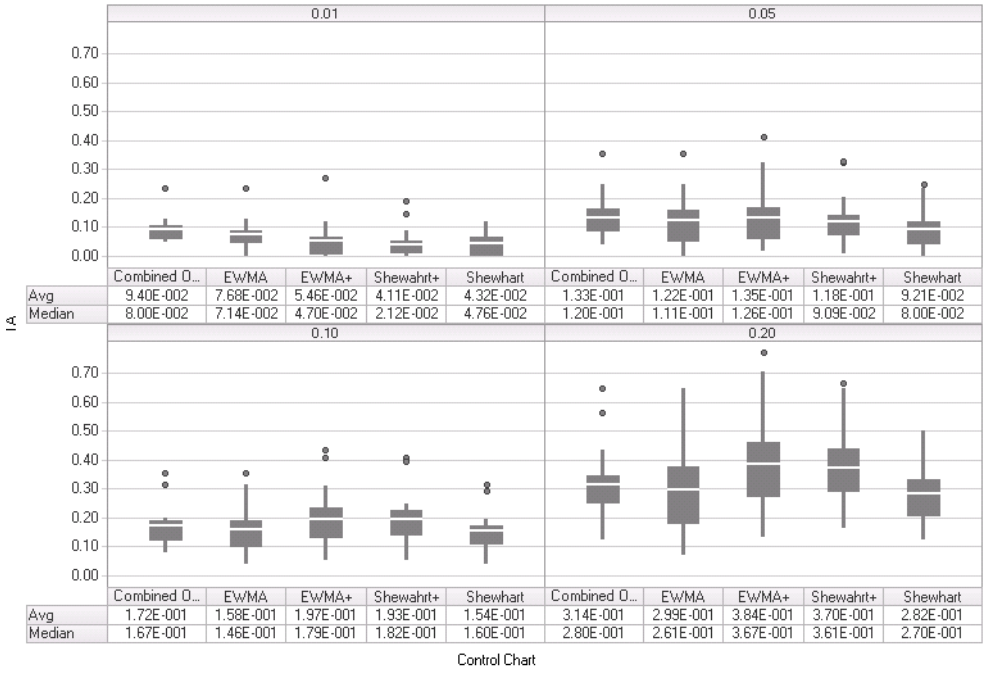

Figure 3.6: True alert rate distribution for select combinations, Panels depict the false alert rate ($\alpha = 0.01, 0.05, 0.10, 0.20$).

3.3. Discussion and conclusions

In this work we proposed a method for improving the performance of univariate monitoring of non-stationary pre-diagnostic data by combining operations at each of the two stage of the outbreak detection task: data preprocessing and residual monitoring. In this work we focus on combining monitoring charts. By setting an objective cost function that takes into account true and false alarm rates, we are able to formulate this as an integer programming problem and to find the weights that optimize the combination method. Initial empirical experiments confirm the advantage of this portfolio approach. But further experimentation is needed.

In the future, in addition to expanding the adaptive combination, we plan to study a machine-learning method that automatically adjusts the combination weights based on current and recent performance, and on the most recent weight vector. The idea is to penalize individual methods whenever the combined method misses a true alarm or detects a false alarm.

Another step is to explore the relationship between linear combinations of control charts and the use of wavelets. It is known that several control charts are related to different resolutions of the Haar wavelet Aradhye et al. (2003). The extent of this relationship to the combination method is therefore an interesting open question.

Chapter 4

Directionally-sensitive multivariate control charts

An important limitation of current detection algorithms is that they monitor each data stream univariately, when in practice the number of data streams is usually large. Among the different sources of biosurveillance data are over-the-counter medication sales, nurse hotlines, and emergency department visits from a single of multiple locations, etc. In this section we focus on multiple time series arriving from a single data source, or from multiple data sources. A central question that arises is whether to monitor each series separately and then to combine the results in some fashion, or instead to monitor the series in a multivariate fashion. Current temporal monitoring in biosurveillance systems is done univariately, by applying univariate control charts to each series separately. This multiple testing results in a very high false alert rate, leading many users to ignore alerts altogether. An alternative is to use multivariate control charts, which have traditionally been used in industry for monitoring multiple series simultaneously. This alternative to employing multiple univariate charts simultaneously helps avoid the multiple testing phenomenon.

Furthermore, multivariate control charts take advantage of the correlation structure between individual series, thereby having a higher potential of detecting small signals that are dispersed across series. However, several theoretical and practical issues arise regarding the usefulness of multivariate control charts in biosurveillance. In particular, the characteristics of biosurveillance data and the conditions under which monitoring is performed usually mean that standard assumptions are not met, thereby rendering theo-

retical derivations questionable. This chapter tackles the challenge of directional-sensitive monitoring *in practice*, in terms of the sensitivity and robustness of several multivariate monitoring methods for detecting outbreak signatures in multivariate biosurveillance-type data.

Fricker (2007) and Fricker et al. (2008) offer comparisons between Hotelling and Multivariate CUSUM charts, and between Multivariate CUSUM and Multivariate EWMA charts. They evaluate performance on simulated multivariate normal data with a seasonal sinusoidal cycle and a random fluctuation, as well as on authentic data. Our work differs from the two papers above in that we examine the robustness of such methods to various assumption violations as well as to practical data conditions.

The novelty of this work is by isolating the properties of authentic data streams that violate the basic assumptions of multivariate monitoring methods, and evaluating the sensitivity and robustness of the methods to these violations. Our work focuses on the evaluation of computationally-feasible, model-based methods for monitoring multivariate data in a directionally-sensitive way. The main contribution is the evaluation of these methods in light of actual data characteristics and application conditions in disease outbreak detection, which often violate the assumptions of these methods. In addition, we assess the effects of practical factors such as the number of series, the amount of training data, and the level of cross-correlation on performance.

The outline of our framework applied to multivariate monitoring is shown in the float chart in Figure 4.1 depicts the outline of this project.

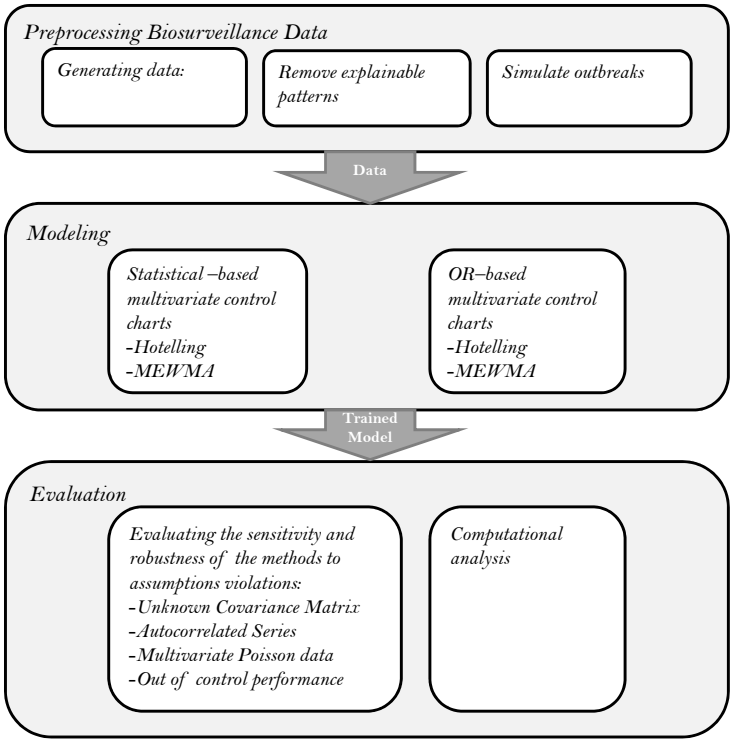

Figure 4.1: Biosurveillance schematic: multivariate monitoring.

4.1. Directional monitoring

Three popular univariate control charts are the Shewhart chart, the Cumulative Sum (CuSum) chart, and the Exponentially-Weighted Moving Average (EWMA) chart. Multivariate extensions exist for each of these: The Hotelling T^2 chart, the MCuSum and the MEWMA, respectively (see e.g. Crosier (1988); Pignatiello and Runger (1990); Hotelling (1947); Lowry et al. (1992); Lowry and Montgomery (1995)). These multivariate charts are aimed at detecting a change in one or more of the process means in any direction. However, in the context of biosurveillance the interest is in detecting only an increase in one or more of the means (indicative of disease outbreak). The hypothesis is that an epidemic will manifest in the series as an increase in daily counts. In the univariate case there are simple corrections of the bi-directional charts to accommodate a one-directional

change. In the multivariate case correcting for directional sensitivity is more complicated. One approach has been to empirically adjust the threshold of ordinary multivariate charts to achieve a given false alarm rate and then to evaluate its true alerting properties (Fricker, 2007). In this chapter, we focus on two approaches that are useful for practical implementation: Follmann (1996) provides a correction for the ordinary Hotelling chart (Hotelling, 1947) and Testik and Runger (2006) present a quadratic-programming approach to estimate the in-control mean vector. These two methods take two different approaches to yield directionally-sensitive Hotelling charts: The former is a statistical approach, while the latter is an operations-research approach. We describe each of these methods in detail in Section 4.3, where we also generalize them to obtain directionally-sensitive MEWMA charts. Section 4.4 compares the performance of the different charts. Using a large array of simulated data, we compare the performance of the directionally-sensitive Hotelling and MEWMA charts as a function of the number of monitored series, the cross-correlation structure, and the amount of training data required for estimating the covariance matrix. We then evaluate the robustness of the charts to underlying assumptions of normality and independence. In Section 4.5 the four charts are applied to a set of authentic biosurveillance data. Conclusions are given in Section 4.6.

4.2. Multivariate control charts

We use the following notation throughout the chapter. Let $\mathbf{X}_t = \{X_t^1, \ldots, X_t^p\}$ be a p-dimensional multivariate normal vector with mean $\boldsymbol{\mu} = \{\mu^1, \ldots, \mu^p\}$ and covariance matrix Σ. We assume that at every time point t, a *single* observation is drawn from each of p series.

In the next sections, we describe the Hotelling control chart, followed by modifications by Follmann (Follmann, 1996) and by Testik and Runger (Testik and Runger, 2006) for directional-sensitivity. We then expand their methods to obtain directionally-sensitive MEWMA charts.

4.2.1 Hotelling's T^2 Control Chart

The multivariate extension of the ordinary Shewhart chart is the χ^2 chart, where the monitoring statistic is (Hotelling, 1947):

$$\chi_t^2 = (\mathbf{X}_t - \boldsymbol{\mu})' \Sigma^{-1} (\mathbf{X}_t - \boldsymbol{\mu}). \tag{4.1}$$

This is the squared statistical distance (also known as *Mahalanobis distance*) of the observation on day t from the in-control mean vector. Under the null no-shift hypothesis the statistic follows a $\chi^2(p)$ distribution. The alarm threshold is therefore $\chi^2_\alpha(p)$, where $\chi^2_\alpha(p)$ is the α quantile of the $\chi^2(p)$ distribution.

When Σ is unknown, it is estimated from data of length tr (referred to as *training data*). The estimated covariance matrix is denoted by S. The statistic, known as the Hotelling T^2 statistic, is given by (Montgomery and Klatt, 1972)

$$T_t^2 = (\mathbf{X}_t - \boldsymbol{\mu})' S^{-1} (\mathbf{X}_t - \boldsymbol{\mu}). \tag{4.2}$$

Under the null hypothesis of no shift, $T^2 \sim \frac{p(tr+1)(tr-1)}{tr(tr-p)} F(p, tr-p)$. The alerting threshold

for the Hotelling T^2 statistic is therefore

$$\frac{p(tr+1)(tr-1)}{tr(tr-p)}F_\alpha(p, tr-p), \tag{4.3}$$

where $F_\alpha(p, tr-p)$ is the α quantile from the $F(p, tr-p)$ distribution.

4.2.2 Multivariate EWMA

The standard univariate EWMA chart is based on the statistic:

$$Z_t = \lambda X_t + (1-\lambda)Z_{t-1}, \tag{4.4}$$

where $0 < \lambda \leq 1$ is the smoothing parameter (typically chosen in the range [0.1,0.3]). Under the null hypothesis of no shift, this statistic follows a normal distribution with mean μ and asymptotic variance $s^2_{EWMA} = (\lambda/(2-\lambda))s^2$, where s is the standard deviation of X_t estimated from historical data. The alerting thresholds are therefore $\mu \pm k \times s^2_{EWMA}$, where the constant k is commonly set to 3.

A multivariate extension of EWMA (MEWMA) is first to create an EWMA vector from each of the univariate p series (Lowry et al., 1992):

$$\mathbf{Z}_t = \Lambda \mathbf{X}_t + (1-\Lambda)\mathbf{Z}_{t-1}, \tag{4.5}$$

where $\mathbf{Z}_t$ is the EWMA p-dimensional vector at time t, $\mathbf{X}_t$ is the p-dimensional observation vector, and Λ is a diagonal matrix with smoothing parameters $\lambda_1, \ldots, \lambda_p$ on the diagonal.

The monitoring statistic is then:

$$\mathbf{Y}_t = \mathbf{Z}_t' \Sigma_Z^{-1} \mathbf{Z}_t \tag{4.6}$$

Lowry et al. (1992) showed that for a sufficiently large t (i.e., after a start up period) and for $\lambda_1 = \cdots = \lambda_p = \lambda$ the covariance matrix, Σ_Z, is given by

$$\Sigma_Z = \frac{\lambda}{2-\lambda}\Sigma. \tag{4.7}$$

The alerting threshold is still $\chi^2_\alpha(p)$.

Note that for small t, the (k,l)th element in the covariance matrix Σ_Z is given by

$$Cov(Z_k, Z_l)_t = \lambda_k \lambda_l \frac{1-(1-\lambda_k)^t(1-\lambda_l)^t}{\lambda_k + \lambda_l - \lambda_k\lambda_l}\sigma_{k,l}, \tag{4.8}$$

where $\sigma_{k,l} = Cov(X_k, X_l)$. If $\lambda_1 = \cdots = \lambda_p$, then (4.8) reduces to

$$\frac{\lambda}{2-\lambda}\left(1-(1-\lambda)^{2t}\right)\Sigma.$$

4.3. Directionally-Sensitive Multivariate Control Charts

In order to use multivariate control charts for detecting a parameter shift in *one* direction (an increase or decrease), there have been several approaches. One approach is to modify the non-directional multivariate monitoring statistic, as suggested by Follmann (1996). We describe this method in Section 4.3.1. Another approach is to construct likelihood-ratio statistics for the alternative hypothesis. Most of the derivations along these lines are theoretical in nature and are hard to implement for more than 2 series.

However, Testik and Runger (2006), taking an operations research approach, proposed an alternative formulation which can be implemented in practice for many series. We describe their method in Section 4.3.2. We then generalize these two approaches to obtain directionally-sensitive MEWMA charts.

4.3.1 Follmann's Approach

Follmann (1996) introduced a correction to the standard Hotelling statistic that adjusts it for directional sensitivity. The corrected statistic is given by

$$\chi_t^2+ = (\mathbf{X}_t - \boldsymbol{\mu})'\Sigma^{-1}(\mathbf{X}_t - \boldsymbol{\mu}), \tag{4.9}$$

Note that χ_t^2+ in Follmann's notation is equivalent to the ordinary χ_t^2 statistics. The '+' sign indicates that we are interested in mean increase only. $\mathbf{X}_t$ in the equation is the sample mean vector at time t. An alert is triggered when $\{\chi_t^2+ > \chi_{2\alpha}^2(p)\ and\ \sum_{j=1}^{p}(X_t^j - \mu^j) > 0\}$. This means that we require the sum of the elements of the mean vector at time t to exceed the series mean in order to alert.

Follmann proves that the procedure has type I error rate equal to 2α whether or not the covariance is known, and uses simulations to illustrate its power and to compare it to more complicated likelihood ratio tests.

Albers et al. (2001) show that the test statistic in (4.9) is not invariant to scale transformations. They propose a simple fix where instead of Σ the correlation matrix is used.

4.3.1.1 Extending Follmann's method to MEWMA charts

We extend the method proposed by Follmann in order to convert the ordinary MEWMA chart to a directionally-sensitive MEWMA chart. This is done by replacing Σ with Σ_Z in equation (4.9) (or equivalently, the original correlation matrix R with R_Z), and $\mathbf{X}_t$ by $\mathbf{Z}_t$. In other words, the alerting statistic is given by:

$$\chi_t^2+ = \mathbf{Z}_t' \Sigma_Z^{-1} \mathbf{Z}_t, \tag{4.10}$$

and an alert is triggered when $\{\chi_t^2+ > \chi_{2\alpha}^2(p) \; and \; \sum_{j=1}^{p} \left(Z_t^j - \mu^j \right) > 0\}$.

Additionally, if an outbreak occurs on day τ, we allow an option of restarting the control chart by setting $\mathbf{X}_\tau = \boldsymbol{\mu}$. The action of restarting reduces the "ringing effect" of the algorithm, by removing the sequence of alerts that follow an initial alert when there is a gradual increase in the vector mean.

A method similar to that of Follmann's was proposed by Joner Jr et al. (2008) and Fricker et al. (2008). The authors propose the following statistic, which is the maximum between the Lowry et al. (1992) MEWMA statistic and 0:

$$Z_t = \max\{\lambda(X_t - \mu_0) + (1 - \lambda)Z_{t-1}, 0\}$$

The method is evaluated using simulated multivariate normal data, with simulated Poisson outbreaks. Although we do not include this method in our sensitivity analysis, we believe that its performance is equivalent to that of Follmann's MEWMA.

4.3.2 Testik and Runger's Quadratic Programming Approach

A different approach for obtaining directional sensitivity is based on deriving the monitoring statistic from the likelihood ratio, which is equal to the maximum likelihood under the alternative hypothesis of a directional shift (either positive or negative) divided by the null likelihood. Nüesch (1966) showed that (twice the) log-likelihood ratio is

$$2l(\boldsymbol{\mu}) = \hat{\boldsymbol{\mu}}\Sigma^{-1}\hat{\boldsymbol{\mu}}, \tag{4.11}$$

where $\hat{\boldsymbol{\mu}}$ is the maximum likelihood estimator. The monitoring statistic $\chi_t^2 = \mathbf{X}_t'\Sigma^{-1}\mathbf{X}_t$ is therefore proportional to the likelihood ratio under the alternative hypothesis. Nüesch proposed $l(\hat{\mu})$ as an alternative monitoring statistic, where $\hat{\mu}$ maximizes the log likelihood. Testik and Runger (2006) showed that this can be formulated as an easily solvable quadratic programming problem if the data are first standardized:

$$\widetilde{\mathbf{X}_t} = \Sigma^{-1/2}\mathbf{X}_t, \tag{4.12}$$

$$\widetilde{\boldsymbol{\mu}} = \Sigma^{-1/2}\boldsymbol{\mu}, \tag{4.13}$$

Nüesch also proved that the corresponding threshold is given by:

$$P(\chi^2 > c^2) = \sum_{i=1}^{p} w(i)P(\chi_i^2 > c^2), \tag{4.14}$$

where $w(i)$ is the probability that $hat\boldsymbol{\mu}$ has exactly i nonzero elements. χ_i^2 is a chi-squared random variable with i degrees of freedom, and c^2 is a constant threshold value.

The problem has been to compute the weights $\omega(1), \ldots, \omega(p)$. While theoretical derivations exist, they are typically hard to implement beyond $p \geq 3$. Testik and Runger (2006) obtained the weights empirically, by simulating p-dimensional multivariate normal data with $\boldsymbol{\mu} = \hat{\boldsymbol{\mu}}$ and Σ and estimating the weights from the simulated data. It is important to note, however, that this approach assumes a known covariance matrix. The problem, in terms of $\widetilde{\mu}$, is then:

$$\hat{\boldsymbol{\mu}}_t = arg \min_{\widetilde{\boldsymbol{\mu}} \geq 0} (\widetilde{\mathbf{X}_t} - \widetilde{\boldsymbol{\mu}})' (\widetilde{\mathbf{X}_t} - \widetilde{\boldsymbol{\mu}}) \tag{4.15}$$

4.3.2.1 Extending TR's Method to MEWMA Charts

We extend Testik and Runger's method to obtain a directionally-sensitive MEWMA chart. This is achieved by replacing Σ with Σ_Z and $\mathbf{X}_t$ by $\mathbf{Z}_t$ in equations (4.12) and (4.13). In other words, our standardized data and means are:

$$\widetilde{\mathbf{Z}_t} = \Sigma_Z^{-1/2} \mathbf{Z}_t, \tag{4.16}$$

$$\widetilde{\boldsymbol{\mu}} = \Sigma_Z^{-1/2} \boldsymbol{\mu}, \tag{4.17}$$

and the problem in terms of $\boldsymbol{\mu}_z$ is therefore:

$$\hat{\boldsymbol{\mu}}_t = arg \min_{\widetilde{\boldsymbol{\mu}} \geq 0} \widetilde{\mathbf{Z}_t} - \widetilde{\boldsymbol{\mu}})' (\widetilde{\mathbf{Z}_t} - \widetilde{\boldsymbol{\mu}}) \tag{4.18}$$

In this formulation we do not implement a restart condition for computational reasons. According to TR's approach the entire series is transformed prior to the monitoring

action. A restart action would therefore require a re-transformation of the data after each alert, thereby increasing the run time by a factor equal to the number of alerts. An alternative on-the-fly implementation would apply the transformation on a daily basis (rather than in retrospect). This would not allow using the simple matrix operations for obtaining $\hat{\boldsymbol{\mu}}_t$, but it would easily incorporate the restart condition.

4.4. Performance and Robustness Comparison

We set out to evaluate and compare the four different directionally-sensitive multivariate control charts: Hotelling and MEWMA using Follmann's method, and Hotelling and MEWMA using Testik and Runger's method. We compare their actual in-control performance as a function of the number of monitored series (p), the covariance structure (Σ) and their robustness to assumption violations that are likely to occur in practice. We first describe the simulation setup, and then examine the different factors and their effect on performance. Finally, we examine performance in the presence of a mean increase. We consider shifts of different magnitude, shape, and their presence in subsets of the series. We also discuss outbreak detection in the presence of mean decreases.

4.4.1 Simulation Setup

We generate multivariate normal data and vary the level of correlation between series ($\rho = 0.1, 0.3, \ldots, 0.9$) and the number of dimensions ($p = 2, 3, \ldots, 20$). The length of each series is set to $T = 1000$ time points. To compute a false alert rate, the number of false alerts is divided by T. The desired false alert rate is set to $\alpha = 0.05$. The threshold is computed according to equations (4.10) (Follmann) and (4.14) (TR). For each combination

of ρ and p, 20 replications are generated. This creates a distribution of false alert rates for each combination. We then examine the sensitivity and the robustness of the control charts as we change the simulation setting.

We use *R*2.4.0 (http://cran.r-project.org/) to implement our simulation.

4.4.2 Impact of Cross-Correlation and Number of Series

We start by evaluating the actual false alert rates of the different charts by assuming that Σ is known and given by:

$$\begin{pmatrix} 1 & \rho & \rho & \rho \\ \rho & 1 & \rho & \rho \\ \rho & \rho & ... & \rho \\ \rho & \rho & \rho & 1 \end{pmatrix},$$

where $\rho \in \{0.1, 0.2...0.5\}$. For the MEWMA charts we set $\lambda_1 = \ldots = \lambda_p = \lambda \in \{0.3, 0.5\}$. Figures 4.4-4.5 compare the distribution of false alerts (FA) as a function of the number of series p (on the y-axes) and correlation ρ (across panels) for the four methods. In all cases the charts were set to an FA of $\alpha = 0.05$. The results are provided as side-by-side boxplots, with the mean FA represented as a solid dot; the whiskers extending to the 5^{th} and 95^{th} percentiles; and 'outliers' represented as hollow dots.

Overall we see that all charts produce false alerts that are centered around $FA = 0.05$. The inter-quartile range is approximately (0.04,0.06) for all methods, although the MEWMA charts have a slightly larger variance when λ is small ($\lambda = 0.3$). Note that the number of series does not appear to affect the false alert rate. This is not surprising as

the thresholds in all four methods are a function of p. Another interesting observation is that the distribution is very stable across the different correlation levels for all charts. Based on these results, in our next experiments we set $\rho = 0.5$ and for MEWMA charts we set $\lambda = 0.3$ (a popular choice in practice.)

4.4.3 Robustness to Assumptions

Next, we study the robustness of the four charts to assumption violations. We examine the length of training data required to estimate an unknown covariance matrix. We then relax the assumption of normality of the underlying observations and examine the behavior of the charts when the series are autocorrelated and when the data come from a multivariate Poisson distribution (a common scenario with count data).

4.4.3.1 Unknown Covariance Matrix

In this setting we set the cross-correlation to $\rho = 0.5$. We assume that the covariance structure is unknown and is approximated from a training data (tr) of varying length (using the Pearson method (Rodgers and Nicewander, 1988)). We examine the false alert rate of the charts as a function of the length of the training data.

Unlike bidirectional monitoring methods, in which the F distribution is used when Σ is unknown (Montgomery and Klatt, 1972), the use of the F distribution in Follmann's and TR's methods is inadequate: In Follmann's method, recall that the χ^2 statistic has a threshold of 2α rather than α (see equation (4.9)). Therefore, replacing the χ^2 statistic with an F statistic requires an additional modification to the alerting threshold, which

is not straightforward (see equation (4.3)). In TR's method, the threshold is computed empirically. Theoretically, we could simply replace the χ^2 distribution in equation 4.14 with an F distribution. However, we find that the χ^2 formulation results in a much lower false alert rate compared to the F distribution. We therefore use a χ^2 test in our simulation study.

Figure 4.6 compares the false alert rate of the four methods. The training data length varies from $tr = 100$ time points (left panel) to $tr = 500$ time points (right panel). We see that Follmann's Hotelling chart has a slightly lower average false alert rate than TR's Hotelling. This improved performance is more significant when the training period is short (e.g., $tr < 300$) and the number of series is high, yet the variance is higher. Follmann's MEWMA, on the other hand, has both a lower average false alarm rate and lower variance.

Overall, we observe that for $p \leq 5$ the average false alert rate is centered around the desired threshold of $FA = 0.05$, independent of the training data length. This implies that 100 data points are sufficient to estimate the covariance matrix accurately. However, for $p > 5$ the performance depends on the length of the training data. When $tr = 100$ the false alert rate increases exponentially in the number of series. When $tr = 200$ the rate increases linearly in the number of series. The average false alert rate converges to 0.05 only when the training data include at least 300 time points (almost one year of data, for daily series!).

Our results coincide with previous literature findings. Jensen et al. (2006) reviewed the effect of parameter estimation on control charts performance. They found that when using univariate EMWA charts, the smaller the value of λ, the larger the required sample

size for ensuring performance similar to that of a chart based on known parameters. Jones et al. (2001) recommended using 100 samples of size $n = 5$ for $\lambda = 0.5$ and 400 samples of size $n = 5$ for $\lambda = 0.1$. For Hotelling charts, Nedumaran and Pignatiello (1999) recommended using sample sizes of at least 200 when the number of observations is $n = 5$ and the dimension is $p = 3$.

4.4.3.2 Autocorrelated Series

In this setting we examine the impact of autocorrelation on the false alert rate of the different charts. It has been shown that biosurveillance daily time series tend to be autocorrelated (see, e.g., Burkom et al. (2007)). We set the cross-correlation to $\rho = 0.5$ and vary the autocorrelation coefficient in the range $\theta \in \{0.05, 0.15...0.35\}$.

To understand the impact of autocorrelated series on chart performance, we first assume a known Σ. Figure 4.7 compares the four charts. The false alert rate of the Hotelling variants appears similar for both methods and centered around the desired threshold of $FA = 0.05$. In contrast, in the MEWMA charts the false alert rate increases significantly as θ increases, and more so for TR's method. We observe, for example, that even when the number of series is $p = 2$ and the cross-correlation is $\theta = 0.35$, Follmann's false alert rate is approximately 0.18 and TR's is 0.2. For $p = 10$ series the corresponding rates are 0.4 and 0.65.

To reduce false alerts, we next examine a version of Follmann's MEWMA where the univariate EWMA statistics are restarted after alerts (i.e., if an alert is set on day t, we set $Z_t = \mu$). As mentioned earlier, for computational reasons, we did not implement TR's method with restarting. Figure 4.8 shows that the false alert decreases by a factor of 4,

compared to the MEWMA without restart (Figure 4.7, 3rd panel).

4.4.3.3 Multivariate Poisson data

We now relax the normal distribution assumption, as it is often violated in authentic data. Instead, we generate multivariate Poisson data (using the method in Chapter 5) with varying arrival rate $\lambda \in \{1, 5, 10, 20\}$. Biosurveillance data are typically daily count data. In some instances the counts are sufficiently high to justify normal-based control charts, while in other cases the counts might be too low. In low count situations a reasonable approximation that has been used in practice is a Poisson distribution (see e.g., Kleinman et al., 2004; Joner Jr et al., 2008). Examples are daily counts of cough complaints in a small hospital, or daily counts of school absences in a local high school. Using a multivariate Poisson structure enables us to evaluate the performance of the control charts in low-count data.

We evaluate and compare the methods when Σ is known (Figure 4.9) and observe that MEWMA outperforms Hotelling in terms of FA rate. The difference in performance is more pronounced as λ decreases, and when the number of series is large (roughly $p \geq 5$). Both Follmann and TR methods have equivalent performance. Similar to the findings in Stoumbos and Sullivan (2002), we find that MEWMA charts are more robust to non-normality than Hotelling charts. As the authors suggest, the robustness property might later cause a decreased rate of true alerts for detecting spike outbreaks.

4.4.4 Out of control performance

In this section we evaluate the performance of the four control charts in the presence of unexpected anomalies. Since the exact shape and magnitude of a disease outbreak manifestation in pre-diagnostic data is unknown, we consider two shapes that represent abrupt and incremental signatures of varying magnitudes. In particular, we consider one-day spikes and multi-day lognormal increases. For each control chart we examine the rate of which it identifies true outbreaks (the ratio of detections, denoted TA), the time to detection (the number of time points until the first true alert), and the false alert rate.

4.4.4.1 Injecting Outbreaks

We consider *iid normal data* with correlation $\rho = 0.5$ and estimate the correlation matrix from a history of $tr = 500$ time points that do not contain outbreak signatures. We vary the number of series in the set $p \in \{4, 8, 12, 16\}$ and inject outbreak signatures into subsets of the p series of size $s \in \{25\%, 50\%, 75\%, 100\%\}p$.

Spikes. In the first experiment we inject single day spikes into the data. The magnitude of the spike varies in the range $o \in \{0.5, 1, 1.5, ..., 4\} \times \vec{\sigma}$, where $\vec{\sigma}$ is the series standard deviation (in our experiments we set $\vec{\sigma} = 1$). In each experiment 20 spikes are injected into a subset of series at different time points, and the resulting true and false alert rates are computed.

Figures 4.10-4.11 show the true alert rate (TA) of the control charts when spikes of different magnitudes are injected into all p series. Figures 4.12-4.13 present the same for spikes injected into 25% of the series. For the MEWMA chart we evaluate the TA

with and without restarting after an alert. As expected, the TA rate is higher for the Hotelling charts. Also, it appears that TR's method outperforms Follmann's in terms of true detections in both Hotelling and MEWMA charts. The difference in performance is more noticeable when the subset is small.

To further analyze the true alert rate and its determinants, we examine the relationship between true and false alerts while controlling for other factors (outbreak size o, subset size s and number of series p). We use a linear regression model to explore the magnitude of the cross effect between false and true alert, as shown by equation (4.19). While the relationship between the alert factors is not necessarily linear, the simplicity of this analysis enables us to clearly illustrate the increase of false alerts in the presence of true alerts.

$$TA = \beta_0 + \beta_1 \times FA + \beta_2 \times s + \beta_3 \times p + \beta_4 \times o + \epsilon \tag{4.19}$$

Table 4.1 shows the output of the estimated model (p-values of the estimates are given in parentheses, coefficients significance at 5% are in bold). We find that controlling for all factors, TR's Hotelling chart performs on average 30% better than Follamnn's in terms of percentage of true alert for every additional 1% in false alert rate. The performance is equal only when the subset size is close to 100%. Similar results are observed for MWEMA without restart. Another observation is the strong correlation between TA and FA in the MEWMA control charts. Controlling for all other factors, a 1% increase in FA rate results in an average 5.35% (Follmann) and 4.24% (TR) increase in TA. MEWMA with restart, on the other hand, has the exact opposite relationship, presumably since the restarting action erases the history and allows the control chart to re-accumulate small deviations

Table 4.1: The relationship between TA and FA rates

Coef	**Follmann's Hotelling**	**TR's Hotelling**	**Follmann's MEWMA**	**Follmann's MEWMA with Restart**	**TR's MEWMA**
$\widehat{\beta}_0$	**0.22**	**0.33**	**-0.31**	-0.05	**-0.21**
	(0)	(0)	(0)	(0.27)	(0)
$\widehat{\beta}_1$	-0.62	-0.44	**5.35**	**-3.78**	**4.24**
	(0.26)	(0.39)	(0)	(0)	(0)
$\widehat{\beta}_2$	0.01	**-0.14**	0.01	-0.02	**-0.11**
	(0.64)	(0)	(0.21)	(0.19)	(0)
$\widehat{\beta}_3$	**0.2**	**0.2**	**0.2**	**0.25**	**0.22**
	(0)	(0)	(0)	(0)	(0)
$\widehat{\beta}_4$	**0.01**	**0.01**	**0**	**0.01**	**0.01**
	(0)	(0)	(0)	(0)	(0)
Adj-R^2	0.58	0.58	0.71	0.7	0.75

from the means.

Log-normal signatures. Next, we inject into the data multi-day lognormal progression outbreaks. As with the spikes, we vary the magnitude of the signature and the fraction of 'infected' series. We inject a single signature in each experiment and examine the number of time points until the first successful detection as well as the true alert rate. Results are shown for subset size $s = 25\%p$ in Figures 4.14-4.15. Panels correspond to different magnitudes (we plot partial magnitudes for brevity). Boxplots show the distribution of time to detection (black dots are average time to detection), *condtional* on outbreak being detected.

We observe that MEWMA chart perform better than Hotelling charts. This result is expected, as MEWMA charts are designed to detect gradually increasing signals. We again see that TR's method outperforms Follmann's both in terms of TA and time to detection.

4.4.4.2 Detection of Mean Increases in the Presence of Mean Decreases

Although we are interested in detecting only increases in the process mean, it is possible that due to data quality anomalies, one or more of the means will decrease (e.g., due to reduced reporting or problems with data recording on a certain day). We therefore evaluate the performance of the control charts when we inject both *positive* and *negative* spikes, where negative spikes represent such anomalies. Recall, however, that for detecting disease outbreaks rather than data quality anomalies, we are only interested in detecting mean *increases* (i.e., positive spikes).

We consider a bivariate dataset ($p = 2$). In each experiment we inject 20 *positive* spikes into the first series and 20 *negative* spikes into the second series. *Positive and negative spikes are injected on the same days.* The magnitude of the spikes varies in the range $o \in \{1, 2, 3\} \times \vec{\sigma}$, where $\vec{\sigma}$ is the series standard deviation.

Figure 4.16 (top panels) depicts the resulting true alert detection rate of the four methods. For comparison, we repeat the same experiments with positive spikes only (middle panels) and with negative spikes only (injected into a single series, bottom panels). Note that 'True Alert' (TA) in the bottom panels refers to the detection rate of decreases in the mean. We see that, surprisingly, TR's Hotelling and MEWMA charts detect mean decreases and alert as if they were also outbreaks (i.e., mean increases). The result is an 'improved' performance when both positive and negative spikes are present, yet poor performance (increased false detection) when only negative spikes are injected into the data.

As for Follmann's charts, a simultaneous decrease and increase in the mean vector

leads to poor performance of Follmann's Hotelling chart. This result is expected, as Follmann's method alerts when the summation of the series is greater than zero (see Equation (4.10)). In contrast, although Follamnn's MEWMA chart performs similar to the Hotelling chart, it performs equally well in the presence and the absence of decreasing spikes.

4.5. Results for Authentic Data

We now examine the behavior of the four control charts when applied to the authentic biosurveillance data described in Chapter 2, Section 2.1.2. The authentic data contain several explainable patterns such as seasonality and day of week effects. We preprocess the series to remove these patterns using Holt-Winter's exponential smoothing to remove day-of-week effects, seasonality, and autocorrelation.

We compute the FA rate by applying the four control charts to the preprocessed data. To evaluate TA rate and time to detection, we inject 32 spikes of magnitude o, where $o \sim u[1,4] \times \vec{\sigma}$ into a random subset of the 13 series. The covariance structure is estimated from the first year of data (365 time points). Applying each control chart to the data with injected signatures we compute their FA and TA rates. The results are shown in Table 4.2. There are two main observations: (1) Follmann's MEWMA chart with restart alerts the least, whether or not there are outbreak signatures, and (2) TR's Hotelling chart is most sensitive: it has the highest TA rate, but also the second highest FA rate. The highest FA rate is obtained with TR's MEWMA, but this is likely due to the lack of restart after an alert. Note also that the FA rate computed before and after the signature injections are similar.

Table 4.2: Performance of the control charts on authentic data

Method	**FA rate in the absence of outbreaks**	**FA rate in the presence of outbreaks**	**TA rate**
Hotelling, Follmann	0.16	0.15	0.80
Hotelling, Testik	0.16	0.20	0.88
MEWMA, Follmann	0.16	0.17	0.84
MEWMA with restart, Follmann	0.08	0.07	0.78
MEWMA, Testik	0.21	0.23	0.81

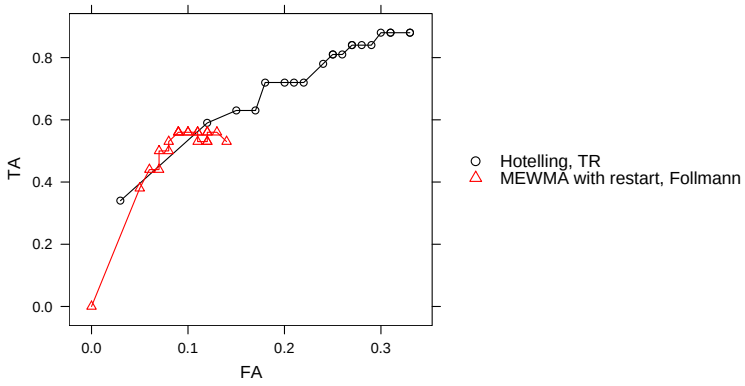

Figure 4.2: True vs. false alert rates for TR's Hotelling chart vs. Follmann's MEWMA chart with restarts

To further explore these two results and the relationship between TR's Hotelling and Follmann's MEWMA (with restart), we examine their performance across a range of FA rates ($[0, 0.2]$). For a higher sensitivity comparison, we examine only outbreaks of smaller magnitude ($o \sim u[0.5, 2.5] \times \vec{\sigma}$). Results are shown in Figure 4.2. We see that the low FA rate is controlled by Follmann's MEWMA chart and the high TA rate is controlled by TR's Hotelling chart. These results are inline with those obtained from the simulated data. The conclusion is therefore that *the choice of chart should be driven by the tradeoff between true and false alerts required by the user.*

Finally, to evaluate the advantages of each of the four multivariate control charts we compare them against univariate monitoring where univariate Shewhart charts are applied

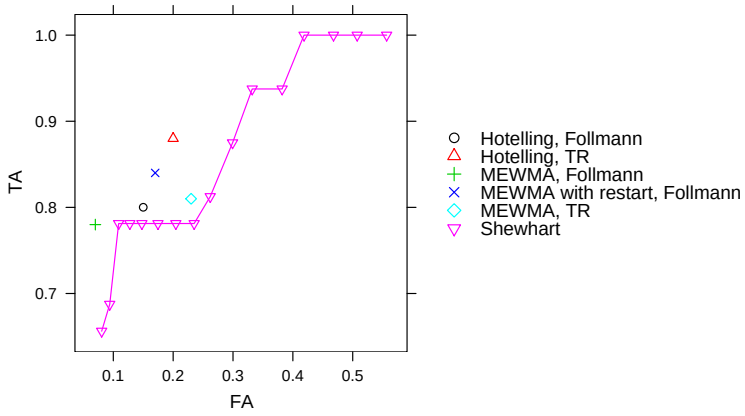

Figure 4.3: True vs. false alert rates; comparing multivariate control charts with multiple-univariate Shewhart charts

simultaneously to each of the $p = 13$ series. The rule for alerting is when at least one of the charts alerts. We vary the *actual* FA rate of the multiple-univariate charts between $[0, 0.6]$ and observe the TA rate. Results are shown in Figure 4.3. We can see that *all of the multivariate charts are Pareto efficient compared to the multiple-univariate Shewhart.*

4.6. Conclusions and Future Directions

We present and evaluate four tools for monitoring multivariate time series for the purpose of detecting anomalies that manifest in a certain direction. The directionally-sensitive multivariate control charts are Follmann's Hotelling, TR's Hotelling, Follmann's MEWMA (with and without restart) and TR's MEWMA. All charts have underlying assumptions such as normality, independence, and knowledge of the covariance structure, which rarely hold in practice. We therefore evaluate and compare their performance when each of the assumptions is violated. We also examine practical issues such as length of the training set, the number of monitored series, and the size of the subset of series in which the outbreak signature appears. All these are manipulated using simulation, where we can

assess the impact of each factor separately. Finally, we apply the charts to authentic data (with and without injected outbreak signatures) and compare their TA and FA rates.

Note that control charts are often applied to raw data rather than pre-processed data in practice. Yet raw data usually violate the normality assumption and also exhibit high levels of autocorrelation. In that case the more robust method (Follmann's) would be preferable. However, clearly the correct approach is to first pre-process the data.

The analysis in this paper is aimed at providing guidelines to biosurveillance systems where multiple time series are monitored. For a given dataset, and based on its characteristics (cross correlation, autocorrelation, etc.) and features (number of series, length of training data, etc.), we can evaluate the performance of each of the multivariate charts in terms of expected false and true alert rates and time to detection. These, in turn, can be used to choose one chart according to costs associated with missed and false alerts. To allow wide implementation of the tools and their incorporation in existing systems, and to be able to compare existing tools to the proposed multivariate charts, we make our code available in the Appendix.

There are several directions for extending this work. First, our simulated data are generated from a mean and covariance structure that do not change over time. In practice, however, data characteristics are subject to changes. To overcome this problem, one can consider estimating the mean and covariance structure repeatedly over time, using a moving window. Our framework is helpful in determining the length of this window.

Secondly, in terms of performance evaluation, once we move from the synthetic environment to authentic data, we no longer have replications of the series. This means

Table 4.3: Summary of performance of multivariate control charts: FA rate as a function of multiple factors

Factor	**Performance**
Follmann vs. TR	Under iid multivariate Normal assumption (known covariance) - no difference in performance
Hotelling vs. MEWMA	Under iid multivariate Normal assumption (known covariance) - higher variance of FA rate for MEWMA
Magnitude of cross correlation	No effect on performance
Number of series	*With no other assumption violations*, the number of series does not affect performance
Length of training data (unknown covariance structure)	• When training data length $tr < 100$, FA rate increases exponentially • When training data length $tr = 200$, FA rate increases linearly • When number of series $p \leq 5$, FA rate is as desired ($\alpha = 0.05$), even for $tr = 100$ • $tr = 300$ is sufficient for accurate estimation of covariance • Follmann's approach results in lower average FA rate but higher variance compared to TR • No significant difference between Hotelling and MEWMA
Autocorrelation	• Hotelling charts not affected by autocorrelation • MEWMA FA rate increases with autocorrelation magnitude and number of series • TR's MEWMA is significantly less robust than Follmann's (almost double FA when high autocorrelation) • Restart condition on Follmann's MEWMA dramatically improves performance (up to 4 times better)
Violation of normal distribution: multivariate Poisson data	• For low count data ($\lambda \leq 5$) FA rate increases by a factor of 3 for Hotelling and 2 for MEWMA • No significant difference between TR and Follmann's approaches

Table 4.4: Summary of performance of multivariate control charts: TA rate as a function of pre-set true alert rate and timeliness

Factor	**Performance**
True alert rate and timeliness	• Overall, TR has a higher detection rate. Especially when signature appears in small subset of the series • Hotelling is better at detecting spikes and high magnitude signatures • MEWMA has higher TA for small magnitudes of a gradual increases

that we cannot assess the statistical significance of the difference between the performance of the different charts (e.g., is TR's Hotelling's TA=0.88 significantly higher than Follmann's MEWMA TA=0.84?). Lotze et al. (2007) propose an elegant approach for creating multiple realizations of authentic data by 'mimicking' the statistical characteristics of an authentic multivariate set of data. Comparing performance over a sample of mimics would then enable assessing statistical significance of differences in false and true alerts as well as time to detection.

Finally, our focus was on comparing four multivariate monitoring tools, and we only briefly touched upon the comparison with multiple-univariate monitoring. Yahav and Shmueli (2006) explore methods for combining univariate algorithms. A more thorough comparison is needed in order to assess under what conditions multivariate monitoring should be preferred over multiple univariate monitoring in practice.

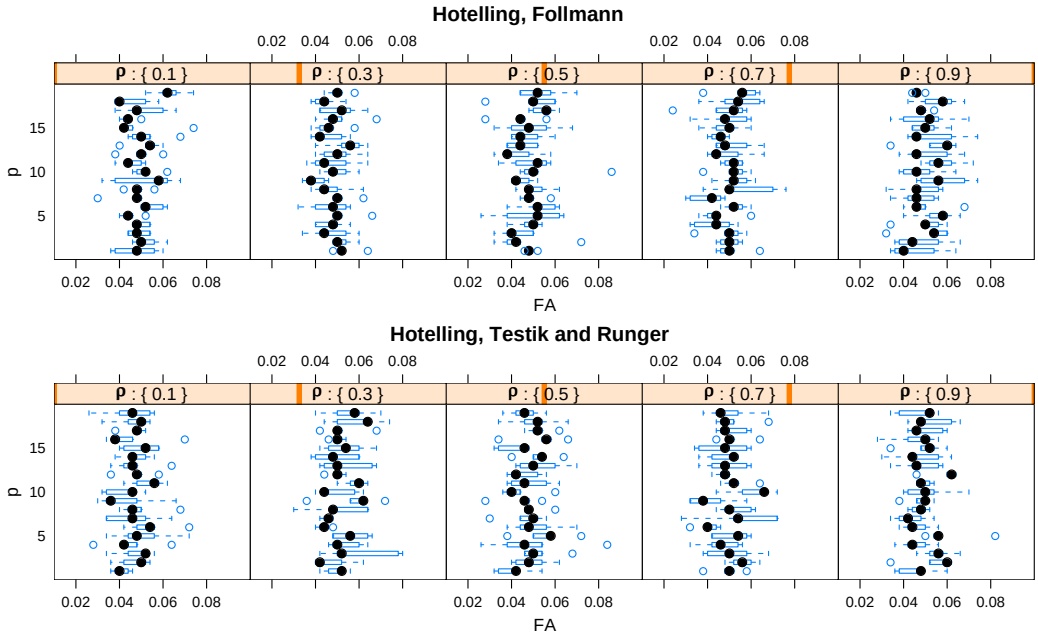

Figure 4.4: Distribution of false alert rate (FA) in directionally-sensitive Hotelling charts as a function of the number of series p and correlation ρ. The charts are all set to FA=0.05

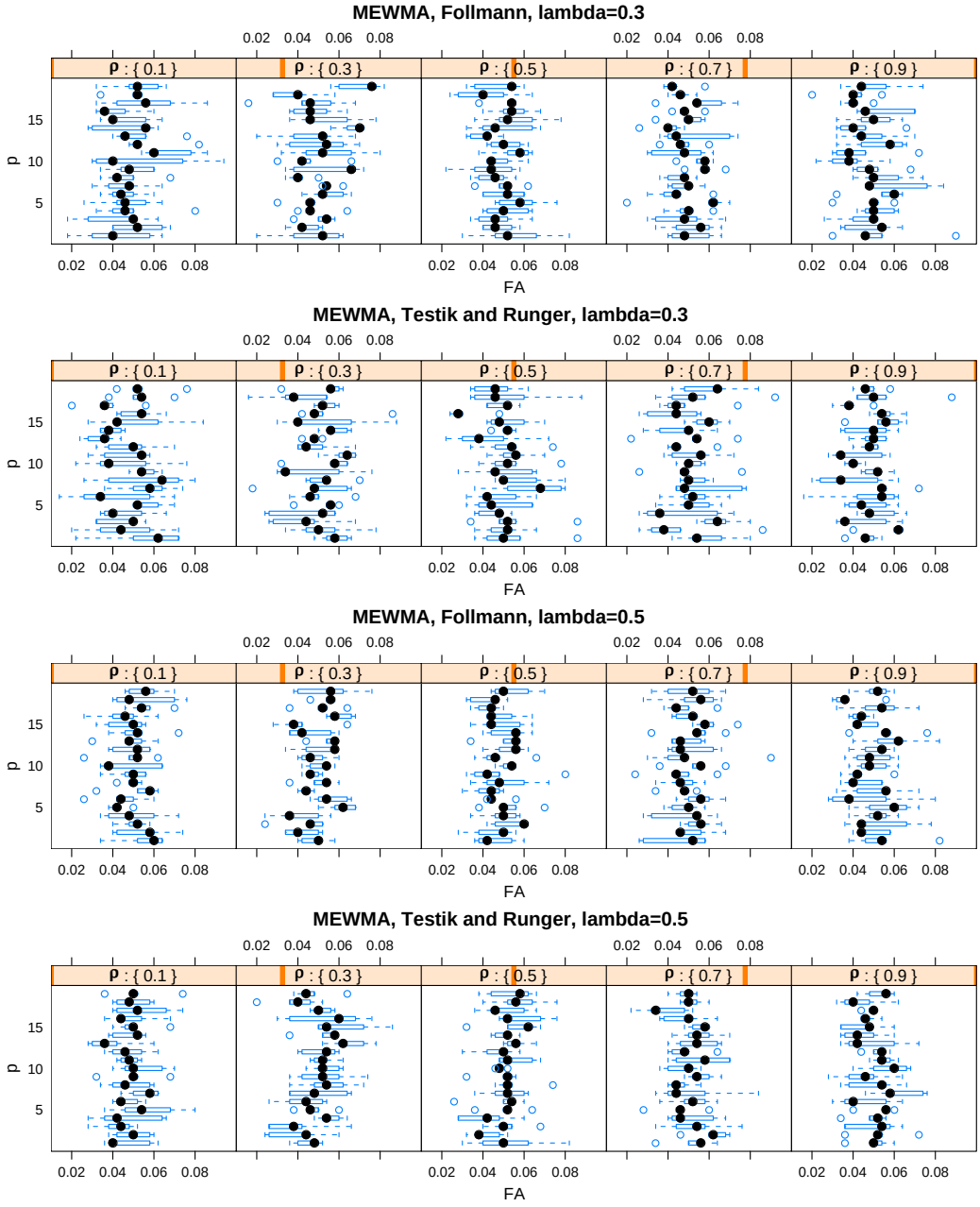

Figure 4.5: Distribution of false alert rate (FA) in directionally-sensitive MEWMA as a function of the number of series p and correlation ρ. The charts are all set to FA=0.05

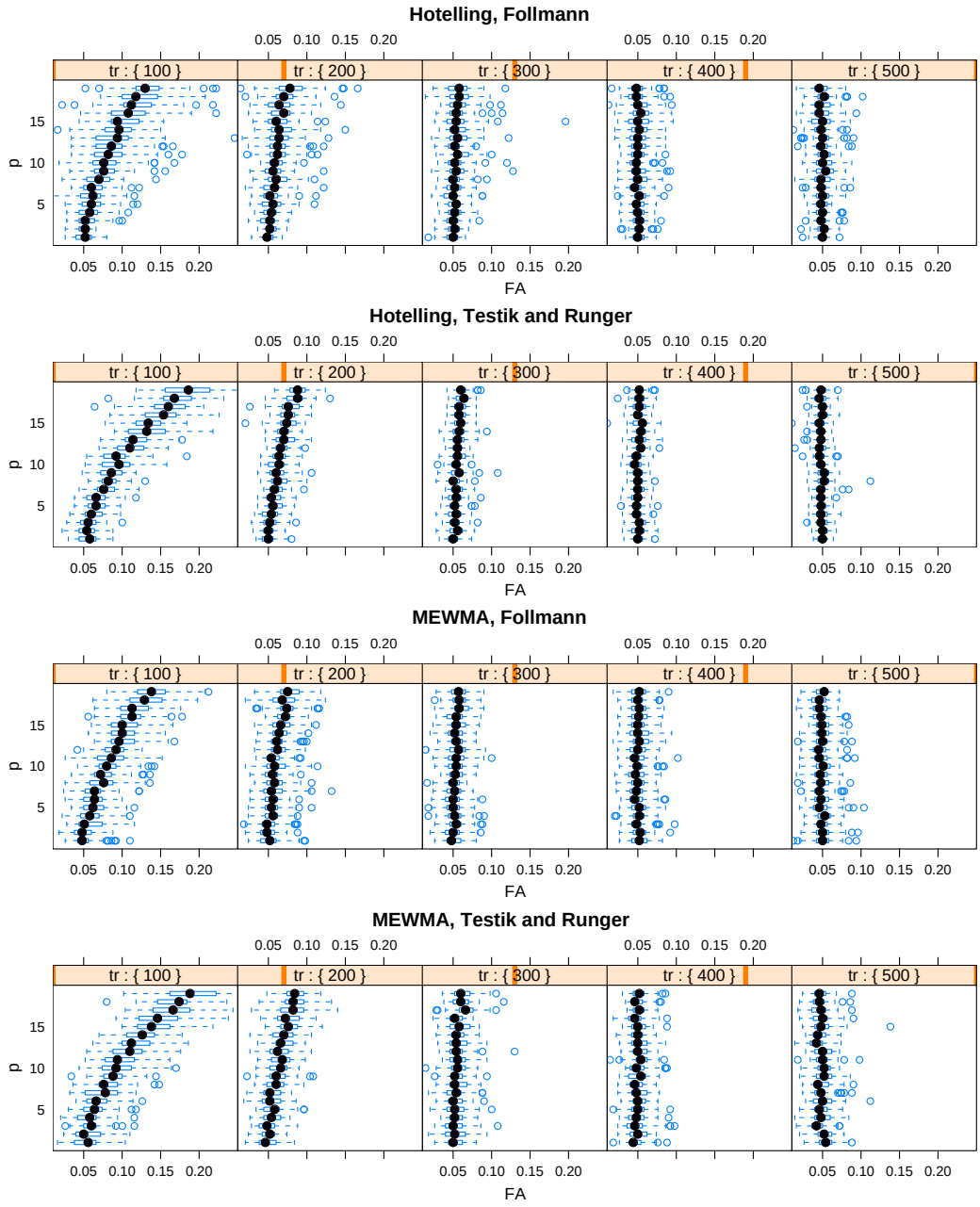

Figure 4.6: Distribution of false alert rate (FA) in directionally-sensitive charts as a function of training data length (tr)

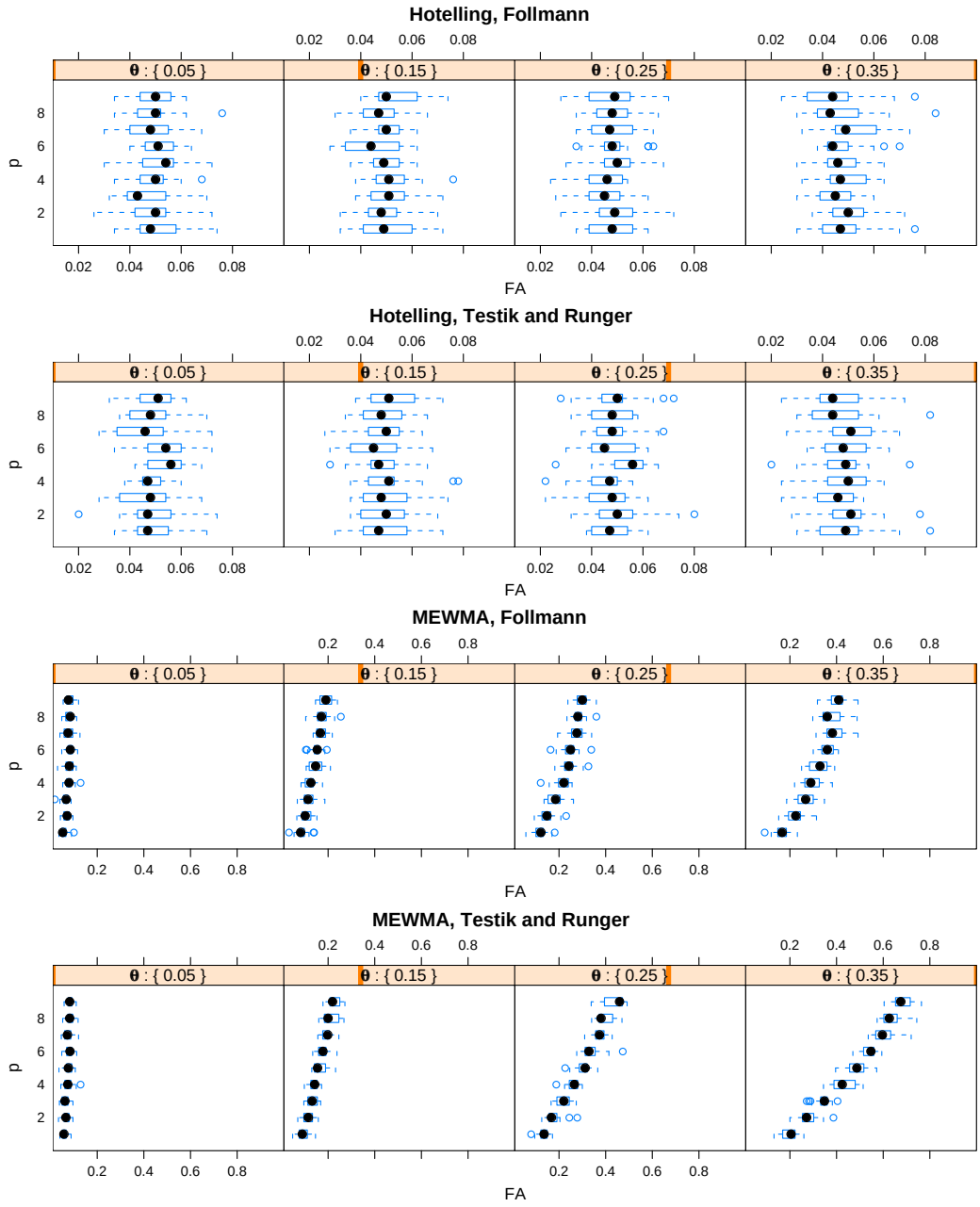

Figure 4.7: Distribution of false alert rates (FA) in directionally-sensitive charts as a function of the autocorrelation (θ), when the covariance matrix is known

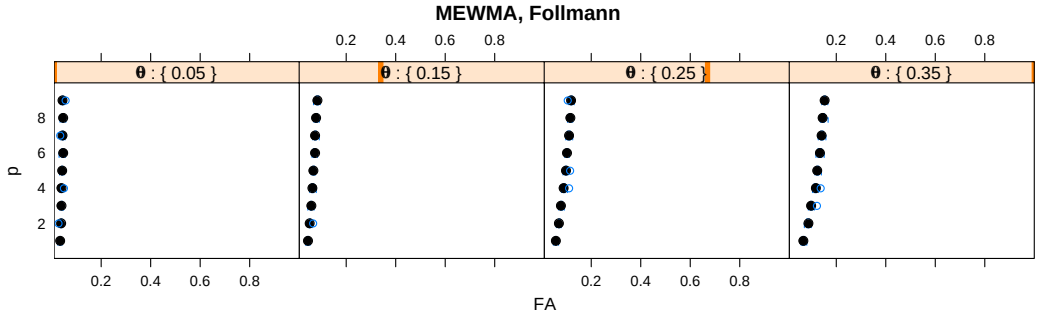

Figure 4.8: Distribution of false alert rates (FA) in Follmann's directionally-sensitive MEWMA chart with restarts, as a function of the autocorrelation (θ), when the covariance matrix is known

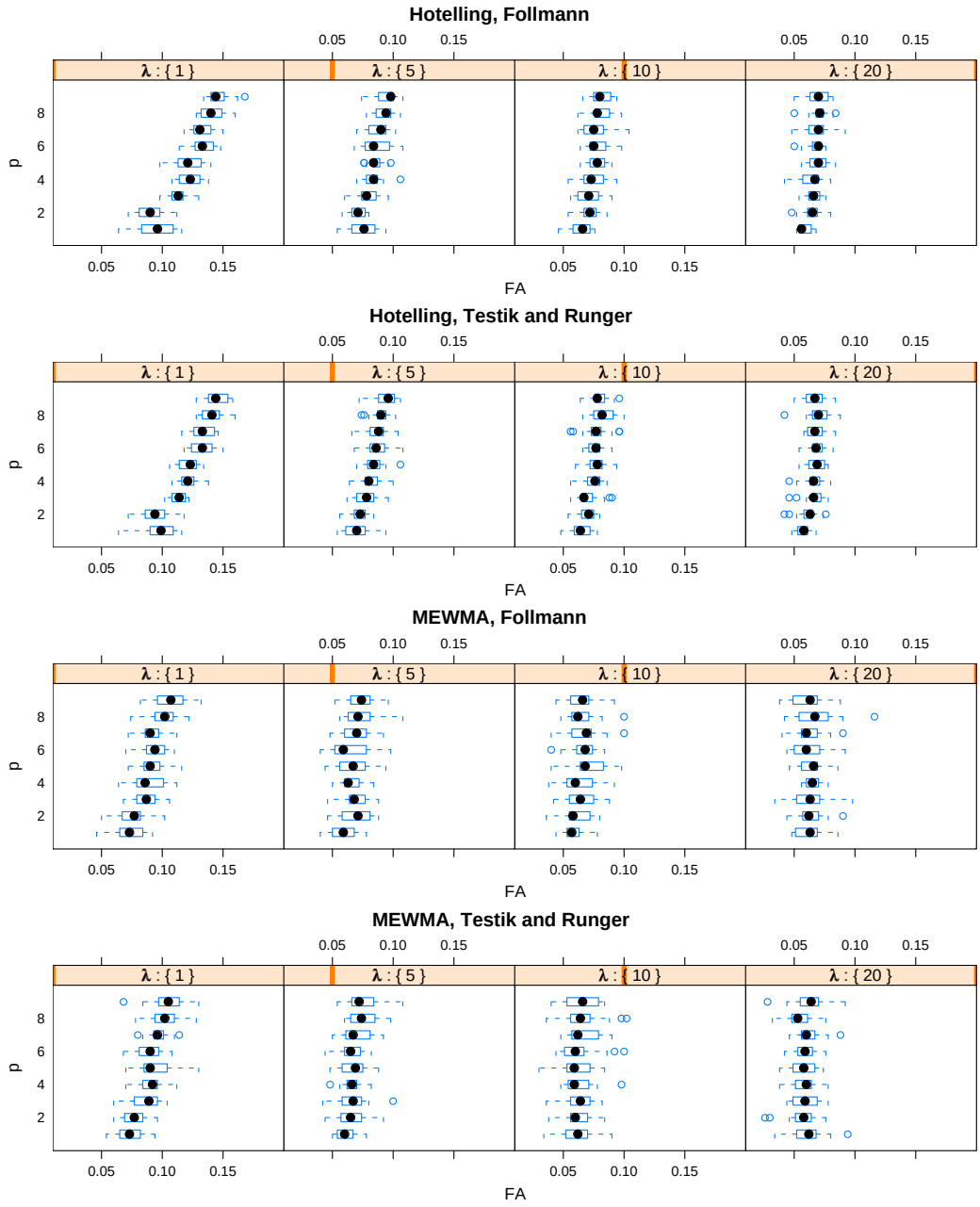

Figure 4.9: Distribution of false alert rates (FA) in directionally-sensitive charts for Poisson counts, as a function of the Poisson parameter (λ), when the covariance matrix is known

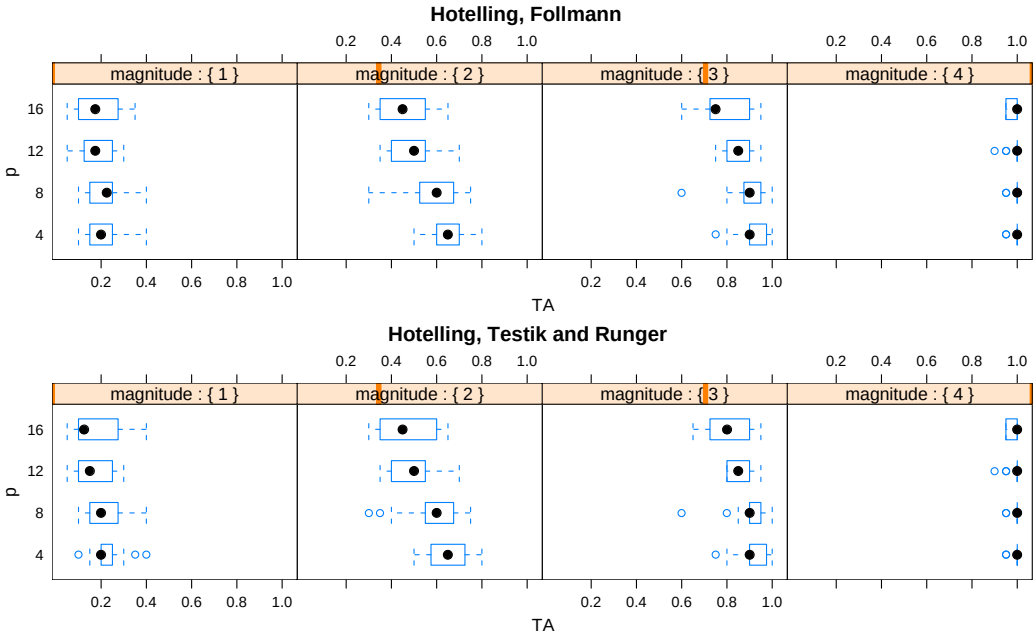

Figure 4.10: Distribution of true alert (TA) rate in directionally-sensitive Hotelling charts as a function of spike magnitude

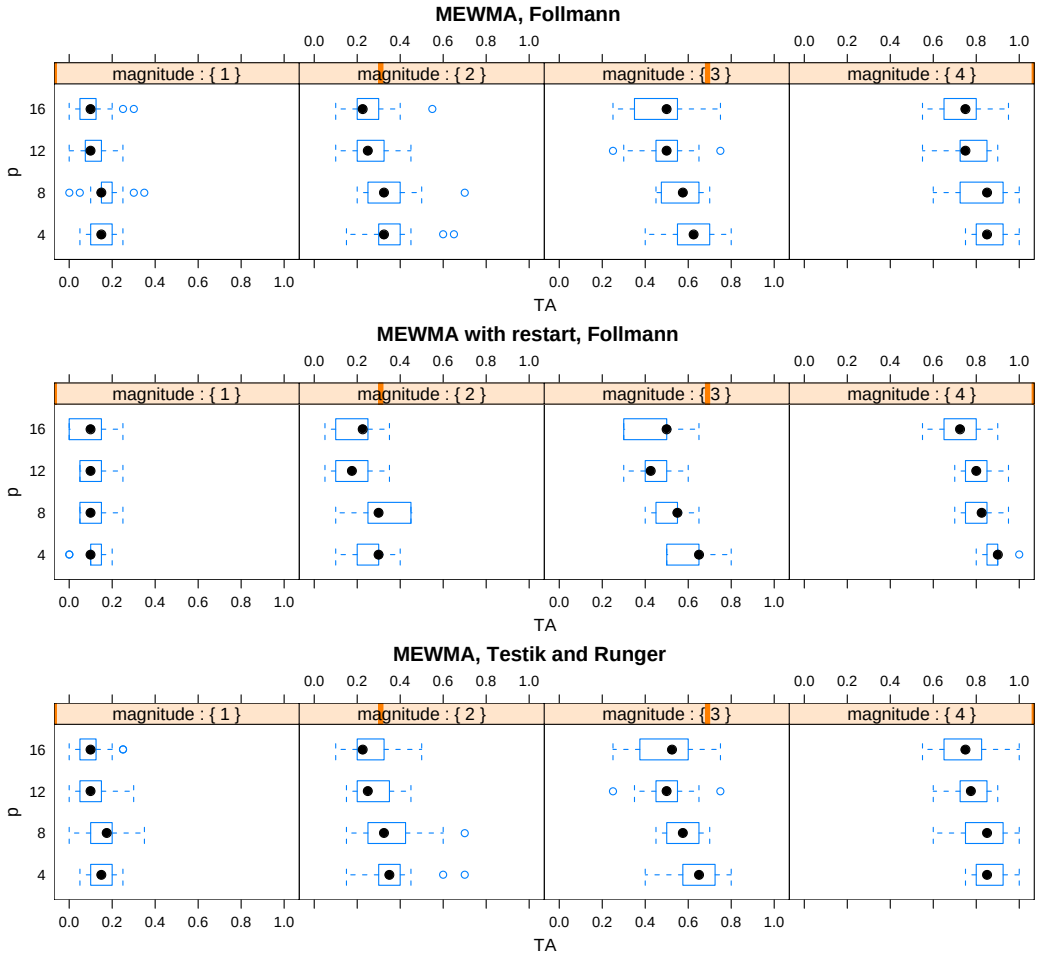

Figure 4.11: Distribution of true alert (TA) rate in directionally-sensitive MEWMA charts as a function of spike magnitude

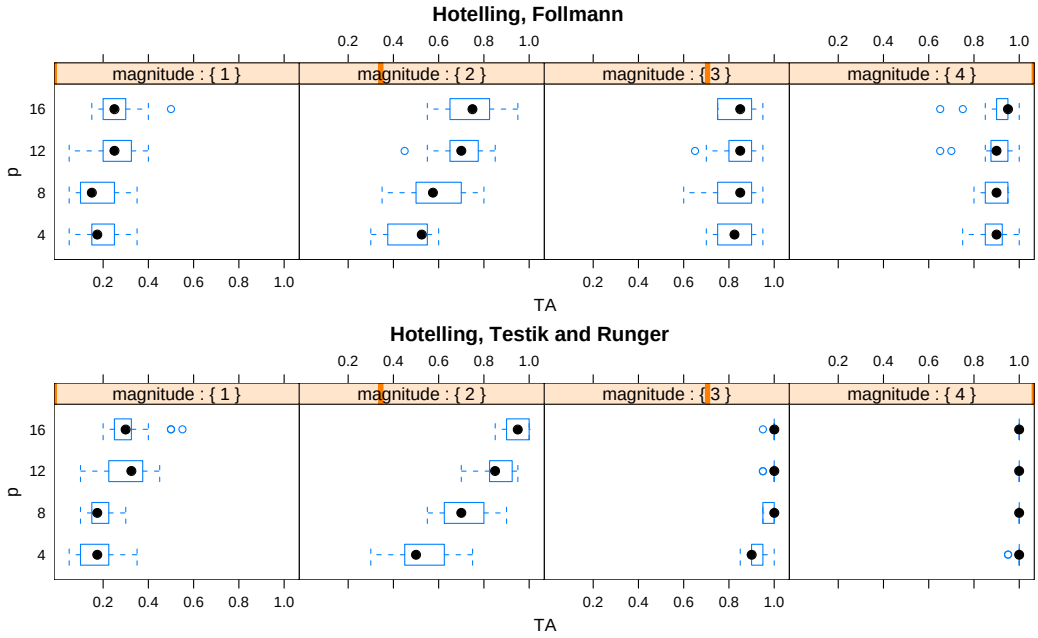

Figure 4.12: Distribution of true alert (TA) rate in directionally-sensitive Hotelling charts as a function of spike magnitude when spike is injected into 25% of the series

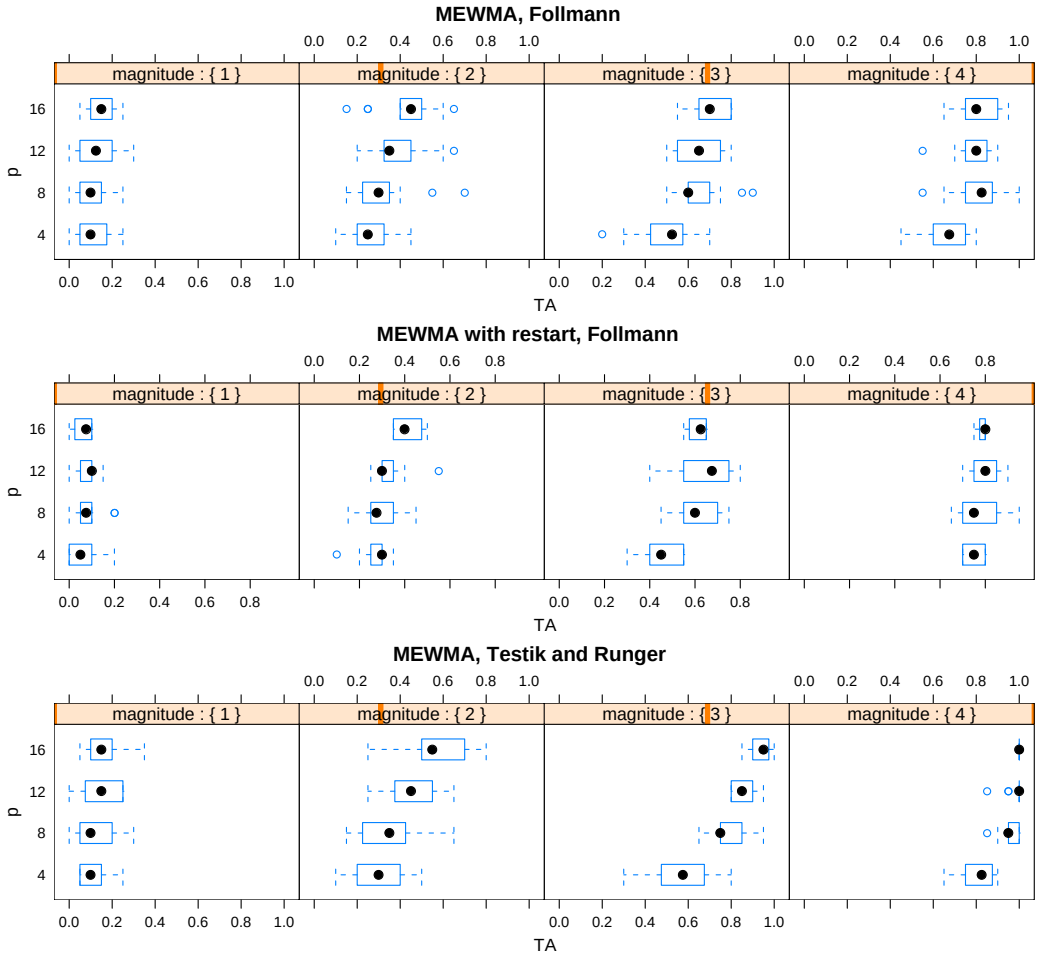

Figure 4.13: Distribution of true alert (TA) rate in directionally-sensitive MEWMA charts as a function of spike magnitude when spike is injected into 25% of the series

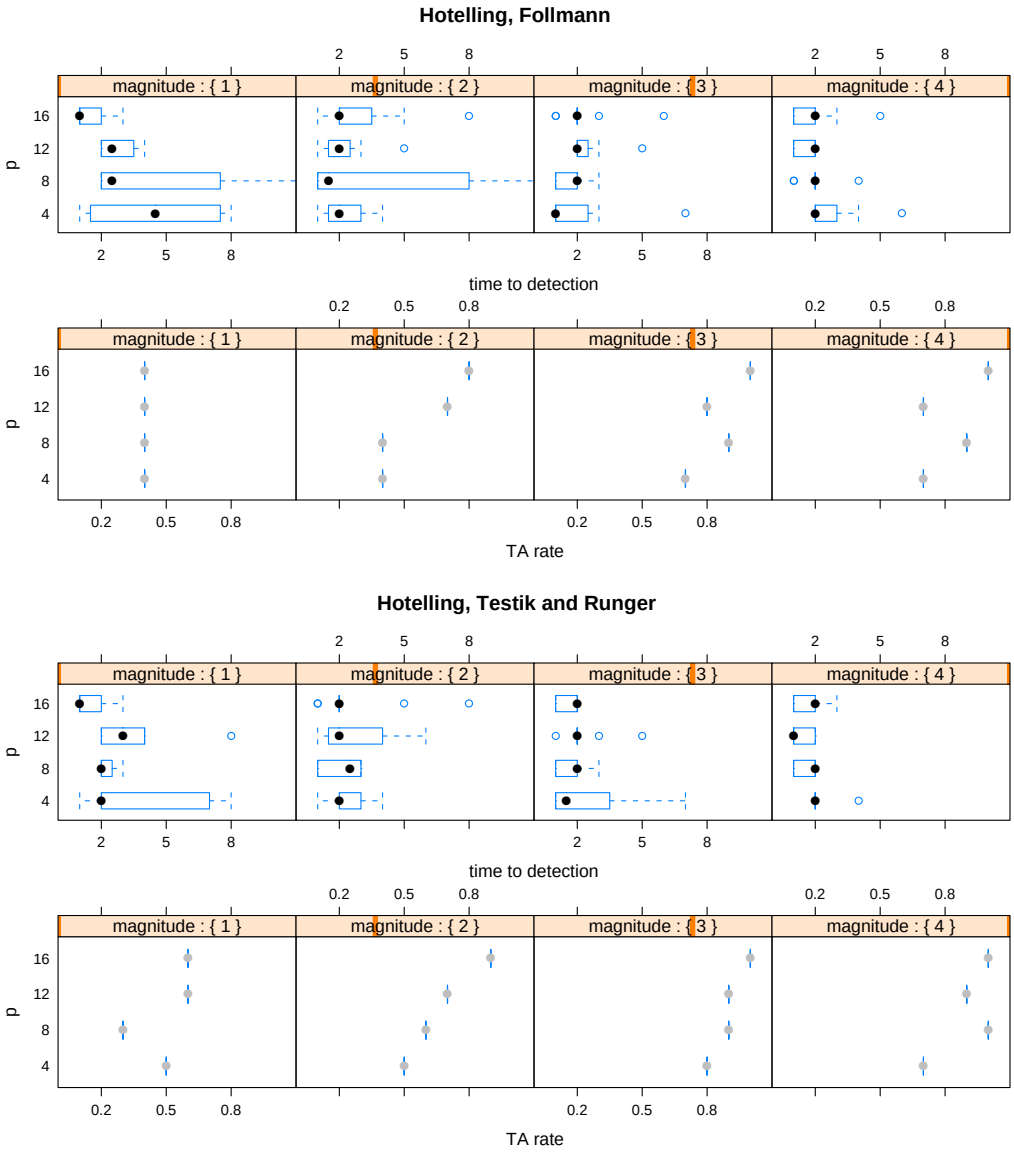

Figure 4.14: Distribution of true alert (TA) rate in directionally-sensitive Hotelling charts as a function of outbreak magnitude when the outbreak is injected into 25% of the series

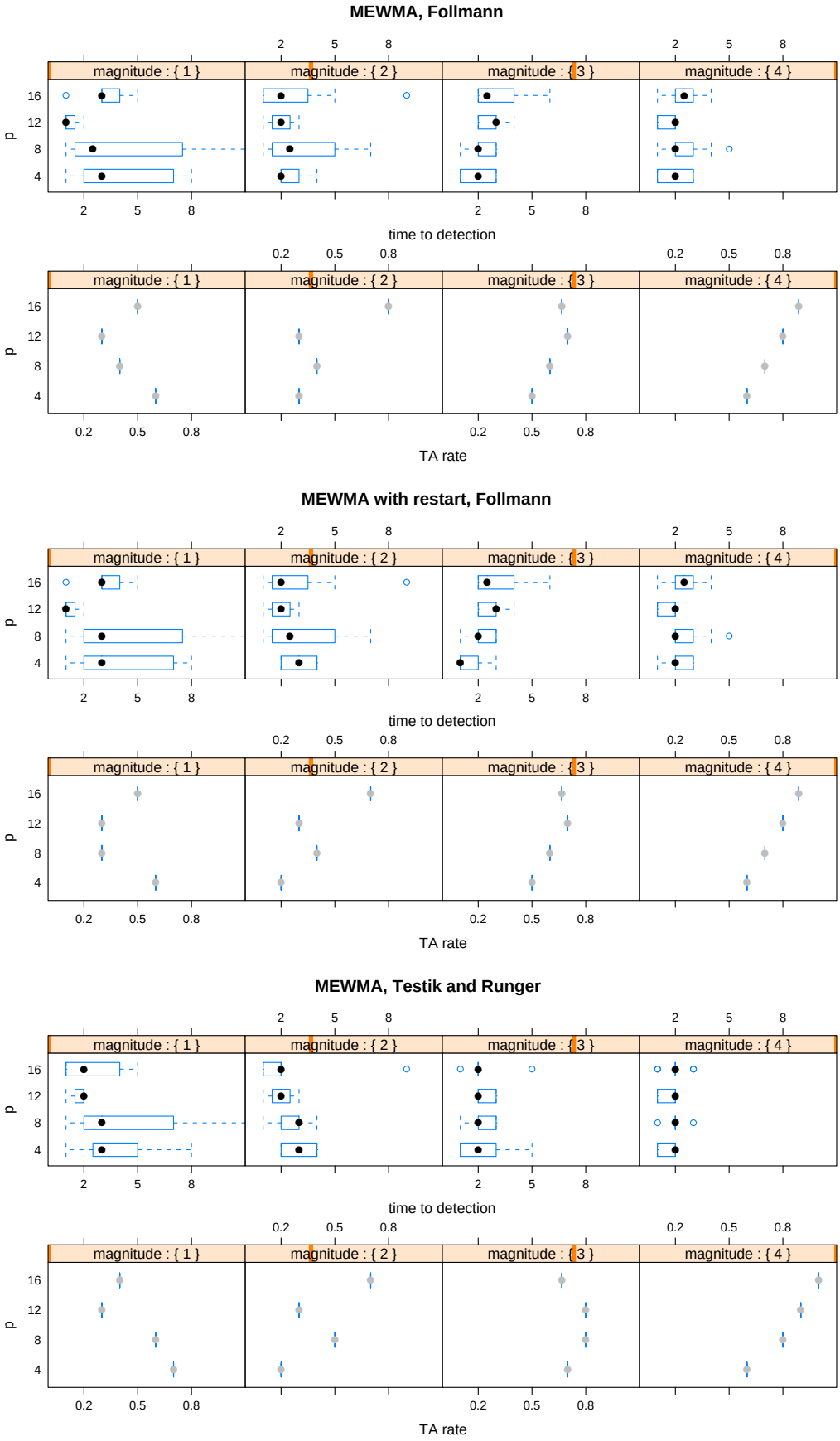

Figure 4.15: Distribution of true alert (TA) rate in directionally-sensitive MEWMA charts as a function of outbreak magnitude when the outbreak is injected into 25% of the series

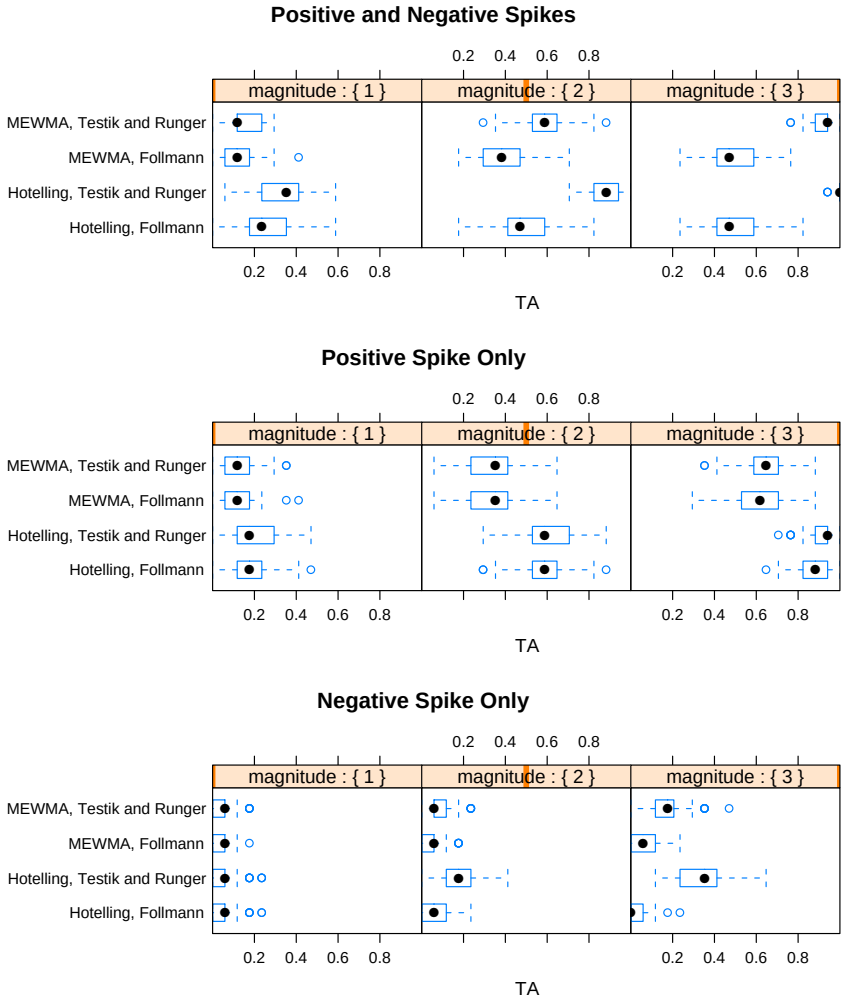

Figure 4.16: Distribution of true alert (TA) rate in directionally-sensitive charts, as a function of spike magnitude in the presence of increasing and decreasing spikes (top), increasing spikes only (middle) and decreasing spikes only (bottom)

Chapter 5

Generating Multivariate Poisson Random Variables

In this Chapter we introduce a conceptually appealing method for generating multivariate Poisson random variable. The multivariate Poisson distribution is essential in simulating biosurveillance multivariate data, in which the daily aggregates are fairly low. Examples are daily counts of cough complaints in a small hospital, or daily counts of school absences in a local high school.

Current simulation methods suffer from limitations ranging from computational complexity to restrictions on the structure of the correlation matrix. We propose a correction for *NORTA* (NORmal To Anything) for generating multivariate Poisson data with flexible correlation structure and rates. *NORTA* is based on simulating a multivariate Normal distribution and converting it to an arbitrary *continuous* distribution with a specific correlation matrix. We show that our method is both highly accurate and computationally efficient.

5.1. Existing Methods

The p-dimensional Poisson distribution is characterized by a mean (or rate) vector $\mathbf{\Lambda}$ and covariance matrix Σ_{Pois} that has diagonal elements equal to $\mathbf{\Lambda}$. It is customary to use the term "multivariate Poisson" for any extension of the univariate Poisson distribution where the resulting marginals are of univariate Poisson form. In other words, the same

term is used to describe different multivariate distributions, which have in common the property that their marginals are univariate Poisson.

One of the best known methods for generating bivariate Poisson data is the *Trivariate Reduction*, which was proposed by Mardia (1970). In this method three independent Poisson random variables Z_1, Z_2, Z_{12} are first generated with rates λ_1, λ_2 and λ_{12} respectively. Then, the variables are combined to generate two dependent random variables in the following way:

$$X_1 = Z_1 + Z_{12}$$

$$X_2 = Z_2 + Z_{12}.$$

It is shown that:

$$X_1 \sim Poisson(\lambda_1 + \lambda_{12}),$$

$$X_2 \sim Poisson(\lambda_2 + \lambda_{12}),$$

$$\rho_{X_1,X_2} = \frac{\lambda_{12}}{\sqrt{(\lambda_1 + \lambda_{12})(\lambda_2 + \lambda_{12})}}.$$

The main drawbacks of the Trivariate Reduction method are that it does not support negative correlation values and it does not cover the entire range of feasible correlations. In a recent paper, Shin and Pasupathy (2007) presented a computationally fast modification to the Trivariate Reduction method that enables generating a bivariate Poisson with a specified negative correlation. Their method first generates two dependent Poisson variables with rates $\lambda_{X_1}, \lambda_{X_2}$ and some correlation $\widetilde{\rho}_{X_1X_2}$ and then iteratively adjusts the rates to achieve the desired correlation $\rho_{X_1X_2}$.

Krummenauer (1998a,b) proposed a convolution based method to generate bivariate Poisson data in polynomial time. The algorithm first generates and then convolves independent univariate Poisson variates with appropriate rates. The author presented a recursive formula to carry out the convolution in polynomial time. This method enables the simulation of multivariate Poisson data with *arbitrary* covariance structure. The main limitation of this method is its high complexity (the recursions become very inefficient when the number of series p increases). Also, the method does not support negative correlation.

Minhajuddin et al. (2004) presented a method for simulating multivariate Poisson data based on the Negative Binomial - Gamma mixture distribution. First, a value k is generated from a Negative Binomial distribution with rate r and success probability $\Pi = \frac{\lambda}{\lambda+\theta}$. Then, conditional on k, a set of p independent Gamma variates are generated $(X_1, \ldots, X_p)$. The sum over k of the joint distribution of k and $X_1, \ldots, X_k$ has a Gamma marginal distribution with rates r and λ. The correlation between a pair X_i and X_j $(i \neq j)$ is $\frac{\theta}{\lambda+\theta}$. There are two main drawbacks to this approach: First, it requires the correlation between each pair of variates to be identical ($\rho_{ij} = \rho$ for all $i \neq j$). Second, it does not support negative correlations.

Karlis (2003) points out that the main obstacle limiting the use of multivariate simulation methods for Poisson data, including the above-mentioned methods, is the complexity of calculating the joint probability function. He mentions that the required summations might be computationally exhausting in some cases, especially when the number of series p is high.

5.1.1 NORTA: NORmal To Anything

A different approach for generating data from a multivariate distribution with given univariate marginals and a pre-specified correlation structure is known as *NORTA*. The idea here is to first generate a p-vector from the multivariate Normal distribution with correlation structure R_N and then to transform it to any desired distribution (say F) using the inverse cumulative distribution functions (Chen, 2001; Nelsen, 2006). The resulting distribution is referred to as *normal-copula*.

When the desired distribution F is continuous, the normal-copula has a well defined correlation structure. However, when F is discrete (as in the Poisson case), the matching between the initial correlation structure R_N and the normal-copula correlation structure R_F is a non trivial problem (Avramidis et al., 2009). For example, consider the following steps for generating a p-dimensional Poisson vector with arbitrary correlation matrix R_{Pois} and rates $\mathbf{\Lambda}$:

1. Generate a p-dimensional Normal vector $\mathbf{X}_N$ with mean $\boldsymbol{\mu} = 0$, variance $\boldsymbol{\sigma} = 1$ and a correlation matrix R_N.

2. For each value $X_{N_i}, i \in 1, 2, ...p$, calculate the Normal CDF:

$$\Phi(X_{N_i}).$$

3. For each $\Phi(X_{N_i})$, calculate the Poisson inverse CDF (quantile) with rate λ_i:

$$X_{Pois_i} = \Xi^{-1}(\Phi(X_{N_i})),$$

where,

$$\Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\sigma^2}} e^{\frac{-u^2}{2}} du \tag{5.1}$$

$$\Xi(x) = \sum_{i=0}^{x} \frac{e^{-\lambda}\lambda^i}{i!}. \tag{5.2}$$

The vector $\mathbf{X}_{Pois}$ is then a p-dimensional Poisson vector with correlation matrix R_{Pois} and rates $\mathbf{\Lambda}$. When $\mathbf{\Lambda}$ is sufficiently high ($\mathbf{\Lambda}$¿$\mathbf{5}$), the Poisson distribution is well known to be asymptotically Normal and $R_{Pois} \approx R_N$. However, when one or more of the rates (λ) is low, the normal-copula correlation deviates from the Normal correlation ($R_{Pois} \neq R_N$). The reason is that the feasible correlation between two random Poisson variables is no longer in the range $[-1, 1]$, but rather (Whitt, 1976)

$$\left[\underline{\rho} = corr(\Xi_{\lambda_1}^{-1}(U), \Xi_{\lambda_2}^{-1}(1-U)), \quad \bar{\rho} = corr(\Xi_{\lambda_1}^{-1}(U), \Xi_{\lambda_2}^{-1}(U))\right]. \tag{5.3}$$

In fact, Shin and Pasupathy (2007) show that when $\lambda_1, \lambda_2 \to 0$ the minimum feasible correlation $\underline{\rho} \to 0$. Therefore, the NORTA transformation maps a correlation range of $[-1, 1]$ (multivariate normal) to a much smaller range $[\underline{\rho} \geq -1, \bar{\rho} \leq 1]$.

To illustrate this reduction in the correlation range, consider Figure 5.1. The figure depicts a bivariate NORTA transformation process with correlation $\rho = 0.9$, and the resulting Poisson bivariates with high (20) and low (0.2) rates. The 'bubble' size in each panel corresponds to the number of bivariates with the same value. Naturally, the bivariate Poisson with high rates has a fairly similar distribution to that of the Normal distribution. The bivariate Poisson with low rates, however, not only takes very few possible values ($\{(X_1, X_2)|X_1, X_2 \in (0, 1, 2, 3)\}$) but is also a much more skewed distribution (the majority

of the bivariate values are the pair $(0, 0)$).

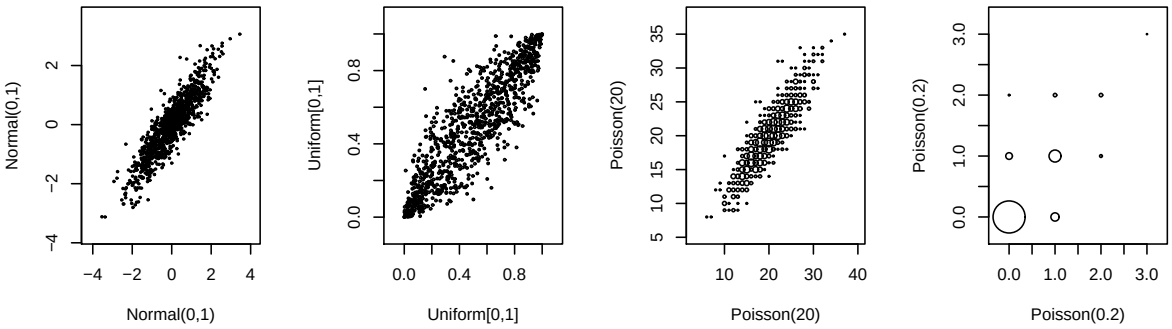

Figure 5.1: Scatter plots for bivariate simulated variables using NORTA, for Normal, Uniform, Poisson($\lambda = 20$) and Poisson ($\lambda = 0.2$)

Figure 5.2 illustrates the relationship between the desired correlation and the resulting actual correlation when generating bivariate Poisson data with low rates ($\lambda < 1$).

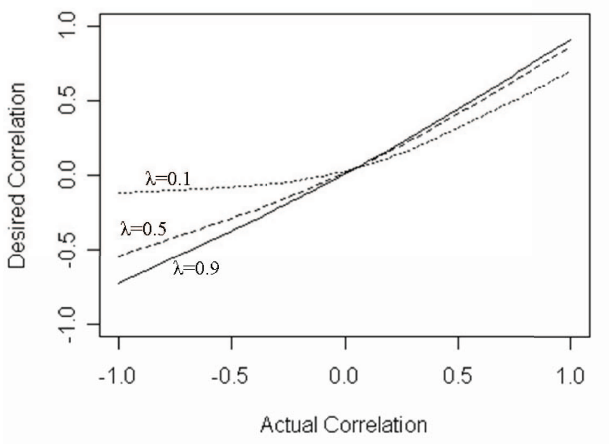

Figure 5.2: Comparing the desired correlation to the resulting actual correlation for Poisson bivariate data with low rates

In a recent paper, Avramidis et al. (2009) studied the NORTA correlation matching problem when the marginal distributions are discrete. The authors provide several approximations for mapping the desired correlation to the actual correlation. Their approximation involves a bivariate normal integral and the approximation of the derivative of the matching function with respect to the actual correlation. They illustrate the per-

formance of their method on multivariate Binomial and Negative Binomial distribution.

In contrast to the approximation by Avramidis et al. (2009), we provide a conceptually simple, empirically based approximation method for mapping R_N to R_{Pois}. We show that our method is highly accurate (with absolute error of less than 6×10^{-2}) and can be computed within milliseconds. We hope that the simplicity of this method, along with the availability of the code will lead to a wider use of simulated multivariate Poisson data, which can be used for studying and evaluating algorithms in the management science field.

5.2. Generating Multivariate Poisson Random Variables

For simplification, we explain our approximation for the bivariate case ($p = 2$). One can easily extend to higher dimension data by simply applying the correlation mapping to each pair.

We define $\mathbf{\Lambda} = \{\lambda_1, \lambda_2\}$ and,

$$\underline{\rho} = corr(\Xi_{\lambda_1}^{-1}(U), \Xi_{\lambda_2}^{-1}(1-U))$$

$$\bar{\rho} = corr(\Xi_{\lambda_1}^{-1}(U), \Xi_{\lambda_2}^{-1}(U))$$

Using a large simulation approach, we find that the relationship between the desired correlation (ρ_{pois}) and the actual correlation (ρ_N) can be approximated by an exponential form:

$$\rho_{Pois} = a \times e^{b\rho_N} + c,$$

where the coefficient a, b and c can be estimated from the points ($\underline{\rho}$, -1), ($\bar{\rho}$, 1) and (0, 0):

$$a = -\frac{\bar{\rho} \times \underline{\rho}}{\bar{\rho} + \underline{\rho}}; \;\; b = log\left(\frac{\bar{\rho} + a}{a}\right); \;\; c = -a$$

To evaluate the performance and computation time of our approximation, we implement the algorithm in R on a 2.6 GHz Intel dual-core 32 bit-processor running windows. Code is available in the Appendix.

Figure 5.3 illustrates the simulation performance when using the above approximation to correct for the distortion in the resulting correlation. This is illustrated for the bivariate Poisson case with rates that range in $(\lambda_1, \lambda_2) \in (0.1, 0.1), (0.1, 0.5), (0.5, 0.5), (0.5, 0.9), (0.9, 0.9)$. Figure 5.4 shows that the mean absolute difference between the actual and desired correlation, for any choice of Λ and ρ is less then 0.06. The method performs better as the rates increase.

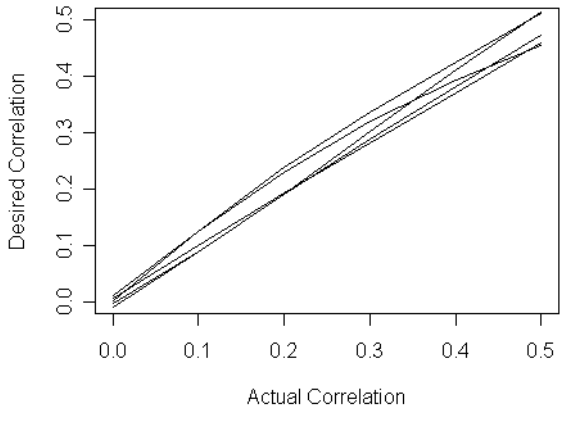

Figure 5.3: Comparing the desired correlation to the corrected actual correlation

Figure 5.4 depict the absolute mean error of our method for rates in the range $[0.1, 10]$ and any correlation value. The left panel presents the error distribution as a function of rate. As stated earlier, correlations between low rates have a slightly higher error. In the left panel, we illustrate the distribution of error when $\lambda_1 = 0.4$ as a function of λ_2 and the *desired* correlation coefficient ρ. We use white to represent infeasible correlation values. It appears that the absolute error is independent of the desired correlation value.

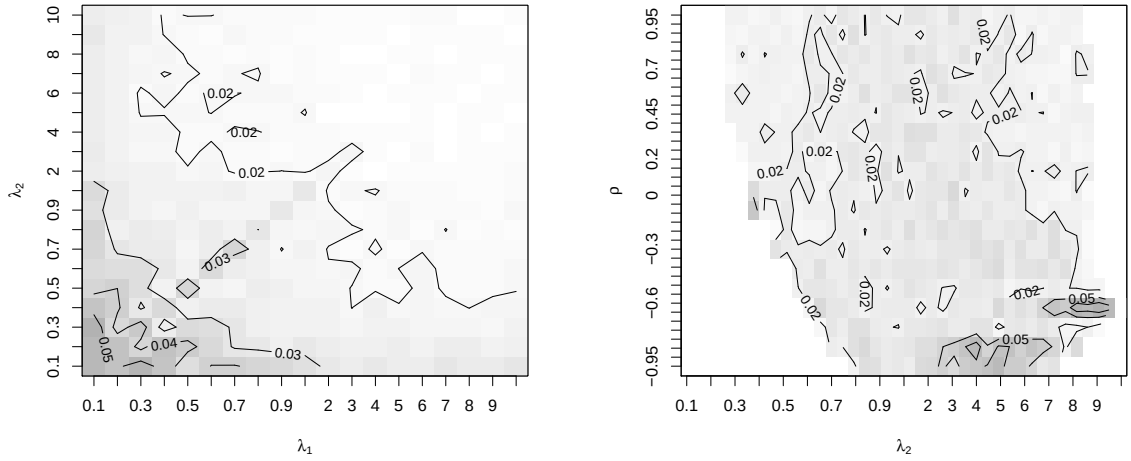

Figure 5.4: Absolute mean error. Left: Error as a function of the Poisson rates (λ_1, λ_2). Right: Error as a function of the Poisson rate (λ_2) and the desired correlation (ρ). ($\lambda_1 = 0.4$)

Apart from simplicity, a very important feature of our generator is the low computation time. Figure 5.5 depicts the computation time (in millisecond) as a function of data dimension p and series length. The running time is shown to be minor even when generating large datasets.

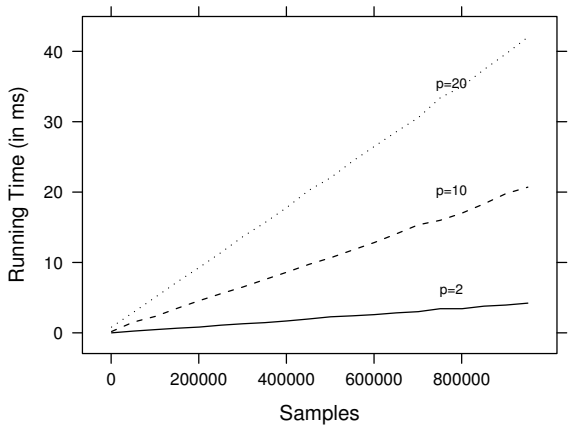

Figure 5.5: Computation time as a function of the data dimension (p) and length (#samples

Part II

Modeling Kidney Allocation: A Data-Driven Optimization Approach

Chapter 6

Background

According to the Scientific Registry of Transplant Recipients' (SRTR) annual statistics, there are more than 90,000 candidates annually with kidney failure, End Stage Renal Disease (ESRD[1]), who are waiting for transplantation in the United States. However, because only about 10,000 deceased donor kidneys are available for transplantation each year, more than 20,000 new candidates are added to the waiting list annually (see Figure 6.1).

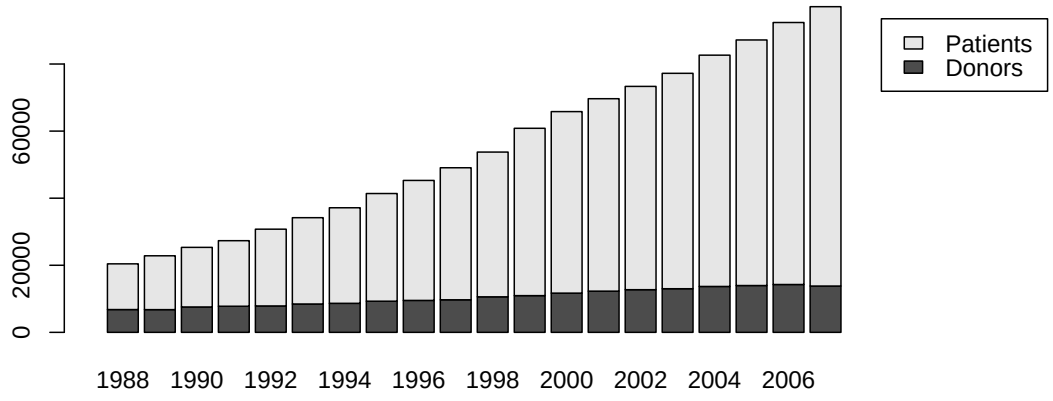

Figure 6.1: Increase in waiting list vs. number of donors.

The current kidney allocation system was developed in the 1980s and 1990s. The system revolves around a set of priority points given to candidates for tissue matching and sensitization level (the level of sensitization to donor antigens, measured by Panel Reactive Antibody (PRA)), combined with waiting time and age. However, during the last 30 years the waiting list has grown and the population of candidates has aged. Consequently, the

[1]For more information see OPTN/UNOS (2008).

allocation process has become dominated by the length of time a candidate has been on the waiting list, effectively becoming a "first come first transplant" policy. Moreover, the current system does not match donors and recipients well. As a result, kidneys with long expected post-transplant survival are commonly allocated to candidates with short expected post-transplant survival (OPTN/UNOS, 2008).

In 2004 the Kidney Allocation Review Subcommittee (KARS) was formed with the goal of analyzing ways to improve kidney allocation in the United States. In 2008 the committee proposed three concepts that, along with matching criteria, would combine to determine a candidate's Kidney Allocation Score (KAS). The concepts are as follows: Life Years From Transplant (LYFT), which determines the estimated additional years a kidney recipient will gain from the transplant; Time spent on dialysis (Dialysis Time); and Donor Profile Index (DPI) that provides a measure of organ quality. According to the proposed method, kidneys would be allocated to patients based on candidates' KAS, rather than priority points.

Compared to the current system, the new allocation policy is expected to increase the overall number of life years gained from the kidney allocation system by over 3,000 years in its first year of operation. Additionally, transplantation rates for minority candidates (such as African-Americans), as well as candidates with high PRA levels, is expected to improve.

6.1. Current Allocation: Priority Points (PP)

The current allocation policy is based on a set of priority points. When an organ becomes available, each candidate registered on the Organ Procurement and Transplantation Network (OPTN) waiting list receives priority points according to a compatibility test between the patient and the donor, his/her waiting time and rank, and age (higher priority for younger and older populations). The compatibility test is a combined measure of tissue matching (also known as Human Leukocyte Antigen (HLA) matching) and the Panel Reactive Antibodies (PRA) test. PRA is a blood test that examines whether a candidate exhibits antibodies to the proteins of the donor. Priority points for compatibility are given with the objective of minimizing the rejection rate after transplant. Priority points for waiting time and age are given to compensate candidates with rare tissue types. The details of the policy are shown in Table 6.1.

Geographic factors are also taken into consideration. There are 11 geographic regions, divided into a total of 69 regional Organ Procurement Organizations (OPO). Patients within the same OPO of a donated organ are referred to as 'local candidates' and have first priority. If an appropriate candidate is not found within the OPO, then the search becomes regional. Lastly, if the search fails at the regional level, it becomes national (Zenios, 2004).

While giving priority points for waiting time is supposed to provide a sense of equity in access to transplantation, over the years those priority points have started to dominate the allocation system and have decreased the efficiency of allocations. The dominance of waiting time on priority points has transformed the system to a first come first trans-

Table 6.1: The UNOS point system. Source: Zenios (2004)

Category	**Points**
Waiting time	1 point for each full year on the waiting list
Rank in the waiting list	1 point for the longest waiting candidate; fraction of points are assigned proportionately to all other candidates
Tissue mismatches	∞ for no mismatches
	7 points for 0 B or DR mismatches
	0 points for 0 A or B mismatches
	5 points for 1 B or DR mismatches
	2 points for 2 B or DR mismatches
	0 points for 3 A or B mismatches
Panel reactive antibodies	4 points for PRA $>$ 80%
Pediatric candidates	4 points when age $<$ 11 years
	3 points when age $>$ 11 years and $<$ 18 years

plant assignment that disregards the potential afterlife of the transplanted kidney itself (OPTN/UNOS, 2008). As a result, the current system poorly matches the candidate and transplanted kidney expected life after transplant, resulting in death with a functioning kidney in many patients, and an increase in the need for retransplantation for many other patients. Moreover, the system also fails to achieve equity for several demographic groups with rare tissue types and high sensitization levels (Eggers, 1995).

According to OPTN/UNOS (2008), even with additional priority given to sensitized candidates (patients with PRA>80%), highly matched kidneys, and children, the current system does not adequately balance equity and efficiency factors, as it utilizes only a few of the medical criteria that are now available to rank candidates. Furthermore, the PRA measure itself is known to be highly variable and inconsistent, as it is measured by different commercially available kits or locally procured cell panels, which often do not represent the entire donor population.

6.2. KARS' Proposed Allocation: Kidney Allocation Score (KAS)

The drawbacks of the current allocation policy have initiated an intensive debate and the need to consider alternative allocation policies (OPTN/UNOS, 2008). In 2004, the Kidney Allocation Review Subcommittee (KARS) was established with the goal of designing an allocation policy that maximizes the tradeoff between equity in access to transplantation and efficiency; that is, maximizing the aggregate health of the transplant candidate pool (Votruba, 2001).

Early in the review process, KARS recognized that kidney allocation is unique as a treatment option, compared to other organ allocations such as liver and lung, due to the availability of dialysis for candidates suffering from renal failure. The Kidney Allocation Review Subcommittee (KARS) defines the main objective of its kidney allocation policy as follows:

> "rankings shall be ordered from most to least medically urgent [...] The kidney allocation policy should have the goal of providing equitable access for kidney transplant candidates to deceased donor kidneys for transplantation while improving the outcomes of recipients of such kidneys."

The allocation policy should also focus on minimizing mortality, maximizing kidney expected life, maximizing the post transplant expected life, minimizing the sensitization level, maximizing equity in access to transplantation, etc. This multiplicity in objectives raises a non-trivial, multi-objective allocation problem.

In 2008, KARS has proposed a multi-criteria objective that ranks patients on the

waiting list according to a kidney allocation score (KAS). The KAS is a multi-criteria objective that seeks to balance the trade-off between allocation outcome and equity in access to transplantation. The formula for determining KAS provided by OPTN/UNOS (2008) is the following:

$$KAS = LYFT \times 0.8 \times (1 - DPI) + DT \times (0.8 \times DPI + 0.2) + (CPRA \times 0.04), \quad (6.1)$$

where,

$LYFT$ (Life Years From Transplant) is the estimated survival duration in years that a recipient of a specific organ may expect to have, versus his remaining years on dialysis (at time of offer). LYFT is a function of the patient's profile (age, tissue type, etc.) as well as the donor's profile (age, cause of death, etc.)

DPI (Donor Profile Index): a continuous measure of organ quality based on clinical information. DPI increases individual matching by providing a better metric for deciding which organs are appropriate for which candidates.

DT (Dialysis Time): the length of time that the patient has been receiving dialysis at the time of the offer.

$CPRA$ (Calculated Panel Reactive Antibody): measures the likelihood that the recipient and donor would be incompatible, based on HLA frequencies in donors. CPRA replaces the PRA measure in the current renal allocation system.

The dialysis survival component of LYFT and DT is adjusted by a factor of 0.8 to

account for the diminished quality of life (QoL) reported by candidates on dialysis. CPRA is adjusted by a factor of 4 to give extra consideration to highly sensitized patients, to ensure that these patients get more opportunities to receive transplants than they would without extra points.

Because research on survival following dialysis and kidney transplantation indicates that nearly all candidates with ESRD are predicted to live longer with kidney transplant than on dialysis, the introduced metric of LYFT is considered an important factor for estimating the final kidney allocation score for every donor-candidate pair. However, mainly utilizing the LYFT might lead to unfair allocations for some patients with short estimated LYFT who have been on dialysis for longer periods. Therefore, in order to make the new allocation policy more equitable, dialysis (DT) is also considered in the calculation of the KAS. Sensitization level (or calculated panel reactive antibody, CPRA) and donor's organ types (DPI) together provide a metric for deciding which organs are appropriate for which candidates.

Similar to the current allocation scheme, upon a kidney arrival, each candidate is assigned with a KAS value and the organ is offered to the candidate with the higher score (herein referred to as *Highest-KAS-first* (HKF) policy).

One of the main disadvantages of this allocation is that it does not account for future prospective donors and candidates' health condition degradation. In other words, the allocation decision is done in a *static* fashion, in which the decision is based only on the match between the *current arriving organ* and the candidates for the kidney.

6.3. Literature Survey

The organ allocation problem, and specifically cadaveric kidney (or *grafts* as they are called by the medical community) allocation, raises a very interesting operations research policy modeling problem that combines supply shortage with ever increasing demand. Unlike liver and lung allocation policies, in which the objective is to minimize the number of deaths on the list, kidney allocation is unique in that there exists the alternative of dialysis for candidates suffering from renal failure (OPTN/UNOS, 2008). This treatment option requires the allocation policy to be based upon many additional factors, such as post transplant expected life, equity in access to transplantation, etc.

For the past two decades, operations research applications of organ transplantation have received great attention. Generally, the organ transplantation is modeled as a matching problem between donors and recipients with the goal of maximizing some reward function. Specifically, two perspective of the problem have been addressed: the patient's perspective of deciding to accept or reject an organ offer, and the policy planner's perspective of designing allocation decision model. Alagoz et al. (2008) provide a comprehensive survey of operations research applications related to organ allocation in general. A specific survey of models for kidney allocation is provided by Zenios (2004). We next survey the key literature on kidney allocation models. We focus on the policy planner's problem and discuss the patient choice problem and the game between the planner and the candidates.

The question of patient choice models in the context of kidney allocation has been repeatedly addressed in the literature of the last two decades. According to Zenios (2004), about 45% of the offered organs are rejected by the first patient who receives it (or by

his/her physician). The rationale for a rejection is that, under the current allocation policy, once a candidate reaches the top of the list, s/he remains there until s/he accepts a graft. Models of patient choice are offered by David and Yechiali (1985, 1995) and Ahn and Hornberger (1996). Patient choice in these papers is modeled as a stopping problem, in which each patient receives an organ offer at random and must decide whether or not to accept the offer. Similar stopping models for the patient choice problem has been studied in the liver and lung transplantation context (see e.g. Howard (2002) and Alagoz et al. (2007)).

One of the most studied models for organ allocation is the sequential stochastic assignment problem introduced by Derman et al. (1972). In the Derman et al. model, as random jobs arrive, they must be assigned to workers. Rewards depend on the match between jobs and workers. Kidney allocation can be viewed as an application of Derman et al.'s model, where jobs are organs and workers are candidates for kidney allocation.

Righter (1989) and David and Yechiali (1995) propose Markov Decision Process (MDP) models for the sequential allocation of kidneys to patients, extending the results from Derman et al. (1972) to random environments. Randomness is reflected in graft arrival and candidate departure (death) rates. The authors raise the question of admission control and allocation of arriving organs and discuss the threshold-based property of the optimal policy.

Su and Zenios (2004, 2005, 2006) incorporate the patient choice question into the allocation problem. They consider a stylized model, in which the patients are homogenous in their preferences. They model the allocation system as an $M/M/1$ queue where the queue is composed of patients, and the donors are "service providers". Here, the objective

of the patients is to maximize their *individual* discounted quality of life (QoL) before and after transplant. The planner's objective is to maximize the *total* QoL. Similar to previous literature, the authors find that the optimal welfare (i.e., aggregated candidates health) is achieved under a threshold-based policy, implying that it is optimal for the planner to reject organs under a certain quality. Su and Zenios (2005, 2006) extend the work to account for patient heterogeneity.

Su and Zenios (2005) present a stylized model that considers the sequential allocation of n kidneys to n patients. Each patient-kidney pair has its own type and rewards depending on the match between them. The planner's objective is to maximize the total expected reward. As noted earlier, the optimal policy is a threshold policy. Su and Zenios compare the optimal policy with and without patient choice, and demonstrate that patient choice may introduce significant inefficiencies. However, the authors also observe that minimizing the variability in the type of offers expected by each patient type reduces those losses. We will use this observation in our model to claim that policies that do not consider patient choice, yet yield minimum variability concerning the assigned kidney type to each group of patients, are consequently robust to patient choice.

For the more general and realistic representation, Zenios et al. (2000) develop a fluid-based model that imitates the actual clinic environment. In this model the patient pool is divided into K classes and the donors into J classes. The division into classes is based on demographic, immunological, and physiological characteristics. The state of the system is defined as a vector of the number of patients in each class. Patients depart from the system through either death or transplant. Zenios et al. consider a control problem that combines three objective functions. The first objective is to maximize the quality of

adjusted life which satisfies the efficiency criteria. Then, two equity criteria are considered: minimize inequity in (absolute) waiting time (first-come-first-serve), and minimize inequity in (relative) likelihood of transplantation. The authors develop a simulation-based study in which the distributions of the patients' and donors' characteristics, mortality rates, and arrival rates are estimated using data from UNOS 1995, the United States Renal Data System and the New England Organ Bank.

In the same paper, Zenios et al. raise a discussion about the number of patient clusters. The authors claim that the appropriate number of clusters is 10^7, which is less than the number of patients in the waiting list. This implies that the cluster analysis in their model is not used to reduce the dimensionality of the problem.

A recent stream of papers considers the problem of maximizing the efficiency of liver transplants through a redesign of UNOS geographic regions (Kong, 2006; Kong et al., 2008). Demirci et al. (2010) extends this problem to account for the trade-off between allocation efficiency (measured by the matching between donors and recipients) and a measure of geographical equity in the allocation process.

Another branch of research focuses on developing simulation tools to examine how different allocation policies affect system outcomes. Such tools are available for kidney allocation (Taranto et al., 2000), liver transplantation (Shechter et al., 2005; Thompson et al., 2004), and heart transplantation (van den Hout et al., 2003).

6.4. Our Approach

The literature survey clearly demonstrates that the problem of kidney allocation is not a trivial problem. Any allocation policy has to take many factors into consideration. We can group these factor into three sub-problems:

The planner's problem: to design a policy that balances the efficiency of the allocation with equity in access to transplantation. Despite the large dimensionality of the problem, the policy has to be computationally efficient. That is, making an allocation decision in a timely manner, and leaving sufficient time for tissue matching tests, organ shipping, and the actual transplant.

The model estimation problem: to develop a framework for estimating candidates' expected quality of life (with and without transplant), mortality rate, organ rejection probability, etc. The accuracy of these estimates is crucial for the efficiency of the allocation policy (that is, maximizing the aggregate health of the transplant candidate pool).

The patient's choice problem: to resolve the inefficiency arising from patient's choise, who can either accept on offered organ or refuse to accept it (seeking a better match). The patient's choice problem adds another dimension to the problem. Apart from the time loss caused by a refusal, the patients choice may cause an allocation policy to be inefficient (from the societal perspective).

Due to the complexity of the problem, we focus on the first two elements.

The remainder of this part of the dissertation is organized as follows. We address the planner's problem in Chapter 7, where we propose a 2-phase policy that is composed of a learning phase and a knowledge-based deployment phase. In the learning phase we develop a semi-offline algorithm that optimally allocates organs to candidates (given a specific allocation objective). As an input to the semi-offline allocation problem we use the entire information of candidate and organ arrivals and a set probabilistic health condition and lifetime scenarios of candidates on the waiting list. We later use the knowledge gained from the learning phase to derive a knowledge-based, real-time allocation policy. The novelty of the 2-phase method is that it incorporates the future uncertainty of allocations into the decision process, yet maintains computational feasibility regardless of the challenges that the large dimensionality of the problem presents.

In Chapter 8 we present a detailed model estimation procedure. We estimate candidates' year-gain from a transplanted organ (LYFT), sensitization level (CPRA), and lifetime. We also estimate organs' quality, measured by donor's profile index (DPI). We use these estimates to deploy the 2-phase policy that we proposed in Chapter 7 on the actual OPTN waiting list, as of August 2008.

Comparing the semi-offline allocation with real-time policies (in specific, with the current priority point (PP) system and the proposed highest-KAS-first (HKF) policy), we find that in terms of *equity*, both policies provide equal access to transplants to candidates of different races, ages, and tissue types. However, in terms of *efficiency*, there is a clear gap between the performance of semi-offline and real-time allocations (HKF and PP), in particular

- Waiting time to transplantation is more than 1 year shorter (on average), according

to the semi-offline allocation.

- The match between organ and recipients improves, resulting in a lower organ rejection rate and a higher correlation between organs' survival years and recipient lifetime. A poor correlation between organ and recipient survival times increases the need for retransplantation on the one hand, and deaths with functioning organs, on the other hand.

- Recipients' year-gain from transplanted kidneys increases by almost 7 months per recipient in the semi-offline allocation, compared to HKF, and more than 3 years compared to the current PP system.

Encouraged by this gap, we examine the actual allocation resulting from each policy and design a heuristic knowledge-based method that combines the knowledge gained from the semi-offline allocation along with the real-time HKF policy. We denote the new policy KB (stands for Knowledge-Based). The performance of the KB policy lies between the performance of the semi-offline and HKF policies: it improves upon HKF, yet, as expected, it is does not perform as well as the semi-offline allocation.

Another important feature of the semi-offline allocation and KB policy, compared to HKF, is that the variability of organ types offered to patients with similar health profiles is relatively small. That implies that the problem of patients' choice becomes less relevant. The reason is that patients have very low incentive to refuse organs: a patient who chooses to refuse an offer is most likely to be offered the same kidney quality all over again. We discuss patient choice in Chapter 9.

A schematic representation of our approach is given by the flow chart in Figure 6.2.

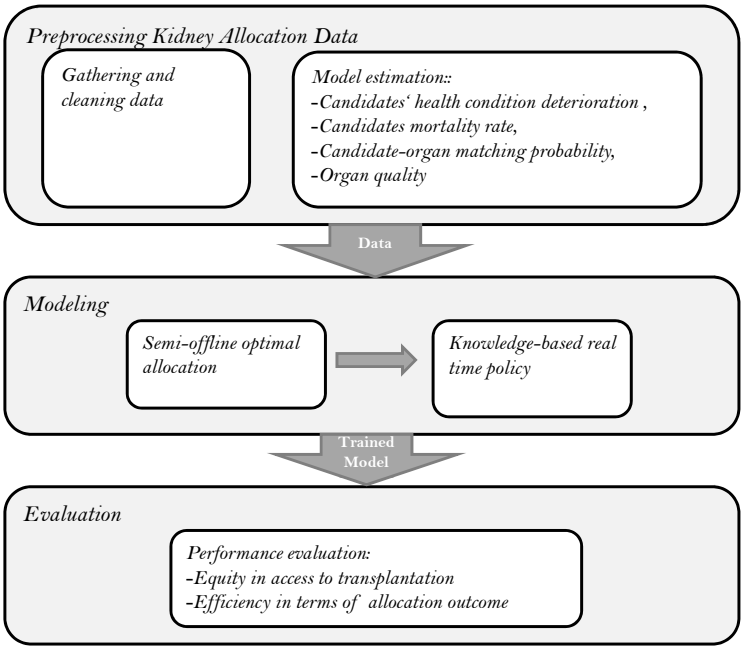

Figure 6.2: Kidney allocation schematic.

Chapter 7

Proposed Model for Kidney Allocation

In this chapter we examine improvements to the current and the proposed policy by KARS. In particular, we propose a novel semi-offline allocation model and derive a real-time knowledge-based allocation policy that outperforms the current and proposed policies.

7.1. Problem Description and Model Formulation

We consider the general problem of allocating kidneys to patients on the kidney waiting list. Organs are allocated upon arrival to patient p_i according to some allocation policy Φ. The patient then can either *accept* the kidney or *refuse* the kidney (seeking a better match). If the patient refuses the offered organ, the organ is re-allocated to another patient, according to the same policy Φ. When a patient accepts the kidney, a transplant is done. If a transplant is successful, the planner receives some reward r_{ij}. Otherwise, the organ is reallocated with some probability p, reflecting the probability that the organ is still qualified for retransplantation. Figure 7.1 illustrates the allocation scheme.

According to OPTN, most transplants are successful (approximately 95% of living-donor transplants are successful and more than 90% of deceased-donor transplants are successful, with increased success rate in recent years). We therefore assume that transplants are always successful. Additionally, we assume that patients always choose to

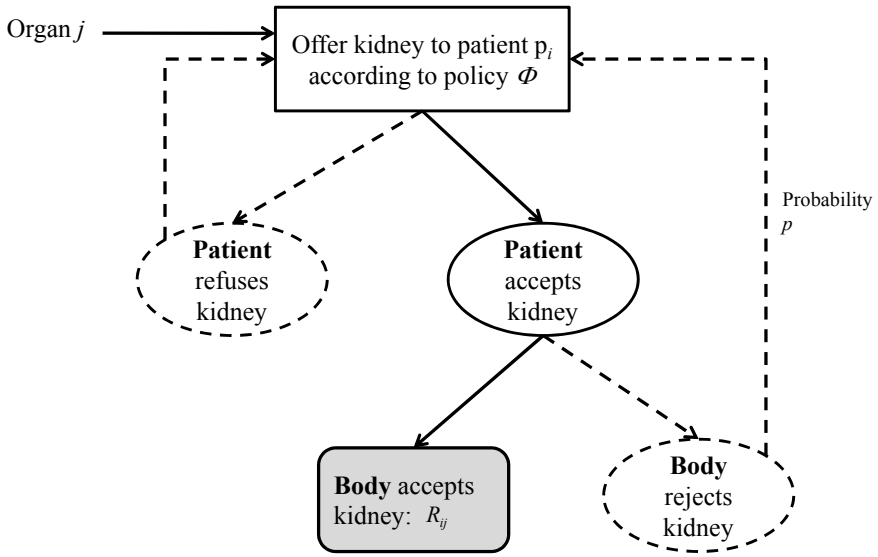

Figure 7.1: Schematic representation of kidney allocation.

accept offered kidneys. Our analyzed model based on these assumptions is marked with solid lines in Figure 7.1.

It is important to note that a kidney from a cadaver donor (approximately 2/3 of the transplants) can be preserved for up to 48 hours only (Hines and Marschall, 2008). Hence it is crucial to make an allocation decision in a timely manner, leaving sufficient time for tissue matching tests, organ shipping, and the actual transplant.

7.1.1 Notation

Let $P^t = \{p_1, p_2, ...\}$ be the pool of patients in the kidney waiting list at time t. Note that the patient pool changes over time, as patients may join the waiting list or depart from it (due to different reasons such as transplant, death, transplant in another country, etc.). We identify each patient by the tuple (t_i, h_i^t, sp_i^t), representing respectively the date s/he was placed on the waiting list, his/her conditional health state information at time

t (e.g., age, dialysis time, weight to height ratio (BMI), sensitization level, etc.), given that s/he has not received a transplant until time t, and the probability that the patient has survived up to time t. The survival probability replaces the patient's departure time information, because the latter remains unknown until the actual departure event.

We use O to denote the set of available organs for transplant, with $o_j = (t_j, d_j)$ being the j^{th} organ, where t_j is its arrival date, and d_j is the profile of the donor who donated the organ. The donor profile includes information on his/ her age, cause of death, whether or not s/he was diabetic, blood pressure at death time, etc. The reward from allocating organ o_j to patient p_i at time t, given the patient's health state and expected mortality rate at time t is given by $r_{ij}^t(h_i^t)$. Without loss of generalization, we assume that the allocation decision time is negligible, and $t = t_j$. We therefore use an abbreviated notation: r_{ij}^t.

We define x^Φ to be the allocation decision under policy Φ, with $x_{ij} = 1$ if organ o_j is allocated to patient p_i and $x_{ij} = 0$ otherwise. The objective is to find a policy that maximizes the total expected time-dependent reward, given by

$$TR^\Phi = \int_t r^{t'} x^\Phi dt \tag{7.1}$$

Note that $r_{ij}^{t_1}$ and $r_{ij}^{t_2}$ $(t_1 \neq t_2)$ may take different values. We do not, however, restrict the *order* of these two values (i.e., $r_{ij}^{t_1}$ can be either greater than, equal to, or less than $r_{ij}^{t_2}$).

7.1.2 Choosing Objectives

We are interested in obtaining the best attainable allocation of deceased donor kidneys to candidates[1]. Unlike liver and lung allocation policies, in which the objective is to reduce waiting list mortality and improve recipient survival during the first year following transplant, kidney failure is not an immediate cause of death and thus the allocation policy must be based upon different considerations as quality of life (with and without a transplant), waiting time, sensitization level, etc. (OPTN/UNOS, 2008).

Following the KARS recommendation, we use the expected KAS (see equation (6.1)) as the allocation reward. The KAS metric is important in that it balances equity in access to transplantation and allocation efficiency factors (i.e., the aggregate health of the transplant candidate pool). However, we propose a dynamic allocation that utilizes the donors and candidates arrival distribution and candidates' expected health condition degradation and mortality rate. Given that, the reward function is given by:

$$r_{ij} = \begin{cases} KAS_{ij} & \text{if } p_i \in P^{t_j} \text{ (i.e. } t_i \leq t_j \text{ and patient } p_i \text{ has not departed by time } t_j) \\ 0 & \text{otherwise} \end{cases} \tag{7.2}$$

As future departure events are unknown, we use the expected reward, denoted by the KAS value multiplied by the probability that the patients have not departed the waiting list by the time of the offer:

$$er_{ij} = KAS_{ij} sp_i^{t_j}. \tag{7.3}$$

[1]We disregard living-donor transplants, as those are allocated to family-related patients.

7.2. Proposed Real-time Dynamic Allocation Policy

We propose a two-phase allocation policy that provides real-time semi-optimal allocation under the critical constraint of making an allocation decision in a timely manner. The policy works as follows: (1) compute the semi-offline allocation of organs to candidates, based on the entire historical information available at hand, and (2) derive a knowledge-based allocation policy that utilizes the properties of the optimal allocation and deploy this knowledge in real-time.

We next explain each of of the two components in further detail.

7.2.1 Semi-Offline Optimization

We formulate the problem of allocating deceased kidneys to candidates in the OPTN waiting list as a semi-offline optimization problem. Here, the entire input of kidneys and candidates (i.e., past and future arrivals) is available from the start. Whereas the information on kidney arrival times and donors profiles are complete, the available information on candidates' health conditions and lifetimes are only partial. The reason is that candidates who received a kidney under the current policy might not receive a kidney under the optimal policy. The health condition and lifetime of such candidates, had they not received a kidney, is unknown. We thus replace the missing information by a set of probabilistic scenarios.

The semi-offline representation of the allocation problem has two roles. First, it provides an upper bound on the best attainable expected performance by any real-time allocation scheme, *given a specific allocation objective* (e.g., KAS). Hence, an upper bound

will enable us to assess the performance of any real-time allocation policy. Second, we utilize the knowledge we gain from semi-offline allocation on the properties of that optimal allocation, and derive a real-time knowledge-based allocation that generates a near-optimal allocation that is computationally fast.

Since allocation is a one-to-one mapping between kidneys and candidates we can restate the allocation problem as a max-weighted matching problem on a bipartite graph[2]. Specifically, consider a bipartite graph in which the organs constitute one set of nodes and the candidates constitute another. An edge between organ j arriving at time t_j, and candidate i is drawn if and only if j arrives later than i ($t_i \leq t_j$). The weight on this edge equals the expected reward of a successful transplant er_{ij}. Note that the reward er_{ij} is a function of the estimated health condition of candidate i at time t_j ($h_i^{t_j}$). A schematic representation of the allocation problem is given in Figure 7.2.

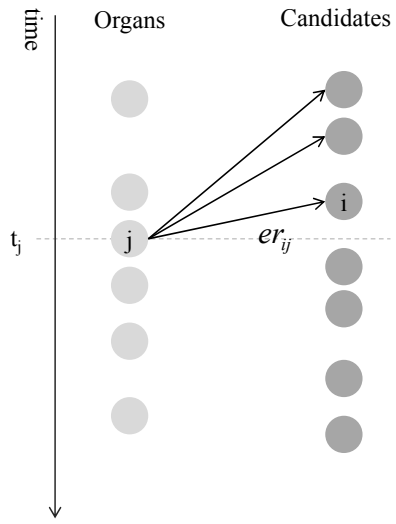

Figure 7.2: Semi offline representation.

The objective is to maximize the overall allocation reward, denoted by the sum of edge weights:

[2]For a definition of max-weighted matching consider Edmonds (1965).

$$\begin{aligned} & \max \sum_{j\in O, i\in P} er_{ij}^{t} x_{ij} \\ \text{s.t.} \quad & \sum_{j\in O} x_{ij} \leq 1, \forall i \in P \\ & \sum_{i\in P} x_{ij} \leq 1, \forall j \in O \\ & x_{ij} \in \{0,1\} \end{aligned} \tag{7.4}$$

The problem of finding the max-weighted matching is known to be NP-complete.

7.2.2 Knowledge-Based Real-time Allocation Policy

The semi-optimal allocation proposed in Section 7.2.1 cannot be obtained in real-time, as future kidney and candidate arrivals are unknown. One way of utilizing the optimal formulation is by evaluating a reward of an assignment based on a probabilistic set of future scenarios (see Bardossy and Yahav (2010)). However, since the size of the problem is extremely large (more than 90,000 candidates are currently waiting for kidney transplantation, with an annual increase of about 20,000 candidates) and the semi-offline problem is NP-complete, such an approach would suffer from heavy computational requirements and thus may perform poorly in terms of solution time. Therefore, rather than designing an optimal allocation, there is need for a computationally efficient policy that can handle the dynamic and stochastic nature of the real-time problem.

We design a knowledge-based, robust allocation algorithm that meets the computational requirement and provides a near-optimal allocation. The algorithm operates in four steps:

1. Divide the dataset into training and holdout sets.

2. Compute the optimal semi-offline allocation, as given by problem (7.4), on the training set

3. Let $A = a_1, ..., a_n$ be the set of paired allocations resulting from Step 2 ($a_1 = (p_1, o_1)$), and $S = s_1, ..., s_n$ be the profile of the donors and recipients in each allocation. Derive a knowledge-based (herein referred to as KB) policy based on the common properties of S. The details of this step are given in Section 8.3.2

4. Deploy policy KB in real-time.

In the next chapter we deploy this policy on the OPTN waiting list.

Chapter 8

Analytical Framework

In this section we describe the analytical framework that integrates the methods from Chapter 7 for deriving an optimal policy and generating knowledge-based rules. We apply this framework to the kidney waitlist data provided by UNOS, and show the performance of the resulting policy compared to the current priority-points (PP) policy and KARS proposed (HKF) policy.

8.1. Overview

The main goal of this section is to develop a real-time policy for allocating deceased kidneys to candidates in the UNOS waiting list. The policy is evaluated based on the equity in access to transplantation it provides and its efficiency in terms of the allocation outcome.

We describe a framework for deriving a knowledge-based allocation policy that is based of a model estimation and a data driven semi-offline optimization analysis that serves as a learning phase for the real-time policy. The framework is composed of several elementary steps. First, we estimate candidates health profile (current and prospect) and organ quality (Section 8.2). In particular, we estimate candidates' sensitization level (CPRA), year gain from a given organ (LYFT) and survival probability, and an organ survival index (measured by DPI value). We use these estimates to compute the KAS

value for each candidate-organ pair.

The KAS score, along with the survival probability, is then used to compute the expected KAS (also referred to as *expected reward*) and serves as an input to the semi-offline allocation in equation (7.4). We use the outcome of the semi-offline allocation to derive our knowledge-based, real-time policy (KB).

To simulate a real-time environment, we first estimate candidates' expected lifetime. Patients in our simulated environment arrive according to their actual arrival time (as recorded by UNOS) and depart according to their estimated lifetime. We use this simulation to compute the outcome of real-time policies (PP, HKF and KB), for the purpose of comparison and evaluation.

A schematic representation of this process is provided in Figure 8.1. We next discuss each step in detail using the UNOS data.

8.1.1 Data

We consider a dataset of waiting list registrations and transplants of kidney and simultaneous kidney-pancreas[1] that have been listed or performed in the U.S. and reported to the OPTN since October 1, 1987. The dataset includes records on both deceased and living-donor transplants. The data were exclusively provided by UNOS.

Preliminary analysis of the data exhibits a rapid increase in kidney donations over the last two decades. Hence, our analysis is based on the last five years only (Jan 1, 2004- Aug 15, 2008), for which the data seem to have a stationary nature (e.g., the

[1]Simultaneous transplantation of the kidney and pancreas is performed for those who have kidney failure as a complication of insulin-dependent diabetes mellitus (also called Type I diabetes).

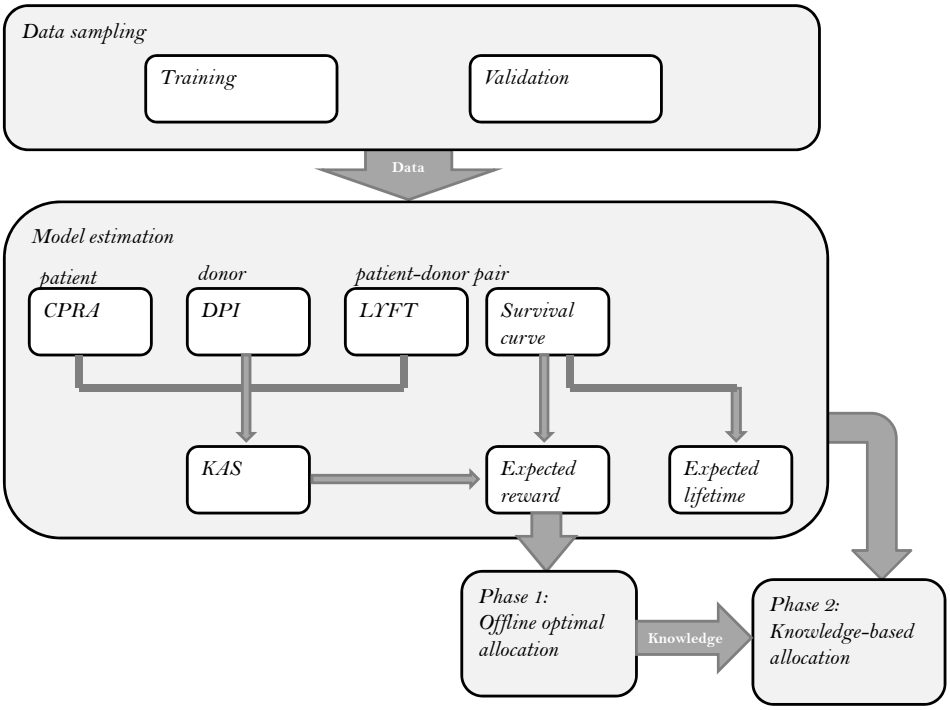

Figure 8.1: Schematic representation of the analytic study.

distributions of candidate and organ arrivals do not change significantly over time)[2]. For the purposes of our study, we consider only deceased-donor transplants of kidneys and simultaneous kidney(s) and pancreas. For computational reasons, we apply the policies and evaluate them for a single geographic region. We randomly chose region #2 that contains the following states: Pennsylvania (PA), New Jersey (NJ), West Virginia (WV), Maryland (MD), Delaware (DE), and Washington DC (DC). The data contain a total of approximately 26,000 candidates, 3,000 donors, and 5,800 kidney donations. A breakdown of candidate and donor arrival by year and state is given in Figure 8.2 (the data available for year 2008 is only up to Aug/15/2008).

We compile the data to extract the key profile components of candidates and donors. We summarize the average profiles in Tables 8.1-8.2. Table 8.3 depicts the kidney trans-

[2]Candidates that were placed in the OPTN waiting list before 01/01/2004 and are still on the waiting list as of that date are included in the study.

plant failure rate.

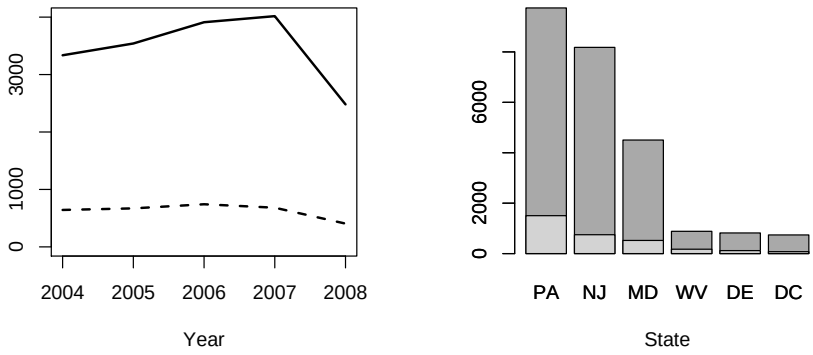

Figure 8.2: Left: Number of candidates (solid line) and donated kidneys (dashed line) added to the waiting list per year. Right: Number of candidates (dark grey) and donated kidneys (light grey) per state.

8.2. Model estimation

8.2.1 Estimating Life Years From Transplant (LYFT)

The concept of LYFT, which is the estimated number of additional years of life that a specific kidney would give to a candidate, was introduced by KARS in 2004. The calculation is based on estimating the remaining lifetimes of candidates with and without transplant, given the medical profile of the candidates and the donors at the time of offer. Evaluation of these two potential lifetimes is complicated for two reasons. First, the characteristics of candidates that have received transplants are different from those who have not. Some of these characteristics have not even been recorded in the past. Second, treatment methods of patients without a transplant have changed over the last two decades, making predictions of future lifetimes less certain[3].

[3]See "Predicting the Life Years From Transplant (LYFT): Choosing a Metric", Scientific Registry for Transplant Recipients working paper, May 16, 2007 at www.unos.org.

Table 8.1: Patient profile statistics

Number of patients	25860
Mortality rate	12% (constant over the studied period)
Average waiting time	3 years (until either departure time or data end date (Aug/15/2008))
Average time to transplant	2 years
Average patient age	49.6 years
Patients with diabetes	11618 patients
Patients on dialysis	18477 patients
Patients waiting for simultaneous kidney-pancreas	1266 patients
Patients who had a previous transplant	4419 patients
Average dialysis years, for patients on dialysis	1.9 years
Average body mass index (BMI)	27.7
Average albumin level	3.9
Average PRA level	19.1

Several methods were proposed for estimating the LYFT (Wolfe, 2007; Wolfe et al., 2008). The idea central to all of the methods is to estimate the remaining lifetimes with and without transplant, given the characteristics of a candidate and donor organ. This is typically done using the Cox proportional hazards regression model (Cox, 1972; Cox and Oakes, 1984). The main metrics considered are the expected lifetime (area under the survival curve), truncated lifetime (area limited to a specified interval, such as up to 30 years), and the median lifetime. The latter was shown to provide the most stable and interpretable estimates. We therefore use the Cox model estimated by Wolfe (2007) to estimate the candidates' median survival times (in years) with and without transplant. The formula Wolfe used to calculate LYFT is given by:

Table 8.2: Donor profile statistics

Number of donors	3144
Number of kidney donations	5828
Average donor age	39.5 years
Average creatinine blood levels	1.14
Donors with history of hypertension	1626 donors
Donors with history of diabetes	134 donors
Donor cause of death	Anoxia: 1155 donors
	Cerebrovascular/stroke: 2338 donors
	Head trauma: 2148 donors
	Center Nervous System tumor: 52 donors
	Other: 159 donors

Table 8.3: Transplant failure rate

Number of transplants	5828
Number of immediate failures	90 (1.5%)
Number of failed transplants in the first week	153 (2.6%)
Number of failed transplants in the first year	612 (11.6%)
Number of failed transplants in the first 5 years	938 (16.1%)

$$\begin{aligned} \text{LYFT} = \quad & 1.0 \times \text{lifespan with functioning graft} + \\ & 0.8 \times \text{lifespan after graft failure} - \\ & 0.8 \times \text{remaining lifespan without transplant} \end{aligned} \tag{8.1}$$

8.2.2 Computing Calculated Panel Reactive Antibody (CPRA)

Panel Reactive Antibodies (PRA) is a metric that determines the degree of sensitization of patients. The value of PRA is obtained via a lab test that examines the antibodies of a patient against a panel of about 100 blood donors. Hence, this metric heavily depends on both the panel composition and the technique used for antibody detection, resulting in a very inconsistent and unreliable metric (Cherikh, 2006).

In 2004, the OPTN Histocompatibility Committee that examines the PRA listing practices proposed an alternative metric, called 'calculated PRA' (CPRA). CPRA measures the *likelihood* that the recipient and donor would be incompatible, based on HLA frequencies in donors (based on more than 12,000 recent donors (Leffell et al., 2007)). CPRA is one of the key components of the KAS value.

To compute the CPRA values of candidates on the waiting list, a computer program was written (in PHP language) that accesses the UNOS CPRA calculator (available online http://www.unos.org/resources/frm_CPRA_Calculator.asp?index=78). For each candidate on the waiting list, the computer program automatically obtains the CPRA value. Interestingly, we find that there is no correlation between the PRA and the CPRA metrics. Furthermore, approximately 60% of the candidates in the waiting list have a recorded PRA of 0, which implies that they are likely to be compatible with any donor, whereas only 0.2% have a CPRA value of 0. Highly sensitized patients form 12.9% of the waiting list according the PRA matric, and 9% according to the CPRA metric. These differences further support that the PRA metric is unreliable. We plot the distributions of the two metrics for patients in our sample in Figure 8.3.

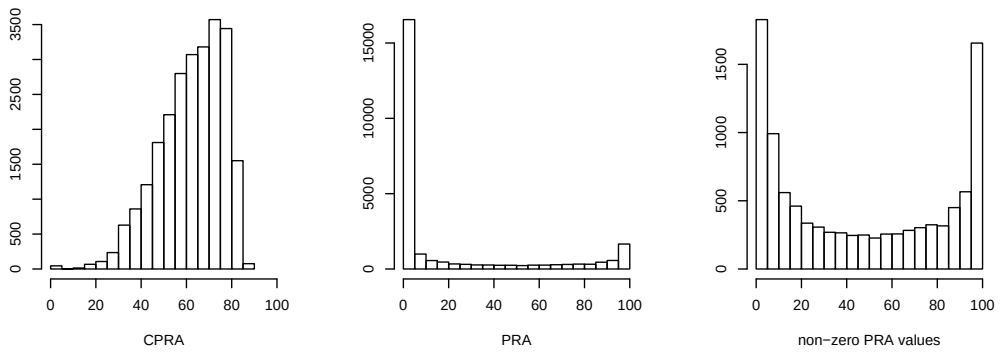

Figure 8.3: From left to right: distribution of CPRA, PRA, and non-zero PRA values.

8.2.3 Modeling Patient Lifetime

The waiting list of kidney transplants changes over time, as patients join the waiting list or depart from it. One of the main reasons for waiting list departure is mortality (over 10% of the patients die yearly). We utilize the OPTN waiting list data to estimate the mortality rate in the patient population given their health profile. Due to changes in recent medicine, we consider a subset of the patients who joined the waiting list after January 2000. This left and right truncated subset (that is, the studied period is in the range [2000, 2008]) ensures that the patients' arrivals are approximately uniformly distributed over the studied interval. Our subset includes over 265,000 patients, out of which about 27,500 have died.

Our first deviation from common practice is to estimate survival times parametrically. Although the Cox proportional hazards models is widely used in medical research (e.g., Wolfe et al. (2008)), we use a parametric Accelerated Failure Time (AFT) model with Weibull distribution to estimate patient survival rates (Bradburn et al., 2003). AFT models are shown to be more robust towards neglected covariates, compared to proportional hazards models (Lambert et al., 2004). Another main advantage of the AFT model over the Cox model in our case is that it enables us to extrapolate survival rates beyond the studied interval (8 years). The Cox model, on the other hand, fits a survival model to the data-at-hand only, and hence does not provide extrapolation capabilities.

The choice of a Weibull distribution (rather than other common survival distributions such as lognormal and exponential) stems from two main reasons. First, Weibull is a very flexible lifetime distribution with shape and scale parameters that enable modeling of a wide range of failure rates.

Second, comparing the fit of the different distributions to our data, the Weibull exhibits by far the least skewed residuals, as shown in Figure 8.4. We see that the AFT with a lognormal distribution and the Cox model both tend to underestimate candidate lifetimes (in fact, the maximum predicted lifetime in both models is 8 years). The reason for the underestimation is that candidates in the training data arrived in the interval [2000, 2008], whereas candidates in the holdout set may have arrived before the year 2000, thereby surviving longer than 8 years. In addition, because candidates' observed lifetimes are measured only up to the data end date (that is, Aug/15/2008), whereas the estimated lifetime (from the survival model) is computed until expected departure time, a right skewed residual distribution is conceptually more appropriate.

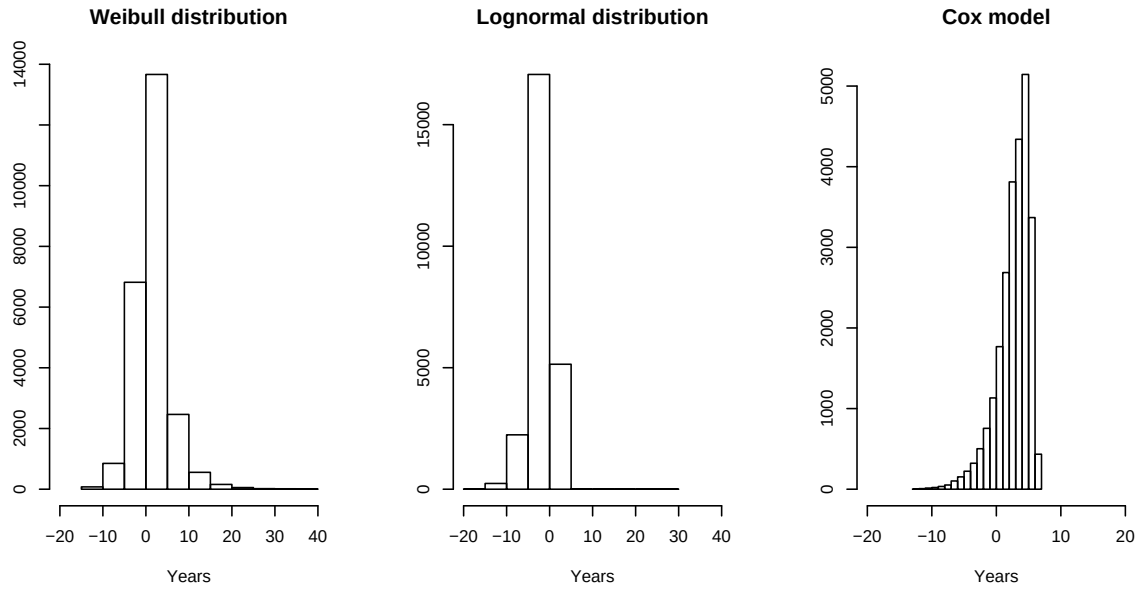

Figure 8.4: Residuals (estimated lifetime - observed lifetime) distribution for different survival models. The Weibull distribution (left panel) presents right skewed residuals, which are most appropriate for estimated lifetime.

The predictors (and their description) that are included in the survival model are given in Table 8.4. The choice of predictors stems from the factors used in approximating the patients survival factors in Wolfe et al. (2008).

The estimated parameters in our model are provided in Table 8.5. All estimates are significant at the 5% significance level due to the large sample size. We therefore focus on the effect size (e.g., the model coefficient values). We observe that CPRA have a minuscule effect (although it is statistically significant) on estimated lifetime, which is reasonable, as CPRA measures the sensitization level and the probability of a successful transplant.

We illustrate the effects of diabetes (DIAB) and need for a simultaneous pancreas-kidney transplant (KP) in Figure 8.5, using the profile of a 30 year old patient with no previous transplants and no dialysis. The solid black curve corresponds to a patient with no diabetes that waits for a kidney only. The grey line is a patient with diabetes that waits for a kidney only. The dashes line represents a patient with diabetes that waits for a simultaneous pancreas-kidney transplant. As expected, the first patient has the highest estimated survival rate, and the third patient has the lowest (although only slightly lower than the second patient, in this case).

The effect of age and the interaction between age and diabetes is plotted in Figure 8.6. We compare three types of patients: a 20-year old patient without diabetes (solid black line), a 60-year old patient without diabetes (dashed line), and a 20-year old patient with diabetes (grey line). All three patients have no other special characteristics (such as previous transplants, antigens, dialysis, etc.). On average, a 20-year old is expected to live approximately 20 years longer than a 60-year old, and 17 years longer than a same age patient with diabetes.

Finally, we illustrate the effect of antigens on survival rate in Figure 8.7. We compare a 30-year old patient with no antigens (and no other special characteristics, solid line),

to a similar patient with a single A antigen, B antigen, and DR antigen. The no-antigen patient in our example is expected to live 10 years longer than the antigen patient.

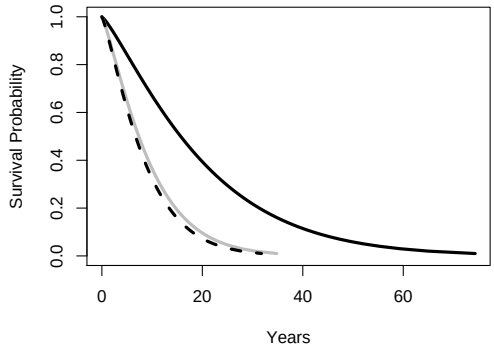

Figure 8.5: Effect of diabetes and simultaneous pancreas-kidney transplant on patient lifetime. Solid (black): no diabetes, kidney only. Dashed: diabetes, simultaneous pancreas-kidney transplant. Grey: diabetes, kidney only.

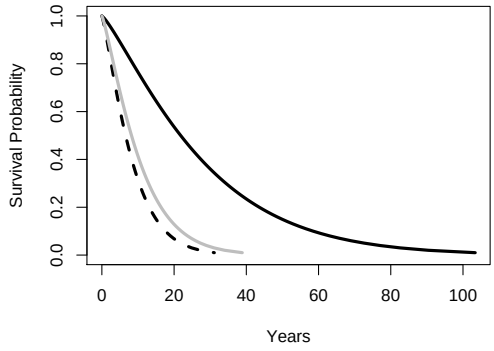

Figure 8.6: Effect of age on patient lifetime. Solid (black): 20 year old patient with no diabetes. Dashed: 60 year old patient with no diabetes. Grey: 20 year old with diabetes.

8.2.4 Computing Donor Profile Index (DPI)

DPI is a proposed metric that provides a continuous measure of organ quality, which predicts the expected lifetime of a kidney transplanted to an average candidate (OPTN/UNOS, 2008). Currently, deceased donors are designated as either standard cri-

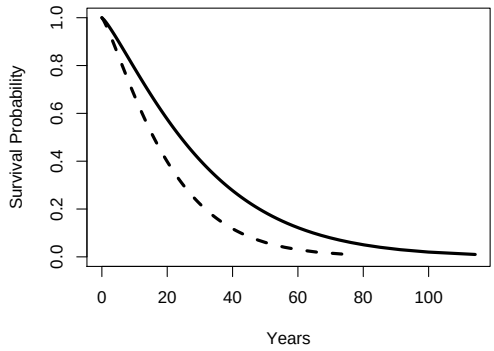

Figure 8.7: Effect of antigens on patient lifetime. Solid: no antigent (ABDR=0). Dashed: a patient with a single antigen A, B and DR.

teria donors (SCD), or expanded criteria donors (ECD), based on the medical characteristics of the donor. ECD kidneys are defined as having a relative risk of graft failure (due to a history of of high blood pressure, high creatinine level, age, etc.). However, the current categories of SCD and ECD kidneys are shown to no longer be adequate, as they do not reflect the recipient outcomes from receiving these organs.

The DPI score is based on the following donor criteria: age, creatinine level, history of hypertension and diabetes, and cause of death. Organs with the longest expected survival time are assigned a DPI score of zero, and those with the shortest expected survival time are assigned a DPI score of one. A method for computing the DPI score was proposed by Freeman (2008). The formula is given by equation (8.2). As this formula results in non-normalized DPI value, we normalize the DPI to the range $[0, 1]$ by subtracting the minimum and dividing by the maximum DPI score in our sample.

$$
\begin{aligned}
\text{DPI} \quad &= \exp(0.01111^{*} \text{ Donor Age-35.8322}) \\
&+ (0.01672^{*} \text{ Donor Creatinine -1.0415}) \\
&+ (0.12012\ I_{\text{Donor Creatinine Missing}}) \\
&+ (0.13863\ I_{\text{Donor History of Hypertension}}) \\
&+ (0.17805\ I_{\text{Donor History of Diabetes}}) \\
&+ (0.05423\ I_{\text{COD Anoxia}}) \\
&+ (0.08832\ I_{\text{COD Stroke}}) \\
&- (0.04476\ I_{\text{COD CNS Tumor}}) \\
&+ (0.04222\ I_{\text{COD Other}})
\end{aligned}
\tag{8.2}
$$

The notation I_c is an indicator variable that obtains the value 1 if the condition c is satisfied and otherwise 0.

8.3. Model Deployment

We generate training and holdout sets by randomly assigning 10% of the patients and donors into the training set and another 10% into the holdout set. The reason for using only 10% in each data set is due to the computational constraints of the software used (GAMS 23.2). The training set is used to build the model, and the holdout set is used for evaluating model performance. Each sample contains a random sample of patients (approximately 2500 patients) in the waiting list and a random sample of donors (approximately 300 donors). We repeat the analysis on the 10 different training and holdout samples, to estimate the performance error. For each set we estimate the model parameters, as discussed in Section 8.2.

In the next sections, we perform an offline analysis and study the properties of the best attainable allocation, under the objective of maximizing the expected KAS score. We then derive a knowledge-based policy, based on these properties, and apply it in real-time settings.

We perform an evaluation study in which we compare our policy with the currect priority point (PP) system and KARS' proposed (HKF) policy. The performance of the policies is evaluated based on the following metrics:

1. Mean time to transplantation
2. Mean time on waiting list (of candidates who have not received a transplant)
3. Mean transplantation utility (measured as mean LYFT)
4. Correlation between graft expected lifetime (measured by donor's DPI) and patient's expected lifetime
5. Mortality rate

8.3.1 Semi-Offline Analysis

We compute the semi-offline allocation (see equation (7.4)) of each training set, using GAMS 23.2 (http://www.gams.com/). The mean performance of the semi-offline allocation is summarized in Table 8.6. In parentheses we report the standard error across different runs.

For a fair comparison between the semi-offline allocation and the current (PP) and proposed (by KARS (HKF)) allocation policies, we generated a computer program (in R

2.8.1, http//cran.r-project.org) that simulates a real-time environment, in which donors and patients arrive according to their actual arrival order in each replica, and depart (due to mortality) according to their lifetime distribution. The simulation's pseudo-code is given in Algorithm 1 below. We compute the performances of the allocation policies (PP and HKF) on these simulated datasets. We report their performances in Table 8.10.

Algorithm 1 Simulation's pseudo-code

Input:$\{P, O, \text{policy}\Phi\}$
Output:$\{A\}$
for all $p_i \in P$ **do**
  $l_i = \int_t sp_i^t dt$ *//Estimate patient's life time*
end for
for all $o_j \in O$ (in their arrival order) **do**
  $P^t \leftarrow \{p_i \in P \mid t_i \leq t_j \ \ and \ \ t_i + l_i \geq t_j\}$
  $a = \{(o_j, p_i) \mid \Phi\}$ *//allocate organ* o_j *to patient* $p_i \in P^t$ *according to policy* Φ
  $A \leftarrow A \cup a$
  $P \leftarrow P - \{p_i\}$
end for

Comparing the semi-offline allocation with the other two policies, we can see improvement in several respects. First, the mean time to transplantation of the semi-offline allocation is shorter compared to either of the other policies (1.34 years compared to 3.96 years and 2.58 years of the PP and HKF policies, respectively). Second, the mean HLA mismatch of the semi-offline allocation is improved compared to HKF, implying that the probability of successful transplantation is improved. Moreover, we find that approximately 65% of the patients that receive an organ under both HKF and the semi-offline policy in our experiments had a better match with the organ they receive under the semi-offline policy, compared to what they receive under HKF. The current (PP) policy has the lowest HLA mismatch, since priority points of an allocation are given based on the match between the donor and the patient. Comparing the utility from an allocation (or in other words, the year-gain from allocating kidneys to candidates), we find that the semi-offline

policy achieves the highest utility. On average, candidates that receive a transplant under the semi-offline allocation in our experiments are expected to live almost 7 months longer than those who receive a kidney under the KAS policy, and almost 3 years longer than those who receive a kidney under the current PP policy. Finally, the semi-offline allocation better matches survival times of a transplanted kidney (measured by DPI, where low DPI corresponds tolong expected organ survival) with expected lifespans of the their recipient. This is a very important feature for two reasons. First, it decreases the probability that a patient would return to the waiting list after a transplanted graft failure. Second, it decreases the probability that a patient would die with a functioning graft, resulting in an increase in the graft utility. We illustrate this property in Figures 8.8 and 8.9. Figure 8.8 illustrates the relationship between organ survival and recipients' LYFT and dialysis time. High (low) LYFT value, as well as short (long) dialysis time, are likely to result in long (short) after-transplant lifetime. Figure 8.9 depict the ages of the matched donors and recipients. It is shown that young donors (less than 20 year old) are more likely to be matched with young candidates (less than 40 year old), whereas older donors are matched with older candidates.

The actual waiting time of candidates in the waiting list increases under the semi-offline policy by an average of approximately 1 year (compared to both PP and HKF policies). Although the increased waiting time may be considered a downside of the semi-offline allocation, it also implies that the semi-offline allocation does not operate as a "first-come-first-transplant" policy. Instead, it gives higher weight to allocation and graft utility than to waiting time.

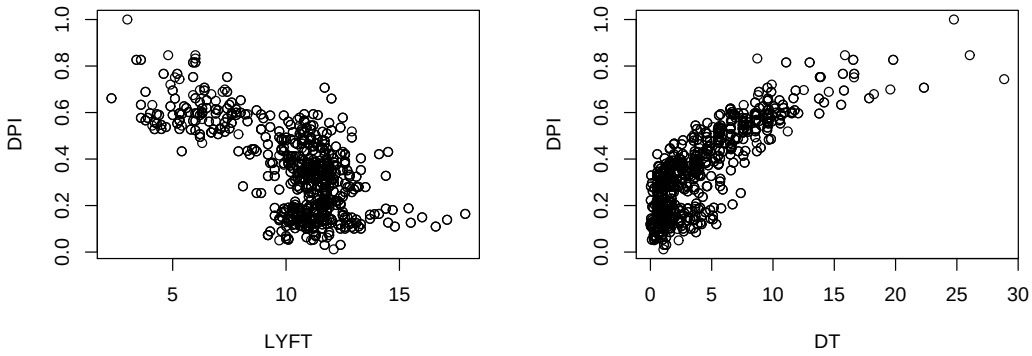

Figure 8.8: Relationship between organ survival and recipient lifespan (left) and dialysis time (right).

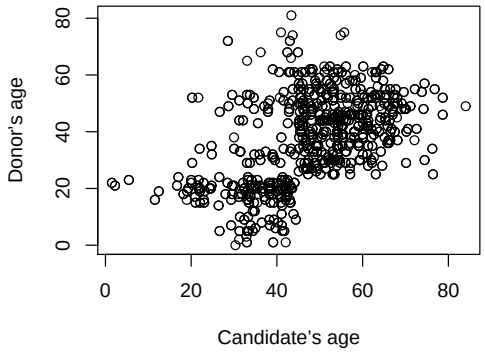

Figure 8.9: Relationship between donors' and recipients' age.

8.3.2 Knowledge-Based Real-Time Policy

We use the knowledge gained from the semi-offline allocation and propose a knowledge-based policy that mimics the improvements that the semi-offline allocation suggests on the HKF policy. We address the questions of *equity* and *efficiency*:

Equity: Does the semi-offline allocation provide a sense of equity in access to transplantation, or are candidates prioritized based on their health profile? Does the semi-offline

allocation provide better or worse equity compared to HKF?

Efficiency: Given that a candidate receive an organ allocation, what type of kidney would a recipient receive under the semi-offline policy compared to HKF, and when?

8.3.2.1 Equity

For the question of equity, we use the Kruskal-Wallis χ^2-test (Kruskal and Wallis, 1952) to compare the median health profile of candidates (e.g., dialysis time, antigens, age, etc.) to those of organ recipients, according to the semi-offline allocation. Table 8.8 summarizes the test results. We illustrate the insightful comparisons in Figure 8.10, where we show the distributions of CPRA and dialysis time on the top panels (with the median values represented as a hollow strip and the whiskers extending to the 5^{th} and 95^{th} percentiles), simultaneous kidney-pancreas and diabetes status at the middle panel, and ethnicity at the bottom panel. We find that the semi-offline allocation gives high priority to candidates with high sensitization level (high CPRA value), diabetes, long dialysis time, and to those who are waiting for simultaneous kidney-pancreas transplant. This result is not surprising, considering the structure of the KAS score (equation (6.1)) that gives priority based on CPRA value (up to 4 'points'), dialysis time (between 0.2 and 1 point per year), and LYFT (between 0.2 and 1 point per additional years gained from the transplant), which increases with diabetes and the need for simultaneous kidney-pancreas transplant (Wolfe, 2007). We also find that unlike the current PP system, the semi-offline allocation does not prioritize based on candidates' antigens, and consequently, does not prioritize based on ethnicity (for the equity problem caused by candidates' antigens and tissue types, consider (Eggers, 1995)). To compare the equity of semi-offline allocation to

that of the HKF policy, we perform a χ^2-test on the two groups of organ recipients in our experiments. We find that the median is statistically equal at the 1% significance level on all the different metrics, implying that **the equity provided by both policies is statistically equal**. In fact, we find that the two recipient groups overlap by more than 70%. Therefore, the semi-offline allocation cannot be used to improve the *equity* of the HKF policy.

8.3.2.2 Efficiency

To study the *efficiency* of the semi-offline allocation, we construct a regression tree on the allocation outcome that maps organ types, that is DPI values (y variable in the tree notation), to recipients' health profile: LYFT, CPRA, DT and Survival curves (x variable in the tree notation). The choice of variables stems from the concepts used to define the KAS value (see Section 6.2). In essence, the regression tree satisfies one main objective: it provides *knowledge* on what type of kidney is allocated to what type of candidates, in terms of their health-related properties. We later use this knowledge to develop our knowledge-based real-time policy. The resulting regression tree is given in Figure 8.11. On the regression tree, the patient pool is divided into 7 groups (7 leaf nodes), based on their LYFT and DT. The number in the leaf nodes represents the mean DPI assigned to each group.

Next, we study the actual distribution of organ quality, measured by the DPI value assigned to each patient type. We summarize the DPI distribution per group in Table 8.9. We also compare the distribution of organ types per patient group of the semi-offline allocation to that resulting from the HKF policy in Figure 8.12. The results are provided

as side-by-side boxplots. In comparison, we find that the ranges of DPI values assigned to each group are smaller, implying that the match between donors and candidates is more accurate under the semi-offline policy. In other words, **the semi-offline allocation is more efficient than the real-time HKF allocation**.

We can backtrack to the efficiency insights of the semi-offline allocation to derive a knowledge-based real-time policy (KB): upon arrival of new organ o_j, with DPI value equals dpi_j we consider patient groups (g_k, $k \in (1, 2, ...7)$) such as dpi_j to be within the [5%, 95%] percentile DPI range of these groups, according to Table 8.9. The organ is then allocated to the candidate with the highest KAS value within these groups. The policy is given in Algorithm 2.

Algorithm 2 KB policy

On new organ o_j arrival, with DPI value= dpi_j**:**
$G_j = \{g_k \mid dpi_j \in [5\%, 95\%]$ percentile DPI of group $k\}$
$P_j = \{p_i \in G_j\}$
$a = \{(o_j, p_i) \mid p_i \in G_j \ \ and \ \ KAS_{ij} = \max_{p_k \in G_j} \{KAS_{kj}\}\}$

Lastly, we compare the performance of the KB policy to that of PP and HKF on 10 different holdout sets. The results are summarized in Table 8.10. We find that KB significantly improves the performance of HKF and PP on four planes: (1) decreased time to transplant, (2) higher utility, (3) better match between graft expected lifetime and patient expected lifetime, and (4) shorter time on waiting list (compared to HKF only). The mortality rate, however, significantly increases by 1.2% yearly (approximately 25 candidates in our sample). A possible reason for that increase is the expected increase in after-transplant lifespan and decrease in reduction rate.

Table 8.4: Predictors used in estimating candidates survival curves

Predictor	Description
DIAB	Indicates whether a patient is diabetes (DIAB=1 if diabetes).
KP	Indicates whether a patient is waiting for simultaneous pancreas-kidney transplant (KP=1 if simultaneous transplant).
DIAL	Indicates whether a patient needs dialysis (DIAL=1 if dialysis).
PrevTRANS	Indicates whether s patient had previous transplants (PrevTRANS=1 if previous transplants).
DT	Dialysis time in years upon arrival.
AGE	Patient's age upon arrival.
POLYCYSTIC	Indicates weather a patient was diagnosed with polycystic kidney syndrome (POLYCYSTIC=1 if yes). Polycystic kidney syndrome is a genetic disorder that results in massive enlargement of the kidneys. The disease can also damage the liver, pancreas, and in some rare cases, the heart and brain.
HYPERTENSION	Indicates whether a patient was diagnosed with malignant hypertension (HYPERTENSION=1 if yes). Malignant hypertension is a complication of hypertension characterized by very elevated blood pressure. Malignant hypertension can damage the kidneys as well as the eyes, brain and heart.
NotSPECIFIED	Indicates whether a patient has no diagnosis (NotSPECIFIED=1 if no diagnosis).
BMI	Patient's Body Mass Index (ratio of weight to square root of the hight). BMI provided a measure of a patient's overweight (BMI¿25) or underweight (BMI¡18.5).
ALBUMIN	Patient albumin level. Low albumin levels reflect possibility of diseases in which the kidneys cannot prevent albumin from leaking from the blood into the urine and being lost.
A	Number of a patient's A antigens.
B	Number of a patient's B antigens.
DR	Number of a patient's DR antigens.
ABDR	Indicates weather the patients has no antigens (ABDR=1 if A+B+DR=0)
CPRA	Patient's CPRA level.

Table 8.5: Estimated coefficients of the AFT models (sample size≈ 270k)

	coef	exp(coef)
(Intercept)	2.65	14.21
DIAB	-0.96	0.38
KP	-5	0.01
DIAL	-0.38	0.68
PrevTRANS	-0.32	0.73
DT	-0.02	0.98
AGE	-0.03	0.97
POLYCYSTIC	-0.18	0.84
HYPERTENSION	0.33	1.39
NotSPECIFIED	-0.03	0.97
BMI	0.02	1.02
ALBUMIN	0.43	1.54
A	-0.25	0.78
B	-0.22	0.8
DR	-0.25	0.78
ABDR	-0.3	0.74
CPRA	0.002	1
DIAB:KP	5.06	158.3
DIAB:AGE	0.02	1.02
DIAB:ALBUMIN	-0.11	0.89
KP:ALBUMIN	0.59	1.81
KP:AGE	0.04	1.04
DIAB:KP:ALBUMIN	-0.63	0.53
DIAB:KP:AGE	-0.04	0.96
Shape	1.20	

Table 8.6: Semi-offline performance on training dataset (standard error in parentheses)

metric	Semi-offline performance
Age at time of offer	47.40 (0.43))
Time to transplantation (years)	1.34 (0.05)
HLA mismatch	3.94 (0.02)
Utility (LYFT)	10.37 (0.07)
Correlation between graft expected lifetime and patient's expected lifetime	0.69 (0.01)
Mean waiting time on waiting list	3.59 (0.03)

Table 8.7: Performance of Real-time policies (on training dataset)

metric	HKF	PP
Age at time of offer	47.47 (0.39)	46.79 (0.73)
Time to transplant (years)	2.58 (0.07)	3.96 (0.06)
HLA mismatch	4.21 (0.02)	2.66 (0.03)
Utility (LYFT)	9.79 (0.08)	7.49 (0.08)
Correlation between graft expected lifetime and patient's expected lifetime	0.66 (0.01)	-0.28
Mean waiting time on waiting list	2.74 (0.03)	2.45 (0.07)
waiting list mortality	15.2% (0.04)	14.7% (0.05)

Table 8.8: Comparison between profile distribution of candidates and recipients (Kruskal-Wallis χ^2-test)

metric	χ^2	p-value	statistically equal?
CPRA	77.46	0	no
DIAB	205.55	0	no
KP	16.05	0	no
PrevTrans	1.73	0.19	yes
DT	171.51	0	no
AGE	31.25	0.07	yes
A antigens	0.09	0.77	yes
B antigens	3.23	0.07	yes
DR antigens	2.12	0.15	yes
BMI	0.53	0.47	yes
ALBUMIN	8.58	0	no
Ethnicity	0.85	0.36	yes

Table 8.9: Patient type and organ allocation

Group	Group properties	$[5^{th}, 50^{th}, 95^{th}]$ percentile DPI	mean DPI
1	$DT \in [0, 0.52)$	[0.05, 0.12, 0.29]	0.14
2	$DT \in [0.52, 2.53)$	[0.10, 0.22, 0.40]	0.24
3	$DT \in [2.53, 5.41)$	[0.13, 0.35, 0.51]	0.33
4	$DT \in [5.41, 8.68)$, $LYFT \in (-\infty, 8.25)$	[0.51, 0.56, 0.62]	0.56
5	$DT \in [5.41, 8.68)$, $LYFT \in [8.25, \infty)$	[0.27, 0.47, 0.58]	0.46
6	$DT \in [8.68, 12.28)$	[0.50, 0.60, 0.71]	0.61
7	$DT \in [12.28, \infty)$	[0.63, 0.74, 0.85]	0.74

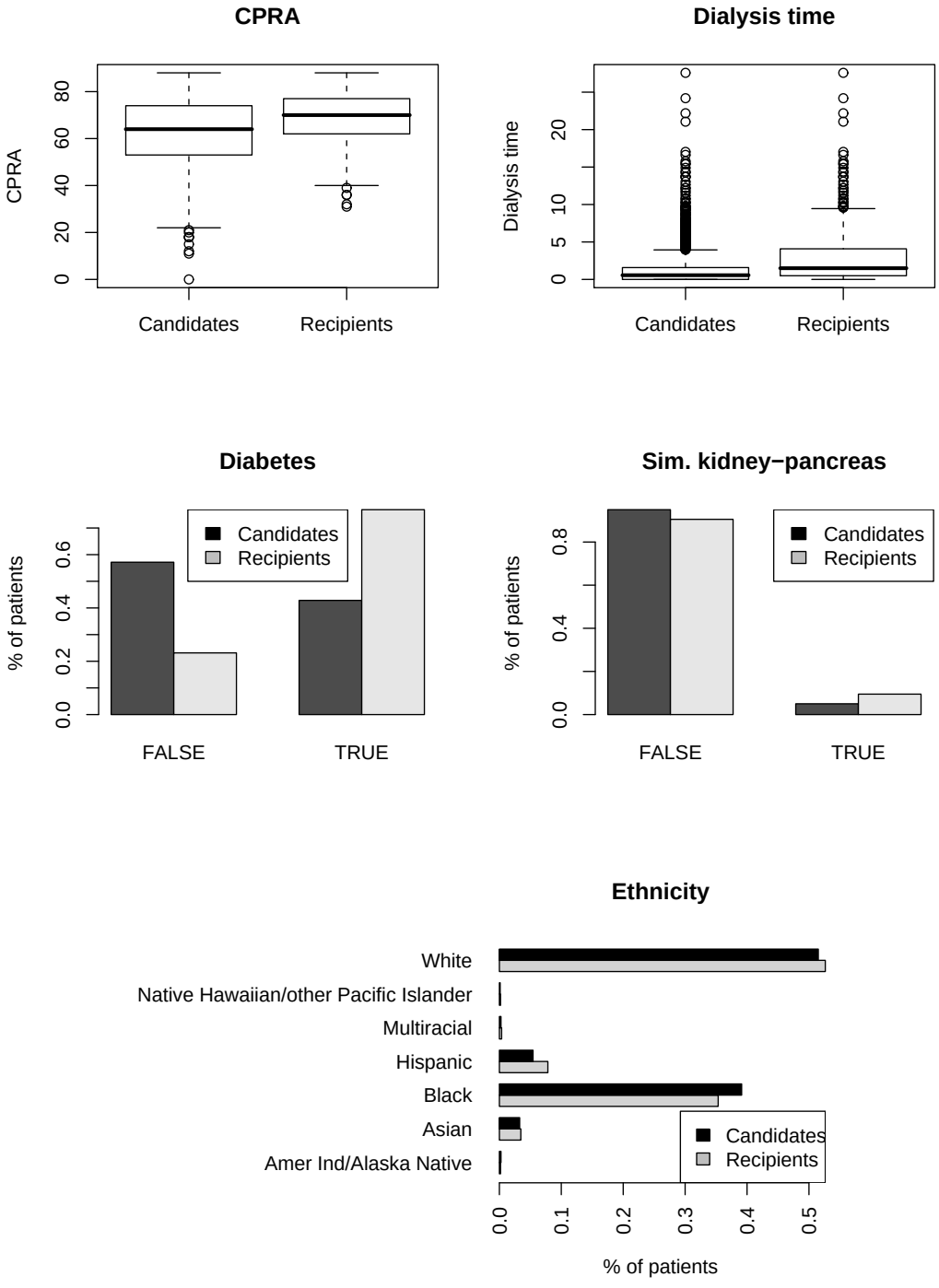

Figure 8.10: Comparison of profile distribution of candidates and recipients.

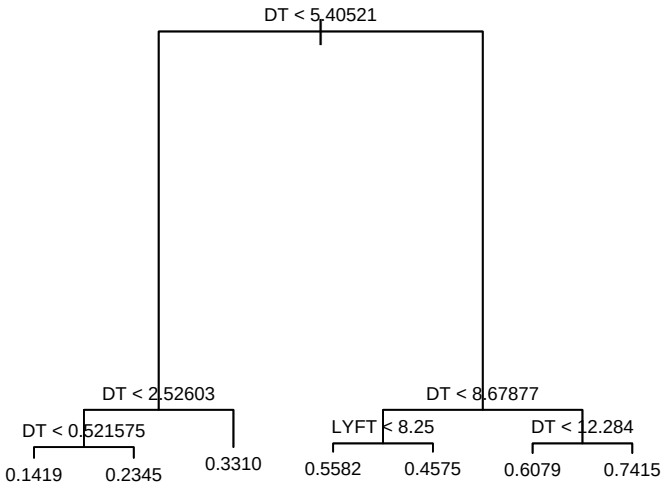

Figure 8.11: Regression tree on the allocation outcome that maps organ types to recipients' health profiles. Leaf nodes give average DPI.

Table 8.10: Real-time policies performance (on holdout dataset)

metric	KB	HKF	PP
Age at time of offer	46.88 (0.41)	46.06 (0.42)	46.73 (0.71)
Time to transplant (years)	2.25 (0.05)	2.50 (0.03)	4.22 (0.06)
HLA mismatch	4.19 (0.02)	4.15 (0.02)	2.67 (0.03)
Utility (LYFT)	9.9 (0.06)	9.88 (0.07)	7.46 (0.08)
Correlation between graft expected lifetime and patient's expected lifetime	0.69 (0.01)	0.64 (0.01)	-0.26 (0.03)
Mean waiting time in waiting list	3.11 (0.05)	3.23 (0.04)	2.79 (0.06)
waiting list mortality	16% (0.04)	14.8% (0.05)	13.7% (0.03)

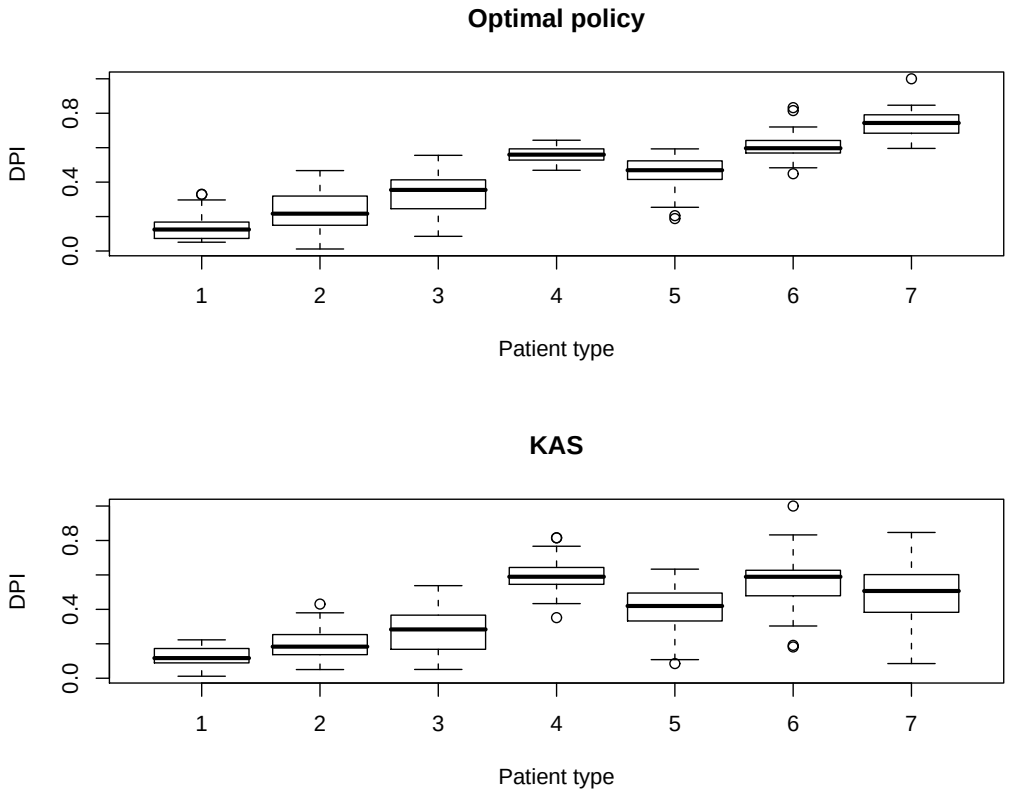

Figure 8.12: Comparing the semi-offline allocation and HKF allocation in terms of DPI (smaller variance is better). The solid line represents the median DPI value, and the whiskers extend to the 5^{th} and 95^{th} percentiles.

Chapter 9

Discussion and Future Work

In this work we discuss policies that allocate deceased donor kidneys to candidates with kidney failure. Following the OPTN report, we addressed the efficiency and equity in access to transplantation of the current allocation policy (Priority Points (PP)) and a new allocation policy, proposed by KARS in 2008 (Higher KAS First (HKF)). We then develop an alternative policy that is tailored to the properties of the kidney waiting list. Using a novel approach that combines data analytics and operations research methods, our allocation policy incorporates future prospect allocations into the decision making process, while accounting for dynamics in the queue, such as that of patients joining or leaving due to mortality.

Our work contributes to the kidney allocation field in both theoretical and applied aspects. On the theoretical side, we define a semi-offline optimization problem that provides the best attainable allocation of organs to candidates in the OPTN waiting list, given a specific objective. The solution to the optimization problem serves as an upper bound to any real-time allocation policy that considers the same allocation objective. In addition, we use data mining tools to study this solution and mimic it in real-time. In other words, we utilize the properties of the semi-offline allocation and derive a knowledge-based policy, which combined with the HKF policy, yields a better allocation at both individual and social levels (compared to PP and HKF). As a by product, we also numerically show the significant advantage of the proposed HKF policy over the existing PP system.

On the applied side, we provide a complete model estimation section in which we estimate candidates' changes in health condition, mortality rate, candidate-organ matching probability, and organ quality. These estimates are then used to tailor our proposed policy to the properties of the kidney waiting list data.

There are several directions for extending this work. First, in this work we focus on the planner problem, disregarding the patients' choice problem. However, literature shows that in practice about 45% of the offered kidneys are rejected by the first patient that receives it (or his/her physician). The rationale for such a decision is that once in the top of the list, it pays to wait for a better offer (Zenios, 2004; Su and Zenios, 2005). Patients' decisions might introduce inefficiencies into the system since transplants are delayed when patients reject offers. Understanding and modeling the patients' choice is hence a natural extension to our work.

Although we do not explicitly model patient choice, our analytical results show that under the proposed policy, the types of kidneys offered to patients with the same health condition has a relatively small variation. Therefore, as suggested in Su and Zenios (2005), our proposed policy is expected to be robust to patient choice. The reason is that if a patient rejects an offer, s/he is most likely to receive a kidney of the same quality in the next offer.

Another possible extension of this work, is evaluating recipients' lifespan and rejection rate after transplant. In this work, we show that our proposed method results in a better correlation between the lifetime of kidneys and patients, implying that the need for retransplantation decreases, as well as the rate of death cases with a functioning graft. On the other hand, we find that the actual tissue match (HLA match) degrades, compared to

the other allocation policies, which may result in higher rejection rates.

On the methodological side, our semi-offline formulation can be extended to optimize the worst-case allocation rather than the expected performance (also referred to as minimax analysis or robust optimization). In the context of kidney allocation, the idea is to account for the uncertainty associated with the model estimation (such as mortality rate and organ quality) and to maximize the allocation outcome while minimizing the possible loss due to mortality. It is expected that worst-case optimization will yield a lower expected mortality rate compared to the current KB performance. To that end, the efficiency of the allocation is expected to increase. On the equity side, however, such a policy might give unproportionately high priority to the older population and to patients with more severe health condition, leaving the less sick patients with low chance of receiving a transplant.

Acknowledgement

This work was supported in part by Health Resources and Services Administration contract 234-2005-370011C. The content is the responsibility of the authors alone and does not necessarily reflect the views or policies of the Department of Health and Human Services, nor does mention of trade names, commercial products, or organizations imply endorsement by the U.S. Government.

Appendix A

R codes

A.1. Multivariate monitoring

```
    # Based on Lowry's algorithm
    # dF = dataframe with s columns of data series
    # L = lambda value (e.g., 0.3)
    # t = alpha level / threshhold (e,g. 0.05)
    # z0 = initialization vector
    # train = fraction of train data
    # sigma = covariance matrix (otherwise estimated from train data)
MultivariateEWMA<-function(dF, L, t, z0=NULL, train, sigma=NULL){
    s = length(dF)          # number of series
    r = length(dF[,1])      # number of days
    n = r*train             # train data
    L.matrix = matrix(rep(rep(0, s),s), ncol=s)
    diag(L.matrix) = L      # lambda
    I = matrix(0,s,s)
    diag(I) = rep(1,s)
    if (is.null(z0)) {      # initialized z0
        z0 = L.matrix %*% t(dF[1,])
```

```
}


z = dF                  # EWMA values
T2 = dF[,1]             # statistic
outbreak = dF[,1]       # binary outbreak vector


# calc sigma
if (is.null(sigma)) {
    sigma = matrix(rep(rep(0, s),s), ncol=s)
    for (k in 1:s) {
        for (l in 1:s) {
            sigma[k,l] = cov(dF[1:n,k],dF[1:n,l])
        }
    }
}
sigma = sigma * L/(2-L)


for (i in 1:r){         # z(i) = Lx(i) + (1-L)*z(i-1)
    ifelse (i==1,
         (z[i,] = z0),
         (z[i,] = L.matrix %*% t(dF[i,]) + (I-L.matrix) %*% t(z[i-1,])))



    T2[i] = as.matrix(z[i,]) %*% solve(sigma) %*% t(as.matrix(z[i,]))
```

```
        ifelse (qchisq(2*t, s, lower.tail=F) < X[i],
             (outbreak[i]=1),
             (outbreak[i]=0))
    }
    remove(dF)
    remove(L.matrix)
    remove(z)
    remove(T2)
    remove(sigma)
    return(outbreak)
}

    # Variation of:
    # A Simple Multivariate Test For One-Sided Alternatives
    # Dean Follmann
    # dF = dataframe with s columns of data series
    # L = lambda vector (size = s). Example: c(1/3, 1/4, 1/5)
    # t = alpha level / threshhold
    # z0 = initialization # train = fraction of train data
    # sigma = covariance matrix (otherwise estimated from train data)
OneSideMEWMA_Follmann<-function(dF, L, t, z0=NULL, train,
sigma=NULL, with.restart=F){
    s = length(dF)          # number of series
    r = length(dF[,1])      # number of days
```

```
n = r*train             # train data
L.matrix = matrix(rep(rep(0, s),s), ncol=s)
diag(L.matrix) = L      # lambda
I = matrix(0,s,s)
diag(I) = rep(1,s)
if (is.null(z0)) {      # initialized z0
    z0 = L.matrix %*% t(dF[1,])
}

z = dF                  # EWMA values
T2 = dF[,1]             # statistic
outbreak = dF[,1]       # binary outbreak vector

# normalize: mu = 0
mu = mean(dF)
for (i in 1:s){
    dF[,i] = dF[,i] - mu[i]
}

# calc sigma
if (is.null(sigma)) {
    sigma = matrix(rep(rep(0, s),s), ncol=s)
    for (k in 1:s) {
        for (l in 1:s) {
```

```
            sigma[k,l] = cov(dF[1:n,k],dF[1:n,l])
        }
    }
}
sigma = sigma * L/(2-L)

for (i in 1:r){         # z(i) = Lx(i) + (1-L)*z(i-1)
    ifelse (i==1,
         (z[i,] = z0),
         (z[i,] = L.matrix %*% t(dF[i,]) + (I-L.matrix) %*% t(z[i-1,])))

    T2[i] = as.matrix(z[i,]) %*% solve(sigma) %*% t(as.matrix(z[i,]))
    ifelse (qchisq(2*t, s, lower.tail=F) < T2[i] && (sum(z[i,])>0),
         (outbreak[i]=1),
         (outbreak[i]=0))

    if (outbreak[i]==1 && with.restart==T){ # Restart
        z[i,]=rep(0,s)
    }
}
return(outbreak)
}

    # Based on:
```

```
    # A Simple Multivariate Test For One-Sided Alternatives
    # Dean Follmann
    # Added correcrion: cov-->correl, based on:
    # Data-Driven Rank Tests for Classes of Tail Alternatives, by Willem Albers et al
    # dF = dataframe with s columns of data series
    # t = alpha level / threshhold
    # train = fraction of train data
    # sigma = covariance matrix (otherwise estimated from train data)
OneSideHotelling_Follmann<-function(dF, t, train, sigma=NULL){
    s = length(dF)          # number of series
    r = length(dF[,1])      # number of days
    n = r*train             # train data

    # normalize: mu = 0
    mu = mean(dF)
    for (i in 1:s){
        dF[,i] = dF[,i] - mu[i]
    }

    X2 = dF[,1]         # statistic
    outbreak = dF[,1]       # binary outbreak vector

    # calc sigma
    if (is.null(sigma)) {
```

```
        sigma = matrix(rep(rep(0, s),s), ncol=s)
        for (k in 1:s) {
            for (l in 1:s) {
                sigma[k,l] = cov(dF[1:n,k],dF[1:n,l])
            }
        }
    }

    for (i in 1:r){
        X2[i] = as.matrix(dF[i,]) %*% solve(sigma) %*% t(as.matrix(dF[i,]))
        #ifelse ((qchisq(2*t, s, lower.tail=F) < X2[i]) && (sum(dF[i,])>0),
        ifelse (qchisq(2*t, s, lower.tail=F) < X2[i] && (sum(dF[i,])>0),
             (outbreak[i]=1),
             (outbreak[i]=0))
    }
    return(outbreak)
}

    # Based on:
    # Multivariate One-Sided Control Charts
    # Murat Caner Testik and George Runger
    # dF = dataframe with s columns of data series
    # t = alpha level / threshhold
    # train = fraction of train data
```

```
    # sigma = covariance matrix (otherwise estimated from train data)
OneSideHotelling_Testik<-function(dF, t, train=, sigma=NULL){
    s = length(dF)          # number of series
    r = length(dF[,1])      # number of days
    n = r*train             # train data

    # normalize: mu = 0
    mu = mean(dF)
    for (i in 1:s){
        dF[,i] = dF[,i] - mu[i]
    }

    X2 = dF[,1]             # statistic
    outbreak = dF[,1]       # binary outbreak vector

    # calc sigma
    if (is.null(sigma)) {
        sigma = matrix(rep(rep(0, s),s), ncol=s)
        for (k in 1:s) {
            for (l in 1:s) {
                sigma[k,l] = cov(dF[1:n,k],dF[1:n,l])
            }
        }
    }
```

```
mu = mean(dF)


# calc sigma^(-1/2)
e = eigen(sigma)
V = e$vectors
# CHECK: V %*% diag(e$values) %*% t(V) = inv_sigma
sigma_square = V %*% diag(sqrt(e$values)) %*% t(V)
inv_sigma_square = solve(sigma_square)


# transform standardized variables
Zt = inv_sigma_square %*% as.matrix(t(dF))
mu_z = inv_sigma_square %*% as.matrix(mu)


mu_z_t = Zt
for (i in 1:r){
    Dmat = matrix(0, s, s)
    diag(Dmat) = 1
    dvec = Zt[,i]
    Amat = sigma_square
    bvec = rep(0, s)
    mu_z_t[,i] = solve.QP(Dmat,dvec,Amat,bvec=bvec)$solution
}
```

```
    mu_t = sigma_square %*% mu_z_t


    # Statistic
    X2 = dF[,1]
    outbreak = dF[,1]
    c = Threshold(t, s, sigma, n)
    for (i in 1:r){
        X2[i] = t(as.matrix(mu_t [,i])) %*% solve(sigma) %*% as.matrix(mu_t [,i])
        ifelse ((X2[i] > c), (outbreak[i]=1), (outbreak[i]=0))
    }
    return(outbreak)
}


    # Based on:
    # Multivariate One-Sided Control Charts
    # Murat Caner Testik and George Runger
    # dF = dataframe with s columns of data series
    # L = lambda value (e.g., 0.3)
    # t = alpha level / threshhold
    # z0 = initialization
    # train = fraction of train data
    # sigma = covariance matrix (otherwise estimated from train data)
OneSideMEWMA_Testik<-function(dF, L, t, z0=NULL, train, sigma=NULL,
with.restart=F){
```

```
s = length(dF)          # number of series
r = length(dF[,1])      # number of days
n = r*train             # train data
L.matrix = matrix(rep(rep(0, s),s), ncol=s)
diag(L.matrix) = L      # lambda
I = matrix(0,s,s)
diag(I) = rep(1,s)
if (is.null(z0)) {      # initialized z0
    z0 = L.matrix %*% t(dF[1,])
}
z = dF                  # EWMA values

# calc sigma
if (is.null(sigma)) {
    sigma = matrix(rep(rep(0, s),s), ncol=s)
    for (k in 1:s) {
        for (l in 1:s) {
            sigma[k,l] = cov(dF[1:n,k],dF[1:n,l])
        }
    }
}
sigma = sigma * L/(2-L)

for (i in 1:r){         # z(i) = Lx(i) + (1-L)*z(i-1)
```

```
        ifelse (i==1,
             (z[i,] = z0),
             (z[i,] = L.matrix %*% t(dF[i,]) + (I-L.matrix) %*% t(z[i-1,])))
    }
    return(OneSideHotelling_Testik(z, t, sigma=sigma))
}

    # Calculating the threshold for Testik's method
    # t = tail (0.05)
    # s = number of series
    # sigma = covariance matrix
    # tr = length of train data
Threshold<-function(t=0.05, s, sigma, tr){
    # Create weights vector - by simulation
    w_values = mvrnorm(10000, rep(0,s), sigma)

    # calc sigma^(-1/2)
    e = eigen(sigma)
    V = e$vectors
    sigma_square = V %*% diag(sqrt(e$values)) %*% t(V)
    inv_sigma_square = solve(sigma_square)

    # transform standardized variables
    Wt = inv_sigma_square %*% as.matrix(t(w_values))
```

```
mu_z_t = Wt
for (i in 1:10000){
    Dmat = matrix(0, s, s)
    diag(Dmat) = 1
    dvec = Wt[,i]
    Amat = sigma_square
    bvec = rep(0, s)
    mu_z_t[,i] = solve.QP(Dmat,dvec,Amat,bvec=bvec)$solution
}

mu_t = sigma_square %*% mu_z_t

mu_t = (abs(mu_t) > 0.0001)
w = rep(0,s)
for (j in 1:10000) {
    count = sum(mu_t[,j])
    if (count>0) {
        w[count] = w[count]+1
    }
}
w = w/10000

# p(X2 > c^2)
```

```
    c = 0
    is.threshold = F
    change.step = F
    step = 5

    while (is.threshold == F){
        p = 0
        for (j in 1:s) {p = p + w[j]*pchisq(c, j, lower.tail=F)}
        if (abs(p - t) < 0.0001) {is.threshold = T}
        if (p < t) {change.step = T}
        if (change.step == T) {step = step/2}
        if (p < t) {c = c - step}
        if (p > t) {c = c + step}
    }
    return(c)
}
```

A.2. Multivariate Poisson

```
    # Generate a p-dimensional Poisson
    # p         = the dimension of the distribution
    # samples   = the number of observations
    # R         = correlation matrix p X p
```

```
    # lambda    = rate vector p X 1
GenerateMultivariatePoisson<-function(p, samples, R, lambda){
        normal_mu=rep(0, p)
        normal = mvrnorm(samples, normal_mu, R)
        unif=pnorm(normal)
        pois=t(qpois(t(unif), lambda))
        return(pois)
}

    # Correct initial correlation between a
    # certain pair of series
    # lambda1   = rate of first series
    # lambda2   = rate of second series
    # r         = desired correlation
CorrectInitialCorrel<-function(lambda1, lambda2, r){
    samples=500
    u = runif(samples, 0, 1)
    lambda=c(lambda1,lambda2)
    maxcor=cor(qpois(u, lambda1), qpois(u, lambda2))
    mincor=cor(qpois(u, lambda1), qpois(1-u, lambda2))
    a=-maxcor*mincor/(maxcor+mincor)
    b=log((maxcor+a)/a, exp(1))
    c=-a
    corrected=log((r+a)/a, exp(1))/b
```

```
    corrected=ifelse ((corrected>1 | corrected<(-1)),
                      NA, corrected)
    return(corrected)
}
```

Appendix 10

Glossary of Terms

In this work we use a variety of methods to achieve an optimal, data-driven performance in the context of the two healthcare domains. We enlist the methods used broken down by building blocks and by discipline in Figures 10.1 - 10.3.

This chapter is organized as follows. In Section 10.1, we introduce the data mining and statistical techniques used, presented for operations research readers. In particular, we describe the methods for data generation, time series preprocessing and statistical monitoring that were used in Part I of this dissertation, and survival models, a statistical test of median equality, and regression trees, which were used in Part II of the dissertation. In Section Section 10.2 we give a brief introduction to OR optimization, for the non-OR (statisticians and data mining) readers.

10.1. Data Mining and Statistical Tools

10.1.1 Data Mimicking

Data mimicking is a method for generating stochastic replicas of a particular dataset. The approach is based on identifying the most important parameters of a given (authentic) dataset, estimating those parameters from the data, and using them to stochastically generate new data (Lotze et al., 2007).

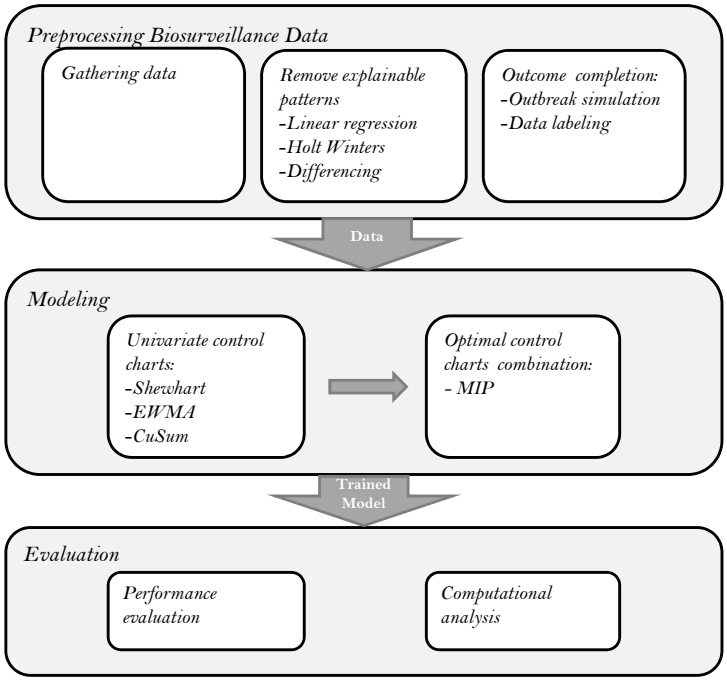

Figure 10.1: Biosurveillance schematic: univariate monitoring.

In this work, we use data mimicking to generate replicates of authentic syndromic datasets. In particular, we simulate multivariate time-series data that includes prominent patterns of syndromic data, such as seasonal and day-of-week (DOW) patterns.

The model operates in two steps. First, an initial multivariate dataset is generated from a multivariate normal distribution that includes autocorrelation and cross-correlation. Specifically, the vector of values on k series at day t is represented as $\mathbf{X}_t = \begin{pmatrix} X_{1,t} \\ \vdots \\ X_{k,t} \end{pmatrix}$, with mean $\boldsymbol{\mu} = \begin{pmatrix} \mu_{1,t} \\ \vdots \\ \mu_{k,t} \end{pmatrix}$ and covariance matrix Σ. The bivariate dis-

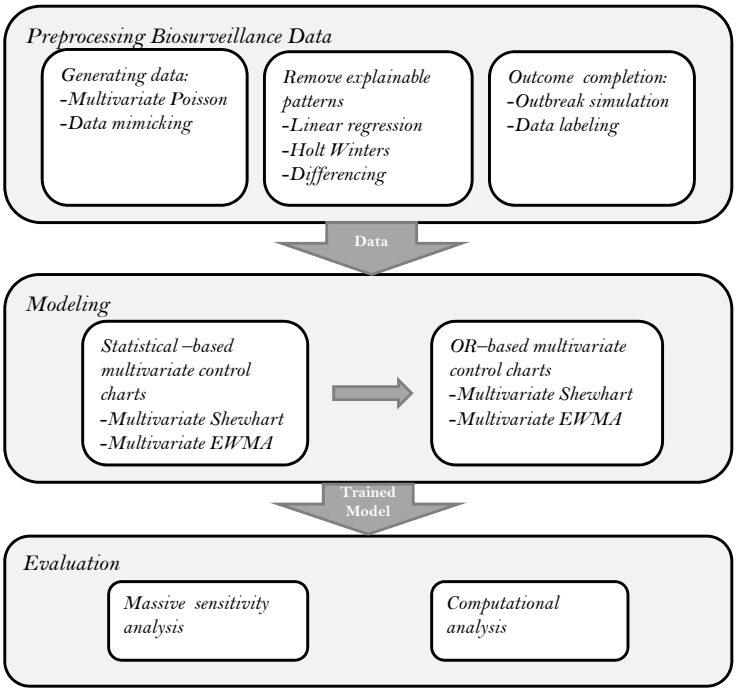

Figure 10.2: Biosurveillance schematic: multivariate monitoring.

tribution of $\mathbf{X}_{t+1}$ and $\mathbf{X}_t$ is

$$\begin{pmatrix} \mathbf{X}_t \\ \mathbf{X}_{t+1} \end{pmatrix} = \begin{pmatrix} X_{1,t} \\ \vdots \\ X_{k,t} \\ X_{1,t+1} \\ \vdots \\ X_{k,t+1} \end{pmatrix} \sim N\left(\begin{pmatrix} \boldsymbol{\mu} \\ \boldsymbol{\mu} \end{pmatrix} \cdot \begin{pmatrix} \Sigma & C \\ C & \Sigma \end{pmatrix} \right),$$

where C is a diagonal matrix with elements c_i $(i = (1, ...k))$ on the diagonal, where $c_i = cov(X_{i,t}, X_{i,t+1})$ is the lag-1 autocovariance of series i.

In the second step, estimated seasonal and DOW patterns are added to the data. The series are then rounded to integers and bounded to be nonnegative, in order to yield

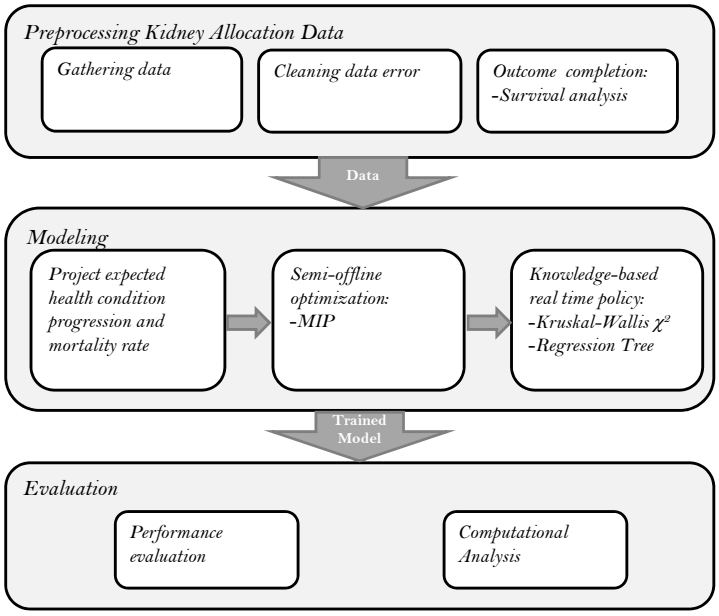

Figure 10.3: Kidney allocation schematic.

valid data.

10.1.2 Linear Regression, Exponential Smoothing, and Differencing

There are a variety of methods for removing explainable patterns from a time series. Methods generally are either model-based or data-driven. Model-based methods remove a pattern by directly modeling the pattern via some specification. An example is a linear regression model with day-of-week indicators. Data-driven methods either suppress certain patterns (e.g., differencing at a certain lag) or 'learn' patterns from the data (e.g., exponential smoothing). In the following we describe three methods that have been shown to be effective in removing the types of explainable effects that are often exhibited in pre-diagnostic daily count series (day-of-week, holidays, seasonality, and autocorrelation). For a more detailed discussion of preprocessing methods see Lotze et al. (2008) and Shmueli and Burkom (2010). The methods we describe produce next-day forecasts. The forecasts

are then subtracted from the actual counts to produce residuals. In particular, we denote the observed data at time t by y_t and the forecasts by $\hat{y}_t$. The residuals R_t are the differences between the observed and forecasted values: $R_t = y_t - \hat{y}_t$.

Linear Regression models are a popular method for capturing recurring patterns such as day-of-week, seasonality, and trends (Rice, 1995; Brillman et al., 2005). The classic assumption is that these patterns do not change over time, and therefore the entire data are used for model estimation. In this work we use log-linear regression of log(daily counts) with the following covariates: daily dummy variables (**DOW**=$Monday$, $Tuesday$, ... $Sunday$) to account for the DOW effect, a holiday indicator ($Holiday$), an index variable ($index$) to capture a linear trend, and daily average temperatures ($Tavg$) and monthly dummy variables (**MON**=Jan, Feb, ... Dec) to capture seasonality. The estimated model is given by:

$$\log(\hat{y}_t) = \hat{\alpha} + \hat{\beta}' \times [\mathbf{DOW}, Holiday, index, Tavg, \mathbf{MON}]$$

Differencing is the operation of subtracting a previous count from a current one. The order of differencing gives the vicinity between the two counts (Brockwell and Davis, 1991). We consider 7-day differencing, as suggested by Muscatello (2004), where forecasts are obtained by using the values from the previous week: $\hat{y}_t = y_{t-7}$.

The main advantage of differencing is that it is computationally efficient and very effective at removing both weekly and monthly patterns. The main drawback of 7-day differencing is that it generates autocorrelated residuals with 7-day autocorrelation.

Exponential Smoothing is a popular scheme for producing a smoothed time series and/

or generate forecasts. Holt-Winter's exponential smoothing is a form of smoothing in which a time series is assumed to consist of few components: a level L_t, a trend T_t, seasonality S_t, and noise (Chatfield, 1978). The k-step ahead forecast is given by

$$\hat{y}_{t+k} = (L_t + kT_t)S_t + k - M,$$

where M is the number of seasons in a cycle (e.g., for a yearly periodicity M = 365). The three components are updated as follows:

$$L_t = \alpha Y_t / S_{t-m} + (1 - \alpha)(Lt - 1 + T_{t-1})$$

$$T_t = \beta(Lt - Lt - 1) + (1 - \beta)T_{t-1}$$

$$S_t = \gamma Y_t / L_t + (1 - \gamma)(S_{t-M}),$$

where α, β, and γ are smoothing parameters that take values in (0, 1). Burkom et al. (2007) show that exponential smoothing is useful for preprocessing pre-diagnostic data, and suggest using $M = 7$ seasons, and smoothing parameter values $\alpha \in [0.1, 0.4]$ (depending on the properties of the series), $\beta = 0$, and $\gamma = 0.15$.

10.1.3 Control Charts

A control chart is a popular statistical tool for monitoring a process with the goal of distinguishing the process common variation from special-caused variation (referred to as 'anomalies'). Control charts were originated by Walter A. Shewhart while working at Bell Labs in 1924, for the manufacturing environment. Taking a statistical approach,

Shewhart showed that measurements in manufacturing data can be approximated by a Normal distribution. Consequently, the underlying assumption of Shewhart's chart and other related charts, is that measurements follow a Normal (or other known) distribution and that the observations are independent.

We use control charts in the first part of the dissertation to monitor biosurveillance processes. In particular, we monitor aggregated daily counts of individual health care seeking behavior (such as daily arrivals to emergency departments or medication sales), for the purpose of early detection of epidemic outbreaks, manifested in the data as shifts from expected baseline behavior.

Three types of control charts, which are widely-used in modern biosurveillance systems, are discussed and used thoroughly in our work: The Shewhart chart, Exponential Weighted Moving Average (EWMA) chart, and Cumulative Sum (CuSum) chart. Each is best at detecting a certain shape of outbreak. We briefly describe each next.

The Shewhart chart is the most basic control chart. A sample statistic at each time point (such as a mean, proportion, or count) is plotted and compared against upper and/or lower control limits (UCL and LCL), and if the limit(s) are exceeded, an alarm is raised. The control limits are typically set as a multiple of standard deviations of the statistic from the target value (Montgomery, 2007). It is most efficient at detecting medium to large spike-type outbreak signatures.

The Exponentially Weighted Moving Average (EWMA) chart plots and monitors a weighted average of the sample statistics with exponentially decaying weights (NIST/SEMATECH, 2006). It is most efficient at detecting exponential changes

in the target value and is widely used for detecting small sustainable changes in the target value.

Cumulative-Sum (CuSum) control charts plot and monitor the cumulative sum of the deviations from the target values. The CuSum chart is known to be efficient in detecting small shifts in the target value (Montgomery, 2007) and step-function type changes (Box et al., 2009).

Table 10.1 summarizes for each of the three charts its monitoring statistic (denoted $Shewhart_t$, $EWMA_t$ and $CuSum_t$), the upper control limit (UCL) for alerting, the parameter value that yields a theoretical 5% false alert rate, and a binary output indicator that indicates whether an alert was triggered on day t (1) or not (0). Y_t denotes the raw daily count on day t. We consider one-sided control charts where an alert is triggered only when there is indication of an increase in mean (i.e., when the monitoring statistic exceeds the UCL). This is because *increases* in health care seeking behavior are of interest.

We also consider multivariate control charts in Chapter 4, where multiple series are monitored simultaneously. Multivariate control charts take advantage of the correlation structure between individual series, thereby having a higher potential of detecting small signals that are dispersed across series.

10.1.4 Survival Analysis

Survival analysis is a field in statistics that examines and models the time it takes for events to occur. In most of the applications of survival analysis the event is death, from which the term 'survival' derives. A broadly applicable and the most widely used

Table 10.1: Features of three main control charts

	Shewhart	**EWMA**	**CuSum**
Monitored Statistic	$S_t = Y_t$	$E_t = \lambda Y_t + (1-\lambda)E_{t-1}$	$C_t = \max(0, C_{t-1} + Y_t - \sigma/2)$
UCL	$UCL = \mu + k\sigma$	$UCL = E_0 + ks$, $s^2 = \lambda/(2-\lambda)\sigma^2$	$UCL = \mu + h\sigma$
Theoretical 5% *threshold*	$k = 1.5$	$k = 1.5$	$h = 2.5$
Output	$1_{S_t>UCL}$	$1_{E_t>UCL}$	$1_{C_t>UCL}$

method of survival analysis is the Cox proportional hazards model (Cox, 1972; Cox and Oakes, 1984). The model estimates the distribution of survival times conditional on a set of one or more predictors. This is done by modeling the survival function $S(t)$, which is the probability of survival as a function of time. In the Cox model, the survival function $S(t)$ (the probability of survival as a function of time) for an individual with set of predictors $\mathbf{z}_i$ is given by

$$S(t|\mathbf{z}_i) = S_0(t)^{exp\{\boldsymbol{\beta}'\mathbf{z}_i\}}, \tag{10.1}$$

with $S_0(t)$ being the baseline survival probability, which can be estimated by the Kaplan-Meier estimate (Kaplan and Meier, 1958), and $\boldsymbol{\beta}$ is a vector of predictor coefficients to be estimated from the Cox model.

The model specifics are as follows. Let T represent the survival time, which is a random variable with cumulative distribution function $P(t) = p(T \leq t)$ and probability density function $p(t) = dP(t)/dt$. Let $S(t)$ be the complement of the distribution function, that is $S(t) = p(T > t)$. Finally, $h(t)$ is the *hazard function*, which assesses the death probability at time t, conditional on survival to that time:

$$h(t) = \lim_{\Delta t \to 0} \frac{p\left[(t \leq T < t + \Delta t) | T \geq t\right]}{\Delta t}. \tag{10.2}$$

The proportional hazards model fits survival data with covariates $\mathbf{z}$ to a hazard function of the form:

$$h(t|z) = h_0(t) exp\{\boldsymbol{\beta}^{'}\mathbf{z}\}, \tag{10.3}$$

where $h_0(t)$ is the non-parametric baseline hazard, and $\boldsymbol{\beta}$ is estimated by a partial likelihood function (a maximum likelihood function, conditional on the death event, see Cox and Oakes (1984)).

An alternative to the Cox proportional hazard model is the Accelerated Failure Time model (AFT). AFT is a fully parametric model with covariates that take the form:

$$S(t|z) = S_0(t \exp\{\boldsymbol{\gamma}^{'}\mathbf{z}\}), \tag{10.4}$$

where $\boldsymbol{\gamma}$ is a vector of parameters and $\mathbf{z}$ is the vector of covariates. This model essentially puts individuals with different covariates on different time scales. The model assumes a linear relationship between the log of failure time ($\log(T)$) and the covariates and parameters ($\boldsymbol{\gamma}^{'}\mathbf{z}$), with an error term ($\varepsilon$) that takes a particular distribution:

$$\log(T) = \alpha + \boldsymbol{\gamma}^{'}\mathbf{z} + \varepsilon. \tag{10.5}$$

10.1.5 The Kruskal-Wallis χ^2 Test

The Kruskal-Wallis (KW) χ^2 test is a non-parametric method for testing the equality of population medians with unknown distributions (Kruskal and Wallis, 1952). Since it is a non-parametric method, the KruskalWallis test does not assume a normal population (unlike the analogous one-way analysis of variance). However, the test assumes an identically-shaped and -scaled distribution for each population examined.

The test operates as follows. First, in the computation of the KW statistic, each observation is replaced by its rank in an ordered combination of all the populations. Then, the sum of the ranks for each of the populations is computed, and compared using a one-way ANOVA test on the ranks.

In this dissertation, we use the Kruskal-Wallis χ^2 test in the kidney allocation project to compare the health condition of candidates in the OPTN waiting list to that of the organ recipients, according to our allocation policy.

10.1.6 Regression Trees

A regression tree is a recursive partitioning method that splits a p-dimensional space (denoted: X variables) using a continuous response variable y (Breiman et al., 1984). The X variables are assumed to be either continuous or categorical. In each of the recursion steps, one variable $x_i \in X$ is chosen to split the tree into two subtrees: numeric variables are divided into $x_i < \alpha$ and $x_i > \alpha$; the levels of categorical factors are divided into two non-empty groups. The split is chosen in a way that maximizes the reduction in impurity of the subtrees. Splitting continues until the terminal nodes are too small or too few to

be split.

There are several methods to measure impurity of a subtree tr. One of the most popular methods (which we also use in this work) is the entropy measure:

$$entropy(tr) = -\sum_{k=1}^{m} p_k \log_2(p_k) \tag{10.6}$$

where p_k is the proportion of samples in subtree k and m is the total number of subtrees.

In the kidney allocation project, we use exploratory regression trees on the semi-offline allocation, with the allocated organ being the response variable (y), and the recipient's health record being the independent variables (X). The regression trees enables us to explore the nature of the optimal allocation, in terms of classifying organ types into classes of candidates, based on candidates' health conditions.

10.2. Operations Research Methods

10.2.1 Offline and Semi-Offline Optimization

An offline optimizer is an algorithm that seeks the optimal solution based on the entire input (i.e., past and future events) available from the start. Similarly, a semi-offline optimizer is an algorithm that seeks the optimal solution based on a probabilistic input, in which missing information is replaced by a set of probabilistic scenarios. A solution obtained by an offline/ semi-offline algorithm is hard to attain in real-time, where future events are unknown.

In the biosurveillance project, we use an offline optimizer to study the optimal offline linear combination of multiple preprocessing techniques and monitoring algorithms, and its properties. We then apply the optimal offline combination in real-time and show that the combined method outperforms any of the single methods in terms of true and false alarms.

In the kidney allocation project, we use a semi-offline optimizer to find the optimal kidney allocation, based on the entire information of patients and donors arrivals, their properties upon arrival, and their probabilistic future health condition and death probability. We then study the properties of the allocation in terms of which kidney is allocated to what patient, expected waiting time, death rate, and other determinants in this complex multi-objective problem. We use these properties to derive allocation rules to be used in real-time deployment.

One of the main drawbacks of this approach is that the offline algorithm is overfitted to a given input, thereby potentially producing inaccurate predictors, and might not be robust to other inputs. To overcome this problem, we train the algorithm on a large set of training inputs, generated in the preprocessing step, and learn the common properties of the different allocations.

10.2.2 Mixed Integer Programming

Linear programming (LP) is a mathematical method for obtaining an optimal outcome represented as a linear function, subject to linear constraints. When some of the variables are restricted to take only integer values, the problem becomes a Mixed integer programming (MIP) problem.

An MIP problem can be expressed as:

$$\max \mathbf{c}^{'}\mathbf{x}_L + \mathbf{c}^{'}\mathbf{x}_I$$

s.t.

$$A(\mathbf{x}_L + \mathbf{x}_I) \leq \mathbf{b},$$

$$\mathbf{x}_I \text{ Integer.}$$

where $\mathbf{x}_L$ and $\mathbf{x}_I$ represents the vector of linear and integer (respectively) variables to be determined, $\mathbf{c}$, $\mathbf{b}$ and A are known coefficients. The expression to be maximized is called the objective function. The equations are the linear and integral constraints.

We use MIP to formulate and solve the offline and semi-offline optimization problems described above. The MIP formulation allows us to obtain an optimal solution to these problems and can be solved by any standard optimization software, such as CPLEX.

Bibliography

Ahn, J.H., J.C. Hornberger. 1996. Involving patients in the cadaveric kidney transplant allocation process: a decision-theoretic perspective. *Management Science* **42**(5) 629–641.

Alagoz, O., L.M. Maillart, A.J. Schaefer, M.S. Roberts. 2007. Determining the acceptance of cadaveric livers using an implicit model of the waiting list. *Operations Research* **55**(1) 24–36.

Alagoz, O., A.J. Schaefer, M.S. Roberts. 2008. Optimizing organ allocation and acceptance. *Handbook of Optimization in Medicine* 1–24.

Albers, W., W.C.M. Kallenberg, F. Martini. 2001. Data-driven rank tests for classes of tail alternatives. *Journal of the American Statistical Association* **96**(454) 685–696.

Aradhye, H.B., B.R. Bakshi, R.A. Strauss, J.F. Davis. 2003. Multiscale statistical process control using wavelets - theoretical analysis and properties. *AIChE Journal* **49**(4) 939–958.

Avramidis, A.N., N. Channouf, P. L'Ecuyer. 2009. Efficient correlation matching for fitting discrete multivariate distributions with arbitrary marginals and normal-copula dependence. *INFORMS Journal on Computing* **21**(1) 88–106.

Bardossy, M.G., I. Yahav. 2010. Stochastic dynamic allocation of kidneys based on historical data logs Working Paper.

Baylis, P. 1999. Better health care with data mining. *white paper, SPSS, Woking* .

Box, G.E.P., A. Luceno, M. del Carmen Paniagua-Quiñones. 2009. *Statistical Control by Monitoring and Adjustment*. Wiley.

Bradburn, M.J., T.G. Clark, S.B. Love, D.G. Altman. 2003. Survival Analysis Part II: Multivariate data analysis–an introduction to concepts and methods. *British journal of cancer* **89**(3) 431.

Brandeau, M.L., F. Sainfort, W.P. Pierskalla. 2004. *Operations research and health care: a handbook of methods and applications*. Kluwer Academic.

Breiman, L., J.H. Friedman, R. Olshen, C.J. Stone, W. Hoeffding, R.J. Serfling, O. Hall, P. Buhlmann. 1984. Classification and regression trees. *Ann. Math. Statist.* **19** 293–325.

Brillman, J.C., T. Burr, D. Forslund, E. Joyce, R. Picard, E. Umland. 2005. Modeling emergency department visit patterns for infectious disease complaints: results and application to disease surveillance. *BMC medical informatics and decision making* **5**(1) 4.

Brockwell, P.J., R.A. Davis. 1991. *Time Series: Theory and Methods*. 1991st ed. Springer-Verlag, New York.

Buckeridge, D.L., H. Burkom, M. Campbell, W.R. Hogan, A.W. Moore. 2005. Algorithms for rapid outbreak detection: a research synthesis. *Journal of biomedical informatics* **38**(2) 99–113.

Burkom, H.S. 2003. Development, adaptation, and assessment of alerting algorithms for biosurveillance. *Johns Hopkins APL Technical Digest* **24**(4) 335–342.

Burkom, H.S., S.P. Murphy, G. Shmueli. 2007. Automated time series forecasting for biosurveillance. *Statistics in medicine* **26**(22) 4202–4218.

Burton, R.M., D.C. Dellinger, W.W. Damon, E.A. Pfeiffer. 1978. A role for operational research in health care planning and management teams. *Journal of the Operational Research Society* 633–641.

Chatfield, C. 1978. The holt-winters forecasting procedure. *Applied Statistics* **27**(3) 264–279.

Chen, H. 2001. Initialization for NORTA: Generation of random vectors with specified marginals and correlations. *INFORMS Journal on Computing* **13**(4) 312.

Cherikh, W. 2006. Variability of PRA levels and reporting of unacceptable antigens among transplant centers. *Report to the Histocompatibility Committee on January* **26**.

Coskun, N., R. Erol. 2010. An Optimization Model for Locating and Sizing Emergency Medical Service Stations. *Journal of Medical Systems* **34**(1) 43–49.

Cox, D.R. 1972. Regression models and life tables (with discussion). *Journal of the Royal Statistical Society. Series B (Methodological)* **34**(2) 187–220.

Cox, D.R., D. Oakes. 1984. *Analysis of survival data*. Chapman & Hall/CRC.

Crosier, R.B. 1988. Multivariate generalizations of cumulative sum quality-control schemes. *Technometrics* **30**(3) 291–303.

Daskin, M., L. Dean. 2004. Location of health care facilities. *Operations Research and Health Care* 43–76.

David, I., U. Yechiali. 1985. A time-dependent stopping problem with application to live organ transplants. *Operations Research* **33**(3) 491–504.

David, I., U. Yechiali. 1995. One-attribute sequential assignment match processes in discrete time. *Operations Research* **43**(5) 879–884.

Demirci, M., A. Schaefer, H. Romeijn, M. Roberts. 2010. An exact method for balancing efficiency and equity in the liver allocation hierarchy. Tech. rep., Technical report, Department of Industrial, Universit of Pittsburgh, 2010.

Derman, C., G.J. Lieberman, S.M. Ross. 1972. A sequential stochastic assignment problem. *Management Science* **18**(7) 349–355.

Edmonds, J. 1965. Paths, trees, and flowers. *Canadian Journal of Mathematics* **17**(3) 449–467.

Eggers, P.W. 1995. Racial differences in access to kidney transplantation. *Health care financing review* **17**(2) 89–103.

Follmann, D. 1996. A Simple Multivariate Test for One-Sided Alternatives. *Journal of the American Statistical Association* **91**(434).

Freeman, Richard B. 2008. The importance of donor risk factors. Online.

Fricker, R.D. 2007. Directionally sensitive multivariate statistical process control procedures with application to syndromic surveillance. *Advances in Disease Surveillance* **3**(0).

Fricker, R.D., M.C. Knitt, C.X. Hu. 2008. Comparing directionally sensitive MCUSUM and MEWMA procedures with application to biosurveillance. *Quality Engineering* **20**(4) 478–494.

Glowacka, KJ, RM Henry, JH May. 2009. A hybrid data mining/simulation approach for modelling outpatient no-shows in clinic scheduling. *Journal of the Operational Research Society* **60**(8) 1056–1068.

Green, L. 2004. Capacity planning and management in hospitals. *Operations Research and Health Care* 15–41.

Hines, R.L., K.E. Marschall. 2008. *Stoelting's Anesthesia and Co-existing Disease*. Churchill Livingstone-Elsevier.

Hotelling, H. 1947. Multivariate quality control, illustrated by the air testing of sample bombsights. *Techniques of statistical analysis* **11** 113–184.

Howard, D.H. 2002. Why do transplant surgeons turn down organs?:: A model of the accept/reject decision. *Journal of Health Economics* **21**(6) 957–969.

Jensen, W.A., L.A. Jones-Farmer, C.W. Champ, W.H. Woodall. 2006. Effects of parameter estimation on control chart properties: a literature review. *Journal of Quality Technology* **38**(4) 349–364.

Joner Jr, M.D., W.H. Woodall, M.R. Reynolds Jr, R.D. Fricker Jr. 2008. A one-sided MEWMA chart for health surveillance. *Quality and Reliability Engineering International* **24**(5) 503–518.

Jones, L.A., C.W. Champ, S.E. Rigdon. 2001. The performance of exponentially weighted moving average charts with estimated parameters. *Technometrics* **43**(2) 156–167.

Kaplan, EL, P. Meier. 1958. Nonparametric estimation from incomplete observations. *Journal of the American statistical association* **53**(282) 457–481.

Karlis, D. 2003. An EM algorithm for multivariate Poisson distribution and related models. *Journal of Applied Statistics* **30**(1) 63–77.

Kleinman, K., R. Lazarus, R. Platt. 2004. A generalized linear mixed models approach for detecting incident clusters of disease in small areas, with an application to biological terrorism. *American Journal of Epidemiology* **159**(3) 217–224.

Kong, N. 2006. Optimizing the efficiency of the United States organ allocation system through region reorganization. Ph.D. thesis, University of Pittsburgh.

Kong, N., A.J. Schaefer, B. Hunsaker, M.S. Roberts. 2008. Maximizing the efficiency of the US liver allocation system through region design. *Management Science (submitted)* .

Krummenauer, F. 1998a. Efficient simulation of multivariate binomial and poisson distributions. *Biometrical Journal* **40**(7).

Krummenauer, F. 1998b. Limit theorems for multivariate discrete distributions. *Metrika* **47**(1) 47–69.

Kruskal, W.H., W.A. Wallis. 1952. Use of ranks in one-criterion variance analysis. *Journal of the American Statistical Association* **47**(260) 583–621.

Lambert, P., D. Collett, A. Kimber, R. Johnson. 2004. Parametric accelerated failure time models with random effects and an application to kidney transplant survival. *Statistics in medicine* **23**(20) 3177–3192.

Leffell, M.S., W.S. Cherikh, G. Land, A.A. Zachary. 2007. Improved definition of human leukocyte antigen frequencies among minorities and applicability to estimates of transplant compatibility. *Transplantation* **83**(7) 964.

Liu, S.S., J. Chen. 2009. Using data mining to segment healthcare markets from patients' preference perspectives. *International Journal of Health Care Quality Assurance* **22**(2) 117–134.

Lotze, T., S.P. Murphy, G. Shmueli. 2008. Preparing biosurveillance data for classic monitoring. *Advances in Disease Surveillance* **6** 55.

Lotze, T., G. Shmueli, I. Yahav. 2007. Simulating Multivariate Syndromic Time Series and Outbreak Signatures. *SSRN eLibrary* .

Lowry, C.A., D.C. Montgomery. 1995. A review of multivariate control charts. *IIE Transactions* **27**(6) 800–810.

Lowry, C.A., W.H. Woodall, C.W. Champ, S.E. Rigdon. 1992. A Multivariate Exponentially Weighted Moving Average Control Chart. *Technometrics* **34**(1) 46–53.

Mardia, K.V. 1970. *Families of Bivariate Distributions*. Griffin, London.

Minhajuddin, A.T.M., I.R. Harris, W.R. Schucany. 2004. Simulating multivariate distributions with specific correlations. *Journal of Statistical Computation and Simulation* **74**(8) 599–607.

Montgomery, D.C. 2007. *Introduction to statistical quality control*. Wiley-India.

Montgomery, D.C., P.J. Klatt. 1972. Economic Design of T^2 Control Charts to Maintain Current Control of a Process. *Management Science* **19**(1) 76–89.

Muscatello, D. 2004. An adjusted cumulative sum for count data with day-of-week effects: application to influenza-like illness. *Presentation at Syndromic surveillance conference*.

Nedumaran, G., J.J. Pignatiello. 1999. On constructing T2 control charts for on-line process monitoring. *IIE transactions* **31**(6) 529–536.

Nelsen, R.B. 2006. *An introduction to copulas*. Springer Verlag.

NIST/SEMATECH. 2006. e-handbook of statistical methods, http://www.itl.nist.gov/div898/handbook/.

Nüesch, P.E. 1966. On the problem of testing location in multivariate populations for restricted alternatives. *The Annals of Mathematical Statistics* **37**(1) 113–119.

OPTN/UNOS. 2008. Kidney Allocation Concepts, Request for Information. *The Kidney Transplantation Comittee* .

Payton, F.C. 2003. Data mining in health care applications. *Data mining: opportunities and challenges* 350.

Pignatiello, J.J., G.C. Runger. 1990. Comparisons of multivariate CUSUM charts. *Journal of Quality Technology* **22**(3) 173–186.

Porter, T., B. Green. 2009. Identifying Diabetic Patients: A Data Mining Approach. *AMCIS 2009 Proceedings* 500.

Price, C., T. Babineau, B. Golden, B. Griffith, E. Wasil. 2008. Maximizing cardiac surgery throughput at a major hospital. *Proceedings of the 2008 Spring simulation multiconference*. The Society for Computer Simulation, International, 513–516.

Reis, B.Y., K.D. Mandl. 2003. Time series modeling for syndromic surveillance. *BMC Medical Informatics and Decision Making* **3**(1) 2.

Rice, J.A. 1995. *Mathematical statistics and data analysis*. 2nd ed. Thomson Brooks/Cole.

Righter, R. 1989. A resource allocation problem in a random environment. *Operations Research* **37**(2) 329–338.

Rodgers, J.L., W.A. Nicewander. 1988. Thirteen ways to look at the correlation coefficient. *American Statistician* 59–66.

Romeijn, H.E., S.A. Zenios. 2008. Introduction to the Special Issue on Operations Research in Health Care. *Operations Research* **56**(6) 1333.

Rosow, E., J. Adam, K. Coulombe, K. Race, R. Anderson. 2003. Virtual instrumentation and real-time executive dashboards: Solutions for health care systems. *Nursing Administration Quarterly* **27**(1) 58.

Royston, G. 2009. One hundred years of Operational Research in HealthUK 1948–2048&star. *Journal of the Operational Research Society* S169–S179.

Sainfort, F., J. Blake, D. Gupta, R.L. Rardin. 2005. Operations research for health care delivery systems. *WTEC Panel Report* .

Shechter, S.M., C.L. Bryce, O. Alagoz, J.E. Kreke, J.E. Stahl, A.J. Schaefer, D.C. Angus, M.S. Roberts. 2005. A clinically based discrete-event simulation of end-stage liver disease and the organ allocation process. *Medical Decision Making* **25**(2) 199.

Shin, K., R. Pasupathy. 2007. A method for fast generation of bivariate Poisson random vectors. *Proceedings of the 39th conference on Winter simulation: 40 years! The best is yet to come*. IEEE Press Piscataway, NJ, USA, 472–479.

Shmueli, G., H. S. Burkom. 2010. Statistical Challenges Facing Early Outbreak Detection in Biosurveillance. *Technometrics* **52**(1) 39–51.

Shmueli, G, S. E. Fienberg. 2006. Current and potential statistical meth- ods for monitoring multiple data streams for bio-surveillance. *Statistical Methods in Counter-Terrorism, Eds: A Wilson and D Olwell, Springer* (2).

Stoumbos, Z.G., J.H. Sullivan. 2002. Robustness to non-normality of the multivariate EWMA control chart. *Journal of Quality Technology* **34**(3) 260–276.

Su, X., S.A. Zenios. 2004. Patient Choice in Kidney Allocation: The Role of the Queueing Discipline. *Manufacturing & Service Operations Management* **6**(4) 280–301.

Su, X., S.A. Zenios. 2005. Patient choice in kidney allocation: A sequential stochastic assignment model. *Operations Research* **53**(3) 443–455.

Su, X., S.A. Zenios. 2006. Recipient Choice Can Address the Efficiency-Equity Trade-off in Kidney Transplantation: A Mechanism Design Model. *Management Science* **52**(11) 1647–1660.

Taranto, S.E., A.M. Harper, E.B. Edwards, J.D. Rosendale, M.A. McBride, O.P. Daily, D. Murphy, B. Poos, J. Reust, B. Schmeiser. 2000. Developing a national allocation model for cadaveric kidneys. *Proceedings of the 32nd conference on Winter simulation*. Society for Computer Simulation International, 1971–1977.

Testik, M.C., G.C. Runger. 2006. Multivariate one-sided control charts. *IIE Transactions* **38**(8) 635–645.

Thompson, D., L. Waisanen, R. Wolfe, R.M. Merion, K. McCullough, A. Rodgers. 2004. Simulating the allocation of organs for transplantation. *Health Care Management Science* **7**(4) 331–338.

Thompson, S., M. Nunez, R. Garfinkel, M.D. Dean. 2009. OR Practice—Efficient Short-Term Allocation and Reallocation of Patients to Floors of a Hospital During Demand Surges. *Operations research* **57**(2) 261–273.

van den Hout, W.B., J. Smits, M.C. Deng, M. Hummel, F. Schoendube, H.H. Scheld, G.G. Persijn, G. Laufer, et al. 2003. The heart-allocation simulation model: a tool for comparison of transplantation allocation policies 1. *Transplantation* **76**(10) 1492.

Votruba, M.E. 2001. Efficiency-equity tradeoffs in the allocation of cadaveric kidneys. Tech. rep., Working Paper. Princeton University.

Whitt, W. 1976. Bivariate distributions with given marginals. *The Annals of Statistics* **4**(6) 1280–1289.

Wolfe, R. 2007. Predicting Life Years From Transplant (LYFT) Focus Upon Methods. *Scientific Registry for Transplant Recipients* .

Wolfe, R.A., McCullough K.P., Schaubel D.E., Kalbfleisch J.D., Murray S., Stegall M.D. Leichtman A.B. 2008. Calculating life years from transplant (LYFT): Methods for kidney and kidney-pancreas candidates. *Am. J. Transplant.* **8** 997–1011.

Yahav, I., T. Lotze, G. Shmueli. 2010. Algorithm Combination for Improved Detection in Biosurveillance. *Infectious Disease Informatics and Biosurveillance: Research, Systems, and Case Studies, Springer (In Press)* .

Yahav, I., G. Shmueli. 2006. Algorithm Combination for Improved Performance in Biosurveillance Systems. *Proceeding of the Second NSF Workshop, BioSurveillance* **4506** 91–102.

Zenios, S.A. 2004. Models for kidney allocation. *Operations Research and Health Care: A Handbook of Methods and Applications* 537–554.

Zenios, S.A., G.M. Chertow, L.M. Wein. 2000. Dynamic Allocation of Kidneys to Candidates on the Transplant Waiting List. *Operations Research* **48**(4) 549–569.