Scenarios Discovery: Robust Transportation Policy Analysis in Singapore Using Microscopic Traffic Simulator

by

Xiang Song

Bachelor of Engineering, Civil Engineering,

Zhejiang University (2011)

Submitted to the Department of Civil and Environmental Engineering

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN TRANSPORTATION

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2013

Signature of Author .....

Department of Civil and Environmental Engineering

May 28, 2013

Certified by .....

Moshe E. Ben-Akiva

Edmund K. Turner Professor of Civil and Environmental Engineering

Thesis Supervisor

Certified by .....

Tomer Toledo

Associate Professor of Faculty of Civil and Environmental Engineering

Technion - Israel Institute of Technology

Thesis Supervisor

Accepted by .....

Heidi Nepf

Chair, Departmental Committee for Graduate Studies

Scenario Discovery: Robust Transportation Policy Analysis in Singapore Using Microscopic Traffic Simulator

by

Xiang Song

Submitted to the Department of Civil and Environmental Engineering

on May 28, 2013, in partial fulfillment of the requirements for the degree of

Master of Science in Transportation

Abstract

One of the main challenges of making strategic decisions in transportation is that we always face a set of possible future states due to deep uncertainty in traffic demand. This thesis focuses on exploring the application of model-based decision support techniques which characterize a set of future states that represent the vulnerabilities of the proposed policy. Vulnerabilities here are interpreted as states of the world where the proposed policy fails its performance goal or deviates significantly from the optimum policy due to deep uncertainty in the future.

Based on existing literature and data mining techniques, a computational model-based approach known as scenario discovery is described and applied in an empirical problem. We investigated the application of this new approach in a case study based on a proposed transit policy implemented in Marina Bay district of Singapore. Our results showed that the scenario discovery approach performs well in finding the combinations of uncertain input variables that will result in policy failure.

Thesis Supervisor: Moshe E. Ben-Akiva

Title: Edmund K. Turner Professor of Civil and Environmental Engineering

Thesis Supervisor: Tomer Toledo

Title: Associate Professor of Faculty of Civil and Environmental Engineering, Technion - Israel Institute of Technology

Acknowledgement

During the two years I have spent at MIT I have been very fortunate to have received the help and support of numerous people.

First, I would like to thank my advisors, Professor Moshe Ben-Akiva and Professor Tomer Toledo for all their support.

I must also thank all the member and assistants of ITS group and the research group in Singapore, too many to list, who make my experience at MIT transportation program memorable.

My special thanks to my friends at MIT and Harvard, especially my girlfriend during my master study at MIT transportation.

My very particular thanks go to my parents, for all their support and love throughout my two years at MIT.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

One of the main challenges of making strategic decisions in transportation is that we always face a set of possible future states due to deep uncertainty in traffic demand. This thesis focuses on exploring the application of model-based decision support techniques which characterize a set of future states that represent the vulnerabilities of the proposed policy. Vulnerabilities here are interpreted as states of the world where the proposed policy fails its performance goal or deviates significantly from the optimum policy due to deep uncertainty in the future. Based on existing literature and data mining techniques, a computational model-based approach known as scenario discovery is described and applied in an empirical problem. We use the Marina Bay district, which is a bay district near Central Area of Singapore, as our empirical setting, and test a proposed transit-oriented policy in this district. This chapter describes the motivation for this thesis and presents the thesis outline.

1.1 Motivation

The performance of proposed policy or strategy is largely impacted by numerous exogenous driving forces. If we let some variables indicate these driving forces, these variables usually do not stay constant, in other words, we can usually find a set of future states with different combinations of these variables.

Traditional planning decisions are usually based on the assumptions that these variables are stable. Thus under deep uncertainty from these varying variables, the performance of the proposed policy may probably deviate significantly from the original optimum state.

In addition, the number of driving forces is not small especially when we deal with the problems in transportation. Large urban transportation network with multiple origin and destination pair demands always forms large complex system and we have to face the challenge of high dimensionality from these complex systems.

The challenge of high dimensionality results in two sub-problems. First, high dimensionality requires computational techniques that can efficiently incorporate all the possible

combinations of variables. Second, we need statistical algorithms that can identify the policy-relevant regions (combinations of variable ranges) of interest which is easy-to-interpret.

With the growing power of information technology, especially emerging algorithms in data mining or machine learning fields and availability of micro-simulation model of these large complex systems, some innovative approaches that can address these challenges to some extent are created.

In sum, there is urgent need to understand and evaluate those innovative computational approaches that can address the robust planning problems efficiently and quantitatively. Complete evaluation requires thorough review of the methods and previous studies and some empirical validation as well. We will go through them in the thesis.

1.2 Thesis Outline

The outline of this thesis is as follows.

Chapter 2 provides an introduction to robust decision making problems. The background of these problems will be shown. In addition, we reviewed previous studies of robust decision making problems, the existing techniques and some of its applications from literature.

Chapter 3 provides a review of techniques and algorithms that can be used in a model-based approach known as scenario discovery. There are two main components of this approach: data "farming" - "farm" a range of possible alternative futures by futures exploration techniques and data mining – identifying vulnerable regions of interest by data mining algorithms. Several exploration techniques and data mining algorithms that can be used in the scenario discovery are reviewed in this chapter.

In Chapter 4, we illustrate the analytical procedure of scenario discovery approach. The methodology of this approach is shown step by step. In first step, we sample from a set of future states by Latin-hypercube-sampling technique and generate output from these samples using a simulation model. Then the candidate scenarios that represent the vulnerabilities of the proposed policy in the future would be identified by patient rule induction method. In the third step, we evaluate the candidate scenarios with some diagnostics and at last we choose a scenario based on the result of the third step.

In Chapter 5, we focus on the application of scenario discovery approach. An empirical case study of a proposed transit-oriented policy in an urban transportation network of a district known as Marina Bay of Singapore will be given. A micro-simulation model of the Marina Bay transportation network will be used to simulate the performance of this system with and without the proposed policy. The relationship between the uncertainty of input model parameters and the failure of the proposed policy will be identified. Conclusions about the policy will be given after the computational results and discussion about them.

In Chapter 6, we finally discuss the overall contributions of my thesis. In addition, we propose ideas for future work including the use of alternative machine learning techniques in scenario discovery and add some dynamic features in the study.

Chapter 2

Introduction to Robust Decision Making Problems

Information technology's growing power offers many new tools and methods to improve human decision-making.

As illustrated in section 1.1, there exists strong motivation to do robust decision making analysis for planning problems especially in transportation. In this chapter, we will first illustrate the background of robust decision making problems including introducing their sources of uncertainty and what kind of uncertainty we will be focusing on in section 2.1. Then we will briefly talk about robust decision making and some of its general features. Finally we will review the existing techniques and applications that used for robust decision making problems from previous literatures.

2.1 Background

Decision making is predicated upon understanding the future. In this context, the field has continuing concerns about uncertainty [1, 2] and deriving robust strategies under the uncertainty. Robust decision making (RDM) is a widely-used iterative decision analytic framework that helps researchers and analysts to identify potential robust strategies, characterize the vulnerabilities of those strategies, and evaluate the tradeoffs among those strategies. RDM always focuses on cases when there is deep uncertainty and is designed and employed as a method for decision support.

We focus on uncertainty in large-scale models, which comes from at least the following sources:

- imperfect data [3].

- imperfect behavioral representations of individuals, markets, etc. [4].

- imperfect knowledge about the future state of exogenous forces impacting an urban area (e.g., national and global-level economic conditions, oil prices) [2].

In this thesis, our interest is in the third source of uncertainty – forecasting exogenous factors. The importance and relevance of this particular source of uncertainty are evidenced by the increasing use of scenario-planning techniques in urban transportation planning. Scenario

planning, famously adapted from military applications to the private sector by Shell in the early 1970s, was being adapted to urban- and transportation-related planning applications by the early 1980s and is increasingly used today [5]. For example, Bartholomew [6] reviews 80 recent applications in over 50 USA metropolitan areas. Bartholomew's review reveals, however, that much of the focus has been on "visioning" – i.e., identifying the future "we want" – rather than identifying uncertain exogenous forces and developing plans that prove to be robust across uncertainty.

Planning for the future inevitably involves accounting for the effects of exogenous driving forces. In the urban context, these forces are generally categorized as economics, social dynamics, politics, technology, and the environment [6]. Thus planners have to define ways to intervene in the urban system in order to achieve certain objectives. This pursuit is considered as constrained optimization– trying to do the best (however "best" is defined) subject to various constraints (legal, financial, technological, etc.).

Optimization and robustness are two main objectives in decision making that are not mutually exclusive. Optimality is a good method for choosing between the two equally robust strategies. However, in cases with highly uncertain probability distributions, robustness provides a good method for choice of strategies. Constrained on some variables like time, strategies can be repeatedly analyzed, with robust strategies selected and improved. Then Choices can be made between similarly performing strategies with weaknesses to different variations in inputs [7].

When confronting high uncertainty, planners should concern themselves with robustness – identifying strategies that perform well across a range of possible future conditions [8]. This thesis focuses upon discovering the areas of the futures space in which a strategy does not perform well as stated in chapter 1.

2.2 Literature Review

2.2.1 Robust Decision Making Overview

Information technology's growing power offers many new tools and methods to improve human decision-making. Robust decision making is a widely-used iterative decision analytic framework that helps researchers and analysts to identify potential robust strategies, characterize the vulnerabilities of those strategies, and evaluate the tradeoffs among those strategies. RDM

always focuses on cases when there is deep uncertainty and is designed and employed as a method for decision support.

Traditionally, policy makers employ expected utility decision framework. The differences between RDM and the traditional expected utility analysis mainly lies in three aspects.

First, RDM characterizes uncertainty with multiple future states. Since there is deep uncertainty when planning for the future, RDM uses sets of plausible probability distributions to describe deep uncertainty.

Second, instead of using optimality, it employs robustness as a criterion in assessing the different policies. It has employed several different definitions of robustness including: trading a small amount of optimum performance for less sensitivity to broken assumptions, good performance compared to the alternatives over a wide range of plausible scenarios, and keeping options open [9].

Third, RDM uses a vulnerability-and-response-option analysis to characterize uncertainty and then evaluate the robust strategies, while the traditional decision analytic approach follows what has been called a predict-then-act approach that first characterizes uncertainty about the future, and then rank the desirability of alternative decision options using this characterization [10].

Scenario discovery [11, 12] is one kind of RDM analysis which assists to identify the vulnerabilities of proposed strategies. There are some other RDM analyses such as exploratory modeling.

2.2.2 Existing Techniques and Applications Overview

As stated in previous section, there are several methods in robust decision making analysis focusing on different aspects of the problem. A new computer-assisted scenario development approach we call scenario discovery helps policy-makers and researchers in identifying groups of data we call scenarios by applying data- mining algorithms to large databases generated by simulation model.

Traditional scenarios provide an appealing means to communicate and characterize uncertainty in supporting robust decision making applications, but can fall short of their potential, especially when used in public sector applications with diverse audiences [13, 14].

Initial applications of scenario discovery, in particular in two high-impact public policy studies, suggest the approach may help overcome some limitations of the purely qualitative approaches to choosing scenarios.

Benjamin Bryant and Robert Lempert first provides a complete description of scenario discovery approach, introduces diagnostic tools to evaluate the statistical significance of the scenarios suggested by the algorithms, and suggests how the approach can address several outstanding challenges faced by traditional scenario approaches when applied in contentious public debates [15].

There are numerous definitions for scenarios and numerous methods that are used to create them [16, 17]. Key approaches derive from the school La Prospective developed in France by Gaston Berger and Michel Godet, the Probabilistic Modified Trends school originally developed at Ted Gordon and Olaf Helmer at RAND, and the intuitive logics or Anglo-American school that originated at RAND in the 1960s now often associated with the scenario groups at Shell Oil and the Global Business Network [18].

One intuitive definition describes scenarios as "internally consistent and challenging descriptions of possible futures" [19]. A small evaluative literature provides empirical evidence that while scenarios and the process of developing them can in some cases produce these claimed benefits, they often fail to do so, particularly when applied for groups with diverse interests and worldviews [14].

In many cases observers see the choice of scenarios as arbitrary or highly subject to the particular interests and values of those choosing them [16]. Observing an exercise by a government agency in the Netherlands, Van 't Klooster and van Asselt noted three distinct and conflicting interpretations of the scenario axes developed by the group. The authors conclude that the diffuse and heterogeneous nature of public agencies' objectives and interests may make it impossible for them to come to consensus about the meaning of scenario axes [20].

Comparative analyses also suggest that many scenario processes systematically exclude surprising or paradoxical developments as inconsistent or logically impossible [21]. Van Notten et al. [22] compare twenty-two scenario studies, some using simulation models and others entirely qualitative, and find that none of the model-based exercises included discontinuities in system behavior. Finally, it often proves difficult to include probabilistic information in a traditional scenario analysis without contaminating the simplicity and sense of possibility, as opposed to prediction, that makes the scenarios useful in the first place [23].

The main challenge is to choose a small number of scenarios to summarize the full breadth of uncertainty about the future. A set of scenarios cannot contain more than a handful of members and remain clear to decision makers, who may face a set of potentially plausible and important futures. Schwartz provides the classic exposition of how the intuitive logics (also called scenario axis) approach aims to reduce many futures to a manageable few [24].

Scenario discovery aims to address this challenge by employing the concept of scenarios and some statistical tools to implementing the concept. The concept defines scenarios as a set of future states of the world that represent vulnerabilities of proposed policies. Vulnerability refers to the states of the world where a proposed policy may fail to meet its performance goals. It can also refer the states where policy's performance deviates significantly from the optimum outcome.

Scenario discovery uses statistical or data-mining algorithms to find those scenarios (policy-relevant regions) in the space of uncertain input parameters to computer simulation models. Since the combination of all uncertain input parameters would be large, simulation models are often run many times over a space defined by the input parameters. Some policy-relevant criterion such as total cost of the project is used to distinguish a subset of the cases. A threshold for the criterion will be applied to the model's outputs in the classification. Statistical or data-mining algorithms applied to the generated database by the simulation model and then find easy-interpret regions of input space that best predict these cases of interest. These regions of input space are considered as scenarios, and the uncertain input parameters used to define these regions are the key driving forces for the proposed policies. Scenario discovery offers a quantitative approach that addresses these difficulties. Particularly, this robust decision making approach improves the efficacy of scenarios for diverse audiences in public sectors [15].

Three practical examples applied scenario discovery and showed its potential benefits. A study in 2007 evaluated alternative policies considered by the United States Congress while debating reauthorization of the Terrorism Risk Insurance Act (TRIA) [25].

Using scenarios made it easier to consider a wide range of assumptions about difficult-to-predict events – in particular any post-attack compensative decisions of a future Congress – thereby enabling this study to reach different conclusions than those of the Congressional Budget Office and Treasury Department [15].

A second scenario discovery analysis helped Southern California's Inland Empire Utilities Agency (IEUA) reduce the vulnerability of its long-range water management plans to potential climate change [26, 27]. Similarly to the TRIA example, this scenario and resulting analysis provided benefits difficult to achieve with other approaches, allowing IEUA's managers, constituents, and elected officials, who did not all agree on the likelihood of climate impacts, to understand in detail vulnerabilities to their plan and discuss ways to ameliorate those vulnerabilities [15].

A third scenario discovery analysis helped in the policy option of a subsidy for low-income households in downtown Lisbon [28]. The study showed different methods in the literature exploring the possible future under vulnerabilities and compared those methods. Scenario discovery is applied to identify the robust urban development strategies. Using the UrbanSim model, it offers the first known example of applying computational scenario-discovery techniques to the urban realm [28]. Data of the input variables including population growth rate, employment growth rate, gas prices and construction costs are sampled by Latin Hypercube Sampling (LHS) experiment design. A data-mining algorithm PRIM (Patient Rule Induction Method) is applied to identify scenarios where the subsidy strategy fails to satisfy the designed objective.

As widely noted, the process of developing scenarios often proves at least as important to decision-makers as the scenarios themselves. Scenario discovery represents a participatory, computer-assisted process that supports Robust Decision Making (RDM), a quantitative decision analytic method that uses available information (such as that contained in computer simulation models), not to improve predictions of a deeply uncertain future, but rather to help decision-

makers craft strategies that can more effectively achieve their goals in the face of these uncertainties [15].

2.3 Conclusion

In the planning field, especially in the urban context, different exogenous driving forces result in deep uncertainty which requires decision makers consider both optimality and robustness when making strategic decisions.

By using the increasing power of information technology especially computer simulation models and data mining techniques, robust decision making especially scenario discovery offers a tool to assist policy makers and analysts. In next chapter, we will illustrate how scenario discovery approach could be performed analytically to assist in the robust decision making process.

Chapter 3

Introduction to Scenario Discovery Analysis

3.1 Introduction

In this chapter, we provide a brief introduction to scenario discovery, a type of robust decision making analysis approach and presents a review of literature for the techniques and algorithms that are used as two main components of this approach.

Scenario discovery aims to identify sets of future states of the world that shows the vulnerabilities in proposed policies and to describe these scenarios for decision makers and other stakeholders. There are four steps when implementing scenario discovery approach, which is shown in Figure 3.1.

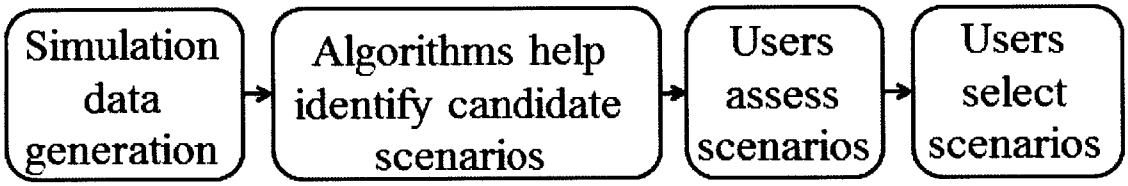


Figure 3.1 Procedure of Scenario Discovery

In the first step, users specify one type of sampling experimental design for the simulation model and also specify the criterion to distinguish policy-relevant regions of interest in the output. Efficient exploration techniques would have less number of samples to represent the uncertainty across all the input parameters. After running the model with the samples, we have a database consists of a number of outputs and their corresponding input combinations. Several exploration techniques will be reviewed.

In the second step, one or more statistical or data-mining algorithms are applied to the resulting database generated from simulation and to identify candidate scenarios that provide a good description of these regions of interest. There are numerous classification and bump hunting algorithms that may fit the requirement of scenario discovery approach. Thus we will give a review of existing algorithms that can be applied in the second step.

In the third step, some statistical diagnostics proposed by Bryant and Lempert [15] are used to evaluate these scenarios. As discussed before, the scenario discovery is an iterative and participatory approach. User can also go from the third step to the first or second step. Different options are given to users showing the tradeoffs among them. By evaluating the proposed scenarios, users can select scenarios.

In section 3.2, various future exploration techniques incorporating uncertain input parameters of the model will be illustrated and discussed. In section 3.3, different data mining algorithms that match the requirement of scenario discovery analysis are presented and compared. In section 3.4, we will summarize this chapter.

3.2 Futures Exploration Techniques

In the first step, one exploration technique or experimental design should be applied to the high dimensional future space of uncertain model input parameters. In this section, we will review the existing exploration techniques.

First, we need one or more built computer simulation models from the existing data. The model could be written in a form as follows.

$$y = f(s, \boldsymbol{x}) \qquad (3.1)$$

In the above model, y is the simulation output of interest which is contingent on a vector of input data $\boldsymbol{x}$ representing a particular point in an M-dimensional space of uncertain model input parameters, s is the policy makers' action, which can be a subsidy policy or a transit-oriented policy based on the study.

For instance, in the urban development in Lisbon example in the previous section, the output of interest is the difference of numbers of low-income households with or without subsidy. The simulation model is built on UrbanSim. The vector of input data includes population growth rate, employment growth rate, gas prices and construction costs. The action is to subsidize the urban area in Lisbon.

Using some policy-relevant criteria, we choose some threshold performance level Y^I that defines a set of cases of interest $I_s = \{x^I | f(s, x^I) \geq Y^I\}$ or $\{x^I | f(s, x^I) \leq Y^I\}$, contingent on that strategy [15].

For instance, in the TRIA example, I_s is the set of cases where the legislation imposes net costs on taxpayers and for IEUA I_s is the set of cases where the agency's costs exceed 20% or more of those assumed in the current plan [25, 26, 27].

To explore those scenarios of interest, numerous known exploration techniques could be applied and Swartz and Zegras [28] compared and evaluated four different exploration techniques for the use in the scenario discovery analysis. Four exploration techniques are discussed in their paper including experience/intuiton-based exploration, orthogonal exploration, Latin-hypercube-sample exploration, and pseudo-full-factorial exploration. The computation intensity increases from the prior to the last one [28].

For experience/intuition-based exploration techniques, they derive from the Shell scenario planning tradition [29], in which experts and/or stakeholders identify combinations of a system's fundamental external driving forces and their likely effects on a particular concern. The idea is to construct scenarios that bound the possible and increase awareness of possible futures.

Zegras et al [6] reviewed several attempts since the early 1980s to apply scenario planning methodology in urban transportation-related applications, including in Sydney, Baltimore, and Seattle.

Orthogonal exploration computes the elasticities of output response to variations in inputs. In order to calibrate an equation based upon these elasticities, model runs that are orthogonal to each other are carried out and the elasticities are derived from these runs.

Bowman et al [30] provide one example of this approach that has been applied to transportation models. Elasticities are estimated based upon reference deviations and effectively predict the local output of more complex models and it saves in computational time obviously.

Latin-hypercube-sample (LHS) experimental design is carried out in the household subsidy example. The experiment design distributes model simulation points across the futures space in a manner that decreases variability of results [31]. The cumulative distribution function for each

input variable is divided into intervals of equal probability. Thus it enables the scenario discovery approach on the range of all possible or even improbable input values. LHS can also be considered as a special type of non-orthogonal sampling, which can be learned from Figure 3.2.

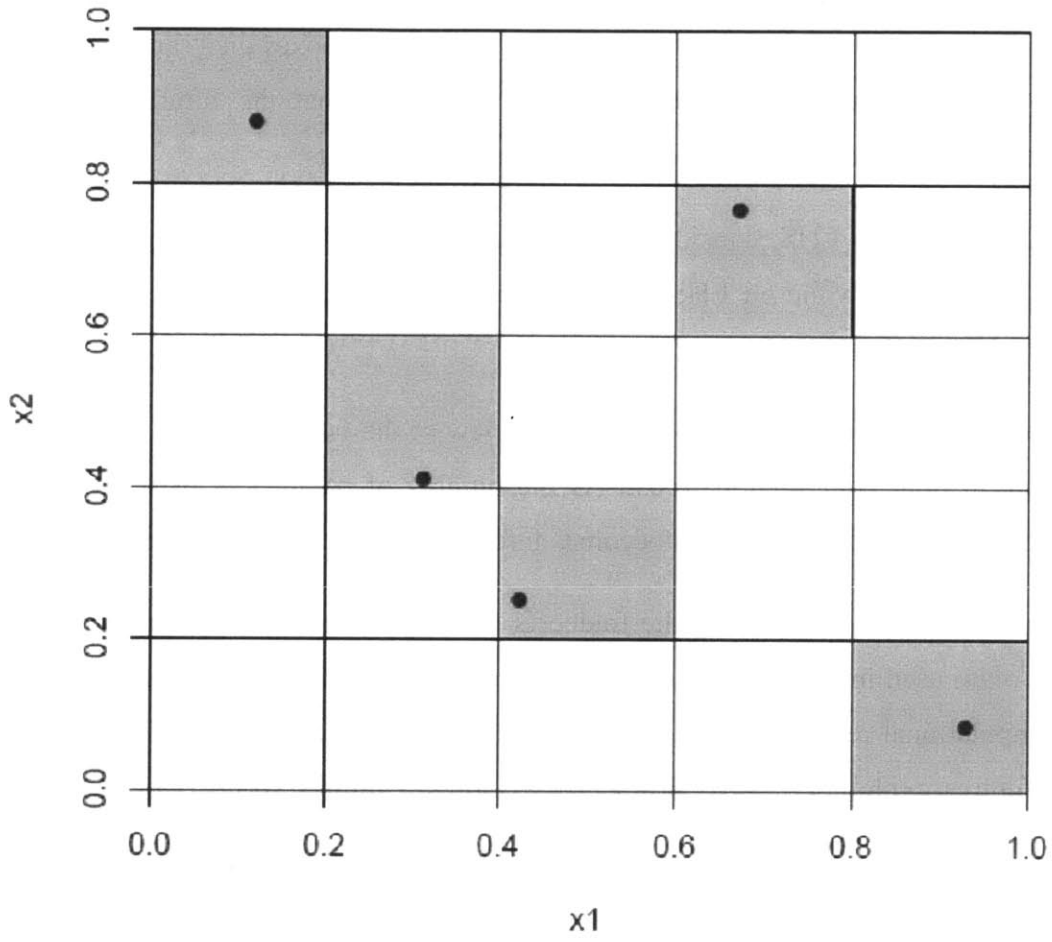


Figure 3.2 LHS for 2 Variables [32]

Figure 3.2 shows an LHS for 2 variables (in 2 dimensions) with ranges of 0 to 1. The range of 0 to 1 are stratified into five stratum with equal probability. The grey represents the area from which each sampled point would be sampled, and the points represent the actual values sampled.

Outputs from the experimental design, characterized into binary failure and success cases, can then be data-mined through classification algorithms such as Patient Rule Induction Method (PRIM) [33] and Classification and Regression Tree (CART) [34]. The classification algorithm

thus identifies the range of input variables for which result in the values of interest (failure or success).

The most computationally intensive of the four approaches is the Pseudo-full-factorial (PFF) exploration, which is similar to an LHS experimental design but samples a greater number of points from the futures space. Construction of the PFF divides the overall hyperspace defined by the range of the parameters that make up into constituent boxes then samples randomly within each box. The "pseudo" qualifier is applied to reflect that the futures space is generally continuous, while the simulations conducted are discrete points.

Different from LHS, considering the cases in Figure 3.2, a PFF would sample one point from all 25 boxes, while an LHS would sample only the points shown. Simulations then are conducted for each point.

A PFF thus gives a much more detailed view of the futures surface than that given by the LHS, but at high computational cost. As the number of parameters increases, any significant stratification of the variable values becomes infeasible for all but the simplest models.

In conclusion, considering the tradeoffs of exhaustive exploration and exploration costs, LHS is often used in the scenario discovery. We also applied it in our study since LHS matched our computational resources while avoiding the pitfalls presented by intuition- and orthogonal-based futures exploration. LHS efficiently explores a futures space [15], economically using computational resources to discover a model's reaction to different input parameters [35].

3.3 Data Mining Algorithms

3.3.1 Existing Algorithms Overview

In the second step, scenario discovery uses statistical or data-mining algorithms to classify the combinations of values of uncertain model input parameters that best predict the set of interesting cases. There is no existing algorithm that exactly fit the requirement of the scenario discovery approach. Thus we will review and compare the existing classification and bump hunting algorithms in this section.

As described previously, in the second step of the scenario discovery approach, a statistical or data-mining algorithm is needed to identify the scenarios. Since classification and bump

hunting algorithm may fit the requirement well, we will explore the some of the existing algorithms that can help identify the scenarios.

There are numerous classification techniques that have been widely employed for identifying different subgroups in the datasets. In the machine learning terminology, there are linear and nonlinear methods to implement it. In addition, there are two different groups of classification methods in machine learning: unsupervised learning and supervised learning.

Next we will explore possible existing techniques and evaluate and select one algorithm for the use of our study. Basically, we will have an overview of most learning algorithms and three common techniques will be introduced and compared: logistic regression, CART (classification and regression tree), and PRIM (patient rule induction method). The first one is linear method and the latter two can be employed in nonlinear cases.

In machine learning, classification is to identify a sub-population or sub-group that a new observation or instance belongs. When we have a training dataset, which contains observations whose sub-group membership is known. Each individual observation belongs and only belongs to one category (sub-group). The individual observations consist of a set of properties called explanatory variables. These explanatory properties can be categorical, ordinal, integer-valued, or real-valued. For example, an email will be assigned to class "spam" or class "non-spam", or a given patient will be characterized into different types of patient by the known characteristics of the patient (gender, temperature, blood pressure, etc).

In machine learning, classification is considered to be a kind of supervised learning, which means in the training data set, the correct category membership of individual observation is known. While in the unsupervised learning domain, there is another procedure known as cluster analysis where observations are grouped into different categories based on the measure of the similarity of the observations (e.g. the distance between instances).

In the statistical terminology, classification is often done by logistics regression or other similar regression techniques, and the properties of observations are known as explanatory variables or independent variables, and the categories are known as outcomes or dependent variables. In machine learning terminology, we also call observation as instances and explanatory variables as features, and the categories as classes.

The classification problem is quite similar to the problem of pattern recognition, which assigns some output values to a given input value, and is recognized as its more general problem.

In the classification problems, the common techniques we used usually fall into two groups: linear and nonlinear methods. In the following sections, we will introduce one linear classification method and two nonlinear (tree-based) methods.

3.3.2 Logistic Regression

In this section, we will introduce three classification or bump hunting algorithms which are widely used in categorizing different subgroups. Most of the materials about algorithms in this section are adapted from the book by Hastie, Tibshirani and Friedman [36].

First, we focus on linear methods for classification. One of most commonly used methods in linear classification is logistic regression.

The logistic regression model arises from the desire to model the posterior probabilities of the K classes via linear functions in x, while at the same time ensuring that they sum to one and remain in [0,1]. The model has the form

$$\log\frac{\Pr(G=1|X=x)}{\Pr(G=K|X=x)}=\beta_{10}+\beta_1^T x \qquad (3.2.1)$$

$$\log\frac{\Pr(G=2|X=x)}{\Pr(G=K|X=x)}=\beta_{20}+\beta_2^T x \qquad (3.2.2)$$

$$\vdots$$

$$\log\frac{\Pr(G=K-1|X=x)}{\Pr(G=K|X=x)}=\beta_{(K-1)0}+\beta_{K-1}^T x \qquad (3.2.K-1)$$

The model is specified in terms of $K-1$ log-odds or logit transformations (reflecting the constraint that the probabilities sum to one). Although the model uses the last class as the denominator in the odds-ratios, the choice of denominator is arbitrary in that the estimates are equivariant under this choice. A simple calculation shows that

$$\Pr(G=k|X=x)=\frac{\exp(\beta_{k0}+\beta_k^T x)}{1+\sum_{l=1}^{K-1}\exp(\beta_{l0}+\beta_l^T x)}, k=1,\dots K-1 \qquad (3.3)$$

$$\Pr(G = K|X = x) = \frac{1}{1 + \sum_{l=1}^{K-1} \exp(\beta_{l0} + \beta_l^T x)} \qquad (3.4)$$

and they clearly sum to one. To emphasize the dependence on the entire parameter set $\theta = \{\beta_{10}, \beta_1^T, \ldots, \beta_{(K-1)0}, \beta_{K-1}^T$, we denote the probabilities $\Pr(G = K|X = x) = p_k(x; \theta)$.

When $K = 2$, this model is especially simple, since there is only a single linear function. It is widely used in bio-statistical applications where binary responses (two classes) occur quite frequently. For example, patients survive or die, have heart disease or not, or a condition is present or absent.

The regression coefficients are usually estimated using maximum likelihood estimation. Unlike linear regression with normally distributed residuals, it is not possible to find a closed-form expression for the coefficient values that maximizes the likelihood function, so an iterative process must be used instead, for example Newton's method.

Conditional likelihood of G given X is used. Since $\Pr(G|X)$ completely specifies the conditional distribution, the multinomial distribution is appropriate. The log-likelihood for N observations is

$$l(\theta) = \sum_{i=1}^{N} \log p_{g_i}(x_i; \theta) \qquad (3.5)$$

where $p_k(x_i; \theta) = \Pr(G = k|X = x_i; \theta)$.

We discuss in detail the two-class case in the following, since the algorithms simplify considerably. To maximize the log-likelihood, we set its derivatives to zero. These score equations are

$$\frac{\partial l(\beta)}{\partial \beta} = \sum_{i=1}^{N} x_i(y_i - p(x_i; \theta)) = 0 \qquad (3.6)$$

which are $p + 1$ equations nonlinear in β.

To solve the equation 3.6, we usually use Newton-Raphson algorithm, which requires the second derivatives of the left hand side of equation 3.6.

It is convenient to write the score and Hessian in matrix notation. Let y denote the vector of y_i values, **X** the N × (p + 1) matrix of x_i values, p the vector of fitted probabilities with ith element $p(x_i; \theta)$, **z** as adjusted response and **W** a N ×N diagonal matrix of weights with ith diagonal element $p(x_i; \theta)(1 - p(x_i; \theta))$. Then we can have

$$\beta^{new} \leftarrow \arg\min_{\beta}(\boldsymbol{z} - \boldsymbol{X}\beta)^T \boldsymbol{W}(\boldsymbol{z} - \boldsymbol{X}\beta)$$

It seems that β = 0 is a good starting value for the iterative procedure, although convergence is never guaranteed. Typically the algorithm does converge, since the log-likelihood is concave, but overshooting can occur. In the rare cases that the log-likelihood decreases, step size halving will guarantee convergence.

For the multiclass case (K $\geqslant$ 3) the Newton algorithm can also be expressed as an iteratively reweighted least squares algorithm, but with a vector of K−1 responses and a non-diagonal weight matrix per observation. The latter precludes any simplified algorithms, and in this case it is numerically more convenient to work with the expanded vector θ directly. Alternatively coordinate-descent methods can be used to maximize the log-likelihood efficiently.

Logistic regression models are used mostly as a data analysis and inference tool, where the goal is to understand the role of the input variables in explaining the outcome. Typically many models are fit in a search for a parsimonious model involving a subset of the variables, possibly with some interactions terms.

3.3.3 Classification and Regression Tree

In the following two subsections, we begin to discuss two specific methods for supervised learning. These techniques each assume a different structured form for the unknown regression function, and by doing so they finesse the curse of dimensionality.

Regression models play a very important role in many data analyses, providing prediction and classification rules, and data analytic tools for understanding the importance of different inputs.

Although attractively simple, the traditional linear model often fails in these situations: in real life, effects are often not linear. In earlier subsection, we described classification methods

with linear form, which is logistic regression. This section describes more automatic flexible statistical methods that may be used to identify and characterize nonlinear regression effects. These methods are called "generalized additive models."

Tree-based methods partition the feature space into a set of rectangles, and then fit a simple model in each one. They are conceptually simple yet powerful. We first describe a popular method for tree-based regression and classification called CART.

Let's consider a regression problem with continuous response Y and inputs X_1 and X_2, each taking values in the unit interval. The top left panel of Figure 3.1 shows a partition of the feature space by lines that are parallel to the coordinate axes. In each partition element we can model Y with a different constant. However, there is a problem: although each partitioning line has a simple description like $X_1 = c$, some of the resulting regions are complicated to describe.

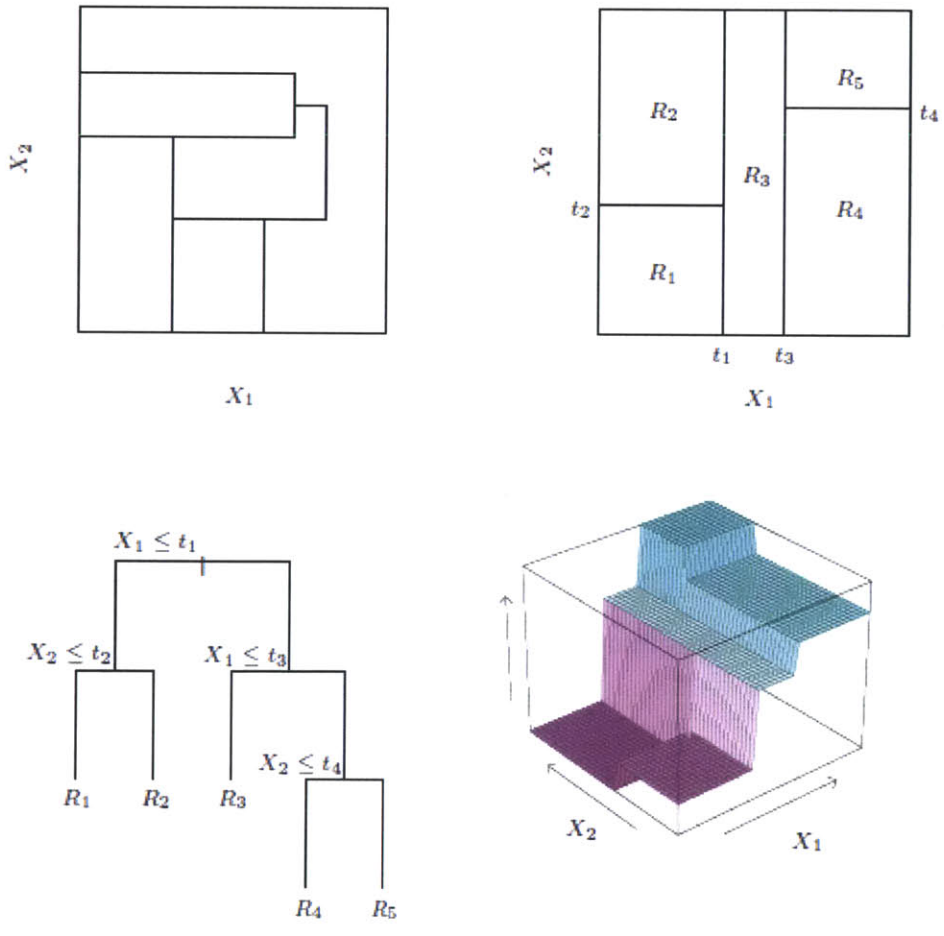


Figure 3.3 Partition and CART [29]

To simplify matters, we restrict attention to recursive binary partitions like that in the top right panel of Figure 3.3. We first split the space into two regions, and model the response by the mean of Y in each region. We choose the variable and split-point to achieve the best fit. Then one or both of these regions are split into two more regions, and this process is continued, until some stopping rule is applied. For example, in the top right panel of Figure 3.1, we first split at $X_1 = t_1$. Then the region $X_1 \leq t_1$ is split at $X_2 = t_2$and the region $X_1 > t_1$ is split at $X_1 = t_{13}$. Finally, the region $X_1 > t_3$ is split at $X_2 = t_4$. The result of this process is a partition into the five regions $R_1, R_2, \ldots, R_5$ shown in the figure. The corresponding regression model predicts Y with a constant c_m in region R_m, that is,

$$\hat{f}(X) = \sum_{m=1}^{5} c_m I\{(X_1, X_2) \in R_m\} \qquad (3.7)$$

This same model can be represented by the binary tree in the bottom left panel of Figure 3.3. The full dataset sits at the top of the tree. Observations satisfying the condition at each junction are assigned to the left branch, and the others to the right branch. The terminal nodes or leaves of the tree correspond to the regions $R_1, R_2, \ldots, R_5$. The bottom right panel of Figure 3.3 is a perspective plot of the regression surface from this model. For illustration, we chose the node means $c_1 = -5, c_2 = -7,\ c_3 = 0,\ c_4 = 2,\ c_5 = 4$ to make this plot.

A key advantage of the recursive binary tree is its interpretability, which fits well for the scenario discovery analysis requirement. The feature space partition is fully described by a single tree. With more than two inputs, partitions like that in the top right panel of Figure 3.3 are difficult to draw, but the binary tree representation works in the same way. This representation is also popular among medical scientists, perhaps because it mimics the way that a doctor thinks. The tree stratifies the population into strata of high and low outcome, on the basis of patient characteristics.

Since regression tree and classification tree are similar. We now go to the question of how to grow a regression tree. Our data consists of p inputs and a response, for each of N observations: that is, (x_i, y_i) for i = 1,2,...,N, with $x_i = (x_{i1}, x_{i2}, \ldots, x_{ip})$. The algorithm needs to automatically decide on the splitting variables and split points, and also what topology (shape)

the tree should have. Suppose first that we have a partition into M regions $R_1, R_2, \ldots, R_M$, and we model the response as a constant cm in each region:

$$f(x) = \sum_{m=1}^{M} c_m I\{x \in R_m\} \qquad (3.8)$$

If we adopt as our criterion minimization of the sum of squares $\sum(y_i - f(x_i))^2$, it is easy to see that the best $\hat{c}_m$ is just the average of y_i in region R_m:

$$\hat{c}_m = ave(y_i | x_i \in R_m) \qquad (3.9)$$

Now finding the best binary partition in terms of minimum sum of squares is generally computationally infeasible. Hence we proceed with a greedy algorithm. Starting with all of the data, consider a splitting variable j and split point s, and define the pair of half-planes

$$R_1(j,s) = \{X | X_j \leq s\} \quad and \; R_2(j,s) = \{X | X_j > s\} \qquad (3.10)$$

For each splitting variable, the determination of the split point s can be done very quickly and hence by scanning through all of the inputs, determination of the best pair (j, s) is feasible.

Having found the best split, we partition the data into the two resulting regions and repeat the splitting process on each of the two regions. Then this process is repeated on all of the resulting regions.

How large should we grow the tree? Clearly a very large tree might overfit the data, while a small tree might not capture the important structure.

Tree size is a tuning parameter governing the model's complexity, and the optimal tree size should be adaptively chosen from the data. One approach would be to split tree nodes only if the decrease in sum-of-squares due to the split exceeds some threshold. This strategy is too short-sighted, however, since a seemingly worthless split might lead to a very good split below it.

The preferred strategy is to grow a large tree T_0, stopping the splitting process only when some minimum node size (say 5) is reached. Then this large tree is pruned using cost-complexity pruning, which we now describe.

We define a sub-tree $T \subset T_0$ to be any tree that can be obtained by pruning T_0, that is, collapsing any number of its internal (non-terminal) nodes. We index terminal nodes by m, with node m representing region R_m. Let $|T|$ denote the number of terminal nodes in T. Letting

$$N_M = \#\{x_i \in R_m\} \tag{3.11.1}$$

$$\hat{c}_m = \frac{1}{N_m} \sum_{x_i \in R_m} y_i \tag{3.11.2}$$

$$Q_m(T) = \frac{1}{N_m} \sum_{x_i \in R_m} (y_i - \hat{c}_m)^2 \tag{3.11.3}$$

We define the cost complexity criterion.

$$C_\alpha(T) = \sum_{m=1}^{T} N_m Q_m(T) + \alpha|T| \tag{3.12}$$

The idea is to find, for each α, the subtree $T_\alpha \subseteq T_0$ to minimize $C_\alpha(T)$. The tuning parameter $\alpha \geq 0$ governs the tradeoff between tree size and its goodness of fit to the data. Large values of α result in smaller trees T_α, and conversely for smaller values of α. As the notation suggests, with $\alpha = 0$ the solution is the full tree T_0. We discuss how to adaptively choose α below.

For each α one can show that there is a unique smallest sub-tree T_α that minimizes $C_\alpha(T)$. To find T_α we use weakest link pruning: we successively collapse the internal node that produces the smallest per-node increase in $\sum_m N_m Q_m(T)$, and continue until we produce the single-node (root) tree. This gives a (finite) sequence of sub-trees, and one can show this sequence must contain T_α. Estimation of α is achieved by five- or tenfold cross-validation: we choose the value $\hat{\alpha}$ to minimize the cross-validated sum of squares. Our final tree is $T_{\hat{\alpha}}$.

For classification tree, it is very similar to the regression tree. If the target is a classification outcome taking values 1, 2, ..., K, the only changes needed in the tree algorithm pertain to the criteria for splitting nodes and pruning the tree. For regression we used the squared-error node impurity measure $Q_m(T)$ defined in Equation 3.11, but this is not suitable for classification. In a node m, representing a region R_m with N_m observations, let

$$\hat{p}_{mk} = \frac{1}{N_m} \sum_{x_i \in R_m} I(y_i = k), \qquad (3.13)$$

the proportion of class k observations in node m. We classify the observations in node m to class $k(m) = arg\max_k \hat{p}_{mk}$, the majority class in node m. Different measures $Q_m(T)$ of node impurity include misclassification error, Gini index, and cross-entropy or deviance. Details of these different measures can be found in the book by Hastie, Tibshirani and Friedman [36].

Tree-based methods (for regression) partition the feature space into box-shaped regions, to try to make the response averages in each box as different as possible. The splitting rules defining the boxes are related to each through a binary tree, facilitating their interpretation.

3.3.4 Bump Hunting Algorithm

The patient rule induction method (PRIM) also finds boxes in the feature space, but seeks boxes in which the response average is high. Hence it looks for maxima in the target function, an exercise known as *bump hunting*. (If minima rather than maxima are desired, one simply works with the negative response values.)

PRIM also differs from tree-based partitioning methods in that the box definitions are not described by a binary tree. This makes interpretation of the collection of rules more difficult; however, by removing the binary tree constraint, the individual rules are often simpler.

The main box construction method in PRIM works from the top down, starting with a box containing all of the data. The box is compressed along one face by a small amount, and the observations then falling outside the box are peeled off. The face chosen for compression is the one resulting in the largest box mean, after the compression is performed. Then the process is repeated, stopping when the current box contains some minimum number of data points.

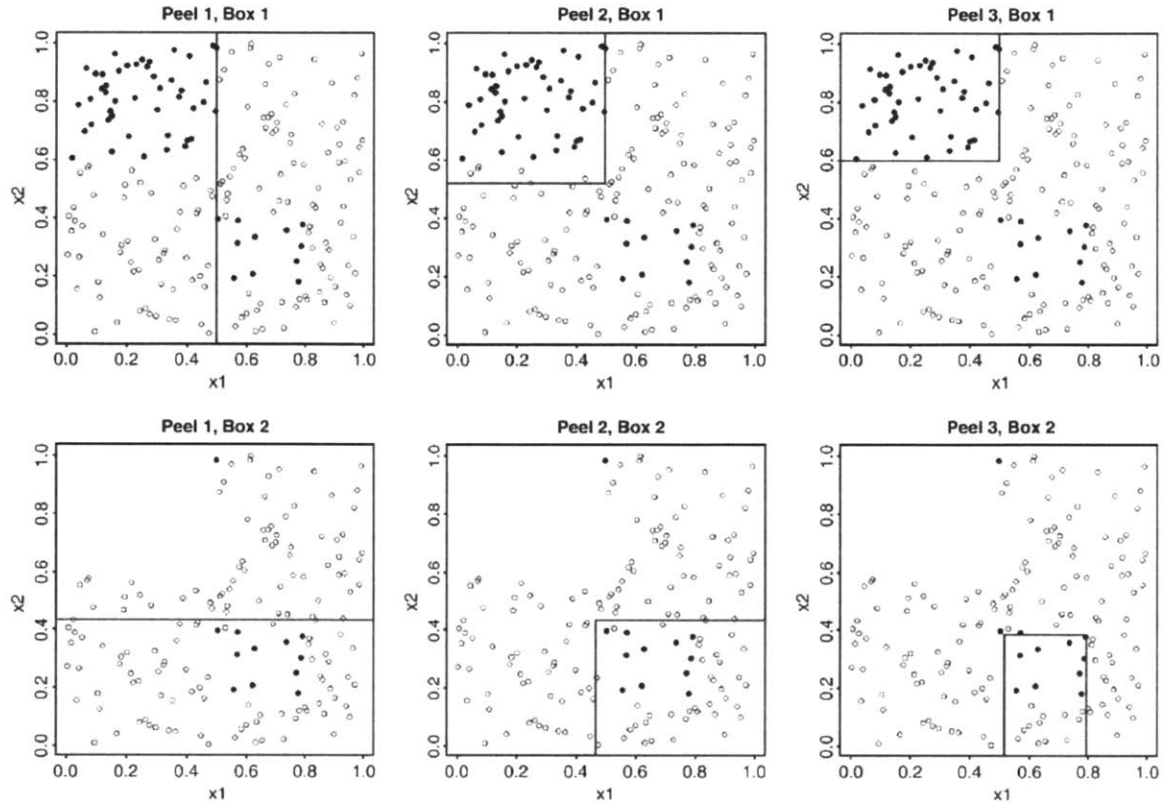


Figure 3.4 Sequence of operations by the PRIM algorithm [15]

As shown in Figure 3.4, PRIM finds each new box by removing a thin low density slice from whichever face of the current box will most increase the mean inside the new (remaining) box. PRIM's developers call the resulting series of boxes a "peeling trajectory."

An advantage of PRIM over CART is its patience. Because of its binary splits, CART fragments the data quite quickly. Assuming splits of equal size, with N observations it can only make $\log_2(N) - 1$ splits before running out of data. If PRIM peels off a proportion α of training points at each stage, it can perform approximately $-\log(N) / \log(1 - \alpha)$ peeling steps before running out of data. For example, if $N = 128$ and $\alpha = 0.010$, then $\log_2(N) - 1$ =6 while $-\log(N) / \log(1 - \alpha) \approx 46$. Taking into account that there must be an integer number of observations at each stage, PRIM in fact can peel only 29 times. In any case, the ability of PRIM to be more patient should help the top-down greedy algorithm find a better solution.

Among existing algorithms, the scenario-discovery task appears most similar to classification approaches. As mentioned in this chapter previously, there are some classification

algorithms that can be applied in the scenario discovery analysis. While to date, there is no existing algorithm performs tasks identical to that required for scenario-discovery [15].

Thus we have provided brief overview of classification algorithms. Three commonly used methods for classification and bump hunting problems are given: logistic regression, CART (classification and regression tree), and PRIM (patient rule induction method).

Each method has its own advantage and disadvantages. In general, logistic regression is often applied in the linear problems (although it can also be applied nonlinearly). For high dimensional cases with mixed data points, logistic regression is not as flexible as CART and PRIM.

For the problems with binary output, CART partition the input space into different regions in which one output class dominantly exists. Bump-hunting algorithms search for regions of input space that has a relatively high mean output value.

PRIM has several advantages over CART. First, it gives a box with relatively high density and high coverage rather than a decision boundary dividing the high dimensional space into two parts. In other words, we have some concentrated states of future from PRIM while we can only get the boundary of policy failure and non-failure. Obviously PRIM gives more flexible and useful results than CART.

In addition, PRIM has better performance in interpretation. The visualization shown in the next chapter will illustrate its highly interactive for the users' scenario choice decision. It also helps users see the tradeoffs between the measures of quality mentioned in section 3.4.1 coverage and density. The interpretability measure poses requirements distinct from most other applications. In addition, while many algorithms seek to maximize coverage, which is equivalent to the success-oriented quantification of the Type II error rate, few consider density, which is related to but does not neatly correspond to the Type I error rate because the denominator in Equation 3.3 refers to the set of scenarios rather than the overall dataset.

In the study by Bryant and Lempert [15], PRIM is applied because it is highly interactive, presents multiple options for the choice of scenarios, and provides visualizations that help users

balance among the three measures of scenario quality: coverage, density, and interpretability. In addition, a toolbox in R known as sdtoolkit was also developed for the use of scenario discovery.

Lempert, Bryant, and Bankes [37] also tested the ability of the classification algorithm CART (Classification and Regression Tree) to perform scenario discovery. CART appears to generate similar results as PRIM, but with less user interactivity and more work required by the analyst to create box sets with high interpretability [15]. CART generates similar results as PRIM, but comparing with user interactivity and concentrated identification results, we choose to use PRIM in our study.

3.4 Summary

In this chapter, we introduced basic steps of scenario discovery analysis. Besides the last step that evaluates the identified scenarios, there are two main steps of the analysis: data "farming" and data mining. Data "farming" incorporates the vulnerabilities of the proposed policy in the future by efficiently sampling from a set of combinations of uncertain input parameters; data mining identifies the policy-relevant regions that represent the vulnerabilities of the future.

Furthermore, we reviewed and discussed some existing exploration techniques and data mining algorithms that fit the requirement of scenario discovery. There are numerous computational tools for exploring the future states and Latin-Hypercube experimental design appears to be feasible and efficient for scenario discovery analysis. Among a number of classification algorithms, PRIM tends to be the best choice for now based on previous discussion. In the next chapter, we will go through the whole analytical procedure of scenario discovery analysis that will be applied in the empirical study in chapter 5.

Chapter 4

Analytical Procedure of Scenario Discovery

4.1 Overview

In this chapter, we will illustrate the procedure of conducting scenario discovery. Recall Figure 3.1, there are four steps in implementing scenario discovery analysis.

In section 4.2, we will first show the model we used for scenario discovery analysis and specify the criterions that distinguish policy-relevant regions of interest in the output. Then we will introduce how to use Latin-hypercube-sampling technique that incorporates the uncertainty of the model input parameters. In section 4.3, we will introduce how to use patient rule induction method to identify the policy relevant regions of interest. Some measures of merits are used to assist in identifying these regions of interest or scenarios. In section 4.4, some statistical diagnostics are illustrated to evaluate the identified scenarios. Summary of scenario discovery will be provided in section 4.5.

4.2 Model and Data Generation

4.2.1 Model

First, we recall the equation (3.1) $y = f(s, \boldsymbol{x})$. In this model, y is the simulation output of interest which is contingent on a vector of input data $\boldsymbol{x}$ representing a particular point in an M-dimensional space of uncertain model input parameters, s is the policy makers' action, which can be a subsidy policy or a transit-oriented policy based on the study.

In this study, we use a microscopic traffic simulation platform known as MITSIMLab. The model used is a traffic simulation model built and calibrated on the traffic sensor data from Marina Bay network Singapore. The input data are the uncertain traffic demand and other data like the network and driving parameters. The details will be shown in the Chapter 5. The action or policy is to convert one lane in the network into bus lane. The output of interest is the difference of travel times with and without bus lane.

To test the robustness of the proposed policy or action of policy makers, s is held in constant while $\boldsymbol{x}$ varies across all the future spaces.

Using some policy-relevant criteria, we choose some threshold performance level Y^I that defines a set of cases of interest $I_s = \{x^I | f(s, x^I) \geq Y^I\}$ or $\{x^I | f(s, x^I) \leq Y^I\}$, contingent on that strategy [15]. Y^I is the outcome threshold for the proposed policy. The set of interest consists of vectors of input parameter which will result into outcome of interest, where we distinguish the cases of interest by the inequality $f(s, x^I) \geq Y^I$ or $f(s, x^I) \leq Y^I$. In general, the direction of inequality is chosen so that the minor parts of the total set are of interest or say are scenarios.

Usually these regions of input parameters in high dimensional space are called scenarios (or boxes). Specifically, the algorithm will search for the scenarios that containing outcomes of interest $I_s = \{x^I | f(s, x^I) \geq Y^I\}$ or $\{x^I | f(s, x^I) \leq Y^I\}$.These scenarios are often one or more sets of limiting constraints $B_k = \{a_j \leq x_j \leq b_j, j \in L_k\}$ on the ranges of a subset of input parameters $L_k \subseteq \{1, \dots, M\}$. Input parameters that are not in L_k is not constrained for B_k. We call each set of simultaneous constraints B_k a box and a set of boxes B a box set.

Although we focus on the states that are in the box, we cannot ignore the all those states not in any box and sometimes they are considered as a scenario [15]. For instance, a single box might represent a scenario where a policy has high costs. All the other states might represent the scenario where the policy has reasonable costs. In addition, the scenario discovery algorithms will in some cases yield boxes that overlap. The situation of scenarios may go much complicated than we illustrated here. It is convenient and intuitively simple to consider such box as distinct scenarios although they might be more usefully viewed as a single scenario with a shape poorly described by a box. Some improvements can be applied to address such situations [15].

4.2.2 Data Generation

LHS is a form of stratified sampling that can be applied to multiple variables. The method is commonly used to reduce the number of runs necessary for a Monte Carlo simulation to achieve a reasonably accurate random distribution. LHS can be incorporated into an existing Monte Carlo model fairly easily, and work with variables following any analytical probability distribution.

With LHS, variables are sampled independently, and then randomly combined sets of those variables are used for one calculation of the target function. LHS construction requires specifying the number of desired model runs and the number of input parameters to be varied

during these runs. For a function with independent inputs, an LHS is created by dividing the cumulative distribution function for each model input x_k into intervals of equal probability. The number of intervals (n_S) is equal to the number of runs to be carried out. Within each interval, a value for the input is drawn based on its cumulative distribution function (CDF_{x1}, CDF_{x2}, ..., CDF_{xk}). The model runs are generated by randomly drawing one value for each input x_j and matching these inputs to create one run. Building additional runs repeats this procedure without replacing previously selected input values. Table 4.1 provides an example of an LHS for a 10 run series with 3 inputs uniformly distributed between 0 and 10.

Sample Run	Variable 1	Variable 2	Variable 3
1	0.9	3.4	4.1
2	2.3	5.7	5.3
3	9.5	2.9	9.4
4	4.5	6.4	2.0
5	7.3	7.1	3.1
6	3.2	9.9	6.3
7	5.1	0.2	8.4
8	6.9	1.7	7.7
9	1.4	8.4	0.5
10	8.8	4.7	1.1

Table 4.1 Sample LHS

4.3 Scenario Identification

4.3.1 Measures of Merit for Scenarios

Choosing among box sets requires measures of the quality of any box and box set. The traditional scenario planning literature emphasizes the need to employ a small number of scenarios, each explained by a small number of "key driving forces," lest the scenario users become confused or overwhelmed by complexity [24]. In addition to this desired simplicity, the quantitative algorithms employed here seek to maximize the explanatory power of the boxes, that is, their ability to accurately differentiate among the cases of interest and the other cases in the database. These characteristics suggest three useful measures of merit for scenario discovery [15].

To serve as a useful aid in decision-making, a box set should capture a high proportion of the total number of policy-relevant cases (high coverage), capture primarily policy- relevant

cases (high density), and prove easy to understand (high interpretability). We define and justify these criteria as follows:

Coverage measures how completely the scenarios defined by box set B capture the cases of interest (I_s) and is analogous to the "sensitivity" or "recall" in the classification and information retrieval fields. With binary output, coverage is simply the ratio of the total number of cases of interest in the set of scenarios B to the total number of cases of interest, that is,

$$\text{Coverage} = \sum_{x_i \in B} y'_i \Big/ \sum_{x_i \in x^I} y'_i \qquad (4.1)$$

Where $y'_i = 1$ if $x_i \in I_s$ and $y'_i = 0$ otherwise.

Density measures the purity of the scenarios and has analogues with "precision" or "positive predictive value" in other fields. With binary output, density can be expressed as the ratio of the total cases of interest in a scenario to the number of cases in that scenario, that is,

$$\text{Density} = \sum_{x_i \in B} y'_i \Big/ \sum_{x_i \in B} 1 \qquad (4.2)$$

Decision makers should find this coverage measure important because they would like the scenarios to explain as many of the cases of interest as possible.

Interpetability measures the ease with which decision makers can understand a box set and use it to gain insight about their decision analytic application. This measure is thus highly subjective, but we can nonetheless approximate it quantitatively by reporting the number of boxes in a box set and the maximum number of model input parameters constrained by any box, equivalent to the size of the set L above. Based on the experience reported by the traditional scenario planning literature [24], a highly interpretable box set should consist of on the order of three or four boxes, each with on the order of two or three constrained parameters.

An ideal set of scenarios would combine high density, coverage, and interpretability. Unfortunately, these measures generally compete, so that increasing one typically comes at the expense of another. There are often tradeoffs between different measures. For instance, increasing coverage often means decreasing the density. Increasing interpretability by

constraining fewer parameters can increase coverage but typically decreases density. For a given dataset, these three measures define a multi-dimensional efficiency frontier. The scenario discovery analysis takes all of these measures into account. The procedure illustrated in Figure 3.1 also envisions that users interactively employ a scenario-discovery algorithm to generate alternative box sets at different points along this frontier and then choose that set most useful for the decision analytic application.

4.3.2 Patient Rule Induction Method

In this section, we will introduce PRIM algorithm and how it works in the scenario discovery analysis. By the three measures of merits, we can apply PRIM to identify a set of scenarios that represent policy-relevant regions of interest.

The main box construction method in PRIM works from the top down, starting with a box containing all of the data. The box is compressed along one face by a small amount, and the observations then falling outside the box are peeled off. The face chosen for compression is the one resulting in the largest box mean, after the compression is performed. Then the process is repeated, stopping when the current box contains some minimum number of data points.

This process is illustrated in Figure 4.1. There are two classes in the figure, indicated by the blue (class 0) and red (class 1) points. The procedure starts with a rectangle (broken black lines) surrounding all of the data, and then peels away points along one edge by a pre-specified amount in order to maximize the mean of the points remaining in the box. Starting at the top left panel, the sequence of peelings is shown, until a pure red region is isolated in the bottom right panel. The iteration number is indicated at the top of each panel. There are 200 data points uniformly distributed over the unit square. The color-coded plot indicates the response Y taking the value 1 (red) when $0.5 < X_1 < 0.8$ and $0.4 < X_2 < 0.6$ and zero (blue) otherwise. The panels shows the successive boxes found by the top-down peeling procedure, peeling off a proportion $\alpha = 0.1$ of the remaining data points at each stage.

Figure 4.2 shows the mean of the response values in the box, as the box is compressed.

After the top-down sequence is computed, PRIM reverses the process, expanding along any edge, if such an expansion increases the box mean. This is called pasting. Since the top-down procedure is greedy at each step, such an expansion is often possible.

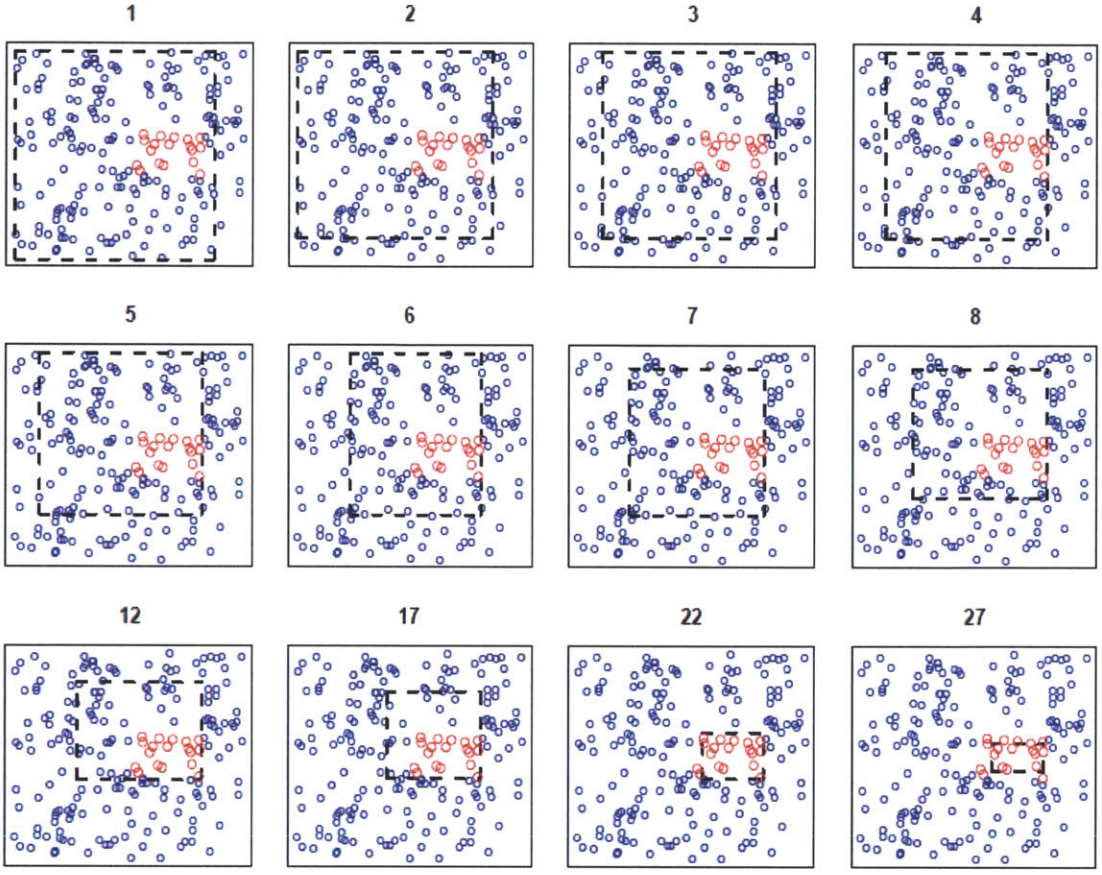


Figure 4.1 Illustration of PRIM Algorithm [36]

The result of these steps is a sequence of boxes, with different numbers of observation in each box. Cross-validation, combined with the judgment of the data analyst, is used to choose the optimal box size.

Denote by B_1 the indices of the observations in the box found in step 1. The PRIM procedure then removes the observations in B_1 from the training set, and the two-step process—top down peeling, followed by bottom-up pasting—is repeated on the remaining dataset. This entire process is repeated several times, producing a sequence of boxes $B_1, B_2, \dots, B_k$. Each box is defined by a set of rules involving a subset of predictors like

$$(a_1 \leq X_1 \leq b_1) \text{ and } (b_1 \leq X_3 \leq b_2).$$

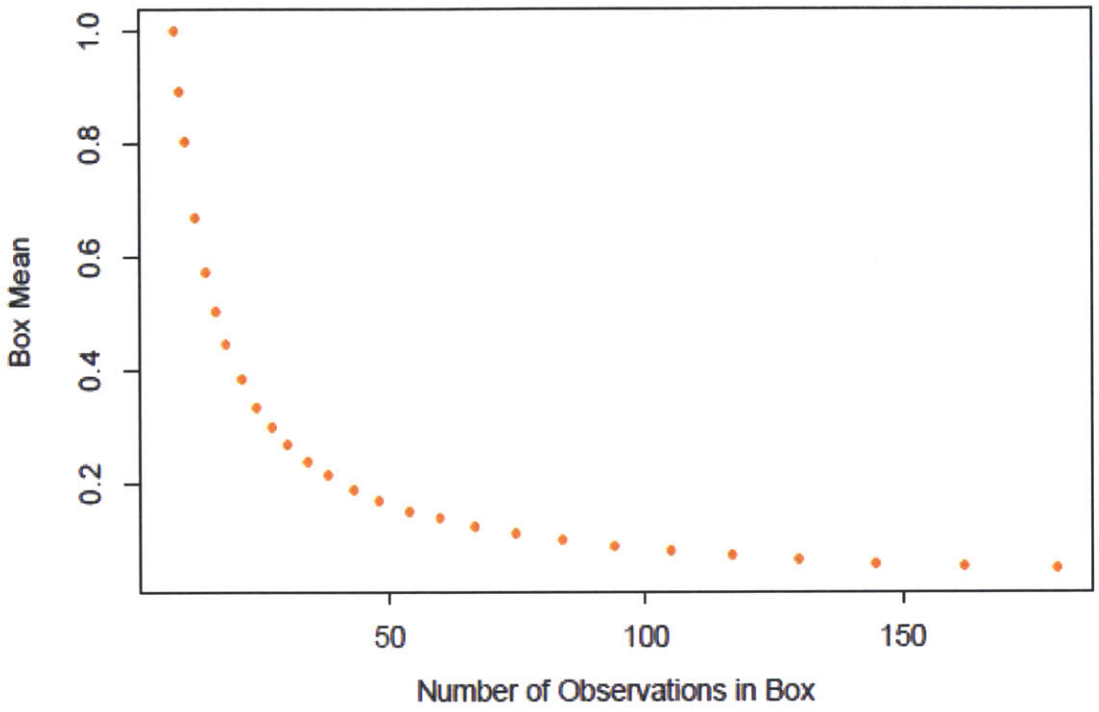


Figure 4.2 Box Mean as a Function of Number of Observations in the Box [36]

A summary of the PRIM procedure is given below.

Step 1. Start with all of the training data, and a maximal box containing all of the data.

Step 2. Consider shrinking the box by compressing one face, so as to peel off the proportion α of observations having either the highest values of a predictor X_j, or the lowest. Choose the peeling that produces the highest response mean in the remaining box. (Typically α =0.05 or 0.10.)

Step 3. Repeat step 2 until some minimal number of observations (say 10) remain in the box.

Step 4. Expand the box along any face, as long as the resulting box mean increases.

Step 5. Steps 1–4 give a sequence of boxes, with different numbers of observations in each box. Use cross-validation to choose a member of the sequence. Call the box B_1.

Step 6. Remove the data in box B_1 from the dataset and repeat steps 2–5 to obtain a second box, and continue to get as many boxes as desired.

4.4 Scenario Evaluation with Diagnostics

As stated in the section 3.3, researchers applied PRIM and CART to datasets with regions of known shape to test the algorithms' strengths and weaknesses for scenario discovery. These tests suggest that both algorithms can perform the scenario discovery task even for relatively complex shapes, but that under some conditions they make several types of errors.

In particular, PRIM may needlessly slice off the end of a parameter's range, incorrectly suggesting that a proposed policy may prove sensitive to even a small variation in some parameter. The potential for such errors is troubling because a policy can be truly sensitive to small variations, as was the case, for instance, with IEUA where scenario discovery properly revealed that the agency's plan was very sensitive to any change in the amount of rain captured as groundwater. In addition, when applied to low-dimensional shapes in high-dimensional data PRIM may erroneously constrain extraneous parameters that do not in fact predict the cases of interest [15].

Such potential errors highlight the importance of using diagnostic tools to evaluate the statistical significance of the parameter constraints proposed by the scenario discovery algorithm.

In the work by Swartz and Zegras [28], they didn't use any diagnostics to assess the scenarios that are data-mined from the simulation results. While some other researchers proposed some simple statistical diagnostics to assess these scenarios, they proposed that users employ a quasi p-value test and resampling test for this purpose [15]. These techniques, commonly employed in the field of statistical learning to diagnose the quality of models fit to data, prove appropriate because the PRIM errors result from the finite and stochastic sampling of the LHS experimental design. Given this stochasticity, the scenario definitions can be considered statistical models with potentially nonzero bias and variance about the true model.

These two tests help detect the errors described above by estimating the probability that any particular parameter constraint is due to chance and by examining the extent to which the scenario definition varies over multiple samples of the original data [15].

The details of both tests are given as follows. We basically follow the same test procedure developed by Bryant and Lempert [15].

4.4.1 Resampling Test

This diagnostic tool evaluates a scenario definition by assessing how frequently the same definition arises from different samples of the same database. The resampling test runs the algorithm on multiple subsamples of the original dataset and notes which of the parameter constraints consistently emerge as important in the resulting scenario definitions.

PRIM complicates automation of this technique because the algorithm is fundamentally interactive, requiring the user to select from a large number of options with different combinations of coverage and density. We thus generate two sets of "reproducibility statistics" – one in which the algorithm generates a scenario matching as closely as possible the coverage of the original box, and one in which it matches the density.

These two criteria will often but not always generate identical results. Ideally for both criteria the parameters constrained in the initial scenario definition will also be constrained in 100 percent of the samples, while the unconstrained parameters will remain so in all the samples.

4.4.2 Quasi-p-value Test

This diagnostic tool uses what is essentially a p-value test to estimate the likelihood that PRIM constrains some parameter purely by chance. Consider a single box β within box set B, defined by limiting constraints on parameters in the set L_β, and which contains H high- value $(y_1' = 1)$ cases out of a total of T cases. To compute this quasi-p-value consider the box β_j defined by constraints on all parameters in L_β except parameter $x_j \in L_\beta$. This box contains T_{-j} total cases and H_{-j} cases of interest, with $T_{-j} \geq T$ and $H_{-j} \geq H$. We then consider the null hypothesis that the cases of interest within β_{-j} are distributed among all cases in β_{-j} according to a binomial distribution with $p(1) = H_{-j}/T_{-j}$. The "qp-value" test thus answers the question: what is the probability that T points drawn from the above binomial distribution would have H or more high valued points? When the ratio H_{-j}/T_{-j} is close to H/T this number is high, the additional contribution of parameter rj is low, and thus possibly due to chance. The opposite is the case when H/T is much larger than H_{-j}/T_{-j}.

Bryan and Lempert [15] call this a quasi-p-value test, because contingent on sampling, it is not an entirely accurate model of the system, since it does not take into account spatial proximity

and its interaction with whatever algorithm is defining the box. Nevertheless, the relative magnitudes of the quasi-p-values provide useful information for comparing parameter relevance.

Due to the limitation of data samples we have, we only employed quasi p-value test in our study.

These diagnostic techniques, combined with the measures of coverage, density and interpretability, help users achieve a more complete understanding of the scenarios and their ability to characterize the cases of interest in the database.

4.5 Summary

Scenario discovery aims to identify sets of future states of the world that shows the vulnerabilities in proposed policies and to describe these scenarios for decision makers and other stakeholders.

There are four steps when implementing scenario discovery approach. In the first step, we specifies a simulation model whose output are based on the proposed policy and a set of input parameters that may bring uncertain. Criterion is chosen to distinguish the policy-relevant regions of interest in the output. We then introduced how to use Latin-hypercube sampling technique to incorporate the uncertainty from the input parameters.

In the second step, patient rule induction method algorithm is applied to the resulting database generated from simulation described in the first step and to identify candidate scenarios that provide a good description of these regions of interest. Several measures of merits are described and present us the tradeoff among coverage, density and interpretability that users may face in choosing scenarios.

In the third step, two simple statistical diagnostics are proposed to evaluate the scenarios from the second step. They are resampling test and quasi-p-value test. By these diagnostics, we can evaluate the selected scenarios and decide which scenario will eventually be chosen.

As previously stated, the whole procedure is adaptive. User can also go from the third step to the first or second step, which means that if use could reselect other scenarios based on the diagnostics in the evaluation step. Different options are given to users showing the tradeoffs among them. By evaluating the proposed scenarios, users can select scenarios.

Chapter 5

Application: New Transit-orient Policy Performance Evaluation

5.1 Background

The city state of Singapore is the second most densely populated country in the world [38]. Since Singapore is a small island with a high population density, the number of private cars on the road is restricted so as to curb pollution and congestion. Car buyers must pay for duties one-and-a-half times the vehicle's market value and bid for a Singaporean Certificate of Entitlement (COE), which allows the car to run on the road for a decade. Car prices are generally significantly higher in Singapore than in other English-speaking countries and thus only one in 10 residents owns a car [39].

Most Singaporean residents travel by foot, bicycles, bus, taxis and train (MRT or Light Rail Transit). Two companies run the public bus and train transport system – SBS Transit and SMRT Corporation. There are almost a dozen taxi companies, who together put out 25,000 taxis on the road. Taxis are a popular form of public transport as the fares are relatively cheap compared to many other developed countries [39].

The policies of the Land Transport Authority are meant to encourage the use of public transport in Singapore. The key aims are to provide an incentive to reside away from the Central district, as well as to reduce air pollution. Singapore has a Mass Rapid Transit (MRT) and Light Rail Transit (LRT) rail system consisting of five lines. There is also a system of bus routes throughout the island, most of which have air conditioning units installed due to Singapore's tropical climate. A contactless smartcard called the EZ-link card is used to pay bus and MRT fares. The public transportation system is the most important means of transportation to work and to school for Singaporeans. According to the Singapore 2000 Census, 52% of Singaporean residents (excluding foreigners) use public transportation for their work commute, 42% use private transportation modes. 42% of school-going residents use public transportation to go to school. 25% use private transportation modes [38].

The Land Transport Authority (LTA) in Singapore reports that roads take up 12% of its total land area [40]. LTA also estimates that demands for land travel will increase by 60%, from the current 8.9 million daily trips to 14.3 million by 2020 [40]. To avoid severe congestion, LTA

plans that much of the future growth in travel demand will be served by public transportation, so that by 2020, 70% of all morning peak hour trips use public transportation [40].

5.2 Problem Statement

Marina Bay is a bay near Central Area in the southern part of Singapore, and lies to the east of the Downtown Core. Marina Bay is set to be a 24/7 destination with endless opportunities for people to "explore new living and lifestyle options, exchange new ideas and information for business, and be entertained by rich leisure and cultural experiences" [41]. It is here where the most innovative facilities and infrastructure such as the underground "Common Services Tunnel" are built and where mega activities take place [41].

There are currently 7 rail stations: City Hall, Raffles Place, Marina Bay, Bayfront, Downtown, Esplanade and Promenade serving Marina Bay. By 2020, the 360 hectares Marina Bay will boast a comprehensive transport network as Singapore's most rail-connected district [41]. By 2018, the Marina Bay district will more than six MRT stations, all no more than five minutes of each other [41]. A comprehensive pedestrian network including shady sidewalks, covered walkways, underground and second-story links will ensure all-weather protection and seamless connectivity between developments and MRT stations [41]. Within greater Marina Bay, water taxis will even double up as an alternative mode of transportation [41].

As a big tourism attraction, there are always needs to improve its public transit system. Although Singapore plans to expand its bus and rail rapid transit networks, future infrastructure funding is uncertain. The government must make the best possible use of existing transit facilities. Marina Bay district is shown in Figure 5.1. It is an area of reclaimed land in the southern part of Singapore. It lies to the east of the Downtown Core. It has mixed residential and business land use. At the center of the area there is a large convention and exhibition center with adjacent hotels and related facilities. The areas close to the coast on the east and especially the south end are leisure destinations with several tourist attractions, such as the Esplanade, floating stadium and Singapore Flyer. The western part of the Marina Bay, adjacent to downtown, has mostly commerce and shopping uses. The area has plans for considerable growth in the next decade.

Figure 5.1 Map of Marina Bay [42]

One of the policies that can help improve the quality of service of public transportation, and attract ridership, is to develop transit priority measures including implementation of bus lanes and providing bus-priority at signalized intersections.

Transit signal priority and bus lanes can play an important role as a foundation for future rapid transit corridors, building corridor-level ridership by improving service until the City can afford (or justify) a major investment in new infrastructure. The city needs to propose a plan that dedicates a section to transit priority and includes other supporting policies.

From Figure 5.1, there is one highway called Nicoll Highway. One lane of this highway would be converted into a bus lane. No other vehicles could drive on this lane except for buses.

In general, the traffic demands in the Marina Bay district are not stable and always fluctuate over time. It is difficult for traditional policy analysis to consider the uncertain traffic demands in this urban network. Our objective is to determine the policy performance under the condition of deep uncertainty of traffic demand.

Scenario discovery described in Chapter 4 will support this decision making process and gives us the relationship between the uncertain traffic demand and the policy performance. The potential impact of this proposed policy will be illustrated in our study.

In the following sections, we will give detailed introduction of how we employ the scenario discovery analysis under the current problem statement. In section 5.4, we will describe the simulation software and model we used in the study. In addition, we will introduce how the input data are prepared. In section 5.5, the whole applications of scenario discovery are illustrated including data generation from simulation model, identifying candidate scenarios, and assessing the scenarios with statistical diagnostics. In section 5.6, the results are summarized and conclusions of this study are shown.

5.3 Framework of Scenario Discovery Application

In this section, we will talk about how the scenario discovery approach will be implemented in this empirical study.

First, we need a built computer simulation models from the existing data of Marina Bay district. Recall equation 3.1 $y = f(s, \boldsymbol{x})$. In this model, $\boldsymbol{x}$ is a vector of input parameters. In this case study, these varying parameters are mainly the traffic demands of different origin and destination pairs. y is the simulation output of interest which is contingent on a vector of input data $\boldsymbol{x}$ representing a particular point in a M-dimensional space of uncertain model input parameters, s is the policy makers' action, which is to implement transit-oriented policy or not.

Recalling what we have illustrated in chapter 3, there are mainly four steps in this approach. Given a simulation model of Marina Bay district and existing data, first we use Latin Hypercube Sampling (LHS) to sample in the space of all combination of input variable distributions. We use

the LHS samples as input rather than using the total combinations of input variables which might be impossible in high dimensional cases. Second, after running simulation model $y = f(s, \boldsymbol{x})$ with and without transit policy numerous times, we have corresponding output of the input variables and by some criterions; we classify different outputs as failure or non-failure. Then we use PRIM algorithm to identify a set of regions of combinations of input variables that result in failure policy. These regions are scenarios in scenario discovery context. Finally we will assess the identified scenarios by some statistical diagnostics proposed in chapter 4.

5.4 Description of Simulation

A microscopic simulation-based laboratory known as MITSIMLab [43] is used for the simulation. The input data including the traffic demand in Marina Bay network and other input parameters such as transportation network are from Future Urban Mobility program. Since the original input and output data are not exactly designed for scenario discovery analysis, some data processing work has been done. In section 5.3.1, we will briefly introduce the MITSIMLab and in section 5.3.2, we will describe briefly about how we prepare the simulation input and process the raw data from the simulation. More detailed descriptions can be found in Appendix B and C.

5.4.1 MITSIMLab

In this section, we will introduce briefly about MITSIMLab. Most of the materials in this section are adapted from the user manual and the website of Intelligent Transportation Systems Program [43].

MITSIMLab is a simulation-based laboratory that was developed for evaluating the impacts of alternative traffic management system designs at the operational level and assisting in subsequent design refinement. Examples of systems that can be evaluated with MITSIMLab include advanced traffic management systems (ATMS) and route guidance systems. MITSIMLab was developed at the MIT Intelligent Transportation Systems (ITS) Program. Professor Moshe Ben-Akiva, Director of the ITS Program at MIT, and Dr. Haris Koutsopoulos, from the Volpe Center, were co-principal investigators in MITSIMLab's development. Dr. Qi Yang, of MIT and Caliper Corporation, was the principal developer.

MITSMLab is a synthesis of a number of different models and has the following characteristics: represents a wide range of traffic management system designs; models the

response of drivers to real-time traffic information and controls; and incorporates the dynamic interaction between the traffic management system and the drivers on the network.

The various components of MITSIMLab are organized in three modules:

1. Microscopic Traffic Simulator (MITSIM)

2. Traffic Management Simulator (TMS)

3. Graphical User Interface (GUI)

A microscopic simulation approach, in which movements of individual vehicles are represented, is adopted for modeling traffic flow in the traffic flow simulator (MITSIM). This level of detail is necessary for an evaluation at the operational level. The Traffic Management Simulator (TMS) represents the candidate traffic control and routing logic under evaluation. The control and routing strategies generated by the traffic management module determine the status of the traffic control and route guidance devices. Drivers respond to the various traffic controls and guidance while interacting with each other.

The role of MITSIM is to represent "the world." Traffic and network elements are represented in detail in order to capture the sensitivity of traffic flows to the control and routing strategies. The main elements of MITSIM are:

1. Network Components: The road network, along with the traffic controls and surveillance devices, are represented at the microscopic level. The road network consists of nodes, links, segments (links are divided into segments with uniform geometric characteristics), and lanes.

2. Travel Demand and Route Choice: The traffic simulator accepts as input time-dependent origin to destination (OD) trip tables. These OD tables represent either expected conditions, or are defined as part of a scenario for evaluation. A probabilistic route choice model is used to capture drivers' route choice decisions.

3. Driving Behavior: The origin/destination flows are translated into individual vehicles wishing to enter the network at a specific time. Behavior parameters (such as desired speed, aggressiveness, etc.) and vehicle characteristics are assigned to each vehicle/driver combination.

MITSIM moves vehicles according to car-following and lane-changing models. The car-following model captures the response of a driver to conditions ahead as a function of relative speed, headway and other traffic measures. The lane changing model distinguishes between mandatory and discretionary lane changes. Merging, drivers' responses to traffic signals, speed limits, incidents, and toll booths are also captured. Rigorous econometric methods have been developed for the calibration of the various parameters and driving behavior models.

The traffic management simulator mimics the traffic control system under evaluation. A wide range of traffic control and route guidance systems can be evaluated, such as:

1. Ramp control

2. Freeway mainline control

    2.1 lane control signs (LCS)

    2.2 variable speed limit signs (VSLS)

    2.3 portal signals at tunnel entrances (PS)

3. Intersection control

4. Variable Message Signs (VMS)

5. In-vehicle route guidance

TMS has a generic structure that can represent different designs of such systems with logic at varying levels of sophistication (from pre-timed to responsive).

The simulation laboratory has an extensive graphical user interface that is used for both, debugging purposes and demonstration of traffic impacts through vehicle animation.

MITSIMLab has been applied in the city of Stockholm, Sweden, for research funded by the City of Stockholm Real Estate and Traffic Administration (GFK), which is responsible for traffic planning and operations within the city. Initially, MITSIMLab was evaluated for its applicability in Stockholm. As part of the project, MIT enhanced the simulation models and

calibrated the model parameters to match the observed conditions in Stockholm. Validation of the simulation model was performed by the Royal Institute of Technology (KTH) in Stockholm.

The network chosen for the evaluation was a ring network around Brunnsviken, north of Stockholm. The network has both freeway and urban sections, and it operates under heavy congestion during the peak periods. MITSIMLab was calibrated by MIT based on traffic data from observations in 1999. The calibrated MITSIMLab was then used to simulate the network conditions in 2000, and validation was performed by KTH using queue lengths and point-to-point travel times within the network. The validation showed that MITSIMLab was able to replicate the actual measurements quite well, and it was concluded that MITSIMLab should be recommended for use in Swedish cities.

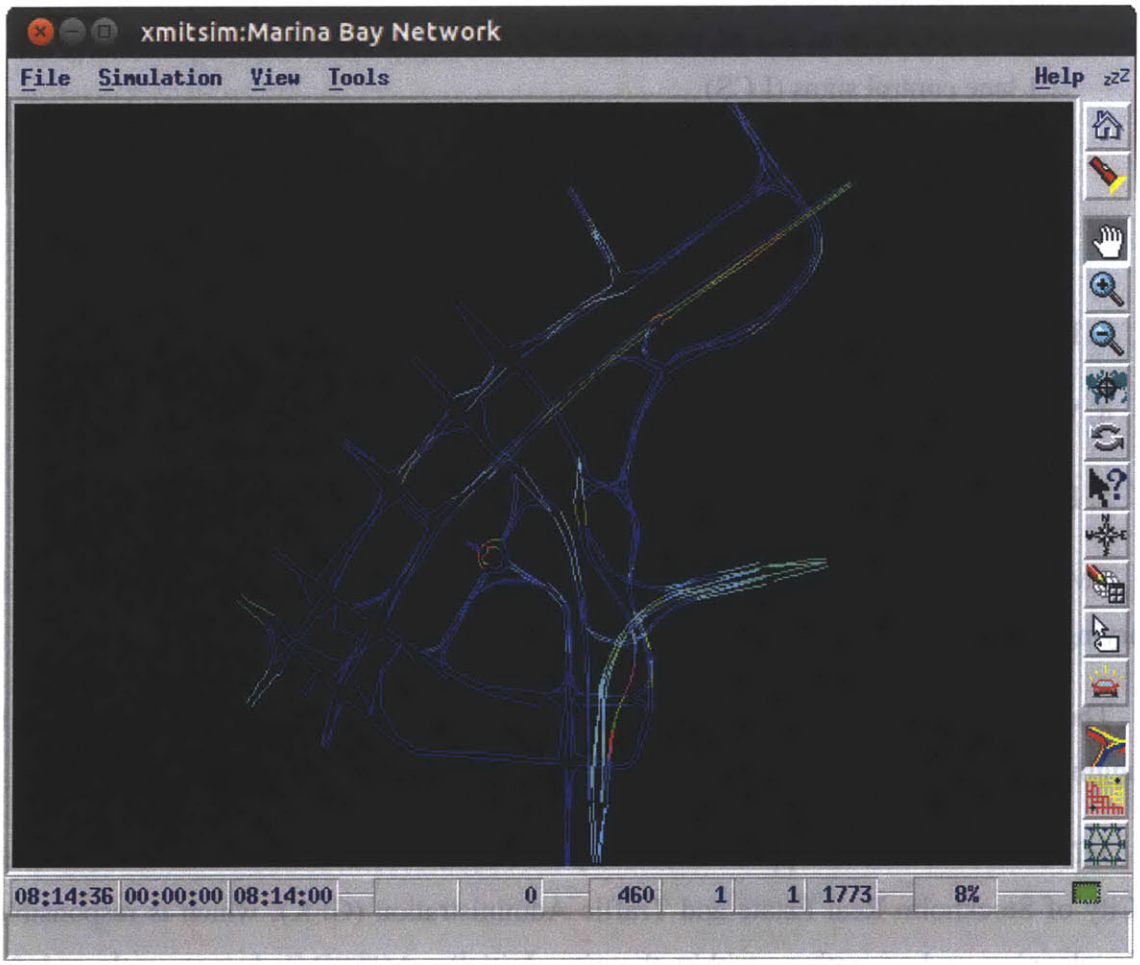


Figure 5.2 GUI of MITSIMLab with Marian Bay Network Loaded

We will use MITSIMLab as simulation platform to implement scenario discovery analysis and show how the uncertainty in traffic demands will impact the performance of the proposed policy. Figure 5.2 shows the GUI (graphical user interface) of MITSIMLab with the Marina Bay network loaded. In Figure 5.2, different colors mean different traffic density in that region.

5.4.2 Data Preparation and Processing

In this section, we will describe the preparation of input files and some assumptions we made under which we prepare the input files.

Figure 5.3 Marina Bay Network under BL in MITSIMLab

There are mainly five types of input files in MITSIMLab including master files, parameter files, network file, demand file, and transit input files if there is public transit system in the loaded network. In general, master files are mostly fixed. For each lane in the real network, the network file includes all the lane information and there is some numbers associated with each lane denoting its functionality (whether it is a bus lane or not). In order to convert one lane into

bus lane, we rewrite the network file and convert the specified lane into bus lane. Thus, we have two cases with or without bus lanes. We denotes them as BL (with bus lane converted) and NBL (without bus lane converted) in the following part of the thesis. Figure 5.3 shows the Marina Bay network under BL.

In addition, we made some transit input files according to the proposed policy and real network. In the demand file, there are numbers associated with each OD (origin and destination) pairs denoting the traffic demand at this time. Figure 5.4 shows the OD nodes in Marina Bay network in MITSIMLab.

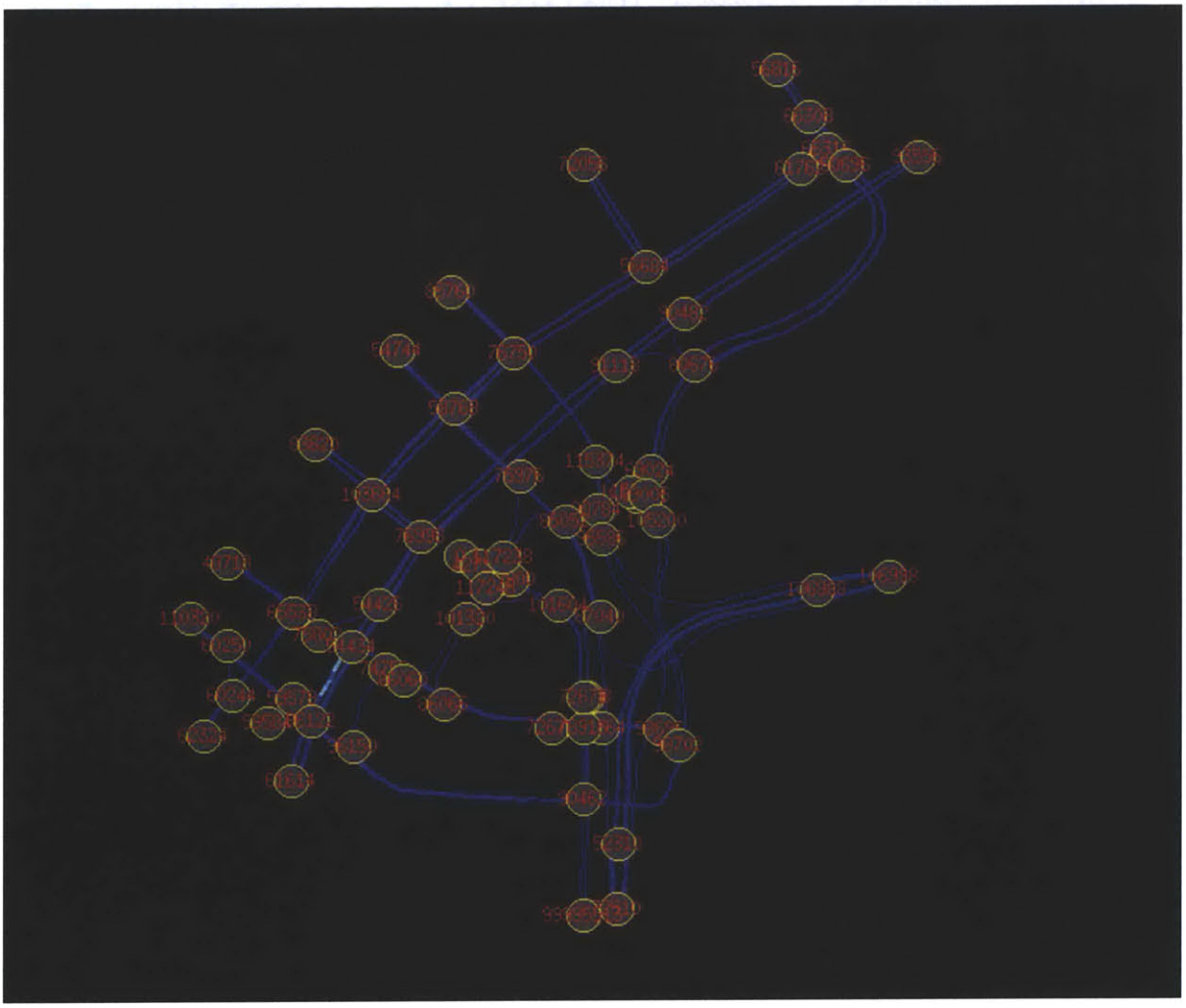


Figure 5.4 OD Nodes in Marina Bay Network in MITSIMLab

The study period of simulation runs are AM peak period (8:00-9:00 AM) in Singapore. With the original demand data of each OD pair, we treat the distribution of demand of each OD

as uniform distribution with original demand value as its mean. The maximum of demand goes up to 60% higher than the original demand and the minimum goes 60% less. By certain experimental design which we will describe in next section, we sample from these demand distributions.

In addition, we treat demand as the only variables that are uncertain, which may not be true in real case. But since demand will be the dominant factor impacting the average travel time or other output variables we use for performance evaluation, this assumption may not hurt our analysis result badly.

For each sample, we run the simulation model only once. Due to the long computational time for each run, only 100 runs are made for BL and NBL case each. In next section, we will describe the output data processing.

Since the simulation model MITSIMLab is a stochastic model, we need to run each model with the same model input many times to control the randomness of output generation. The discussion about this issue will be made in section 5.6

After we run the simulation model in MITSIMLab for designed cases, some output files are generated. Since our goal is to evaluate the performance of proposed policy, some variables are computed from the raw output data. The Table 5.1 shows the descriptions of these variables. The detailed output result tables are attached in Appendix C.

	Variable Names	Descriptions
Input Variables	X1	Proportion of total expected demands whose destination are in the south-west of the Marina Bay area
	X2	Proportion of total expected demands whose destination are in the north of the Marina Bay area
	X3	Proportion of total expected demands whose destination are in the south-east of the Marina Bay area
Output Variables (seconds)	BCT	Total car travel time with policy implemented
	NCT	Total car travel time without policy implemented
	BBT	Total bus travel time with policy implemented
	NBT	Total bus travel time without policy implemented
	BVT	Total vehicle (car + bus) travel time with policy implemented
	NVT	Total vehicle (car + bus) travel time without policy implemented
	BCPT	Total car passenger travel time with policy implemented
	NCPT	Total car passenger travel time without policy implemented
	BBPT	Total bus passenger travel time with policy implemented
	NBPT	Total bus passenger travel time without policy implemented
	BVPT	Total vehicle (car + bus) passenger travel time with policy implemented
	NVPT	Total vehicle (car + bus) passenger travel time without policy implemented
	YTEST	Binary variables (0,1) denote cases of interest: 1 means output of interest, 0 means not of interest; used for illustrating how the algorithm works in Appendix 3

Table 5.1 Descriptions of Output Variables Processed from MITSIMLab

Some thresholds are made to distinguish failure regions (scenarios) and details are illustrated in section 5.4.

Some assumptions are made when dealing with output data and preparing input data. Some may be loosed in the future study. Since the MITSIMlab is not designed exactly for the use of scenario discovery, we did some preliminary work before what we stated in previous sections. Several computer programs are written in Java to handle the raw input data and output data. Some of the codes are attached in Appendix B. After some computation in Excel, we have

transformed the raw data into clear data tables. The main input and output data are attached in Appendix C.

5.5 Application of Scenario Discovery

5.5.1 Data Generation from Simulation

Based on the discussion in chapter 3, Latin-Hypercube-Sampling experimental design is employed to sample data from the demand distribution.

To deal with dimensionality, since we have around forty numbers of OD demands, we categorized these OD demands into three groups by different destinations, which are southwest, southeast and northeast. Figure 5.5 shows the OD groups in Marina Bay network.

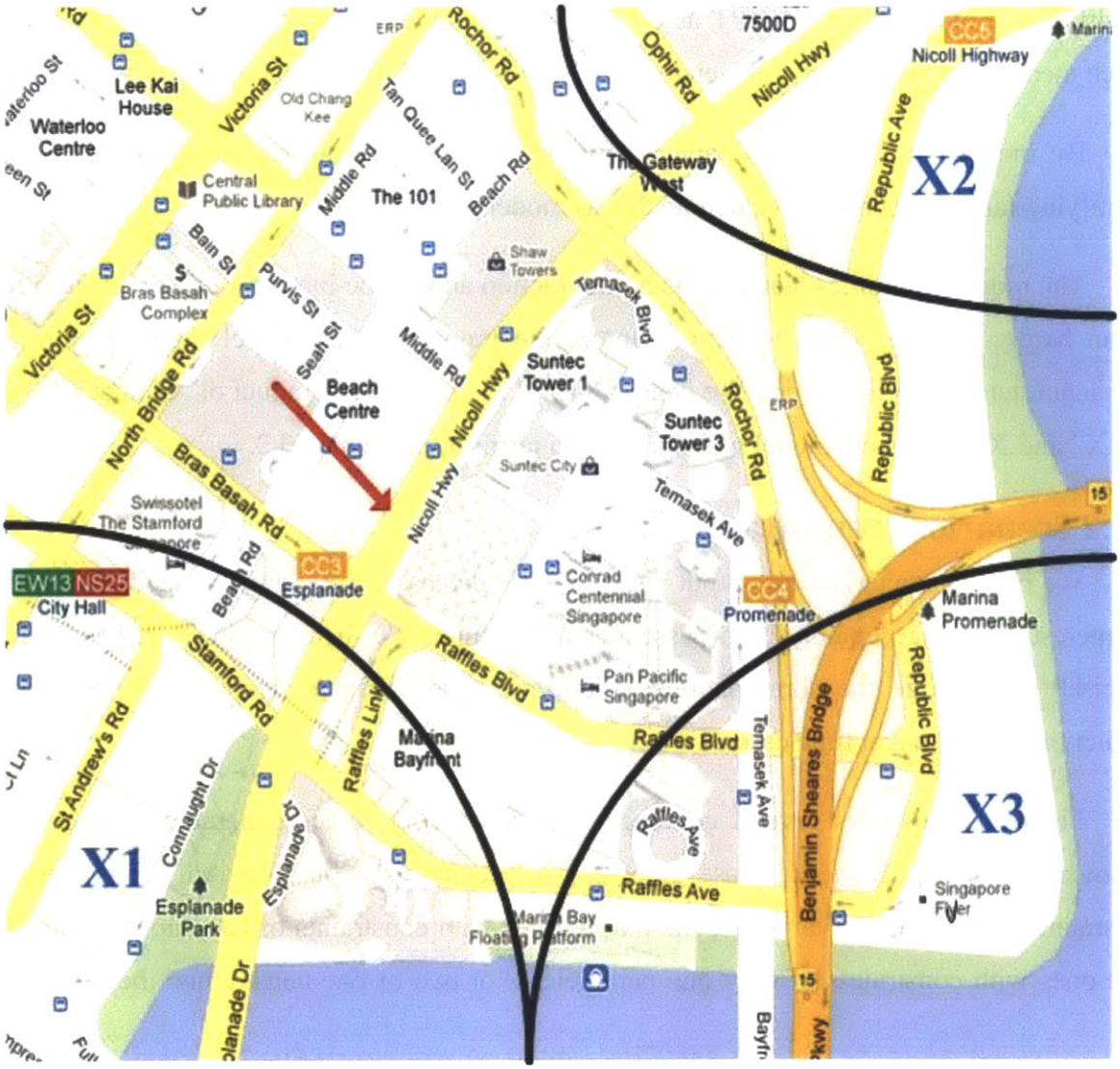


Figure 5.5 OD Groups in Marina Bay Network

X1, X2, and X3 showed the sum of demands with destinations in its region. Clearly, these three variables may be closely related to the performance of the transit-oriented policy. We also assume the distributions of them to be uniform distribution for simplicity.

We use a Latin Hypercube sample (LHS) to create an experimental design over the space defined by these three uncertain model input parameters. Running this sample through the simulation creates a database that explores the implications all combinations of the full range of expert opinion about the values of the three uncertain parameters.

Assuming X1, X2 and X3 are independent, we now have X1, X2, and X3 as input variables and adding bus lane as action s and a simulation model f. In the current study, we use the difference of BVPT and NVPT as output y. Zero is the threshold to classify policy failure, which means if in BL the total passenger travel times go higher than that in NBL, the policy fails.

By model $y = f(\boldsymbol{x}, s)$ in Chapter 3, we run the simulation and get the output data for identifying failure scenarios. We denote this as model 1.

The model 1 assumes that there is no interaction among the input parameters. But in reality, it can hardly be true. Thus, to better capture the property of the travel time, we introduced interaction terms into the model. We denote X12, X13, X23 as the product of X1 and X2, X1 and X3, X2 and X3. This new model with interaction terms is called model 2.

5.5.2 Scenarios Identification

We next characterize the output values in this database, differentiating between the cases of interest with unacceptably high passenger travel time. We then use the PRIM algorithm to concisely summarize the combinations of uncertain model input parameter values that best predict these high travel time cases.

Figure 5.6 displays several coverage-density tradeoff curves generated by the scenario discovery toolkit from the database described in previous section. The red points mean with constraints of three input variables, the purple ones with constraints of two input variables, the blue ones with constraints of only one parameter. For two or one constraints, they thus do not

represent a complete or optimal search, but do serve to illuminate tradeoffs between the scenario quality measures of coverage, density and interpretability.

The algorithm starts from the 100% coverage and 30% density. A box representing a perfect scenario would be defined by constraints on only one or two parameters and would lie in the upper right-hand corner with 100% coverage and 100% density, and thus capturing all the cases of interest and excluding all the other cases. Since such a box is not available, users must choose one with the combination of coverage, density, and interpretability that best supports their decision application. In general, dimensionality increases with density and decreases with coverage, and both decrease with interpretability. For the purposes of this example, we initially consider Scenario 14, which uses four parameters to achieve 66% coverage and 73% density. After evaluating this scenario, one could still modify this choice, possibly improving interpretability by dropping parameters deemed less important or choosing another scenario entirely.

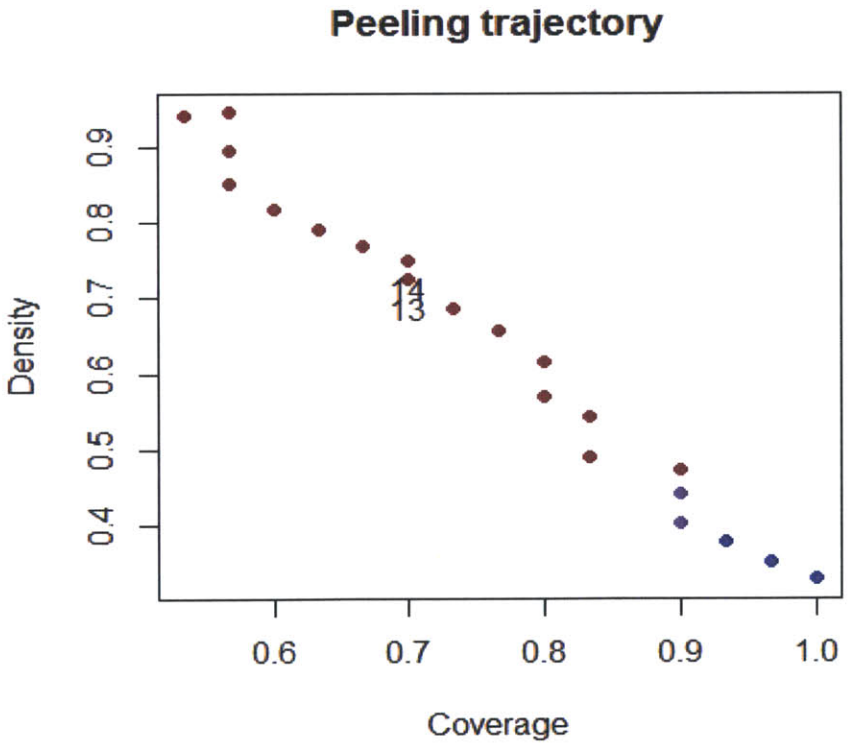


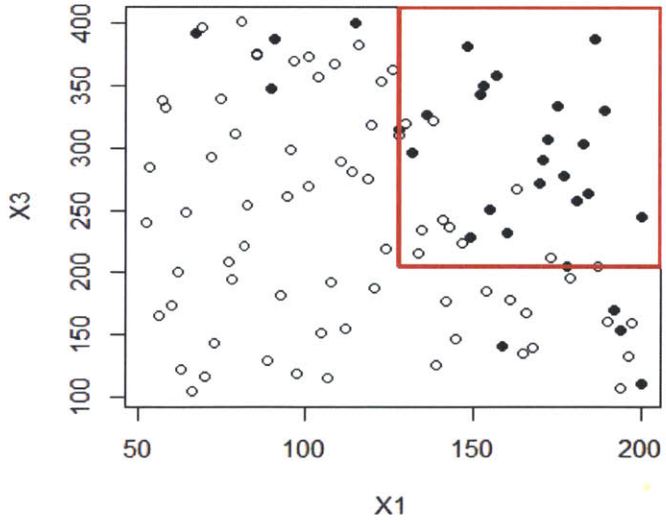
Figure 5.6 Peeling Trajectory for Model 1

Dimension	Constraints for input variables	Density	Coverage
1	X1 >128.0	30%	100%
2	X3 >205.0	52%	87%
3	X2> 250.5	67%	73%
4	X2< 483.0	NA	NA

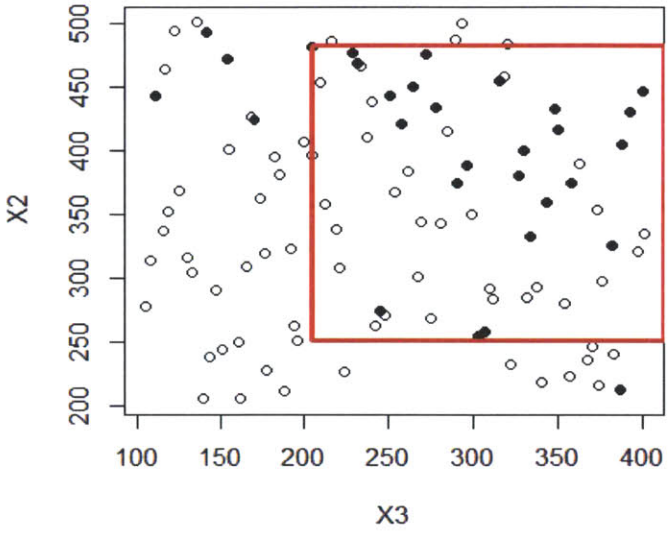
Table 5.2 Combination of Parameters Values in Scenario 14

The scenario includes potential future states of the world where X1 and X3 are at the upper half of their ranges, X2 is almost over all range of its lowest value to highest. Overall, 67% of the cases in the dataset that meet these three constraints have high costs (i.e., 67% density). Of all the high-cost cases in the dataset, 73% meet these three constraints (i.e. 73% coverage).

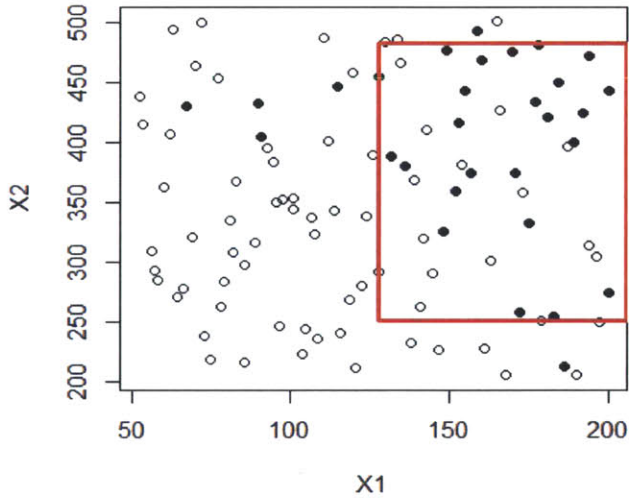
As shown in Table 5.2, PRIM also reports each parameter's marginal contribution to explaining the high travel time cases. With no parameters constraints the box would have 30% density and 100% coverage, since we have defined 30% of the cases in the database to have high travel time. After three input variables constraints are introduced, the density goes up and coverage goes down and it's a trade-off.



(a)



(b)



(c)

Figures 5.7 Visualization Results of PRIM in Model 1

Figure 5.7 illustrate cases in database plotted as function of a) first two parameters and b) first and second parameters and c) second and third parameters shown in Table 5.2.

Black and open dots show high travel time and lower travel time cases, respectively. Red lines show parameters values corresponding to the boundaries of Scenario 14. Figure 5.7a also

suggests that Scenario 14's lack of 100% coverage owes to a small number of high travel time cases with high X1 and low X3 or low X1 and high X3.

By normalizing all three parameters to one, we can have Table 5.3 and Figure 5.8 showing the failure regions in the space of input variables. We call scenarios failure clusters in Figure 5.8.

	X1	X2	X3
Total Space	-60% ~60%	-50% ~40%	-40% ~60%
Scenario 1	-2% ~60%	21% ~40%	-60% ~26%
Scenario 2	-50% ~30%	-7% ~40%	27% ~60%

Table 5.3 Scenarios in the Space of Input Variables in Model 1

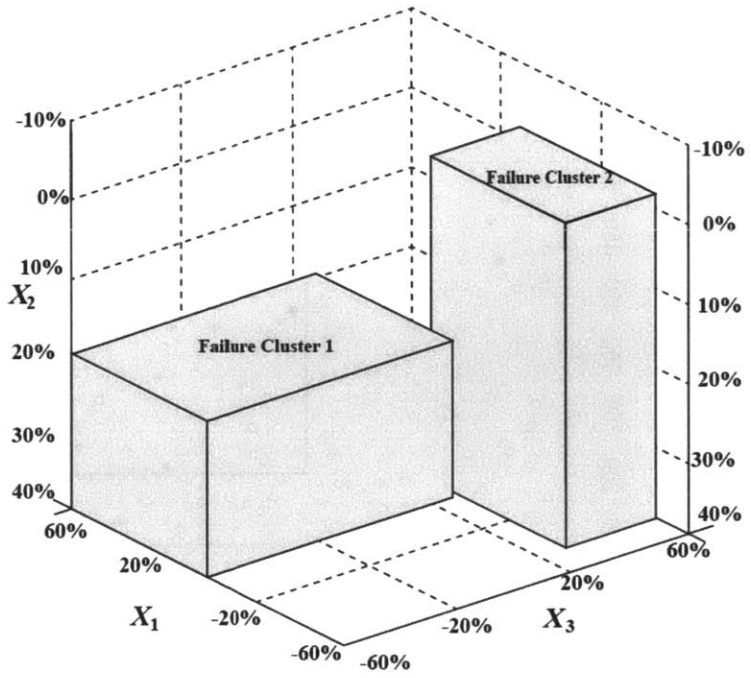


Figure 5.8 Failure Clusters in the Space of Input Variables

In model 2, we employed same methods and results are showed as follows. Figure 5.9a and 5.9b show the peeling trajectory of model 2.

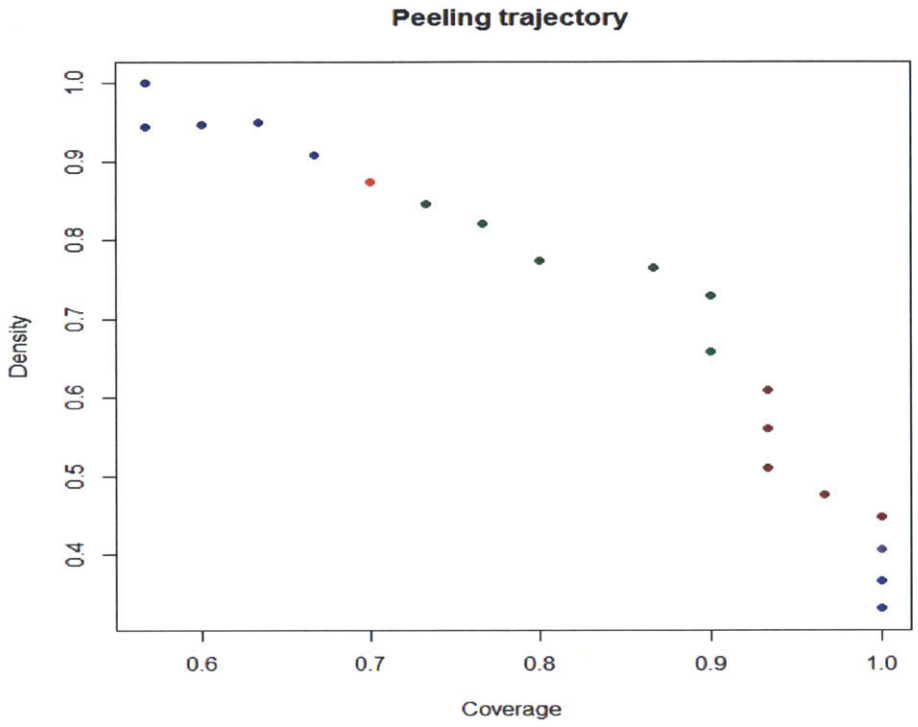


Figure 5.9a Peeling Trajectory of Model 2

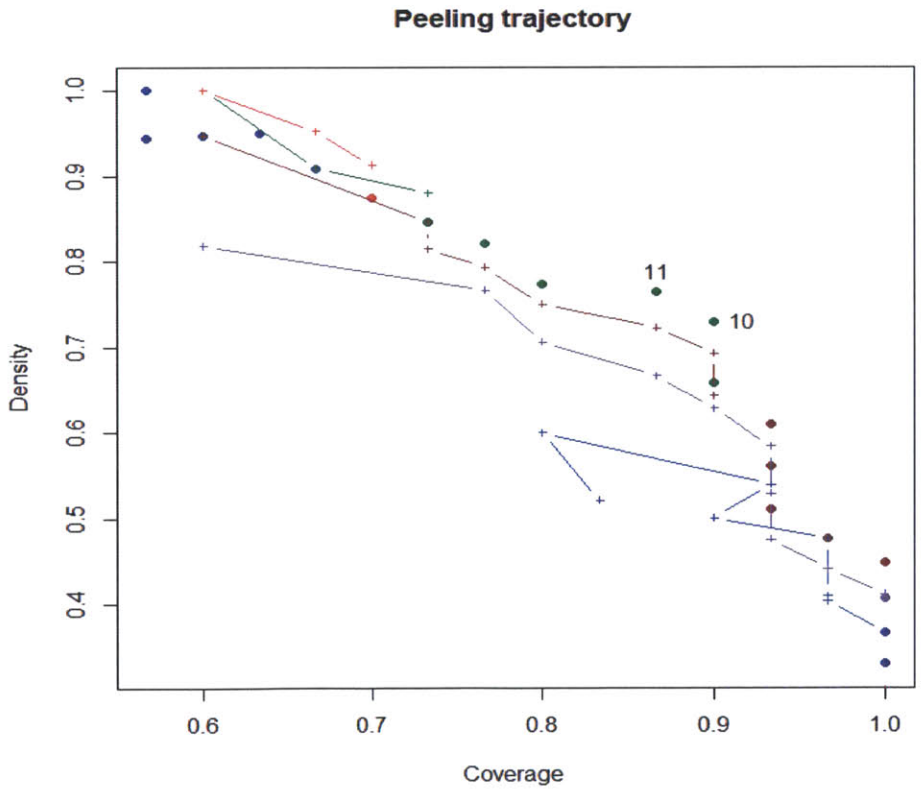
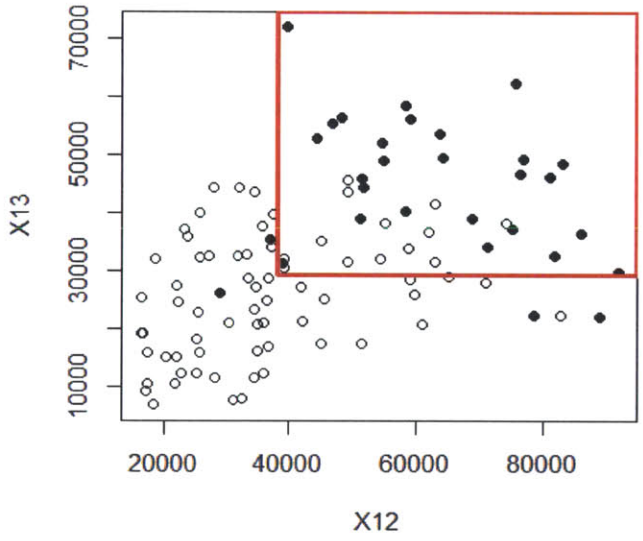


Figure 5.9b Peeling Trajectory of Model 2

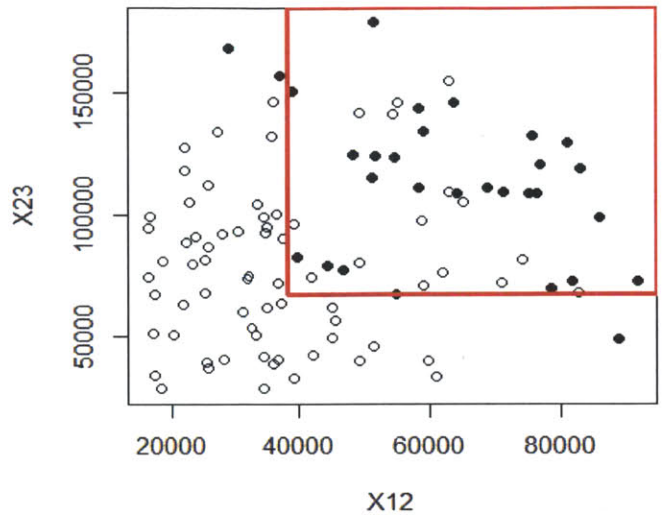
Dimension	Constraints for input variables	Density	Coverage
1	X12 >38173	30%	100%
2	X13 >29183	54%	93%
3	X23 >67130	67%	87%
4	X2 < 483.0	72%	87%

Table 5.4 Scenarios in the Space of Input Variables in Model 2

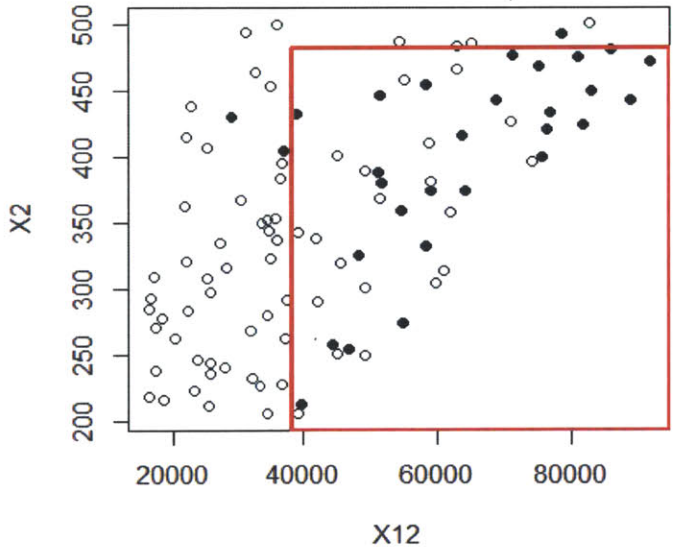
Figure 5.10 illustrate cases in database plotted as function of different pairs of parameters shown in Table 5.4. Black and open dots show high travel time and lower travel time cases, respectively. Red lines show parameters values corresponding to the boundaries of selected scenarios.



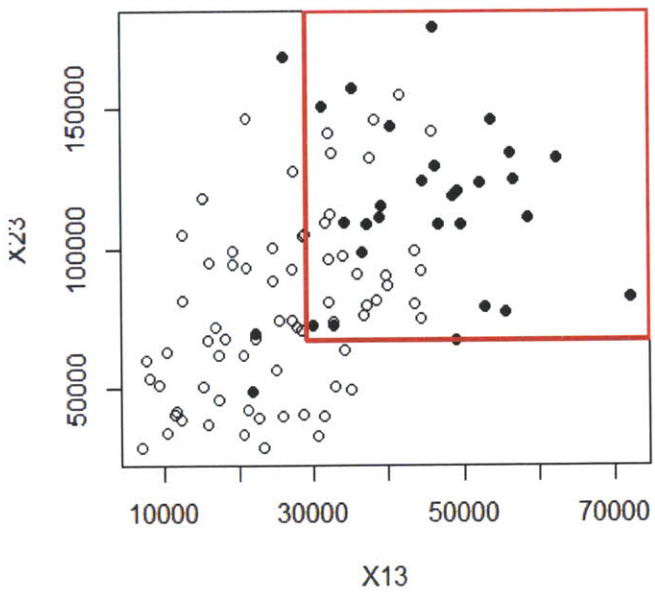
(a)


X23
150000
100000
50000
20000
40000
60000
80000
X12

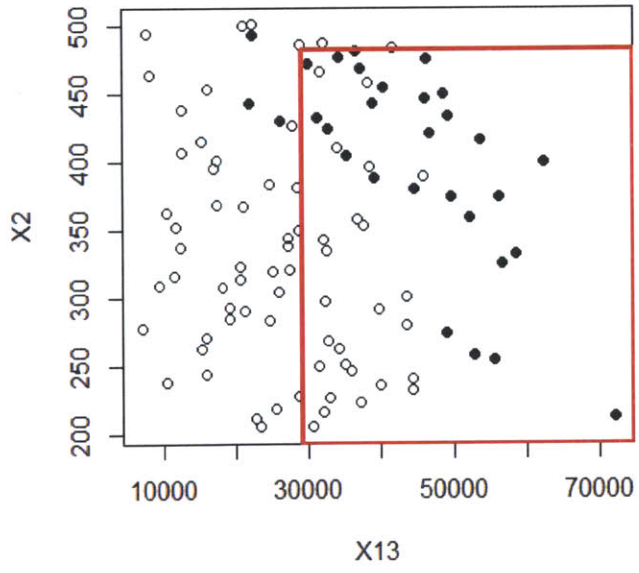
(b)


X2
500
450
400
350
300
250
200
20000
40000
60000
80000
X12

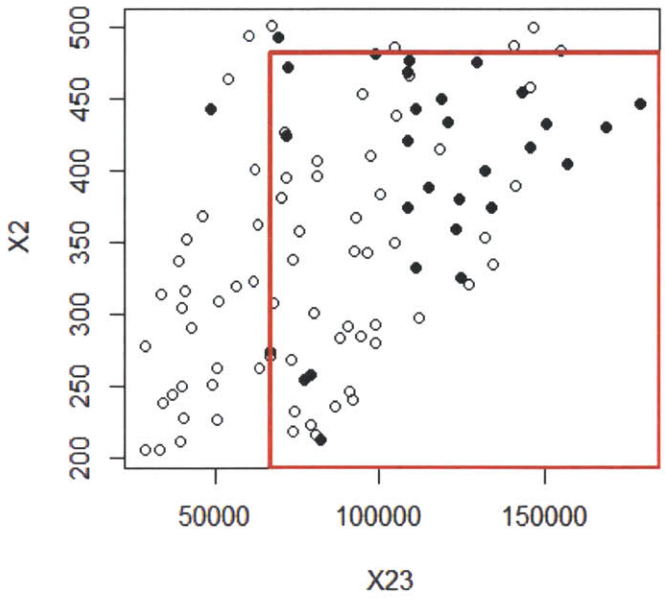
(c)


X23
150000
100000
50000
10000
30000
50000
70000
X13

(d)


X2
500
450
400
350
300
250
200
10000
30000
50000
70000
X13

(e)



(f)

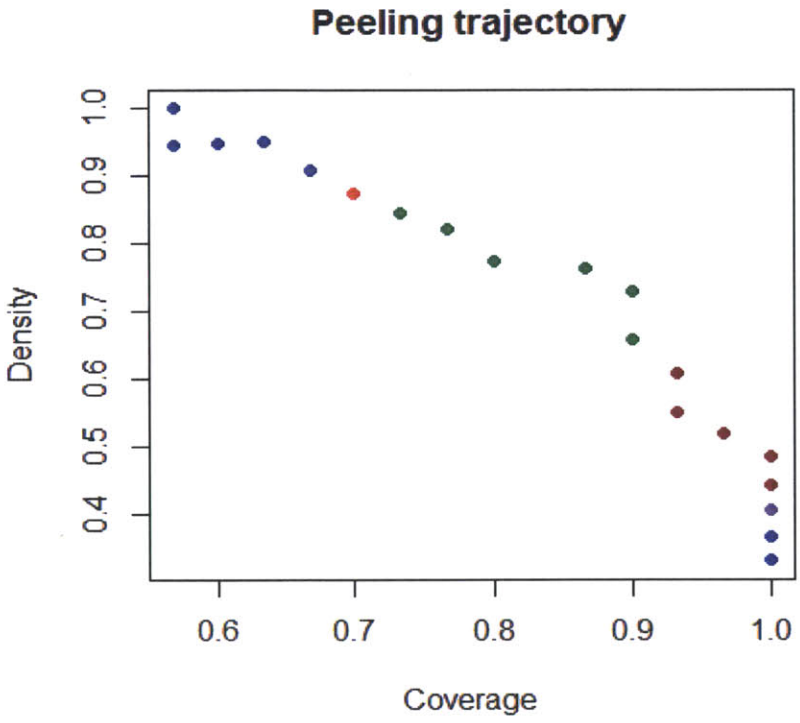
Figure 5.10 Visualization Results of PRIM in Model 2

Input Variables	Coverage	Density	Quasi-p-value
X12	0.9	0.9	0.017
X13	0.7	0.7	0.28
X23	0.9	1.0	0.28
X2	0.6	0.6	0.36
X1	0.2	0.3	
X3	0.2	0.2	

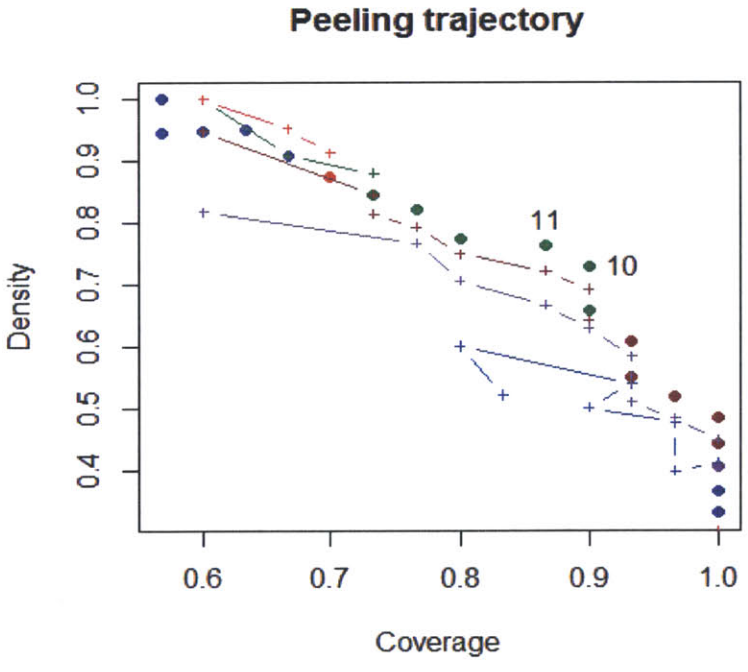
Table 5.5 Coverage, Density and Quasi-p-value of Associated Variables in Model 2

Since the interactions are too big, we do some normalization to these terms and get new model specification. Figure 5.11, 5.12 and Table 5.6, Table 5.7 showed the PRIM results.

Figure 5.11 illustrate the peeling trajectory of the normalized model. Figure 5.12 illustrate cases in database plotted as function of different pairs of parameters shown in Table 5.6. Black and open dots show high travel time and lower travel time cases, respectively. Red lines show parameters values corresponding to the boundaries of selected scenarios.



(a)

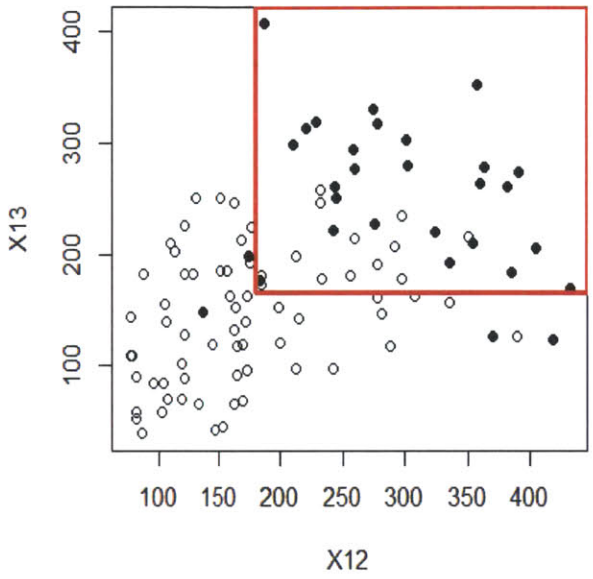


(b)

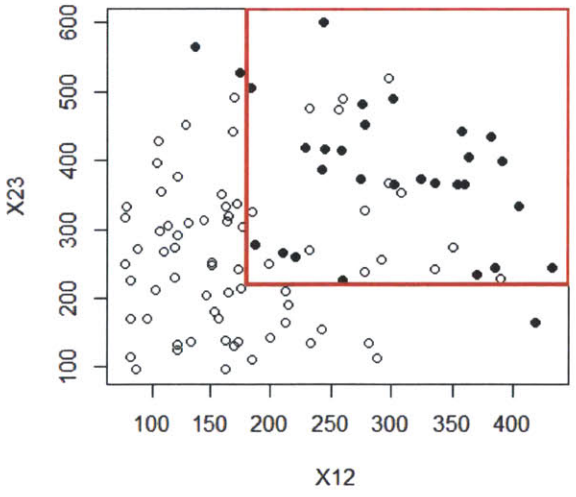
Figure 5.11 Peeling Trajectory of Normalized Model 2

Dimension	Constraints for input variables	Density	Coverage
1	X12 >180	30%	100%
2	X13 >165	54%	93%
3	X23 >219.5	67%	87%
4	X2 < 483.0	72%	87%

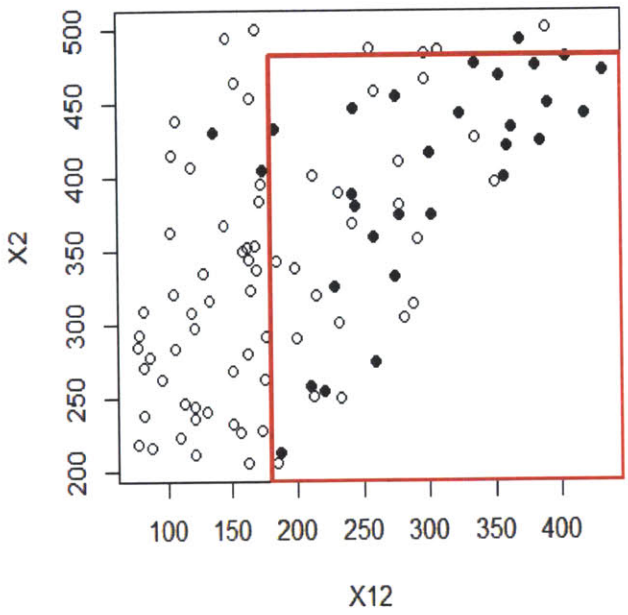
Table 5.6 Scenarios in the Space of Input Variables in Normalized Model 2



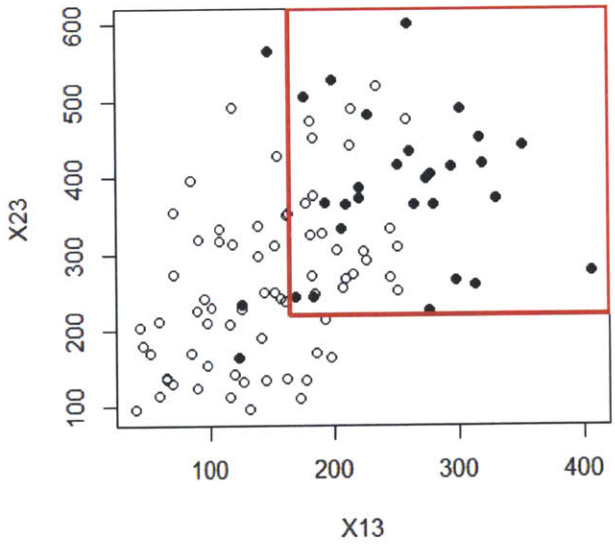
(a)



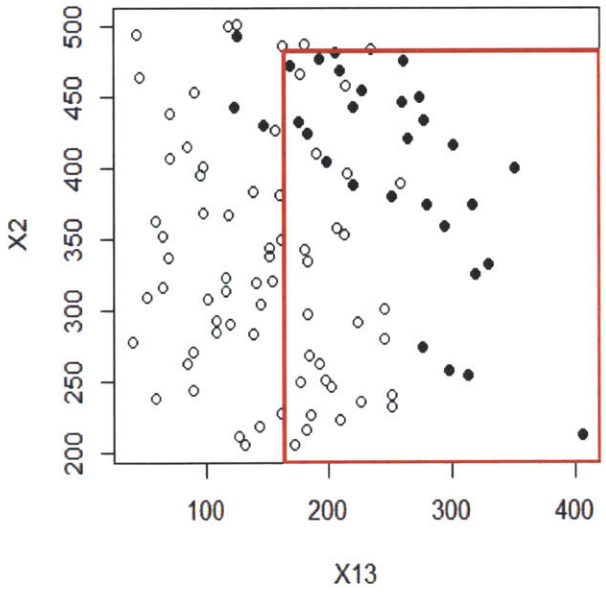
(b)


X2
200
250
300
350
400
450
500
100
150
200
250
300
350
400
X12

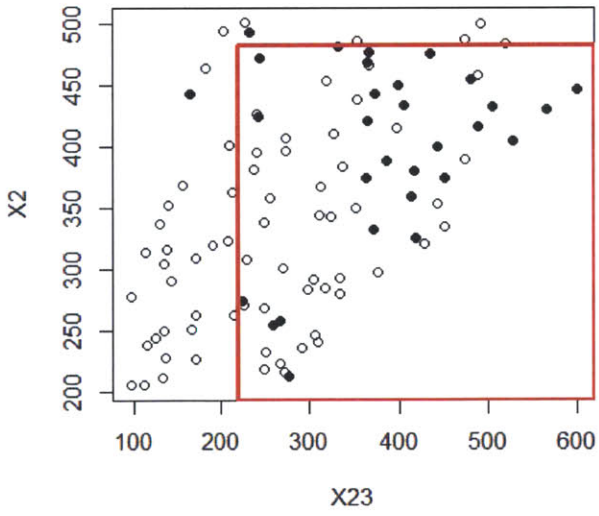
(c)


X23
100
200
300
400
500
600
100
200
300
400
X13

(d)



(e)



(f)

Figure 5.12 Visualization Results of PRIM in Normalized Model 2

Input Variables	Coverage	Density	Quasi-p-value
X12	1.0	1.0	0.017
X13	0.8	0.8	0.28
X23	0.8	0.8	0.28
X2	0.7	0.7	0.36
X1	0.1	0.1	
X3	0.2	0.2	

Table 5.7 Coverage, Density and Quasi-p-value of Associated Variables in Normalized Model 2

5.5.3 Evaluating and Choosing Scenarios

Based on the information from Section 5.5, scenario 14 appears to provide a useful description of the vulnerabilities of the transit-oriented policy for model 1.

The Table 5.8 shows the frequency with which PRIM uses each model input parameter when run on ten different N/2 sized resampling's of the dataset.

Input variables	Coverage	Density	Quasi-p-value
X1	0.9	0.9	0.002
X2	0.8	0.9	0.23
X3	0.7	0.7	0.01

Table 5.8 Coverage and Density associated with Input Variables in Model 1

Table 5.8 also shows the quasi-p-values associated with each of the three parameters used to define Scenario 14. Note that each additional parameter is orders of magnitude less significant than its predecessor, though all but the last would qualify as highly significant. A standard threshold for significance would reject using X3 in the scenario definition because it has a significance level of only 0.23.

Together, these reproducibility statistics and quasi-p-values provide high confidence that the first two parameters in Figure 5.7 each play an important role in defining the high travel time scenario. While the resampling statistics also support the importance of the parameter X3, the quasi-p-value test suggests the inclusion of this parameter may be due to chance.

The quasi-p-values in model 2 didn't show very good coverage and density comparing with those in model 1. In addition, the visualization also implied the same thing. Thus, we didn't select scenarios in model 2.

From the above discussion, one could have chosen an initial scenario with less than 100% coverage, and one could identify a second scenario to characterize some or all of the remaining points. In this example however, we judged Scenario 14 has sufficiently high coverage to render a second scenario unnecessary.

From Section 5.5.1, we selected scenario 14. From Table 5.3, we can know that when X1 and X2 go higher than expected average, the policy goes to failure; when X2 and X3 go higher than expected, the policy fails too. The bad bump hunting result of model 2 may imply that there is no or little interaction between proposed input variables.

5.6 Discussions and Conclusions

We demonstrate scenario discovery approach, using LHS experimental design and a micro traffic simulation model to "farm" alternative futures and applying PRIM to "data mine" those futures, with the goal of evaluating strategically robust policy.

We investigate the general influence of exogenous forces, varying traffic demands in this case, in determining regions of failure and identify strategy-performance regions of high interpretability, with high density of scenarios of interest and moderate-to-high purity.

Under deep uncertainty of traffic demand in the studied district, the performance of proposed policy under some scenarios is not better than the situation when there is no bus lane. In other words, the study shows that the proposed policy may deviate significantly from the optimum performance or even fail to meet its performance goal.

There are still a number of issues remain for investigation. First, the input parameters used in this analysis did not fully capture the exogenous forces that may influence the proposed policy performance. Instead of using all the OD demands as input variables, we simplified the input parameter by grouping these demands by destinations and use the sum within the group. Due to the curse of dimensionality, taking all the OD pairs into account may result in big simulation samples and in turn requires much longer simulation time. Second, in MITSIMLab, there is

limited number of driving forces we can consider while there are more that may impact the policy especially some social and economic factors. More complex simulation models are needed to fully consider the different sources of uncertainty.

Furthermore, in terms of scenario identification, the simple criterion of policy performance (e.g, average passenger travel time) may be unrealistic and inappropriate for actually policy making which requires satisfying a number of objectives environmentally and economically. A multi-criteria analysis combined with current approach may be more realistic.

Stochasticity, which is often inevitable for most simulation models, has been neglected here. As stated in this chapter, MITSIMLab is a stochastic simulation platform. To better capture the performance in reality, we have to perform same simulation run with same input parameter sample many times to reduce the impact of randomness. This will also largely increase the simulation time.

Despite the problems stated previously, we believe that this quantitative approach of scenario discovery would be promising in identifying strategically robust futures and would be increasingly applied in the transportation planning realm.

Chapter 6

Contributions and Future Work

6.1 Thesis contribution

In this thesis, we demonstrate a method that combines the power of emerging data mining algorithms and exploration techniques with the simulation models of large and complex transportation systems in determining robust strategic regions.

A robust decision making analysis approach which has been implemented in some other planning realms like energy is first implemented in transportation planning realm. The previous studies on robust decision making and applications are reviewed.

The approach consists of data farming that incorporate the impact from a number of exogenous driving forces and data mining the future states that represent the vulnerability of the proposed policy. The thesis compared different exploration techniques and data mining algorithms that can be used in the approach. The benefits and limitations of alternative techniques are illustrated.

Thesis implements an empirical study based on a micro traffic simulation platform calibrated from Singapore urban network. In the empirical study, we showed how scenario discovery can help evaluating policy performance and representing vulnerabilities of proposed policies and showed how the new statistical tools can generate easy-to-interpret results visually. These together can be used to better understand the proposed policies and the tradeoffs between them.

Finally, we evaluate the limitations of our approach and empirical study and we will propose some future work directions in next section.

6.2 Future work

As stated in section, there are some limitations of current methods. So focusing on overcoming these limitations of our empirical study, we propose some future work as follows.

First, we may propose a better way to deal with the high dimensional input parameters that influence the performance of the policy. By factor analysis or some other techniques, we can use fewer variables to capture all the varying input parameters for the simulation models.

Second, the criterion of the study may be revised to be more realistic. Instead of simple travel time metric, we would combine some other environmental and economic criterions into the study.

Third, due to the limitation of simulation model, more appropriate simulation platform may be used in the empirical study. Since building simulation model is often costly, some methods incorporating the influence of other driving forces without inputting into the simulation models may be proposed to address this challenge. To better deal with stochasticity, we may need more simulation runs to reduce the randomness.

In addition, we also proposed some future work from technical perspectives. Alternative machine learning algorithms could be implemented to identify the vulnerabilities of proposed policies. In the view of some literature, PRIM would additionally peel away more data at the last step. So more accurate algorithms need to be tested and compared with PRIM.

Finally, from the policy making perspective, there will be some model input variables that cannot easily be quantified. So how to incorporate qualitative information into this quantitative approach will be another challenge to implement this approach widely. Some of the future work may focus on solving this problem.

Appendix A – Glossary of Acronyms

ATMS	Advanced Traffic Management Systems
BL	Bus Lane
BBT	Bus-lane Bus Travel-time
BBPT	Bus-lane Bus Passenger Travel-time
BCPT	Bus-lane Car Passenger Travel-time
BCT	Bus-lane Car Travel-time
BVPT	Bus-lane Vehicle Passenger Travel-time
BVT	Bus-lane Vehicle Travel-time
CART	Classification and Regression Tree
GUI	Graphical User Interface
LHS	Latin Hypercube Sampling
MITSIM	Microscopic Traffic Simulator
NBL	No Bus Lane
NBT	Non-bus-lane Bus Travel-time
NBPT	Non-bus-lane Bus Passenger Travel-time
NCPT	Non-bus-lane Car Passenger Travel-time
NCT	Non-bus-lane Car Travel-time
NVPT	Non-bus-lane Vehicle Passenger Travel-time
NVT	Non-bus-lane Vehicle Travel-time

OD	Origin and Destination
PFF	Pseudo Full Factorial
PRIM	Patient Rule Induction Method
RDM	Robust Decision Making
TMS	Traffic Management Simulator

Appendix B – Simulation Output

ID	X1	X2	X3	BCT	NCT	BBT	NBT
1	0.584	0.68	0.572	761626	1161215	31345	32307
2	0.496	1.163	0.8	1290356	1362027	32676	39121
3	0.48	1.037	0.696	1305748	1660751	33242	33310
4	1.424	1.377	0.82	1669153	1588228	34850	37959
5	1.024	0.834	1.24	1303150	1417164	34616	35819
6	0.864	0.923	0.768	1428739	1648504	31693	38617
7	0.536	1.229	1.568	1840158	1774830	34666	35612
8	0.464	0.814	1.328	1040149	1423327	30418	32601
9	0.616	1.297	0.836	1481411	1473530	31409	39948
10	1.488	0.609	1.548	1260461	1423487	41498	42934
11	0.656	0.88	0.884	1150149	1363990	31928	32864
12	1.52	0.589	0.644	925078	1684628	41531	31010
13	0.688	0.851	1.504	1511103	1900795	36492	32970
14	1.328	1.22	0.672	1391327	1465345	34765	31844
15	0.912	0.98	1.124	1165278	1474670	38063	31912
16	0.888	1.394	1.156	1815335	1809131	32904	37150
17	0.808	1.011	1.492	859053	1129591	40335	42452
18	1.6	0.783	0.98	1249986	1554537	38630	33685
19	0.96	1.309	1.272	1585352	1467558	34612	44688
20	1.344	0.586	0.556	736906	1182727	37130	33160
21	0.552	0.917	1.588	1606699	1705088	34986	36894
22	1.192	1.363	0.916	1296882	1350894	42262	42620
23	0.576	1.429	1.172	1726279	1591077	31874	42038
24	1.6	1.266	0.44	1159339	1191794	43550	44422

25	1.072	1.389	0.864	1661932	1713605	37695	43245
26	0.504	1.414	0.488	1022916	1231794	32083	33175
27	1.552	0.897	0.428	1165341	1395402	36260	33158
28	0.528	0.794	0.416	773909	1051506	31461	34217
29	1.512	1.143	1.32	1768641	1647590	39841	52252
30	0.776	0.703	1.48	753144	1287784	39439	34109
31	1.552	1.349	0.616	1717335	1700514	40031	44005
32	1.04	1.383	1.28	1350148	1418301	32215	39444
33	1.536	1.214	0.68	1071760	1018190	42356	47455
34	0.76	1.097	1.044	989014	1132305	35412	49714
35	0.744	1.129	0.728	1377919	1324925	32094	40667
36	0.72	1.237	1.392	1667792	1531219	36216	50667
37	1.432	0.717	0.784	982485	1459083	40650	33457
38	0.424	1.186	1.14	1536487	1623835	34220	39467
39	1.104	0.663	1.288	1117328	1371401	39301	32390
40	1.288	0.651	0.712	881435	1133021	37136	33144
41	0.416	1.254	0.96	1570298	1657974	30006	39015
42	0.832	0.637	1.428	1298399	1627533	32046	32998
43	1.16	0.829	0.588	880984	1234581	39282	33130
44	1.384	1.023	0.848	799544	1024364	36611	43881
45	1.112	1.054	0.5	1080294	1396052	40502	32798
46	0.992	0.966	0.876	862458	1232844	34872	34782
47	0.856	0.963	0.46	966844	1181956	32535	35687
48	1.568	0.869	0.528	1601996	1793047	33717	33482
49	0.92	1.277	1.6	1698296	1175264	37599	50788
50	1.128	0.751	0.968	1298241	1541870	32832	31230

51	1.496	1.131	0.82	988537	1289408	37252	35484
52	1.08	1.331	0.936	1581546	1720491	33980	38311
53	1.008	1.114	1.452	1647245	1574367	34201	41685
54	1.32	1.431	0.54	1259374	1068788	37221	49495
55	1.184	0.931	1.528	1421694	1282725	38503	38082
56	0.872	0.674	1.468	1681296	1851232	31964	32888
57	0.632	0.809	1.248	1400372	1670859	33345	33882
58	1.144	1.174	0.948	1576662	1635691	33220	41034
59	1.4	0.949	1.336	1491334	1392473	36810	44978
60	1.376	0.737	1.228	1396259	1824203	42084	34180
61	0.664	1.049	1.016	1266886	1690000	32759	32996
62	0.768	1	1.196	1573941	1702151	34035	32050
63	0.512	0.774	0.992	1330939	1391895	28852	32518
64	1.216	1.026	1.372	1154284	1509615	43229	40164
65	1.176	0.646	0.896	1435077	1601814	33342	31441
66	0.624	0.749	0.776	984955	1330943	31556	33380
67	1.224	1.189	1.4	2017596	1761364	38845	45031
68	0.968	0.603	0.752	1102664	1493844	33512	31504
69	1.272	1.409	0.564	1205473	1413371	41298	39656
70	1.36	1.36	1.088	1458600	1556851	44387	41198
71	0.784	1.006	0.472	1169108	1364958	31500	35779
72	1.416	1.24	1.112	1649264	1349159	38266	52570
73	1.088	1.086	1.308	1564627	1767685	39581	33517
74	0.56	1.326	0.464	1250350	1535591	32465	36266
75	1.256	1.071	1.432	1760382	1496138	38458	44674
76	0.84	0.697	0.604	904868	1406986	29907	31218

77	1.24	1.266	1.004	1122332	1460465	37665	39284
78	0.984	0.8	1.416	1043545	1531635	42392	36060
79	0.688	0.617	1.496	1218788	1714646	35798	32022
80	0.808	0.983	1.076	1545637	1757389	32505	32988
81	1.464	0.729	1.212	877036	1276212	40828	34265
82	0.6	0.623	1.36	800367	1190122	29987	32690
83	0.952	0.766	1.1	1679504	1759912	32857	35040
84	0.896	1.146	0.62	1045410	1225548	32417	37583
85	1.024	1.3	1.26	1351085	670334	43027	68099
86	0.928	0.686	1.532	1350877	1563798	36004	36899
87	0.728	1.157	1.552	1143565	1476631	37155	34737
88	1.28	1.34	0.928	1075298	1538937	39778	33334
89	0.648	0.957	1.604	924262	1875479	37026	34936
90	1.448	1.203	1.032	1063835	1333820	38714	37615
91	0.448	0.883	0.66	866647	1268170	30513	32245
92	0.712	0.903	0.516	1243361	1686389	32908	35033
93	0.456	0.837	1.352	1447891	1662588	33683	34268
94	1.136	0.914	0.708	1033325	1077606	32773	43860
95	1.472	1.286	1.056	1528678	1339935	46155	45935
96	1.056	1.109	1.184	1578774	1794927	34820	36622
97	1.304	0.86	1.068	1537108	1857755	35520	32486
98	1.232	1.091	0.74	1385876	1540546	39770	42454
99	1.368	1.069	1.16	1376071	1619651	39812	41770
100	1.576	0.714	0.64	1467113	1715527	34440	35411

ID	BVT	NVT	BCPT	NCPT	BBPT	NBPT	BVPT	NVPT
1	792972	1193522	1148484	1741822	940356	969216	2088840	2711038
2	1323032	1401148	1926414	2043040	980268	1173624	2906682	3216664
3	1338989	1694061	1960266	2491126	997248	999300	2957514	3490426
4	1704003	1626187	2504471	2382341	1045488	1138776	3549959	3521117
5	1337766	1452983	1968717	2125746	1038480	1074564	3007197	3200310
6	1460432	1687121	2157687	2472756	950784	1158516	3108471	3631272
7	1874824	1810442	2765484	2662245	1039980	1068360	3805464	3730605
8	1070567	1455928	1556337	2134990	912540	978024	2468877	3113014
9	1512820	1513478	2227250	2210295	942264	1198428	3169514	3408723
10	1301960	1466421	1890341	2135231	1244952	1288008	3135293	3423239
11	1182077	1396854	1732707	2045985	957828	985908	2690535	3031893
12	966609	1715639	1394496	2526942	1245936	930312	2640432	3457254
13	1547595	1933765	2279829	2851193	1094772	989088	3374601	3840281
14	1426092	1497189	2084679	2198018	1042956	955320	3127635	3153338
15	1203341	1506582	1740113	2212005	1141884	957372	2881997	3169377
16	1848239	1846281	2711374	2713696	987108	1114512	3698482	3828208
17	899389	1172043	1300079	1694387	1210056	1273560	2510135	2967947
18	1288616	1588222	1860265	2331806	1158900	1010544	3019165	3342350
19	1619963	1512246	2379587	2201337	1038348	1340628	3417935	3541965
20	774036	1215887	1114582	1774090	1113888	994800	2228470	2768890
21	1641686	1741982	2411812	2557632	1049592	1106820	3461404	3664452
22	1339145	1393514	1936333	2026341	1267872	1278600	3204205	3304941
23	1758153	1633115	2576975	2386616	956232	1261140	3533207	3647756
24	1202889	1236216	1747870	1787690	1306500	1332672	3054370	3120362
25	1699627	1756850	2505619	2570407	1130844	1297356	3636463	3867763

26	1054999	1264969	1529370	1847691	962484	995256	2491854	2842947
27	1201601	1428560	1753282	2093103	1087812	994740	2841094	3087843
28	805370	1085722	1159866	1577258	943836	1026504	2103702	2603762
29	1808482	1699842	2650780	2471385	1195224	1567572	3846004	4038957
30	792583	1321893	1137925	1931676	1183176	1023264	2321101	2954940
31	1757366	1744519	2580090	2550771	1200924	1320156	3781014	3870927
32	1382363	1457745	2026972	2127452	966444	1183308	2993416	3310760
33	1114115	1065645	1596248	1527285	1270668	1423644	2866916	2950929
34	1024426	1182018	1475650	1698457	1062372	1491408	2538022	3189865
35	1410013	1365592	2081005	1987388	962808	1220004	3043813	3207392
36	1704007	1581887	2490020	2296829	1086468	1520016	3576488	3816845
37	1023136	1492540	1473553	2188624	1219512	1003704	2693065	3192328
38	1570707	1663302	2296350	2435753	1026612	1184004	3322962	3619757
39	1156629	1403792	1683322	2057102	1179036	971712	2862358	3028814
40	918571	1166164	1315456	1699531	1114092	994308	2429548	2693839
41	1600304	1696989	2349598	2486962	900180	1170444	3249778	3657406
42	1330445	1660531	1962295	2441300	961380	989940	2923675	3431240
43	920266	1267711	1311420	1851872	1178460	993900	2489880	2845772
44	836155	1068245	1197447	1536546	1098324	1316436	2295771	2852982
45	1120797	1428851	1606634	2094078	1215072	983952	2821706	3078030
46	897330	1267627	1280767	1849267	1046172	1043472	2326939	2892739
47	999379	1217644	1435631	1772935	976044	1070616	2411675	2843551
48	1635713	1826529	2414127	2689571	1011504	1004460	3425631	3694031
49	1735895	1226052	2561987	1762897	1127976	1523628	3689963	3286525
50	1331073	1573099	1935625	2312804	984972	936888	2920597	3249692
51	1025789	1324892	1473382	1934113	1117572	1064520	2590954	2998633

52	1615526	1758802	2386836	2580737	1019400	1149336	3406236	3730073
53	1681446	1616051	2478117	2361550	1026036	1250544	3504153	3612094
54	1296595	1118283	1881044	1603182	1116636	1484844	2997680	3088026
55	1460198	1320807	2130051	1924088	1155096	1142472	3285147	3066560
56	1713260	1884121	2535211	2776848	958920	986652	3494131	3763500
57	1433717	1704742	2108869	2506289	1000356	1016472	3109225	3522761
58	1609882	1676725	2363084	2453537	996600	1231020	3359684	3684557
59	1528144	1437451	2224292	2088710	1104300	1349328	3328592	3438038
60	1438344	1858383	2083358	2736305	1262532	1025400	3345890	3761705
61	1299645	1722996	1904569	2534999	982764	989892	2887333	3524891
62	1607976	1734201	2348460	2553226	1021056	961512	3369516	3514738
63	1359791	1424413	1993551	2087843	865548	975540	2859099	3063383
64	1197513	1549779	1731410	2264422	1296864	1204932	3028274	3469354
65	1468418	1633255	2152081	2402721	1000248	943224	3152329	3345945
66	1016511	1364324	1467722	1996415	946668	1001412	2414390	2997827
67	2056441	1806395	3025732	2642046	1165356	1350924	4191088	3992970
68	1136176	1525348	1643464	2240766	1005348	945132	2648812	3185898
69	1246771	1453028	1811104	2120057	1238940	1189692	3050044	3309749
70	1502987	1598049	2180372	2335277	1331616	1235928	3511988	3571205
71	1200608	1400737	1752511	2047436	945000	1073376	2697511	3120812
72	1687530	1401729	2484161	2023739	1147980	1577088	3632141	3600827
73	1604208	1801202	2346570	2651528	1187424	1005516	3533994	3657044
74	1282815	1571857	1875955	2303386	973956	1087980	2849911	3391366
75	1798840	1540812	2652841	2244208	1153728	1340220	3806569	3584428
76	934775	1438204	1364213	2110480	897216	936540	2261429	3047020
77	1159997	1499749	1684513	2190698	1129956	1178508	2814469	3369206

78	1085937	1567696	1552904	2297453	1271760	1081812	2824664	3379265
79	1254585	1746669	1818010	2571969	1073928	960672	2891938	3532641
80	1578143	1790377	2332360	2636084	975156	989640	3307516	3625724
81	917864	1310477	1301961	1914318	1224840	1027944	2526801	2942262
82	830354	1222812	1201968	1785182	899604	980700	2101572	2765882
83	1712360	1794952	2531710	2639869	985704	1051188	3517414	3691057
84	1077827	1263132	1582007	1838323	972516	1127496	2554523	2965819
85	1394113	738433	2018285	1005501	1290816	2042964	3309101	3048465
86	1386880	1600697	2038838	2345697	1080108	1106976	3118946	3452673
87	1180720	1511368	1718616	2214946	1114644	1042116	2833260	3257062
88	1115076	1572271	1613442	2308405	1193328	1000020	2806770	3308425
89	961288	1910416	1399469	2813219	1110792	1048092	2510261	3861311
90	1102549	1371435	1591429	2000729	1161420	1128456	2752849	3129185
91	897159	1300415	1286858	1902254	915384	967356	2202242	2869610
92	1276269	1721423	1851491	2529584	987252	1050996	2838743	3580580
93	1481574	1696856	2179479	2493882	1010484	1028052	3189963	3521934
94	1066098	1121465	1560333	1616409	983196	1315788	2543529	2932197
95	1574833	1385870	2282032	2009902	1384656	1378044	3666688	3387946
96	1613594	1831549	2366371	2692390	1044600	1098672	3410971	3791062
97	1572628	1890241	2319237	2786632	1065600	974580	3384837	3761212
98	1425645	1583000	2082001	2310819	1193088	1273632	3275089	3584451
99	1415883	1661420	2053103	2429476	1194372	1253088	3247475	3682564
100	1501553	1750939	2202824	2573291	1033200	1062336	3236024	3635627

Bibliography

[1] Lee D B J, 1973, "Requiem for large-scale models" Journal of the American Planning Association 39(3) 163 – 178

[2] Rodier C J, Johnston R, 2002, "Uncertain socioeconomic projections used in travel and emissions models: could plausible errors result in air quality nonconformity?" Transportation Research A 36 613–619

[3] Fragkias M, Seto K C, 2007, "Modeling urban growth in data-sparse environments: a new approach" Environment and Planning B: Planning and Design 34(5) 858 – 883

[4] Kockelman K, Krishnamurthy S, 2003, "Propagation of uncertainty in transportation-land use models: investigation of DRAMEMPAL and UTPP predictions in Austin, Texas" Transportation Research Record 1831 219–229

[5] Zegras C, Sussman J, Conklin C, 2004, "Scenario planning for strategic regional transportation planning" Journal of Urban Planning and Development-Asce 130(1) 2-13

[6] Bartholomew K, 2007, "Land use-transportation scenario planning: promise and reality" Transportation: Planning, Policy, Research, Practice 34(4) 397-412

[7] R.J. Lempert, D.G. Groves, S.W. Popper, S.C. Bankes, "A general, analytic method for generating robust strategies and narrative scenarios", Manage. Sci. 52 (4) (2006) 514–528.

[8] Serra D, Ratick S, ReVelle C, 1996, "The maximum capture problem with uncertainty" Environment and Planning B: Planning and Design 23(1) 49 – 59

[9] R.J. Lempert, M.T. Collins, "Managing the risk of uncertain threshold response: Comparison of robust, optimum, and precautionary approaches", Risk Anal. 27 (4) (2007) 1009–1026.

[10] Lempert, Robert J., Nebojsa Nakicenovic, Daniel Sarewitz, Michael Schlesinger, 2004: "Characterizing Climate-Change Uncertainties for Decision-makers," Climatic Change, 65, 1-9

[11] D.G. Groves, R.J. Lempert, "A new analytic method for finding policy-relevant scenarios", Glob. Environ. Change 17 (1) (2007) 73–85.

[12] R.J. Lempert, D.G. Groves, S.W. Popper, S.C. Bankes, "A general, analytic method for generating robust strategies and narrative scenarios", Manage. Sci. 52 (4) (2006) 514–528.

[13] E.A. Parson, V. Burkett, K. Fischer-Vanden, D. Keith, L. Mearns, H. Pitcher, C. Rosenweig, M. Webster, "Global-change scenarios: their development and use, synthesis and assessment product 2.1b. ", US climate change science program, Washington, DC, 2007.

[14] European Environmental Agency, "Looking back on looking forward: A review of evaluative scenario literature", Technical Report No 3/2009., European Environment Agency, Luxembourg, 2009.

[15] Bryant B P, Lempert R J, 2010, "Thinking inside the box: A participatory, computer-assisted approach to scenario discovery" Technological Forecasting and Social Change 77(1) 34-49Reference

[16] P. Bishop, A. Hines, T. Collins, "The current state of scenario development: an overview of techniques", Foresight 9 (1) (2007) 5–25.

[17] L. Börjeson, M. Hojer, K.-H. Dreborg, T. Ekvall, G. Finnveden, "Scenario types and techniques: Towards a user's guide", Futures 38 (7) (2006) 723–739.

[18] R. Bradfield, G. Wright, G. Burt, G. Cairns, K. van der Heijden, "The origins and evolution of scenario techniques in long range business planning", Futures 37 (8) (2005) 795–812.

[19] K. Van der Heijden, Scenarios: The Art of Strategic Conversation, John Wiley and Sons, Chichester, UK, 1996.

[20] S.A. Van 't Klooster, M.B.A. van Asselt, "Practicing the scenario-axis technique", Futures 38 (1) (2006) 15–30.

[21] T.J.B.M. Postma, F. Liebl, "How to improve scenario analysis as a strategic management tool?" Technol. Forecast Soc. Change 72 (2) (2005) 161–173.

[22] P. van Notten, A.M. Sleegers, M.B.A. van Asselt, "The future shocks: on discontinuity and scenario development", Technol. Forecast Soc. Change 72 (2) (2005) 175–194.

[23] E. Best (Ed.), Probabilities. Help or hindrance in scenario planning? Deeper news: exploring future business environments, 2(4) (Summer 1991) Topic 154.

[24] P. Schwartz, The Art of the Long View, Doubleday, New York, New York, 1996.

[25] L. Dixon, R.J. Lempert, T. LaTourrette, R.T. Reville, The federal role in terrorism insurance: evaluating alternatives in an uncertain world, MG-679-CTRMP, RAND Corporation, Santa Monica, California, 2007.

[26] D.G. Groves, D. Knopman, R.J. Lempert, S.H. Berry, L. Wainfan, Presenting uncertainty about climate change to water resource managers, TR-505-NSF, RAND Corporation, Santa Monica, California, 2007.

[27] D.G. Groves, R.J. Lempert, D. Knopman, S.H. Berry, Preparing for an uncertain future climate in the Inland Empire: identifying robust water-management strategies, DB-550-NSF, RAND Corporation, Santa Monica, California, 2008.

[28] Gooding Swartz, P. and C. Zegras. (2011). "Strategically Robust Urban Planning? A Demonstration of Concept." working paper

[29] Wack, P, 1985, "Scenarios: Uncharted waters ahead" Harvard Business Review 85(5) 72–89

[30] Bowman J L, Gopinath D, Ben-Akiva M, 2002, "Estimating the probability distribution of a travel demand forecast", http://jbowman.net/#Papers

[31] Helton J C, Davis F J, 2002 Latin Hypercube Sampling and the Propagation of Uncertainty in Analyses of Complex Systems (Sandia National Laboratories, Albuquerque, New Mexico)

[32] Carnell R, 2009, "LHS: Latin Hypercube Samples", http://cran.r-project.org/web/packages/lhs/lhs.pdf

[33] Friedman J H, Fisher N I, 1998, "Bump hunting in high-dimensional data", http://www-stat.stanford.edu/~jhf/ftp/prim.pdf

[34] Breiman L, 1993 Classification and Regression trees (Chapman & Hall, Boca Raton, FA)

[35] Helton J C, Davis F J, 2002 Latin Hypercube Sampling and the Propagation of Uncertainty in Analyses of Complex Systems (Sandia National Laboratories, Albuquerque, New Mexico)

[36] Hastie,T., Tibshirani,R. and Friedman,J. (2001) The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer-Verlag, New York

[37] R.J. Lempert, B.P. Bryant, S.C. Bankes, Comparing Algorithms for Scenario Discovery, WR-557-NSF, RAND Corporation, Santa Monica, California, 2008.

[38] Singapore Census of Population 2000. Statistics Singapore. Retrieved 2008-03-26. http://en.wikipedia.org/wiki/Transport_in_Singapore

[39] Transport in Singapore. Retrieved May 1, 2013, from

http://en.wikipedia.org/wiki/Singapore#Transport

[40] Overview of Public Transport. (2012). Retrieved May 14, 2012, from http://www.lta.gov.sg/content/lta/en/public_transport/overview_.html

[41] Marina Bay, Singapore. Retrieved May 1, 2013, from

http://en.wikipedia.org/wiki/Marina_Bay,_Singapore

[42] Map of Marina Bay, Retrieved May 14, 2012, from

https://maps.google.com/

[43] MITSIMLab, Retrieved May 1, 2013, from http://mit.edu/its/mitsimlab.html