Probabilistic Forecasting in Decision-Making: New Methods and Applications

Xiaojia Guo

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

of

University College London.

UCL School of Management

University College London

November 10, 2020

I, Xiaojia Guo, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the work.

To my family for their invaluable support throughout the years.

Abstract

This thesis develops new methods to generate probabilistic forecasts and applies these methods to solve operations problems in practice. The first chapter introduces a new product life cycle model, the tilted-Gompertz model, which can predict the distribution of period sales and cumulative sales over a product's life cycle. The tilted-Gompertz model is developed by exponential tilting the Gompertz model, which has been widely applied in modelling human mortality. Due to the tilting parameter, this new model is flexible and capable of describing a wider range of shapes compared to existing life cycle models. In two empirical studies, one on the adoption of new products and the other on search interest in social networking websites, I find that the tilted-Gompertz model performs well on quantile forecasting and point forecasting, when compared to other leading life-cycle models. In the second chapter, I develop a new exponential smoothing model that can capture life-cycle trends. This new exponential smoothing model can also be viewed as a tilted-Gompertz model with time-varying parameters. The model can adapt to local changes in the time series due to the smoothing parameters in the exponential smoothing formulation. When estimating the parameters, prior information is included in the regularization terms of the model. In the empirical studies, the new exponential smoothing model outperforms several leading benchmark models in predicting quantiles on a rolling basis. In the final chapter, I develop a predictive system that predicts distributions of passengers' connection times and transfer passenger flows at an airport using machine learning methods. The predictive system is based on regression trees and copula-based simulations. London Heathrow airport currently uses this proposed system and has reported significant accuracy improvements over their legacy systems.

Impact Statement

The research I conducted during my Ph.D. covers both methodology development and practical applications. From an academic perspective, my research contributes to the development of product diffusion models, exponential smoothing techniques and regression tree models. When developing new forecasting models, I mainly focus on distributions, quantiles and probabilities, moving beyond point estimates. This way, these new models can provide better support for business decision making, where uncertainties often arise. It is noteworthy that my research on forecasting product and service life cycles discussed in Chapter 2 and 3 has received the INFORMS Decision Analysis Society Best Student Paper Award in 2018, and was selected as the finalist of the POMS-JD.com Best Data-Driven Research Paper Competition in 2018.

My research also has significant impact in practice. The predictive system I created and described in Chapter 4 has been implemented in London Heathrow airport since 2017. This system allows for predicting which passengers are likely to miss their flights, improving passenger experience and reducing the likelihood that flight departures will be delayed while waiting for late passengers. By aggregating estimated passenger arrival times at various points throughout the airport, the system also assists in reducing queuing and delays at immigration and security desks by enabling modifications to staffing levels in advance. The Head of Airport Research at Eurocontrol writes that this "ground breaking" study has "become a reference", and "is used by a number of major European airports with a first deployment since 2017 in London Heathrow's Airport Operations Centre" (Graham 2018). The research was also selected as a finalist for the 2019 MSOM Practice-Based Research Competition and for the 2019 Daniel H. Prize for Excellence in Operations Research Practice, which both recognize high-impact applications of management science and operations research in practice.

Acknowledgements

I am deeply grateful to my supervisor, Bert De Reyck, for his guidance and support throughout my Ph.D. study. I am also thankful to my co-authros, Yael Grushka-Cockayne and Kenneth C. Lichtendahl, for their contribution to my academic development.

Special thanks to Ioannis Fragkos, Chia-Yu Kou-Barret, Yufei Huang, Hang Ren and other fellow Ph.D. students for their friendship and advice. I am also fortunate to be surround by friendly and supportive faculty members at UCL. I am grateful to Bilal Gokpinar, Rouba Ibrahim, Steve Yoo, Dongyuan Zhan and Ersin Korpeoglu for their feedback and suggestions on my research and job search.

Last but not least, I would like to thank my parents and my husband for their company and support over all these years. I would also like to thank my son who makes my Ph.D. study longer than expected. He has always been a source of joy and constantly reminds me of what is more important than publishing papers. He is the flower in the garden of my heart.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Predictive analysis is a prevalent activity used in business operations. It provides a guide for future uncertainties and assists managers in developing efficient plans. With the ever-growing volume of data available in various fields of operations management, predictive analysis and forecasting methods face new challenges as well as opportunities. This thesis consists of three chapters that contribute to the development and application of statistical and machine learning models, with a focus on generating accurate probabilistic forecasts.

Forecasting methods that produce accurate point forecasts have been well studied in many contexts. Probabilistic forecasts, which are often more important in business decision making, have received limited attention in the literature. My thesis mainly contributes to the methodological development of product diffusion models, exponential smoothing techniques and regression tree models. Product diffusion models have been widely studied in the marketing and operations management literature (Bass 1969, Easingwood et al. 1983, Bemmaor 1994, Van den Bulte and Joshi 2007, Hu et al. 2017). These existing diffusion models, however, are limited in their abilities to capture skewed diffusions. Another inherent problem in forecasting product life cycles is that model parameters are often difficult to estimate with only pre-peak data (Xie et al. 1997, Venkatesan et al. 2004). Exponential smoothing is another forecasting method that has been well studied in the literature (Holt 1957/2004, Winters 1960, Gardner and McKenzie 1985, Taylor 2003). Existing exponential smoothing techniques, however, cannot predict non-monotonic trends, such as a life cycle. For some exponential smoothing models, it is also difficult to make reasonable probabilistic forecasts because they suffer from the infinite variance problem and are numerically unstable (Hyndman et al. 2008). Moreover, when only a few data points are available, exponential smoothing models can easily

overfit to the noise and make poor out-of-sample forecasts. Finally, machine learning methods, such as regression trees, are known to be accurate in generating point forecasts (Eliashberg et al. 2007, James et al. 2013, Xue et al. 2015, Tassone and Rohani 2017). However, since these methods usually do not rely on distributional assumptions, very few of them can forecast distributions (Meinshausen 2006).

Additionally, forecasting methods developed in the marketing and operations management literature are often tested using traditional forecasting accuracy measures. Without an estimate of the economic impact of improved forecasts, it is difficult to conclude whether the forecasting improvements are meaningful in decision-making processes. In this thesis, I formulate several newsvendor-based decision-making problems to study the implications of forecasting on decision making using the proposed models.

In Chapter 2, I study the problem of forecasting the distribution of demand for a new product that evolves according to a skewed life cycle. Firms need accurate distributional forecasts of product life cycles to make operational decisions about capacity and inventory. These forecasts often need to be made prior to launch, and updated frequently thereafter. The tilted-Gompertz model I propose here is based on truncating and exponential tilting the Gumbel distribution and its reflection. In examining the skewness of the new model and the benchmarks, I identify a bias in the leading measure of skewness or asymmetry in the diffusion literature. In light of this, I introduce a new measure of skewness, and reveal that the new model can capture a wider range of shapes than existing models. In two empirical studies, one on forecasting Dell computer sales prior to launch, and the other on generating rolling forecasts of search interests of social networking websites, the tilted-Gompertz model outperforms leading diffusion models in both point forecasting and quantile forecasting.

In Chapter 3, I develop a new exponential smoothing model with a life cycle trend. Although exponential smoothing techniques have been widely applied in business decision making, this is the first model that is able to describe a time series that initially grows, slows in its growth until it peaks, and then begins to decline.

The life-cycle trend in the model follows the density of the tilted-Gompertz model developed in Chapter 2. Consequently, the new exponential smoothing model with a life cycle trend can be viewed as the tilted-Gompertz model with time-varying parameters. In addition to the ability of fitting a wider range of skewed life cycles, the new model can also react to local changes in the environment due to its smoothing parameters. The model also includes prior distributions on its parameters. These prior distributions become regularization terms in the model and allow managers to make accurate forecasts from the beginning of a life cycle, which is a notoriously difficult problem. In two empirical studies, I test accuracies of the exponential smoothing model with a life cycle trend (or the tilted-Gompertz model with time-varying parameters) as well as other benchmark models, using the data of computer sales and search interests. Finally, I further illustrate the impact of applying more accurate quantile forecasts in solving newsvendor problems, where the new model in comparison with the benchmarks leads to an increase in firms' profits or a reduction in their costs.

Chapter 4 focuses on applying machine learning techniques to generate distributional forecasts of transfer passenger movements in an airport. In collaboration with Heathrow, I develop a two-phased predictive system that produces forecasts of transfer passenger flows. In the first phase, the system predicts the entire distribution of transfer passengers' connection times. In the second phase, the system samples from the distribution of individual connection times and produces distributional forecasts for the number of passengers arriving at the immigration and security areas. The predictive system developed is based on a regression tree combined with copula-based simulations. I generalize the tree method to predict complete distributions, moving beyond point forecasts. Additionally, I formulate a newsvendor-based resourcing problem to evaluate decisions made by applying the new predictive system. In comparison with the benchmark models, the two-phased approach is more accurate in predicting both connection times and passenger flows. This new approach could also potentially reduce resourcing costs at the immigration and transfer security areas.

Overall, this dissertation aims to contribute to the methodology and application of probabilistic forecasting techniques, within the field of operations management. I hope that this dissertation will add to the growing stream of research on developing new predictive models, and encourage other researchers and practitioners working on applying advanced predictive analysis in tackling decision-making problems.

I am the first author of all chapters. Chapter 2 and 3 were undertaken as joint work with Kenneth C. Lichtendahl and Yael Grushka-Cockayne. Chapter 4 draws on joint work with Yael Grushka-Cockayne and Bert De Reyck. In all chapters, I performed all analyses and wrote all parts of the chapters myself.

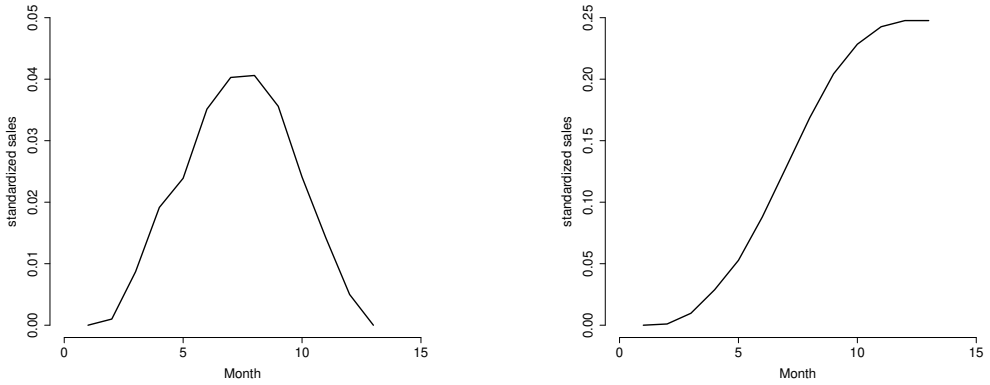
Chapter 2

A Tilted Gompertz Model for Forecasting New Product and Service Life Cycles

2.1 Introduction

Most innovations go through life cycles (Rogers 2010). Customers' demand of a new product often evolves according to a bell-shaped curve that grows initially, slows in its growth until it peaks, and then begins to decline (Figure 2.1(a)). The cumulative demand of the new product can then be described by an S-shaped curve (Figure 2.1(b)). Many important business decisions rely on a manager's forecast of a product or service's life cycle. For instance, the decision about when a social network, such as Twitter, should add server capacity to handle network demand will depend, in part, on predictions of growth in new users. Other decisions inside a firm may occur more frequently and on a large-scale, such as managing the inventory levels of thousands of new products. To support these decisions, managers need a life-cycle model that produces accurate forecasts (Hu et al. 2017, Baardman et al. 2017).

Decisions related to a product or service life cycle often require information about its uncertainty. For example, in a newsvendor setting, the optimal production level of a new product depends on a quantile of the demand distribution. Probabilistic forecasts of tail events, such as how high the network demand could go, are often of interest for setting server capacities. In fact, inaccurate forecasts of network demand could cause severe impacts on technology companies' services in practice. As reported in Zaveri 2019, more than a dozen Google services such as



(a) The period sales of a dell workstation over time

(b) The cumulative sales of a dell workstation over time

Figure 2.1: The Bell-shaped Curve and the S-shaped Curve that Describe a Product's Life Cycle (Data is from Hu et al. 2017).

Gmail and Youtube were disrupted for a few hours on June 2, 2019, due to high levels of network congestion.

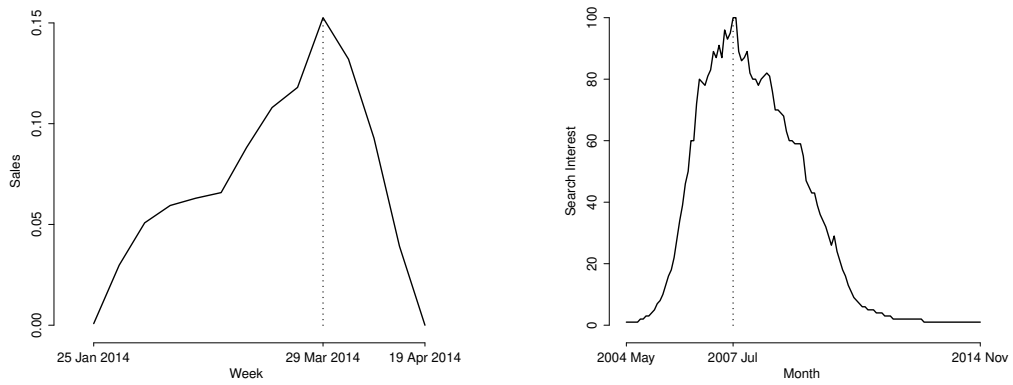
Recently, distributional forecasts have gained popularity in the energy forecasting domain where it is critical to understand the uncertainty in peak demand and build capacity accordingly. Hong et al. (2016) report on the results from the Global Energy Forecasting Competition 2014, which attracted 581 participants from 61 countries. In one of the tracks, participants were asked to make rolling quantile forecasts of electricity demand. Similarly, the recent M4 Competition from 2018 asked participants to make quantile forecasts of various types of time series, including those of microeconomic, macroeconomic and financial quantities (Makridakis et al. 2018). Despite the advances these competitions generated, to our knowledge, how to make accurate quantile forecasts of life cycles remains an open question.

Forecasting product life cycles have been widely studied in the literature (Bass 1969, Easingwood et al. 1983, Bemmaor 1994, Van den Bulte and Joshi 2007, Baardman et al. 2017). These studies, however, mainly focus on point estimates without considering uncertainties. In addition, the most popular life cycle models, such as the Bass diffusion model, are limited in forecasting skewed life cycles. In this chapter, we develop a new model to describe product and service life cycles. The model's trend follows the probability density function of a new distribution, the

tilted-Gompertz distribution, which results from reflecting, truncating, and tilting a Gumbel distribution. Reflecting (or not) and truncating a Gumbel distribution results in a Gompertz distribution, so when we tilt the Gompertz distribution, we call it the tilted-Gompertz distribution. When the model is applied to the adoption of a new product, the tilted-Gompertz distribution describes the uncertain time it takes a consumer to adopt the new product. Its density can fit a wide range of left-skewed, nearly symmetric, and right-skewed shapes. Our model is an extension of the Gompertz diffusion model presented in Mahajan et al. (1986).

When we fit a diffusion model to data, we apply the maximum a posteriori approach to incorporate prior information about the model's parameters learned from past similar products. This approach is new to the literature of forecasting product life cycles. Prior information about the parameters can come from past related series, as demonstrated in our empirical studies, or from managerial judgement. The benefit of including prior information into the model's estimation is that it allows us to forecast a life cycle from its beginning or very early on its evolution, when no or few observations from the time series itself are available. From a planning perspective, decision makers need to order inventory or plan capacity both at the beginning of the life cycle and partway through its evolution. These decision makers will likely want a single forecasting model to make these decisions.

From the marketing literature, we know that the ability to fit skewed diffusions is important as many new-product diffusions exhibit long right or left tails (Dixon 1980, Van den Bulte and Lilien 1997, Van den Bulte and Joshi 2007). Bemmaor and Lee (2002) say these diffusions show "extra-Bass" skew compared to the popular Bass (1969) model. As shown in Figure 2.2, the sales of the Dell workstation grows slower than it declines, making its diffusion left skewed, and the search interests of the social networking website Myspace grows faster than it declines, making its diffusion right skewed. As extensions, several models have been proposed to describe skewed diffusions. Notable among them are the nonuniform influence model (Easingwood et al. 1983), the Gompertz model (Mahajan et al. 1986), the gamma/shifted-Gompertz model (Bemmaor 1994), and the asymmetric influence



(a) Sales of a Dell Workstation (Hu et al. 2017)

(b) Google Search Interest of Myspace (Google Trends 2016)

Figure 2.2: Examples of Processes that Go Through Life Cycles.

model (Van den Bulte and Joshi 2007).

In these studies, two models emerge as natural benchmarks to our model: the Bass and gamma/shifted-Gompertz models. The Bass model has only three parameters that represent the market potential, coefficient of innovation and coefficient of imitation. These parameters all have intuitive implications in the marketing context. Although the Bass model is limited in fitting skewed life cycles, we consider it here because it is intuitively appealing, tractable, and often accurate in practice. Of the models proposed to capture "extra-Bass" skew, the gamma/shifted-Gompertz model that has four parameters stands out because it is a tractable extension of the Bass model. Our tilted-Gompertz model also has four parameters. Although our new model is shown to be more accurate in the empirical studies, it does not have parameters with intuitive implications.

From the operations literature, another benchmark emerges—the trapeziod model, recently posed and applied by Hu et al (2017). The trapezoid model, which has five parameters, is both easy to estimate and easy to explain. Its density follows the shape of a triangle, but with a flat peak corresponding to the life cycle's maturity stage. Hu et al. (2017) note that the trapezoid model applies well to left-skewed life cycles of new products like the high-technology products described in Goldman (1982).

In two empirical studies, we compare the out-of-sample forecasts of the Bass, gamma/shifted-Gompertz, trapezoid, and tilted-Gompertz models. We estimate each of these models using a regularization approach that allows us to incorporate prior information into the model. With prior information as part of the model, we can forecast a life cycle from its very beginning, which is a notoriously difficult problem (Venkatesan et al. 2004).

The data in our empirical studies are time series of sales of 170 new Dell computers and Google Trends search interest in 122 social networks. In these studies, we find that the tilted-Gompertz model performs favorably in out-of-sample forecasting when compared to the benchmark models. The tilted-Gompertz model has, in most instances, more accurate quantile and point forecasts. We believe the tilted-Gompertz model performs well because it can fit a wider range of skewed diffusions.

2.2 Tilted-Gompertz Diffusion Model

In this section, we introduce the tilted-Gompertz diffusion model. At the core of the derivation of the model is a function F of time. Let $F(t)$ be the proportion of eventual adopters that have adopted a product by time t out of a population of m eventual adopters. $N(t)$ is the number of adopters that have adopted by time t, and $E[N(t)] = mF(t)$ is the expected number of adopters that have adopted by time t. The rate of adoption, $mf(t)$, where $f(t)$ is the first derivative of $F(t)$, is the rate of change over time in the expected number of adopters that have adopted by time t.

In a micro-level view, each individual in the population adopts a new product at an uncertain time. The individuals' adoption times are assumed to be independent and identically distributed according to a cumulative distribution function (cdf) F. Hence, the number of adopters that have adopted by time t follows a binomial distribution with probability $F(t)$ and trials m, and $E[N(t)] = mF(t)$ (Schmittlein and Mahajan 1982, Meade and Islam 2006). To fit this view of the process, one maximizes the likelihood of observing the data $y_1, y_2, \ldots, y_n$ by choosing m and the parameters of F where each $y_i = N(t_i) - N(t_{i-1})$ (Srinivasan and Mason 1986).

Next, we derive a new distribution $F(t)$from other well-known distributions in the diffusion literature, such as the Gumbel and Gompertz distributions, to describe the diffusion of a new product. We first reflect the Gumbel distribution to obtain both right and left skewed distributions. We then truncate the resulting distribution below zero since the uncertain time it takes a consumer to adopt a new product should be based on the positive real line. Finally, we apply the exponential tilting technique to introduce a new parameter in the distribution so that it can be more flexible in capturing skewed life cycles.

To begin the construction, we reflect the Gumbel distribution about zero (i.e., take its mirror image) and truncate its reflection below zero. The Gumbel (*Gu*) distribution has the cdf $F_{Gu}(x) = e^{-\rho e^{-\lambda x}}$ for $-\infty < x < \infty$ where $0 < \rho, \lambda < \infty$. We denote the random variable X drawn from a Gumbel distribution by $X \sim Gu(\lambda, \rho)$. For the possibly left-skewed Gompertz (*LeGo*) distribution, we define an adoption time T as a reflected Gumbel random variable $-X$ truncated below zero: $T = (-X|-X \geq 0)$. It has the cdf given by

$$F_{LeGo}(t) = \frac{F_{Gu}(0) - F_{Gu}(-t)}{F_{Gu}(0)} = 1 - e^{-\rho(e^{\lambda t}-1)},$$

for $0 < t < \infty$. Its pdf is $f_{LeGo}(t) = \lambda\rho e^{\lambda t} e^{-\rho(e^{\lambda t}-1)}$.

For the always right-skewed Gompertz (*RiGo*) distribution, we define an uncertain adoption time T as a Gumbel random variable X truncated below zero: $T = (X|X \geq 0)$. It has the cdf given by

$$F_{RiGo}(t) = \frac{F_{Gu}(t) - F_{Gu}(0)}{1 - F_{Gu}(0)} = \frac{e^{-\rho e^{-\lambda t}} - e^{-\rho}}{1 - e^{-\rho}},$$

for $0 < t < \infty$. Its pdf is $f_{RiGo}(t) = \lambda\rho e^{-\lambda t} e^{-\rho e^{-\lambda t}}/(1 - e^{-\rho})$.

We put possibly left-skewed and always right-skewed Gompertz distributions together and define the Gompertz distribution to be the distribution with the pdf

$$f_{Go}(t) = \frac{\lambda\rho}{I_{(0,\infty)}(\lambda) - e^{-\rho}} e^{-\lambda t} e^{-\rho e^{-\lambda t}},$$

for $-\infty < \lambda < \infty$, $\lambda \neq 0$, $0 < \rho < \infty$, and $0 \leq t < \infty$ where the indicator $I_A(z) = 1$ if $z \in A$ and is zero otherwise. The Gompertz distribution's moment generating evaluated at a point θ is given by

$$M_{Go}(\theta) = \int_0^\infty e^{\theta t} f_{Go}(t)\, dt = \frac{\lambda\rho}{I_{(0,\infty)}(\lambda) - e^{-\rho}} \int_0^\infty e^{(\theta-\lambda)t} e^{-\rho e^{-\lambda t}}\, dt$$
$$= \begin{cases} \frac{\rho}{1-e^{-\rho}} \int_0^\rho (\frac{u}{\rho})^{-\frac{\theta-\lambda}{\lambda}} e^{-u} \frac{1}{u}\, du & \text{for} \quad 0 < \lambda < \infty \\ \frac{\rho}{e^{-\rho}} \int_\rho^\infty (\frac{u}{\rho})^{-\frac{\theta-\lambda}{\lambda}} e^{-u} \frac{1}{u}\, du & \text{for} \quad -\infty < \lambda < 0 \end{cases},$$

where $u = \rho e^{-\lambda t}$ and $du = -\rho\lambda e^{-\lambda t}\, dt$. To simplify the result, we let $\delta = -(\theta - \lambda)/\lambda$ and therefore $\theta = \lambda(1-\delta)$. The moment generating function them becomes

$$M_{Go}(\theta) = \begin{cases} \frac{\rho}{1-e^{-\rho}} \int_0^\rho (\frac{u}{\rho})^{\delta} e^{-u} \frac{1}{u}\, du & \text{for} \quad 0 < \lambda < \infty \\ \frac{\rho}{e^{-\rho}} \int_\rho^\infty (\frac{u}{\rho})^{\delta} e^{-u} \frac{1}{u}\, du & \text{for} \quad -\infty < \lambda < 0 \end{cases}$$
$$= \begin{cases} \frac{\rho^{1-\delta}}{1-e^{-\rho}} \int_0^\rho u^{\delta-1} e^{-u}\, du & \text{for} \quad 0 < \lambda < \infty \\ \frac{\rho^{1-\delta}}{e^{-\rho}} \int_\rho^\infty u^{\delta-1} e^{-u}\, du & \text{for} \quad -\infty < \lambda < 0 \end{cases}$$
$$= \rho^{1-\delta} \frac{\gamma(\delta,\rho) - I_{(-\infty,0)}(\lambda)\Gamma(\delta)}{I_{(0,\infty)}(\lambda) - e^{-\rho}},$$

where $\gamma(\delta, s) = \int_0^s z^{\delta-1} e^{-z}\, dz$ is the lower incomplete gamma function, and $\Gamma(\delta) = \int_0^\infty z^{\delta-1} e^{-z}\, dz$ is the gamma function. Note that $\gamma(1,\rho) = 1 - e^{-\rho}$. Next, we exponential tilt the Gompertz distribution so that the new distribution can capture a wider range of skewness. Exponential tilting is a technique that is often used in rare event simulation to bias the distributions so that the likelihood of rare events increases (Siegmund 1976, Butler 2007). In the next session, we will discuss the range of skewness that can be captured by the new model as well as other well-known diffusion models. To tilt the Gompertz distribution, we multiply its pdf by $e^{\lambda(1-\delta)t}$ and divide by the moment generating function $M_{Go}(\lambda(1-\delta))$, which yields the pdf of the tilted-Gompertz (*TiGo*) distribution

$$f_{TiGo}(t) = \frac{e^{\lambda(1-\delta)t} f_{Go}(t)}{M_{Go}(\lambda(1-\delta))} = \frac{\lambda(\rho e^{-\lambda t})^{\delta} e^{-\rho e^{-\lambda t}}}{\gamma(\delta,\rho) - I_{(-\infty,0)}(\lambda)\Gamma(\delta)}, \tag{2.1}$$

for $0 < \delta < \infty$, $-\infty < \lambda < \infty$, $\lambda \neq 0$, $0 < \rho < \infty$, and $0 < t < \infty$. Its cdf is given by

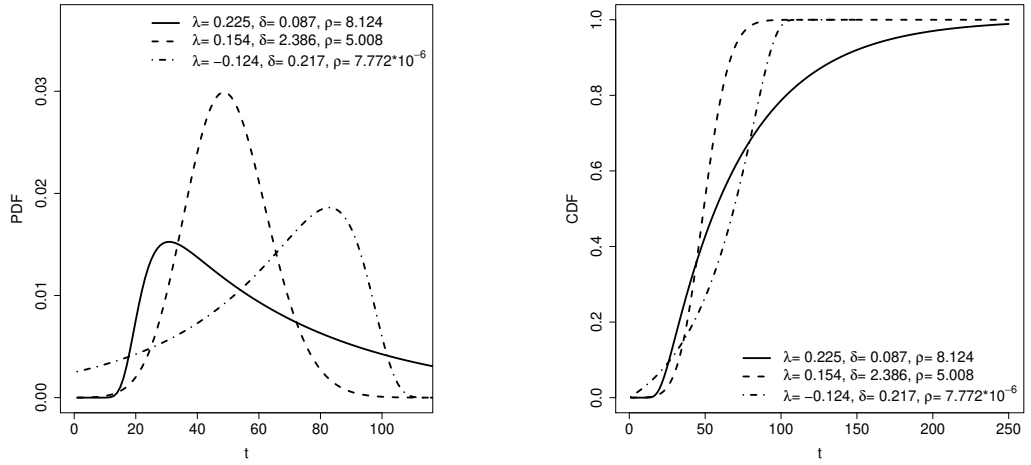
$$F_{TiGo}(t) = \frac{\lambda}{\gamma(\delta,\rho) - I_{(-\infty,0)}(\lambda)\Gamma(\delta)} \int_0^t (\rho e^{-\lambda s})^\delta e^{-\rho e^{-\lambda s}} ds$$
$$= \begin{cases} -\frac{1}{\gamma(\delta,\rho)-\Gamma(\delta)} \int_\rho^{\rho e^{-\lambda t}} u^{\delta-1} e^{-u} du & \text{for} \quad -\infty < \lambda < 0 \\ \frac{1}{\gamma(\delta,\rho)} \int_{\rho e^{-\lambda t}}^{\rho} u^{\delta-1} e^{-u} du & \text{for} \quad 0 < \lambda < \infty \end{cases},$$

where again $u = \rho e^{-\lambda s}$ and $du = -\rho\lambda e^{-\lambda s} ds$. The result of the steps described above is the tilted-Gompertz distribution, which has the cdf

$$F_{TiGo}(t) = \frac{\gamma(\delta,\rho) - \gamma(\delta,\rho e^{-\lambda t})}{\gamma(\delta,\rho) - I_{(-\infty,0)}(\lambda)\Gamma(\delta)},$$

for $-\infty < \lambda < \infty$, $\lambda \neq 0$, and $0 < \delta, \rho < \infty$. The distribution has three parameters: λ, δ, and ρ. We call these parameters the scale, tilting, and shape parameters, respectively. The tilted-Gompertz diffusion model can now be described as $mF_{TiGo}(t)$, where m can be considered as market potential or the number of eventual adopters. Figure 2.3(a) and (b) present a few examples of the pdf and cdf of the tilted-Gompertz distribution.

When the tilting parameter δ equals one, the tilted-Gompertz distribution specializes to the Gompertz distribution. The left-skewed (right-skewed) Gompertz



(a) pdf of the tilted-Gompertz distribution

(b) cdf of the tilted-Gompertz distribution

Figure 2.3: Examples of the pdf and cdf of the tilted-Gompertz Distribution.

distribution is the tilted-Gompertz distribution with $\lambda < 0$ ($\lambda > 0$) and $\delta = 1$. The left-skewed Gompertz distribution was introduced by Gompertz (1825) and is well known as a life distribution (Johnson et al. 1995, Marshall and Olkin 2007). It has also been applied in modelling product diffusions in the literature (Mead and Islam 2006).

The peak time, or the mode of the distribution, can be derived from setting the first derivation of Eq. (2.1) to zero

$$f'_{TiGo}(t) = (\rho e^{-\lambda t} - \delta)\lambda f(t) = 0,$$

To summarize, the tilted-Gompertz model has the following properties:

$$\begin{aligned}
\text{Cumulative adoptions at } t: &\quad m\frac{\gamma(\delta,\rho) - \gamma(\delta,\rho e^{-\lambda t})}{\gamma(\delta,\rho) - I_{(-\infty,0)}(\lambda)\Gamma(\delta)} \\
\text{Rate of adoptions at } t: &\quad m\frac{\lambda(\rho e^{-\lambda t})^{\delta} e^{-\rho e^{-\lambda t}}}{\gamma(\delta,\rho) - I_{(-\infty,0)}(\lambda)\Gamma(\delta)} \\
\text{Peak time } t^*: &\quad -\log(\delta/\rho)/\lambda \text{ if } \lambda(\rho-\delta) > 0; 0 \text{ otherwise} \\
\text{Rate of adoptions at } t^*: &\quad m\frac{\lambda\delta^{\delta}e^{-\delta}}{\gamma(\delta,\rho) - I_{(-\infty,0)}(\lambda)\Gamma(\delta)} \text{ if } \lambda(\rho-\delta) > 0 \\
&\quad m\frac{\lambda\rho^{\delta}e^{-\rho}}{\gamma(\delta,\rho) - I_{(-\infty,0)}(\lambda)\Gamma(\delta)} \text{ otherwise,}
\end{aligned} \tag{2.2}$$

For details on the mathematical properties of the tilted-Gompertz diffusion model and other leading diffusion models (Bass, gamma/shifted-Gompertz, and trapezoid), see the Appendix .

2.3 Measuring a Diffusion's Skewness Around its Mode

Next we introduce a new measure of skewness and show how the tilting parameter δ and the shape parameter ρ work in conjunction to control the degree of skewness of the tilted-Gompertz distribution. For modelling diffusions with long takeoffs or with different rates of growth before the peak and decline afterward, such as the consumer durable goods studied in Golder and Tellis (1997), the shapes that the

tilted-Gompertz distribution can achieve may be useful.

The motivation for proposing the new skewness measure is to measure a diffusion's "extra-Bass" skew, starting from a baseline of zero. The Bass distribution is always symmetric around its peak. Based on existing skewness measures, however, the skewness of the Bass distribution is often positive. Using our new measure, the Bass distribution always has zero skewness, and any distributions with positive (negative) skewness values are considered to be right (left) skewed.

To apply our new measure of skewness, we assume that F is a cdf for an adoption time T, taking values on the non-negative interval $[0,t_{max})$, and that F has a unique mode at time $t^* > 0$. The mode here is the time $t^* > 0$ at which the pdf f reaches its single peak. Specifically, $f(t) > 0$ for $t \in (0,t_{max})$, either $f(0) > 0$ or $f(0) = 0$, $f'(t) > 0$ for $t \in [0,t^*)$, $f'(t) = 0$ for $t = t^*$, and $f'(t) < 0$ for $t \in (t^*,t_{max})$. Let $t^{**} > t^*$ be the adoption time (possibly infinite) such that $f(t^{**}) = f(0)$. For distributions that satisfy these assumptions, we define our measure of local skewness with respect to the mode t^* as

$$Skew(F) = 1 - 2\frac{F(t^*)}{F(t^{**})}.$$

A value of $Skew(F)$ in the interval $(-1,0)$ indicates left skewness, and a value in $(0,1)$ indicates right skewness.

For the tilted-Gompertz distribution, the mode is at $t^* = -\log(\delta/\rho)/\lambda$, and $t^{**} = -\log(-(\delta/\rho)W(-(\delta/\rho)e^{-\delta/\rho})/\lambda$ where W is the Lambert W function. For $\lambda > 0$, we use the principal branch W_0 of the Lambert W function, and for $\lambda < 0$, we use the non-principal branch W_{-1}. (See Corless et al. 1996 for details on the Lambert W function.) The local skewness of the tilted-Gompertz distribution is

$$Skew(F_{TiGo}) = 1 - 2\frac{\gamma(\delta,\rho) - \gamma(\delta,\delta)}{\gamma(\delta,\rho) - \gamma(\delta,-\delta W(-(\delta/\rho)e^{-\delta/\rho}))}, \tag{2.3}$$

for $\rho < \delta$ if $\lambda < 0$ (left-skewed) and $\rho > \delta$ if $\lambda > 0$ (right-skewed). Note that the tilted-Gompertz distribution's mode is at zero when $\lambda(\rho - \delta) < 0$; and therefore, its local skewness is undefined in this case.

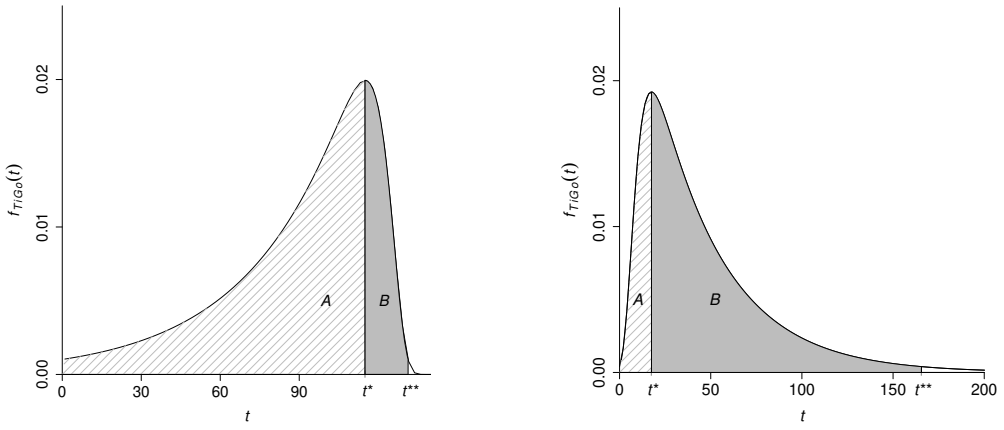
The cdf evaluated at its inflection point t^* has a long history as a measure of skewness or asymmetry. Pearl and Reed (1925) and Winsor (1932) use $F(t^*)$ to describe the skewness of a general growth curve mF. Easingwood et al. (1983) and Mahajan et al. (1990) use $F(t^*)$ to describe asymmetric diffusions. Arnold and Groeneveld (1995) use the related $1-2F(t^*)$ to measure skewness with respect to the mode of any distribution with a single-peaked density.

Our new measure is a localized version of Arnold and Groeneveld's measure $1-2F(t^*)$, localized to the interval $[0,t^{**}]$ when t^{**} is finite. Using our new skewness measure, the Bass distribution has zero local skewness with respect to its mode: $Skew(F_{Ba}) = 0$ for $p < q$. This starting point makes sense because the Bass distribution's pdf is formally symmetric around its mode on the interval $[0,2t^*]$. For the Bass distribution, $t^{**} = 2t^*$.

And yet, Arnold and Groeneveld's measure of the Bass distribution's skewness is always positive. This indication of right skewness is an artifact of truncation. The Bass distribution is a logistic distribution truncated below zero. This truncation makes sense because an adoption time is naturally non-negative. A manager, however, who hears a summary statistic indicating that a new product's diffusion will be right-skewed—merely because a left tail was truncated—may jump to the conclusion that sales will grow more quickly up to its peak than it will decline thereafter.

To understand how our measure works in more detail, we express it as the difference between two conditional probabilities: $Skew(F) = P(t^* < T \leq t^{**} | 0 < T \leq t^{**}) - P(0 < T \leq t^* | 0 < T \leq t^{**})$. Figure 2.4 illustrates the difference between these probabilities using two areas, labeled A and B. The probability $P(0 < T \leq t^* | 0 < T \leq t^{**})$ is equal to $A/(A+B)$, and the probability $P(t^* < T \leq t^{**} | 0 < T \leq t^{**})$ is equal to $B/(A+B)$. With $A > B$, we have left skewness, as depicted in Figure 2.4(a). With $A < B$, we have right skewness, as depicted in Figure 2.4(b).

Our new measure compares the conditional probabilities of adoption in two time intervals: a left interval from zero to the mode and a right interval from the mode to t^{**}. Specifically, $Skew(F) = B/(A+B) - A/(A+B) = 1 - 2A/(A+B)$. In contrast, Arnold and Groeneveld's measure $1-2F(t^*) = 1-2A$ compares the



(a) Left skewness: $\lambda = -0.2$, $\delta = e^{-2}$, $\rho = e^{-25}$, and $Skew(F_{TiGo}) = -0.59$

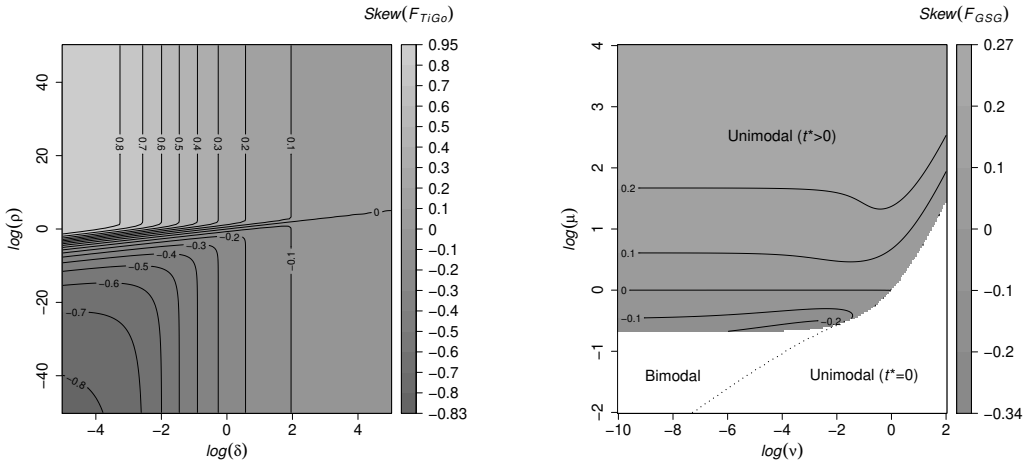
(b) Right skewness: $\lambda = 0.2$, $\delta = e^{-2}$, $\rho = e^{1.5}$, and $Skew(F_{TiGo}) = 0.60$

Figure 2.4: Two Probability Density Functions from the Tilted-Gompertz Distribution.

conditional probabilities in the same left interval and a right interval from the mode to infinity. Because their right interval is longer, Arnold and Groeneveld's measure is biased toward indicating right skewness. In fact, $Skew(F) < 1 - 2F(t^*)$ always holds.

In Figure 2.5(a), we see the contours of local skewness values that the tilted-Gompertz distribution can achieve over a large space of settings for δ and ρ. According to the expression in Eq. (2.3), only the parameter values of δ and ρ and the sign of λ affect its local skewness. Here in the figure, the local skewness values range between -0.83 and 0.95. The effect of decreasing δ is to widen the range of skewness values that the tilted-Gompertz distribution can achieve, approaching $(-1,1)$ as δ gets small.

In Figure 2.5(b), we see the contours of local skewness values that the gamma/shifted-Gompertz distribution can achieve. These values are always between -0.34 and 0.27. Thus, the gamma/shifted-Gompertz distribution is limited in the amount of "extra-Bass" skew it can capture. We also see in Figure 2.5(b) the regions for which the gamma/shifted-Gompertz distribution is unimodal with its mode at zero, unimodal with a positive mode, or bimodal with one mode at zero. Recall that our measure of local skewness does not apply to distributions with a



(a) Tilted-Gompertz Distribution

(b) Gamma/shifted-Gompertz Distribution

Figure 2.5: Contours of Local Skewness for the Tilted-Gompertz and Gamma/shifted-Gompertz Distributions.

mode at zero. The condition for a bimodal gamma/shifted-Gompertz model and other properties of this model are given in Table 2.5 in the Appendix.

With regard to skewness, Bemmaor and Lee (2002, p. 211) report that "As shown in [their] Figure 1, the [gamma/shifted-Gompertz] model captures 'extra-Bass' skew in the data for given levels of p and q. When $[\mu]$ is less than 1, there is more right skew than Bass, and when $[\mu]$ is larger than 1, there is more left skew than Bass." But here we find the opposite for the gamma/shifted-Gompertz distribution with μ greater than 1. As μ increases, the gamma/shifted-Gompertz distribution becomes more right-skewed than the Bass distribution. Furthermore, it approaches a limit of around 0.27. Therefore, the gamma/shifted-Gompertz model may not be suitable for strongly right-skewed diffusions with long takeoffs.

A recent proposal that can capture the full range of skewed diffusions is the trapezoid model of Hu et al. (2017). Its rate of adoption is specified by five parameters as shown in the Appendix. Note that for any applicable growth curve mF, $Skew(F) = Skew(mF)$, since the eventual number of adopters m cancels out in our skewness measure.

2.4 Model Estimation and Prediction Distribution

In this section, we introduce the method that is applied to estimate the parameters of our tilted-Gompertz model and other benchmark models. This method applies to a life cycle that has already begun to evolve or to one that has yet to kick off. Because life cycles are notoriously difficult to forecast from pre-peak data or with no data at all (Mahajan et al. 1990, Xie et al. 1997, Bemmaor and Lee 2002, Venkatesan et al. 2004), we incorporate prior information about the model's parameters into our estimation procedure. This prior information may come from similar previously launched products as discussed in Baardman et al. (2017), Hu et al. (2018), and Ban et al. (2018). Sometimes, it will not be possible to identify similar past life cycles. A manager will need to rely on his or her judgement to inform the prior distributions in the model.

The per-period adoptions $N(t_i) - N(t_{i-1})$ are often estimated with additive errors in the literature:

$$\begin{aligned} N(t_i) - N(t_{i-1}) &= E[N(t_i) - N(t_{i-1})] + \varepsilon_i \\ &= mF(t_i) - mF(t_{i-1}) + \varepsilon_i, \end{aligned} \tag{2.4}$$

where each time interval $(t_{i-1}, t_i]$ is one period and the errors $\varepsilon_1, \ldots, \varepsilon_n$ are independent and identically distributed according to a normal distribution with mean zero and variance σ^2. The model in (2.4) is fit by choosing the parameters so as to maximize the model's normal likelihood function. This type of maximum likelihood estimation is called non-linear least-squares in the literature (Srinivasan and Mason 1986, Van den Bulte and Lilien 1997, Bemmaor and Lee 2002).

The typical problem with maximum likelihood estimation is that, before the peak, with only a few data points observed, the model is sensitive to each of the data points. If the last observed data point falls well above the previous few data points, the model's forecast will overshoot the eventual peak. In this situation, if the last observed data point is high due to noise, the model will be overfit to this noise (or error) and will not be well fit to the signal (or trend). Overfit models make poor out-of-sample forecasts. One way to avoid this overfitting problem is to use a

Bayesian approach.

A popular way to avoid model overfitting is to include a regularization term in a maximum likelihood approach. In a linear regression setting, the objective is to choose parameters that minimize the negative of the log-likelihood function plus some regularization terms. For instance, in a ridge regression (Hoerl and Kennard 1970), the regularization terms are the scaled sum of the squares of the coefficients in a linear regression model. In the lasso (Tibshirani 1996), the regularization terms are the scaled sum of the absolute values of the coefficients in a linear regression model. These regularization terms penalize parameters when they stray too far away from zero. A priori, one may have a strong belief that coefficients in a linear regression model are near zero, so the regularization terms in these objective functions also have a Bayesian interpretation. In the ridge regression, each regularization term is proportional to the logarithm of a normal prior distribution (with mean zero) of a coefficient. In the lasso, each regularization term is proportional to the logarithm of a Laplace prior distribution (with mean zero) of a coefficient.

The approach where one maximizes the log-likelihood function plus the logarithm of the prior distribution is called maximum a posteriori (Geman and Geman 1984). Maximum a posteriori (MAP) involves choosing a model's parameters so as to maximize the posterior distribution of the parameters. The posterior distribution is proportional to the product of the prior distribution of the parameters and the likelihood of the observed data. Consequently, ridge regression and the lasso can be viewed as MAP estimation procedures.

When there is no data and hence no likelihood, one chooses the model's parameters so as to maximize the prior distribution of the parameters, which is still consistent with MAP. In this case, forecasts come from the model with parameters set to the mode of the prior distribution. When there is data available, forecasts come from the model with parameters set to the mode of the posterior distribution.

To estimate the tilted-Gompertz model, we start with a prior distribution of the tilted-Gompertz distribution's parameters. We let parameters $(\lambda, \log(\delta), \log(\rho), \log(m))$ be jointly normally distributed with density denoted

by f_N. In addition, we let each error's precision σ^{-2} be independent of $(\lambda, \log(\delta), \log(\rho), \log(m))$ and distributed according to a gamma distribution with shape $a_\sigma > 1$ and rate $b_\sigma > 0$. The parameters λ, δ, ρ, m, and σ^{-2} are chosen to maximize the logarithm of these parameters' posterior distribution

$$\max_{\lambda,\delta,\rho,m,\sigma^{-2}} \Bigg\{ \log\left(f_N(\lambda, \log(\delta), \log(\rho), \log(m))\right) + \log\left|\frac{1}{\delta\rho m}\right| + (a_\sigma - 1)\log(\sigma^{-2}) - b_\sigma\sigma^{-2} + \frac{1}{2}\sum_{i=1}^{t}\left(\log(\sigma^{-2}) - \sigma^{-2}(y_i - \hat{y}_i)^2\right)\Bigg\},$$

where y_i is the ith observation in the time series (the period demand during $(i-1, i]$), $\hat{y}_i = \hat{N}_i - \hat{N}_{i-1}$ with $\hat{N}_0 = 0$, and $\hat{N}_i$ is calculated according to the expression of cumulative adoptions in Eq. (2.2). Later in the empirical studies, we also build models to forecast the cumulative adoption as it is required in the decision-making problem. In this case, we simply replace y_i and $\hat{y}_i$ in the above function with N_i and $\hat{N}_i$, respectively.

The first two lines in the objective above represent the logarithm of the prior distribution (up to a constant). The second term in the first line is the Jacobian associated with the transformation from $(\lambda, \log(\delta), \log(\rho), \log(m))$ to $(\lambda, \delta, \rho, m)$. The third line in the objective represents the logarithm of the likelihood of the observed data according the model (up to a constant).

Once the parameters are estimated using MAP, the h-step-ahead prediction distribution, made at t, is described by the equation

$$y_{t+h} = m\frac{\gamma(\delta, \rho e^{-\lambda(t+h-1)}) - \gamma(\delta, \rho e^{-\lambda(t+h)})}{\gamma(\delta, \rho) - I_{(-\infty,0)}(\lambda)\Gamma(\delta)} + \varepsilon_{t+h},$$

where ε_{t+h} is normally distributed with mean zero and variance σ^2. Consequently, y_{t+h} is normally distributed with mean $m(\gamma(\delta, \rho e^{-\lambda(t+h-1)}) - \gamma(\delta, \rho e^{-\lambda(t+h)}))/(\gamma(\delta, \rho) - I_{(-\infty,0)}(\lambda)\Gamma(\delta))$ and variance σ^2. If the model is built to estimate the cumulative adoptions up to time $t+h$, the prediction distribution can

be described by

$$N_{t+h} = m\frac{\gamma(\delta,\rho)-\gamma(\delta,\rho e^{-\lambda(t+h)})}{\gamma(\delta,\rho)-I_{(-\infty,0)}(\lambda)\Gamma(\delta)}+\varepsilon^*_{t+h},$$

where ε^*_{t+h} follows normal distribution with mean zero and variance σ^{*2}. Consequently, N_{t+h} is normally distributed with mean $m(\gamma(\delta,\rho)-\gamma(\delta,\rho e^{-\lambda(t+h)}))/(\gamma(\delta,\rho)-I_{(-\infty,0)}(\lambda)\Gamma(\delta))$ and variance σ^{*2}.

For the Bass and gamma/shifted-Gompertz models, we also use the approach above to express our likelihood, following Srinivasan and Mason (1986) and Bemmaor and Lee (2002). For the trapezoid model, we use the pdf approach to express our likelihood, which follows Hu et al. (2017). In this approach, period demand is approximated using $mf_i(t)$, where $f_i(t)$ is the pdf of the trapezoid model.

To estimate the benchmark models, we include prior distributions on their parameters and apply the MAP procedure as well. For the Bass model, we assume $\log(p)$, $\log(q)$, and $\log(m)$ are jointly normally distributed. For the gamma/shifted-Gompertz model, we assume $\log(\lambda)$, $\log(\nu)$, $\log(\mu)$, and $\log(m)$ are jointly normally distributed. And for the trapezoid model, we assume $\log(a)$, $\log(b)$, $\log(-c)$, $\log(\tau_1)$, and $\log(\delta)$ are jointly normally distributed where $\tau_2=\tau_1+\delta$. Also, we assume σ^{-2} is gamma distributed as above.

Thus, to estimate the Bass model using MAP, we choose the parameters p, q, m, and σ^{-2} to maximize the logarithm of these parameters' posterior distribution

$$\max_{p,q,m,\sigma^{-2}}\Bigg\{\log\left(f_N(\log(p),\log(q),\log(m))\right)+\log\left|\frac{1}{pqm}\right|+(a_\sigma-1)\log(\sigma^{-2})-b_\sigma\sigma^{-2} +\frac{1}{2}\sum_{i=1}^{t}\left(\log(\sigma^{-2})-\sigma^{-2}(y_i-\hat{y}_i)^2\right)\Bigg\},$$

where y_i is the ith observation in the time series, $\hat{y}_i=\hat{N}_i-\hat{N}_{i-1}$ with $\hat{N}_0=0$, and $\hat{N}_i$ is calculated as m times the cdf of the Bass distribution shown in Table 2.5. To estimate the other two benchmark models, the approach is similar. Note that for the trapezoid model, $\hat{y}_i$ is equal to $mf(t_i)$. Again, we also build models to forecast cumulative adoptions in the empirical studies using the benchmark models. In this

case, we simply replace y_i and $\hat{y}_i$ in the above function with N_i and $\hat{N}_i$, respectively.

Similarly to the tilted-Gompertz model, when we generate predictions from the benchmark models, the distribution of the period adoption at $t+h$ is a normal distribution with mean $mF(t+h)-mF(t+h-1)$ and variance σ^2. The cumulative adoptions at $t+h$ can be predicted by a normal distribution with mean $mF(t+h)$ and variance σ^{*2}. The expression of F for the Bass, gamma/shifted-Gompertz and the trapezoid model can be found in the Appendix.

In the empirical studies below, we notice that the proposed tilted-Gompertz model runs slightly slower than the Bass model, but is similar to the gamma/shifed-Gompertz and trapezoid models. It also runs faster than the Bass model that is estimated by extended Kalman filter by an order of magnitude.

2.5 Empirical Studies

In this section, we present the results of two empirical studies. These studies represent evaluations of our model's forecasting performance on real data, when compared to other benchmark models. In the first study, we forecast the entire life cycles of 170 different new Dell computers. These Dell computers are made-to-stock products, and their life cycles are complete and relatively short. Therefore, accurate pre-launch estimates of the entire life cycle are important for capacity planning and inventory management purposes. In the second study, we generate rolling forecasts of search interest in 122 social networks. The forecasting problem in our second study is similar to the problem faced by firms, such as Google and Facebook, who need to forecast search queries or views of videos frequently and on a rolling basis. Accurate forecasts of these quantities are critical in handling network demand and planning advertising inventories.

Before we present these results, we provide some detail on how we evaluate the competing models' forecasts. After describing the accuracy measures, we explain how we develop the prior distributions for the models. Note that in this section we only show that our model can provide improvements in forecasting accuracy. Later in Section 3.5, we extend our evaluations by examining newsvendor-related

decisions made using the quantile forecasts generated from the models.

2.5.1 Model Evaluation

To evaluate the accuracy of our quantile forecasts, we use the popular pinball loss function (Koenker and Bassett 1978). The pinball loss of the p quantile (Q_p) given the realization y is

$$L(Q_p, y) = \begin{cases} p(y - Q_p) & \text{for} \quad Q_p \leq y \\ (1-p)(Q_p - y) & \text{for} \quad Q_p > y \end{cases}.$$

This scoring rule has been shown to correspond to the payoffs in a newsvendor problem, when applied to the critical quantile (Jose and Winkler 2009, Grushka-Cockayne et al. 2017). Thus, it has both statistical and economic foundations.

In our empirical studies, we use the pinball loss to evaluate the 0.01, 0.02, ..., 0.99 quantiles that can roughly describe the entire distribution. We report hit rate, the percentage of realizations that fall within a central prediction interval, to measure the 50% and 90% prediction intervals. Hit rate is the most intuitive measure to evaluate quantiles. A 50% (or 90%) prediction interval should contain the realizations 50% (or 90%) of the time, and thus has a hit rate of 50% (or 90%). A low hit rate indicates overconfidence while a high hit rate indicates underconfidence. We also evaluate the median as a point forecast using other common evaluation metrics: mean absolute error (MAE) and root mean squared error (RMSE). One important connection between the pinball loss and the MAE is that the pinball loss of the median is equal to one half of the MAE of the median.

2.5.2 Development of Prior Distributions

We use a prior-information fold to generate prior distributions for our estimation procedure. The method of creating the prior-information fold will be discussed in Section 2.5.3 and 2.5.4. To specify the prior distributions of the models, we take two passes over the prior-information fold. In the first pass, we fit the models using maximum likelihood estimation to the time series in the prior-information fold. To estimate the normal prior distribution's mean, we fit a second-stage model to the

average of the first-stage models' fitted life cycles. The parameters of this second-stage model become the normal prior distribution's mean for the out-of-sample fold. From the matrix of estimated parameters (which has one row of parameter estimates for each time series in the prior-information fold), we then calculate the robust covariance and use it as the normal prior distribution's covariance.

Next, we take a second pass through the prior-information fold to estimate the prior gamma distribution for the error terms' precision σ^{-2}. A grid search to minimize the overall average pinball loss of the 99 quantiles described above determines the final values for the prior distributions' hyperparameters. Specifically, in the second pass, we fit the models using MAP and the prior distributions specified in the first pass. For each setting of the hyperparameters, we then make out-of-sample forecasts and record their average pinball losses. We then find the set of hyperparameters that gives us the lowest average pinball loss. The steps of this two-fold process are summarized in Figure 2.6.

Instead of estimating the prior distributions for σ^{-2} over the first pass, we use this two-pass approach because σ^{-2} determines the width of a prediction interval, which has a significant impact on the performance of the distributional forecasts. In addition, the prior distributions obtained by minimizing the pinball loss should

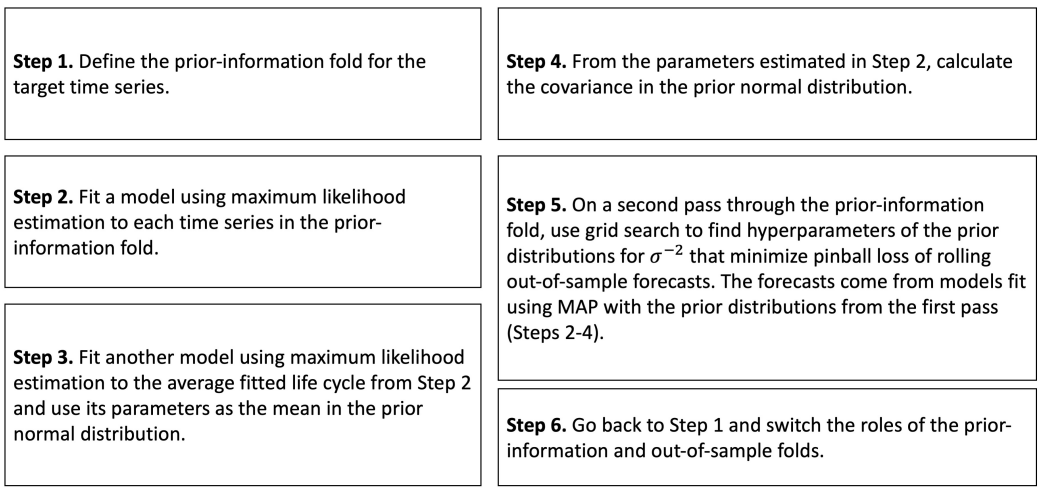


Figure 2.6: Steps of the Two-Fold Process Used to Develop Prior Distributions.

provide better quantile forecasts because the same loss function is used to measure the out-of-sample forecasting accuracy.

2.5.3 Study 1: Forecasting New Computer Sales

For this study, we used the computer sales data described in Acimovic et al. (2018) and analyzed in Hu et al. (2017). These data include weekly sales of 170 different new Dell computers (fixed and mobile workstations, laptops, and desktops) in the United States from 2013 to 2016. During this period, 4,037,825 units of these products were sold, worth over one billion dollars in revenue. The publicly available data from Acimovic et al. (2018) were already normalized so that the total sales over any complete life cycle is equal to one. The mean and median values of the per-period sales in this data set are 0.0166 and 0.0167, respectively.

The typical practice at Dell is to make 34-weeks-ahead forecasts on a quarterly basis from time zero. However, since the median length of the life cycles in this data set is short (40 weeks), we follow Hu et al. (2017) and forecast the complete life cycle of each new product from its very beginning. In their approach, Hu et al. (2017) compare clustering products by feature (processor, RAM, etc.), category (workstation, laptop, etc.), and by data-driven approach (using only the time series). Since we do not have access to the proprietary data on product features and categories, we apply their data-driven algorithm to cluster the products into four categories as suggested in their study. This algorithm, as also outlined in Chouakria and Nagabhushan (2007), is based on hierarchical clustering using a distance measure that addresses the proximity of magnitude in two time series at the same point as well as their first-order temporal correlation.

For each of the products in the data set, we define its prior-information fold as the products whose end date is on or before the launch date of this product. We use the prior-information fold to inform our prior distributions, and then make out-of-sample forecasts of the testing time series. We test model accuracies on only 137 time series in the data set, as no products' life cycles ended before the launch date of the remaining 33.

We first follow Hu et al. (2017) and forecast per-period sales. Table 2.1

presents the pinball losses and point forecasting errors. Each entry in the first column of numbers is a model's pinball loss averaged over the 99 quantiles, over the 137 time series, and over all steps ahead. The next two columns present the hit rates of the central 50% and 90% prediction intervals. In the last two columns, we provide two popular point forecasting error measures.

The tilted-Gompertz model has the lowest overall average pinball loss. The difference in the overall average pinball loss between the tilted-Gompertz model and the second best model (the gamma/shifted-Gompertz model) is statistically significant at the 1% level. The tilted-Gompertz model also has the best hit rates for the 50% and 90% central prediction intervals. Among the 137 time series tested, our new model performs better than the others 71% of the time based on the pinball loss. In terms of point forecasting errors, our model outperforms the second-best model for 73% of the time.

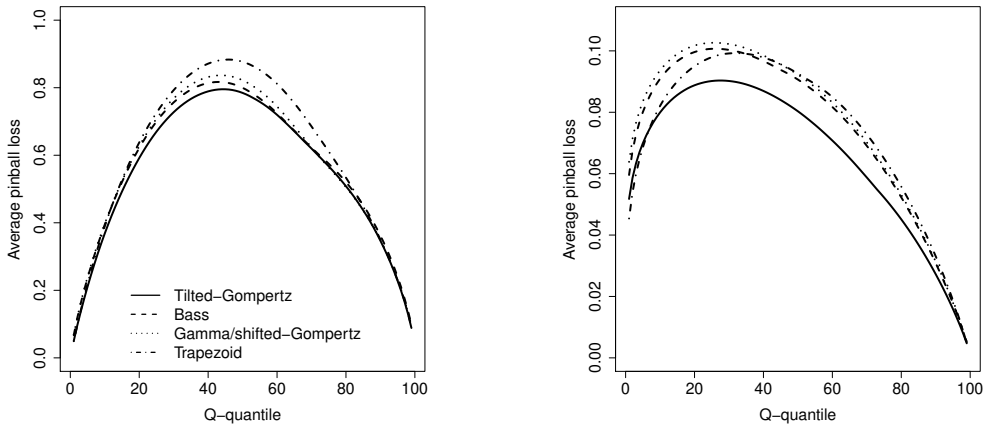
Figure 2.7(a) shows the models' average pinball losses for each of the 99 quantiles. The pinball losses are first averaged over the 137 time series and then over all steps ahead. The gamma/shifted-Gompertz model is the best at the 0.72 to 0.85 quantiles, and the tilted-Gomperz model is the best at the remaining 85 tested quantiles. In terms of point forecasting, the tilted-Gompertz model outperforms the other benchmarks in both MAE and RMSE. The differences between our method and the second best model (the Bass model), however, are not statistically significant.

In addition to the per-period sales, we also test the models' ability to forecast cumulative sales. As described in Hu et al. (2017), demand planners at Dell typi-

	Quantile Forecasting Errors			Point Forecasting Errors	
	100×Avg. pinball loss	Hit rate (0.5)	Hit rate (0.9)	MAE	RMSE
Bass	0.579	0.490	0.857	0.0163	0.0207
Gamma/shifted-Gompertz	0.577	0.409	0.858	0.0164	0.0207
Trapezoid	0.611	0.364	0.842	0.0175	0.0216
Tilted-Gompertz	**0.561**	**0.506**	**0.882**	**0.0157**	**0.0200**
Significance[a]	**	-	-		

[a] Here we check whether the difference of each column's Tilted-Gompertz loss (except for the hit rates) and the loss of the next best model is significant. The symbol ** indicate significance at the 1% level. Values in bold indicate the lowest errors.

Table 2.1: Errors of Forecasts for the Per-period Sales of Dell New Computers.



(a) Pinball losses for Per-period Sales Forecasts

(b) Pinball losses for Cumulative Sales Forecasts

Figure 2.7: The Average Pinball Losses for Each of the 99 Quantile Forecasts of New Computer Sales.

cally make 34-weeks ahead forecasts before the product is launched on the market. They then make decisions on order quantities from overseas manufacturing partners based on the aggregate demand over the next 34 weeks. In Table 2.2, we present the accuracy results of the forecasts for the cumulative sales, made before the product is launched on the market. The tilted-Gompertz model outperforms the benchmark models in all five measures. The differences in the pinball loss and point forecasting errors between the tilted-Gompertz model and the second best models are statistically significant at the 0.1% and 10% levels, respectively. Our tilted-Gompertz model performs better than the others 69% and 71% of the time based on the pinball loss and point forecasting errors, respectively.

Figure 2.7(b) shows the models' average pinball losses of predicting cumulative sales for each of the 99 quantiles. The trapezoid model is the best at the first six (0.01 to 0.06) quantiles, and the tilted-Gompertz model is the best at the remaining 93 quantiles.

Since life cycles in the data set are all completed, this study is ideal for examining the range of skewness values that each of the competing models can capture. See Figure 2.8 for histograms of the skewness values of the Bass, gamma/shifted-Gompertz, trapezoid, and tilted-Gompertz models fit to the 170 new products. The

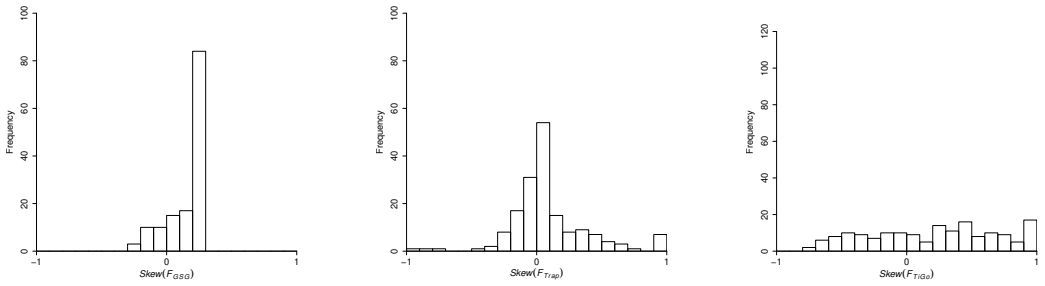
	Quantile Forecasting Errors			Point Forecasting Errors	
	Avg. pinball loss	Hit rate (0.5)	Hit rate (0.9)	MAE	RMSE
Bass	0.075	0.335	0.569	0.181	0.220
Gamma/shifted-Gompertz	0.077	0.313	0.559	0.185	0.223
Trapezoid	0.075	0.333	0.602	0.185	0.222
Tilted-Gompertz	**0.066**	**0.419**	**0.635**	**0.161**	**0.197**
Significance[a]	***	-	-	·	·

[a] Here we check whether the difference of each column's tilted-Gompertz loss (except for the hit rates) and the loss of the next best model is significant. The symbol *** and · indicate significance at the 0.1% and 10% level.

Table 2.2: Errors of Forecasts for the Cumulative Sales of Dell New Computers.

tilted-Gompertz model takes on a wider range of skewness values than the Bass and gamma/shifted-Gompertz models, which suggests there is more "extra-Bass" skew in these new product life cycles than the gamma/shifted-Gompertz models can capture. Although the trapezoid model's skewness values are also on a wider range, the model may not perform as well, because its shape is piecewise linear. Its shape cannot capture exponential growth in a life cycle's growth stage, deceleration in its maturity stage, and exponential decline in its decline stage. Its rate of adoption is linear in the growth stage, flat in the maturity stage, and linear in the decline stage.

To study how well our estimation procedure can identify the true values of the parameters in the tilted-Gompertz model, we perform a simulation study. We simulate 10,000 life cycles generated from our tilted-Gompertz model. The time of the last observation in each life cycle is randomly sampled from the histogram of the



(a) Gamma/shifted-Gompertz Model

(b) Trapezoid Model

(c) Tilted-Gompertz Model

Figure 2.8: Histograms of Local Skewness Values from Three Models Fitted to 170 New Computer Sales Life Cycles.

life cycles' actual lengths in this data set. We set the true values of the parameters λ, δ, ρ, m, and σ^{-2} to the mean estimates from the Computer Sales study. The true values are $\lambda = 0.166$, $\delta = 0.115$, $\rho = 2.082$, $m = 1.767$, and $\sigma^{-2} = 9.152$. In Figure 2.9, we see histograms of parameter estimates using MAP. We notice that, for the most part, these histograms are centered on the true values. For different settings of the true values we tried, the results look similar.

2.5.4 Study 2: Forecasting Search Interest in Social Networks

For this study, we gathered data from Google Trends (2016) on weekly search interest in 122 social networks from January 2004 to June 2016. The names of these networks came from a list of 211 major, active social networking websites (Wikipedia 2016). In the Appendix, we include a description of how these data were gathered. The Google trends data represent normalized search volume, a proxy for the network's demand. The search volume of each term is normalized by Google such that the volume's peak equals 100.

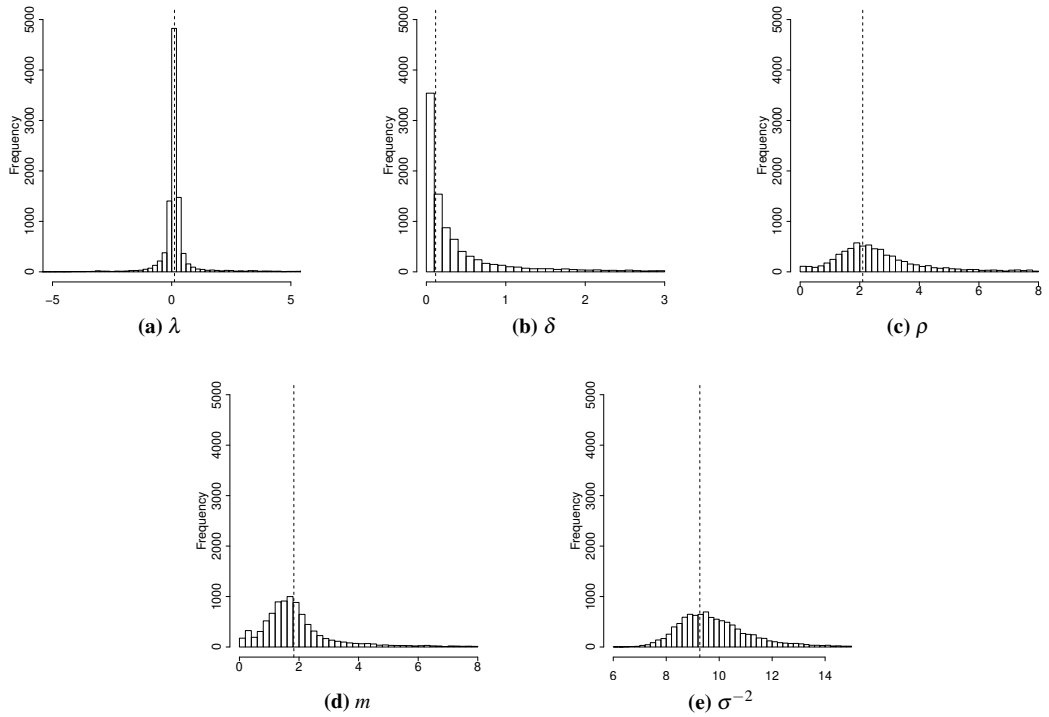


Figure 2.9: Histograms of Parameter Estimates from Simulation of the Tilted-Gompertz Model.

For the 122 networks in this study, we disaggregated their weekly data from Google Trends to equal daily amounts each week, and then aggregated these daily data up to the monthly level. We chose to aggregate up to the monthly level because the weekly data were very noisy. The minimum, median, mean, and maximum of these stretches were 28, 108, 104.2, 150 months, respectively. We then deseasonalized the resulting monthly time series using the ratio-to-moving-average approach (Taylor 2003). Finally, we renormalized the resulting deseasonalized time series so that each time series's peak was at 100.

Social networking websites often need to plan server capacity in advance, based on the forecasts of their per-period web traffic. Therefore, in this study, we focus on forecasting per-period search interests. We do not compare the models' performances on forecasting cumulative demand as we illustrated in Study 1.

In this study, we use a two-fold approach to assess the accuracy of the models. We split across time series so that each of the two folds has half of the total number of time series randomly assigned to it. We use one fold to inform our prior distributions, called the prior-information fold. Then, we make out-of-sample forecasts on the time series in the other fold, called the out-of-sample fold. We go through this process a second time (switching the roles of the two folds the second time) and report the average accuracy for both fold's out-of-sample predictions. We chose two folds for computational convenience, but the approach can be easily extended to more folds.

Here, unlike in study 1, we randomly split the time series into the prior-information and testing fold without considering the websites' launch sequence. If the launch sequence is considered, we would construct a social network's prior-information fold using observations collected up to its launch date. For most social networks in this data set, however, very few time series in the prior-information fold contain peak information. This way, the prior information obtained from these time series may not be reliable, since parameter estimation using only pre-peak data and maximum likelihood is often unstable and has convergence issues (Bemmaor and Lee 2002, Venkatesan et. al 2004). Consequently, in this study, we assume that

the entire time series assigned to the prior-information fold are available before the launch date of the time series in the testing fold. In practice, when there is a new social networking website launching on the market, all these 122 time series can be used to construct prior distributions.

Tables 2.3 and 2.4 present the pinball losses and point forecasting errors for shorter-term forecasts (1-12 steps ahead) and longer-term forecasts (13-24 steps ahead) in Study 2. Each entry in the first column of numbers is a model's pinball loss averaged over the 99 quantiles, over each time series' rolling training sets, over the 122 time series, and over the 12 different steps ahead. The next two columns show the hit rates of the 50% and 90% central prediction intervals. In the last two columns, we provide the MAE and RMSE for measuring the point forecasts.

We can see from these tables that the tilted-Gompertz model has the lowest overall average pinball loss, MAE and RMSE at both forecasting horizons. At the bottom of each table, we report the results of the t-tests for the differences between the tilted-Gompertz model and the next best model. All these differences are significantly different from zero. The tilted-Gompertz model also has the best hit rate in most cases. Figure 2.10(a) and (b) present the average pinball losses for each of the 99 quantiles. The pinball losses are averaged over the rolling training sets, over the 122 time series, and over the 12 different steps ahead. For short-term forecasts (1-12 steps ahead), the gamma/shifted-Gompertz model is the best at the first 19 (0.01 to 0.19) quantiles, and the tilted-Gompertz model is the best at the 0.20 to 0.99 quantiles. For long-term forecasts (13-24 steps ahead), the tilted-Gompertz

	Quantile Forecasting Errors			Point Forecasting Errors	
	Avg. pinball loss	HR (0.5)	HR (0.9)	MAE	RMSE
Bass	6.136	0.533	**0.833**	16.015	17.488
Gamma/shifted-Gompertz	6.415	0.441	0.744	17.124	18.818
Trapezoid	5.760	0.467	0.760	14.754	16.338
Tilted-Gompertz	**5.057**	**0.527**	0.815	**12.879**	**14.478**
Significance[a]	***	-	-	***	***

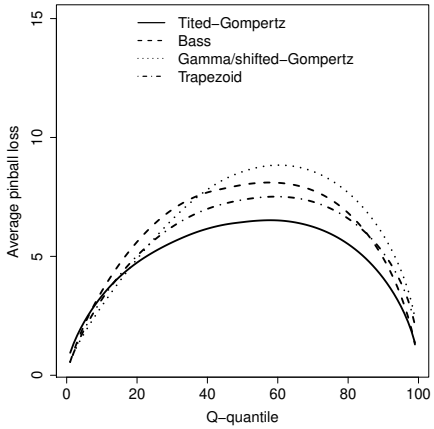
[a] Here we check whether the difference of each column's tilted-Gompertz loss (except for the hit rates) and the loss of the next best model is significant. The symbol *** indicate significance at the 0.1% level.

Table 2.3: Errors of 1-12 Steps Ahead Rolling Forecasts of Search Interest in Social Networks.

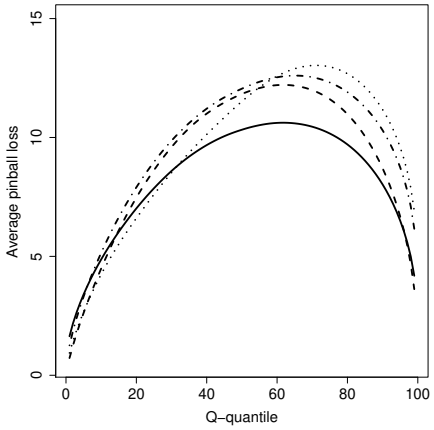
	Quantile Forecasting Errors			Point Forecasting Errors	
	Avg. pinball loss	HR (0.5)	HR (0.9)	MAE	RMSE
Bass	9.185	0.416	0.710	23.660	24.794
Gamma/shifted-Gompertz	9.563	0.332	0.579	23.033	24.309
Trapezoid	9.848	0.320	0.575	24.104	25.310
Tilted-Gompertz	**8.197**	**0.448**	**0.711**	**20.809**	**22.094**
Significance[a]	***	-	-	***	***

[a] Here we check whether the difference of each column's Tilted-Gompertz loss (except for the hit rates) and the loss of the next best model is significant. The symbol *** indicate significance at the 0.1% level.

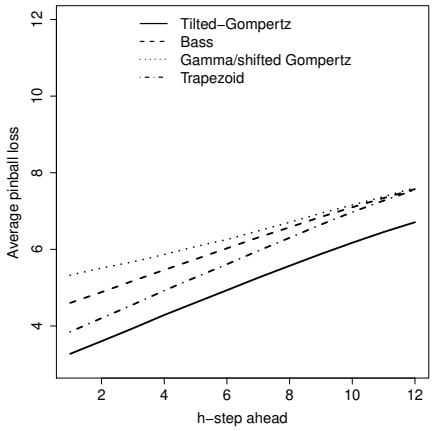
Table 2.4: Errors of 13-24 Steps Ahead Rolling Forecasts of Search Interest in Social Networks.



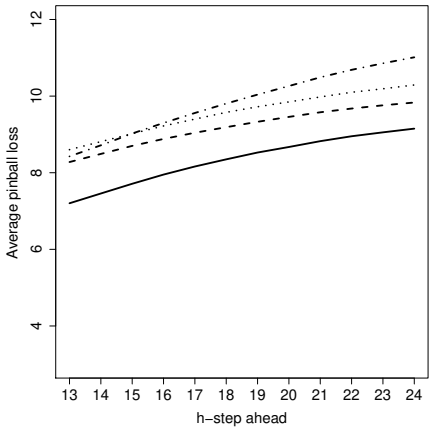
(a) Pinball losses for each of the 99 quantiles at the short-term horizon



(b) Pnball losses for each of the 99 quantiles at the long-term horizon



(c) Pinball losses for each of the 1-12 steps ahead forecasts

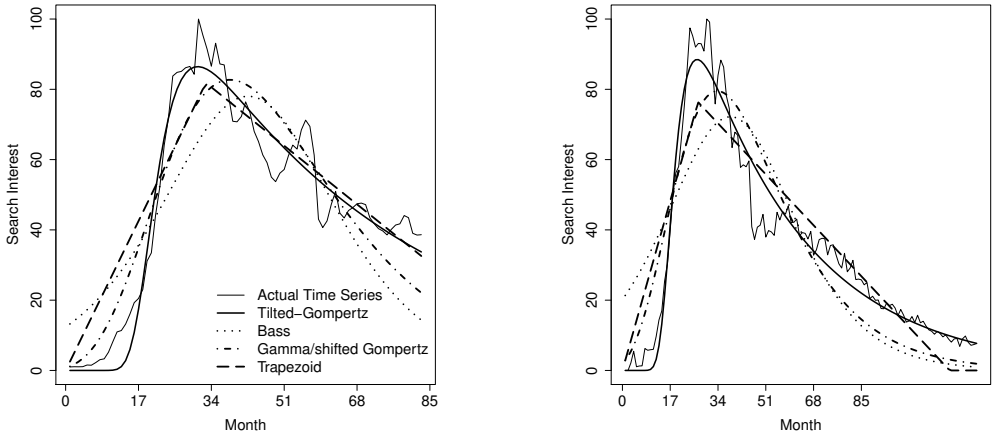


(d) Pinball losses for each of the 13-24 steps ahead forecasts

Figure 2.10: The Average Pinball Losses for the 99 Quantiles and the 1-24 Steps Ahead Forecasts of Search Interest in Social Networks.

model is also the best in most quantiles. The Bass model is the best for the 0.01 to 0.05 quantiles and the gamma/shifted-Gompertz model is the best for the 0.06 to 0.32 quantiles. Finally, Figure 2.10(c) and (d) present the average pinball losses for each of the 24-steps ahead forecasts, where the pinball losses are averaged over the 99 quantiles, over the rolling training sets, and then over the 122 time seres. Apparently, the pinball loss increases with the forecasting horizon, and the tilted-Gompertz model outperforms the others in all different steps ahead. The MAE and RMSE of the 24-steps ahead forecasts follow a similar pattern.

We consider the ability of capturing the "extra Bass" skewness as an important reason why our new model outperforms the benchmark models. The histogram of local skewness values from the competing models in Study 1 has shown that the tilted-Gompertz model can take on a wider range of skewness values than the benchmark models. Here, in Figure 2.11, we present two examples of the models' fitted values for two strongly positively skewed time series in Study 2: search interests of Sina Weibo and DailyStrength. In both cases, the tilted-Gompertz model is the most accurate model. In addition, we can see that the tilted-Gompertz model is more positively skewed than either the gamma/shifted-Gompertz or the Bass model. For Sina Weibo, the local skewness value for the gamma/shifted-Gompertz model is



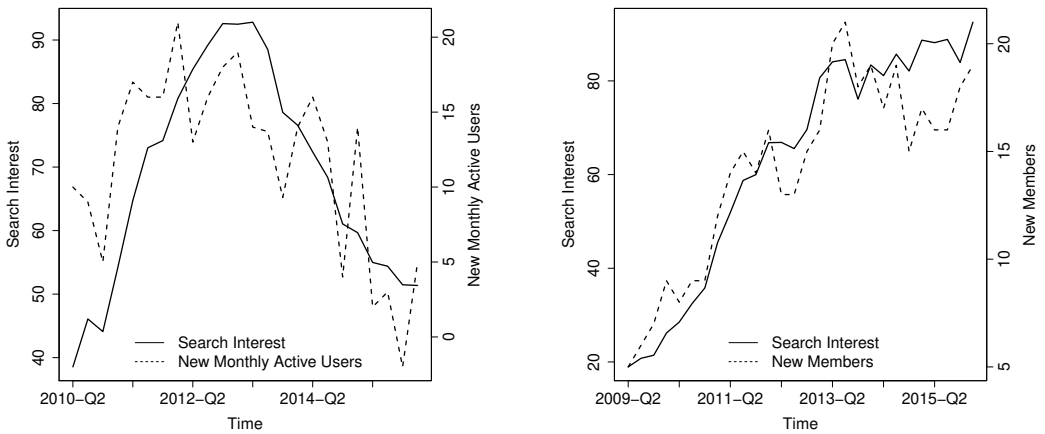
(a) Sina Weibo **(b)** DailyStrength

Figure 2.11: In-sample Fitted Values for Right-skewed Adoptions of Search Interest in a Social Network.

0.273, which is near its upper limit. For the tilted-Gompertz model, the local skewness value is 0.678. For DailyStrength, the local skewness values are 0.275 and 0.650 for the gamma/shifted-Gompertz and tilted-Gompertz models, respectively. The trapezoid model does not fit well as shown in the figures because its shape is piecewise linear.

Finally, we note that Bauckhage et al. (2014) conduct a related study of social media services and web businesses using data from Google Trends. They conclude that "the dynamics of collective attention apparent from search frequency data can be accurately described in terms of diffusion models." (p. 224) Although searches for a service, business, or social network may be done by potential users or may represent repeat searches by current users, the search interest in some social networks appears to closely track new users or members.

Figure 2.12 shows an overlay of new Twitter users and new LinkedIn members on their respective search interest. Search interest data in this figure are the averages by quarter of the monthly search interest from Study 2. Each point in the new users/members time series is the difference in consecutive quarterly averages of monthly active users in the case of Twitter (from the company's SEC filings) and of monthly members of LinkedIn (from the company's Quarterly Re-



(a) Twitter **(b)** LinkedIn

Figure 2.12: Search Interest in Two Social Networks and Their New Users/Members.

sults at `https://investors.linkedin.com`). While new users/members and search interest are not perfectly correlated in Figure 2.12, they appear to move together to a large degree.

2.6 Conclusions

We proposed a tilted-Gompertz model that can be used to model product or service life cycles. The model can predict per-period adoptions as well as cumulative adoptions. The life cycle trend it captures follows the density of a tilted-Gompertz distribution. To study the range of skewness that can be captured by the life cycle models, we propose a local skewness measure, and show that the trend of our new model has the ability to fit a wide range of skewed life cycles. When estimating the parameters of our new model and other benchmarks, we apply the maximum a posteriori method to incorporate prior information from similar products. In two empirical studies, one on forecasting pre-launch new product adoptions and the other on generating rolling forecasts of search interest in social networks, our tilted-Gomperz model performs favorably in both quantile forecasting and point forecasting, compared to other diffusion models.

The use of our modelling approach in practice will depend, in large part, on access to relevant training sets. In our second empirical study, we used half of the time series to form prior distributions in the model. For each new time series in another setting, a manager will need to identify life cycles that are similar to the upcoming life cycle. In a large-scale setting, it may be possible to identify large sets (or clusters) of comparable life cycles, as discussed in Hu et al. (2017).

In this chapter, we apply the MAP method to incorporate prior information into parameter estimations. From the marketing literature, the Kalman Filter (Xie et al. 1997) has been used to include prior beliefs on the parameters of the Bass model. This method also enables the Bass model to have time-varying parameters that are updated over time. In the next chapter, we will develop a time-varying version of the tilted-Gompertz model using the exponential smoothing technique.

2.7 Appendix

This appendix includes derivations of several results in this chapter, as well as properties of various models mentioned in this chapter. We also include a list of the social networks in Study 2.

2.7.1 Trapezoid Diffusion Model and its Skewness

The pdf of the trapezoid model (up to a constant m) is

$$mf_{Trap}(t) = \begin{cases} at+b,\ 0 \le t < \tau_1 \\ a\tau_1+b,\ \tau_1 \le t < \tau_2 \\ c(t-\tau_2)+a\tau_1+b,\ \tau_2 \le t \le t_{max} \\ 0,\ t_{max} < t \end{cases},$$

where $t_{max} = \tau_2 - (a\tau_1+b)/c$ and $0 < a, b, -c, \tau_1, \tau_2 - \tau_1 < \infty$. The shape of its rate of adoption follows a trapezoid with a flat peak between τ_1 and τ_2.

The growth curve of the trapezoid model is given by

$$mF_{Trap}(t) = \begin{cases} \frac{at^2}{2}+bt,\ 0 \le t < \tau_1 \\ \frac{a\tau_1^2}{2}+b\tau_1+(a\tau_1+b)(t-\tau_1),\ \tau_1 \le t < \tau_2 \\ -\frac{a\tau_1^2}{2}+(a\tau_1+b)\tau_2+\frac{c(t^2-\tau_2^2)}{2}+(a\tau_1+b-c\tau_2)(t-\tau_2),\ \tau_2 \le t \le t_{max} \\ -\frac{a\tau_1^2}{2}+(a\tau_1+b)\tau_2+\frac{c(t_{max}^2-\tau_2^2)}{2}+(a\tau_1+b-c\tau_2)(t_{max}-\tau_2),\ t_{max} < t \end{cases},$$

where m is the eventual adopters and $F_{Trap}(t)$ is the cdf of the trapezoid model. Although this model does not have a single peak, we define its local skewness using our local skewness measure, but with t^* set to the trapezoid model's middle peak $(\tau_1+\tau_2)/2$. For this model, $t^{**} = \tau_2 - a\tau_1/c$ so that $f_{Trap}(t^{**}) = f_{Trap}(0)$.

	Bass	Gamma/shifted-Gompertz	Tilted-Gompertz[a]
Density	$f(t)=\frac{(p+q)^2}{p}\frac{e^{-(p+q)t}}{(1+\frac{q}{p}e^{-(p+q)t})^2}$	$f(t)=\frac{\lambda}{\nu}\frac{e^{-\lambda t}(\mu+\nu+(1-\mu)e^{-\lambda t})}{(1+\frac{1}{\nu}e^{-\lambda t})^{\mu+1}}$	$f(t)=ce^{-\lambda\delta t}e^{-\rho e^{-\lambda t}}$, where $c=\frac{\lambda\rho^{\delta}}{\gamma(\delta,\rho)-I_{(-\infty,0)}(\lambda)\Gamma(\delta)}$
Parameters	$0<p,q<\infty$	$0<\lambda,\nu,\mu<\infty$	$-\infty<\lambda<\infty$, $\lambda\neq 0$, and $0<\delta,\rho<\infty$
Distribution	$F(t)=\frac{1-e^{-(p+q)t}}{1+\frac{q}{p}e^{-(p+q)t}}$	$F(t)=\frac{1-e^{-\lambda t}}{(1+\frac{1}{\nu}e^{-\lambda t})^{\mu}}$	$F(t)=\frac{\gamma(\delta,\rho)-\gamma(\delta,\rho e^{-\lambda t})}{\gamma(\delta,\rho)-I_{(-\infty,0)}(\lambda)\Gamma(\delta)}$
Mode(s)[b]	$t^*=-\log(p/q)/(p+q)$ if $q>p$ $t^*=0$ otherwise	$t^*=-\log(\nu)/\lambda$ if $\mu=1$, $\nu<1$ $t^*=-\log(x_1)/\lambda$ if $0<x_1<1\leq x_2$ $t_l^*=0$, $t_r^*=-\log(x_1)/\lambda$ if $0<x_1<x_2<1$ $t^*=0$ otherwise where $x_1=\frac{-B+\sqrt{B^2-4AC}}{2A}$, $x_2=\frac{-B-\sqrt{B^2-4AC}}{2A}$, $A=-(\mu-1)^2$, $B=\mu^2+3\mu\nu-2\nu$, and $C=-\nu(\mu+\nu)$.	$t^*=-\log(\delta/\rho)/\lambda$ if $\lambda(\rho-\delta)>0$ $t^*=0$ otherwise
Peak Rate[c]	$f(t^*)=(p+q)^2/(4q)$ if $q>p$ $f(t^*)=p$ otherwise	$f(t^*)=\lambda(\nu+1)/4$ if $\mu=1$, $\nu<1$ $f(t^*)=f(-\log(x_1)/\lambda)$ if $0<x_1<1\leq x_2$ $f(t_l^*)=\lambda\left(\frac{\nu}{1+\nu}\right)^{\mu}$ or $f(t_r^*)=f(-\log(x_1)/\lambda)$ if $0<x_1<x_2<1$ $f(t^*)=\lambda\left(\frac{\nu}{1+\nu}\right)^{\mu}$ otherwise	$f(t^*)=c(\delta/\rho)^{\delta}e^{-\delta}$ if $\lambda(\rho-\delta)>0$ $f(t^*)=ce^{-\rho}$ otherwise
Skewness[d]	$Skew(F)=0$ if $q>p$ Not applicable otherwise	$Skew(F)=0$ if $\mu=1$, $\nu<1$ $Skew(F)$ numerically evaluated if $0<x_1<1\leq x_2$ Not applicable otherwise	$Skew(F)=1-2\frac{\gamma(\delta,\rho)-\gamma(\delta,\delta)}{\gamma(\delta,\rho)-\gamma(\delta,-\delta W(-(\delta/\rho)e^{-\delta/\rho}))}$ if $\lambda(\rho-\delta)>0$ Not applicable otherwise

[a] $\gamma(\delta,s)=\int_0^s z^{\delta-1}e^{-z}dz$ is the lower incomplete gamma function, and $\Gamma(\delta)=\int_0^{\infty} z^{\delta-1}e^{-z}dz$ is the gamma function. The lower incomplete gamma function is $\Gamma(\delta)$ times the cumulative distribution function (cdf) of the gamma distribution with shape δ and scale one.

[b] The adoption time s is a mode of F, or a local maxima of its density f on $[0,\infty)$, if there exists an $r>0$ such that $f(s)>f(t)$ for all $t\in[0,\infty)$ where $|s-t|<r$. If there is only one mode, we denote it by t^*. If there are two modes, we denote them by t_l^* (the left mode) and t_r^* (the right mode).

[c] We cancel m from both sides of these equations for convenience.

[d] Skewness is measured by $1-2F(t^*)/F(t^{**})$, where $f(t^{**})=f(0)$.

Table 2.5: Properties of Three Smooth Diffusion Models.

2.7.2 Data Gathered on Social Networks in Study 2

In Table 2.6, we list the social networks in Study 2. We did not gather data on 89 of the 211 major, active social networking websites (Wikipedia 2016) for following reasons: (i) no data were available from Google Trends, (ii) the network's name was too generic to associate its name with interest in the network, (iii) search interest was high well before the network's launch date, or (iv) the network was founded before January 2004 and the first observation is greater than one.

google+	tuenti	asmallworld	experience project
facebook	busuu	reverbnation	exploroo
twitter	taringa	italki.com	friendica
qzone	nasza-klasa.pl	gather.com	gentlemint
sina weibo	soundcloud	cross.tv	gogoyoko
instagram	buzznet	socialvibe	govloop
vk	stickam	biip.no	jaiku
tumblr	about.me	skoob	kaixin001
linkedin	livemocha	doximity	mog
renren	iwiw	identi.ca	ning
bebo	ibibo	indaba music	pinterest
tagged	tylted cellufun	travellerspoint	playfire
orkut	43 things	gamerdna	playlist.com
netlog	ravelry	smartican	plurk
hi5	mocospace	spot.im	qapacity
flixster	jiepang	laibhaari	raptr
sonico.com	couchsurfing	mubi	sciencestage
douban	fetlife	stage 32	shelfari
odnoklassniki	draugiem.lv	gays.de	spaces
viadeo	etoro	elixio	teachstreet
flickr	itsmy	zooppa	termwiki
last.fm	grono.net	poolwo	tsu
myspace	getglue	streetlife.com	virb
myheritage	vampirefreaks.com	cloob	vox.com
mixi	nexopia	crunchyroll	wattpad
we heart it	librarything	cucumbertown	wer-kennt-wen
foursquare	cafemom	dailybooth	yammer
academia.edu	uplike	dailystrength	yelp
studivz	zoo.gr	disaboom	yookos
geni	influenster	dreamwidth	
goodreads	focus.com	ello	

Table 2.6: Social Networks in Study 2.

To retrieve the data from Google Trends, we manually entered the network's name and, whenever possible, chose the topic "Social network" or a related topic, rather than the generic topic "Search term". For networks with generic names, the

idea was to separate search interest in the network from search interest in some other item associated with that name. For example, in Google Trends, the term "Delicious" was listed with a topic of "Search term" and separately with a topic of "Social bookmarking website".

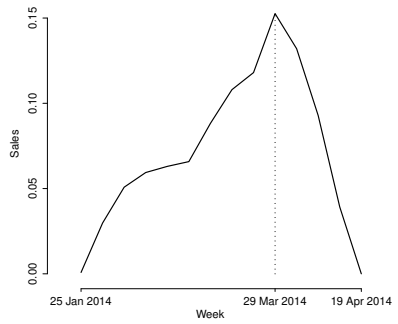
Chapter 3

An Exponential Smoothing Model with a Life Cycle Trend

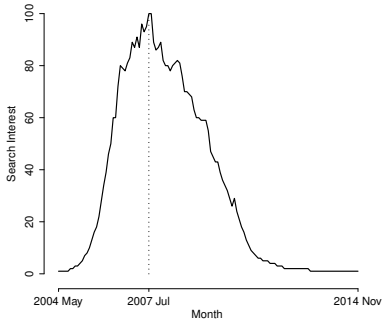
3.1 Introduction

Many business processes go through life cycles. Examples of such processes include sales of an innovative product (Figure 3.1(a)), people's interests in social networking websites (Figure 3.1(b)), the viral nature of news topics (Figure 3.1(c)), and the number of out-of-service customers after a nature disaster (Figure 3.1(d)). A manager's forecast of such life cycles is critical in supporting many decision-making processes. For instance, firms such as Dell need to make decisions on weekly inventory levels of thousands of new products, e.g. next-generation computer or phone products, based on the forecasts of customers' demand. The predictions of growth in monthly active users can help social networking website make decisions on ordering servers. TV producers would be interested in how the news will peak for planning advertising inventories. Moreover, power companies need to determine the number of crews to dispatch after a storm based on the forecasts of customers experiencing power outages. It is also well known that many such decisions rely on the quantile of a demand forecast's distribution (Petruzzi and Dada 1999). In these cases, managers need accurate quantile forecasts.

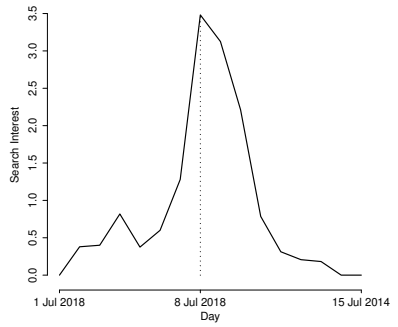
Since their introduction in the 1950s and 1960s, exponential smoothing models have become one of the most widely used forecasting techniques in business (Holt 1957/2004, Brown 1959, Winters 1960, Brown 1963, Gardner and McKenzie 1985). Their popularity stems largely from how accurate they are in practice (Hyndman 2015). Exponential smoothing models are also efficient to estimate using open-source statistical computing software. In addition, they benefit from the



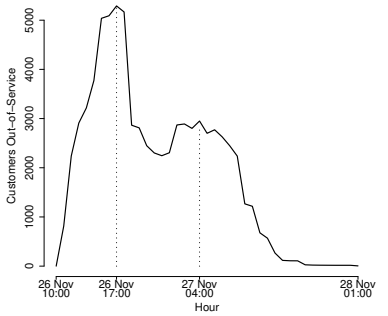
(a) Sales of a Dell Workstation (Hu et al. 2017)



(b) Google Search Interest of Myspace (Google Trends 2016)



(c) Google Search Interest of the Topic "Thailand Rescue" (Schema 2019)



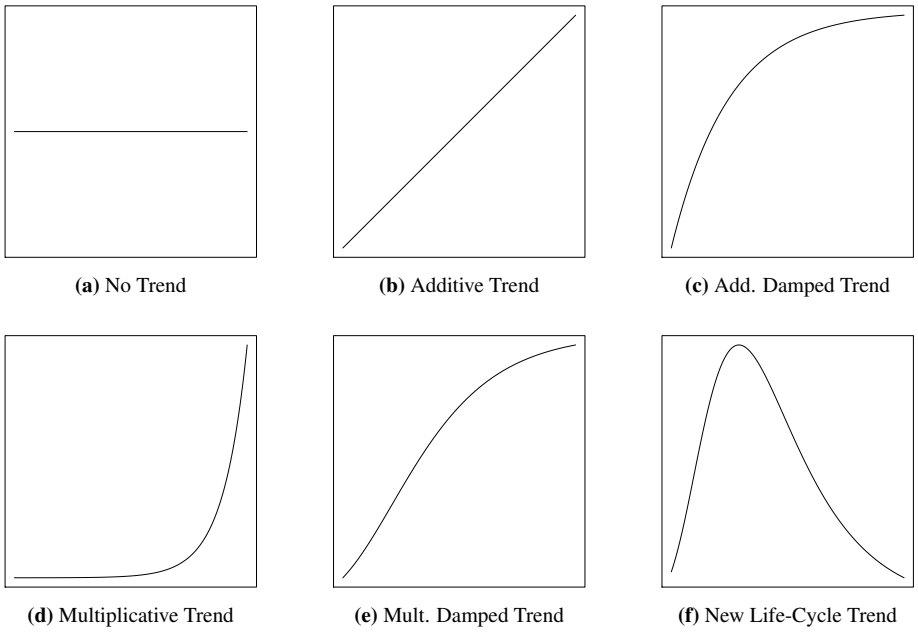
(d) Number of a Utility Firm's Out-of-Service Customers after a Storm in 2016 (Data shared by a North American utility firm)

Figure 3.1: Examples of Processes that Go Through Life Cycles.

simple and intuitive way in which their parameters are updated and their forecasts are generated. In this study, we introduce an exponential smoothing model to forecast life cycles and show that this model can produce accurate point forecasts, as well as accurate quantile forecasts.

In exponential smoothing, a time series is decomposed into three components: error, trend, and seasonality. In the well-known class of exponential smoothing models (Hyndman et al. 2008), there are five possible trend types: (a) no trend, (b) additive trend, (c) additive damped trend, (d) multiplicative trend, and (e) multiplicative damped trend. See Figure 3.2(a)-(e) for a schematic of these five trend types.

The five trends are either increasing, decreasing or constant, and therefore ex-



(a) No Trend **(b)** Additive Trend **(c)** Add. Damped Trend

(d) Multiplicative Trend **(e)** Mult. Damped Trend **(f)** New Life-Cycle Trend

Figure 3.2: Five Existing Trends (a)-(e) and New Life-Cycle Trend (f) in Exponential Smoothing Models.

isting exponential smoothing models can not forecast a non-monotonic trend such as a life cycle. Moreover, without incorporating prior information into model estimation, exponential smoothing models, especially those with multiplicative trends, may wildly overshoot the actuals. In this Chapter, We introduce an important new trend to the above list—the life-cycle trend in Figure 3.2(f). This is a key contribution because exponential smoothing techniques have never been applied to life-cycle models before, and their advantages described above are especially useful for life-cycle modelling in practice.

An additional contribution of this chapter is the framework for incorporating prior information about the model's parameters into the model's estimation procedure via regularization. Prior information about the parameters can come from past related series, as demonstrated in our empirical studies, or from managerial judgement. The benefit of including prior information into the model's estimation is that the procedure takes into account the pattern of the target time series as well as the pattern of past similar time series.

Our life-cycle trend is most closely related to the two multiplicative trends.

Pegels' (1969) multiplicative trend applies to a series that is growing exponentially. Taylor's (2003) multiplicative damped trend applies to a series where the exponential growth dampens and eventually plateaus. Our life-cycle trend captures the notion that a time series initially grows exponentially, slows in its growth until it peaks, and then begins to decline.

Our model's trend follows the probability density function of the tilted-Gompertz distribution that is developed in the previous chapter. In other words, our extension of Taylor's exponential smoothing model results in a trend that is characterized by the tilted-Gompertz distribution. Therefore, the new exponential smoothing model with a life cycle trend can also be viewed as the tilted-Gompertz model with time varying parameters that are estimated using exponential smoothing techniques.

In two empirical studies, we compare the out-of-sample forecasts of our model with a few benchmark models. Similar to the previous chapter, we estimate the Bass model, the gamma/shifted-Gompertz model and the tilted-Gompertz models with time-invariant and time-varying parameters using a regularization approach that allows us to incorporate prior information into the model. This regularization approach for incorporating prior information is both computationally more efficient and more accurate than traditional Bayesian methods such as the extended Kalman filter with continuous state and discrete observations (EKF-CD) introduced by Xie et al. (1997). Consequently, our regularization approach, which is new to the literature on forecasting product life cycles, may be attractive for use at scale, on tens of thousands of time series.

Many firms, such as Google and Facebook, forecast demand for long-term horizons, at scale, and on a frequent rolling basis. Moreover, many time series in these settings go through life cycles. Tassone and Rohani (2017) describe Google's forecasting challenges as follows, "Our typical use case was to produce a time series forecast at the daily level for a 12-24 month forecast horizon based on a daily history two or more years long... We wanted to forecast a variety of quantities: overall search query volume and particular types of queries; revenue; views and minutes

of video watched on Google-owned YouTube... our forecasting task was easily on the order of tens of thousands of forecasts. And then we wanted to forecast these quantities every week, or in some cases more often." In addition, at Google, there was "often interest in probabilistic estimates of tail events, i.e., how high could the forecast reasonably go?"

The data used in our empirical studies are time series of sales of 170 new Dell computers and Google Trends search interest in 122 social networks. In these studies, we find that the tilted-Gompertz model, with either time-invariant or time-varying parameters, performs favorably in both point and quantile forecasting when compared to other models. As discussed in the previous chapter, the tilted-Gompertz model with time-invariant parameters performs well because it can fit a wider range of skewed diffusions. With time-varying parameters, the model performs even better because it can both fit a wider range of skewed diffusions and react to local changes in the environment.

3.2 Exponential Smoothing Model with a Life-Cycle Trend

To model a time series $y_1,\ldots,y_n$ using exponential smoothing, the level states $\ell_1,\ldots,\ell_n$ and the growth states $b_1,\ldots,b_n$ are used to describe the time series' trend in a recursive fashion. For example, Taylor's (2003) exponential smoothing model with additive errors and a multiplicative damped trend has the following formulation, called a state-space formulation:

$$\begin{aligned} \text{Measurement Equation:} \quad & y_t = \ell_{t-1} b_{t-1}^{\phi} + \varepsilon_t \\ \text{Transition Equation for the Level:} \quad & \ell_t = \ell_{t-1} b_{t-1}^{\phi} + \alpha\varepsilon_t \\ \text{Transition Equation for the Growth:} \quad & b_t = b_{t-1}^{\phi} + \beta\varepsilon_t/\ell_{t-1}, \end{aligned} \tag{3.1}$$

where $0 < \ell_0 < \infty$ and $0 < b_0 < \infty$ are the initial level and growth, respectively, $0 \le \alpha \le 1$ and $0 \le \beta \le 1$ are the smoothing parameters, and $0 < \phi < 1$ is a damping parameter. The errors $\varepsilon_1,\ldots,\varepsilon_n$ are independent and identically distributed accord-

ing to a normal distribution with mean zero and variance σ^2. With $\phi = 1$ in (3.1), the model becomes Pegel's (1969) multiplicative trend model.

The exponential smoothing model we introduce below is an important extension of Taylor's (2003) model. To construct our model, we multiply each occurrence of b_{t-1}^{ϕ} in Taylor's model by a positive parameter τ. When this parameter is strictly less than one, the trend will eventually turn down as in Figure 3.2(f). To gain some intuition for why this turn down occurs, suppose the errors are all zero in Taylor's model and each occurrence of b_{t-1}^{ϕ} is multiplied by τ. In the case where $\tau = 1$ (which is Taylor's model with no error), the growth b_t decreases and approaches one as t grows large, which means the trend in Taylor's model will eventually plateau. With $\tau < 1$, however, the growth b_t decreases and approaches zero as t grows large. Hence, our trend will turn down at some point.

The extension of Taylor's model with additive errors to include the turn-down parameter τ results in the same trend at the core of Chapter 2, specifically a trend that is proportional to the probability density function of the tilted-Gompertz distribution. However, when generating prediction intervals, an exponential smoothing model with a multiplicative trend (damped or not) and additive errors suffers from an infinite variance problem and is numerically unstable (Hyndman et al. 2008, Chap. 15.1.1). Therefore, we also replace the additive errors in Taylor's model as discussed in the next subsection.

3.2.1 State-Space Formulation of the Proposed New Exponential Smoothing Model

We define an exponential smoothing model with a life-cycle trend using the state-space formulation:

$$\begin{aligned} \text{Measurement Equation:} \quad & y_t = \ell_{t-1} b_{t-1}^{\phi} \tau (1+\varepsilon_t) \\ \text{Transition Equation for the Level:} \quad & \ell_t = \ell_{t-1} b_{t-1}^{\phi} \tau (1+\varepsilon_t)^{\alpha} \\ \text{Transition Equation for the Growth:} \quad & b_t = b_{t-1}^{\phi} \tau (1+\varepsilon_t)^{\beta}, \end{aligned} \tag{3.2}$$

where $0 < \ell_0 < \infty$ and $0 < b_0 < \infty$ are the initial level and growth states, respectively, $0 \leq \alpha \leq 1$ and $0 \leq \beta \leq 1$ are smoothing parameters, $0 < \phi < \infty$ (and $\phi \neq 1$) is a damping parameter, and $0 < \tau < 1$ is a turn-down parameter. With $\tau = 1$, (3.2) specializes to a model with multiplicative damped trend, which is similar to Taylor's (2003) model, but assumes a different distribution on the errors. Instead of a normal distribution, we assume the random variables $1+\varepsilon_1, \ldots, 1+\varepsilon_n$ are independent and identically distributed according to a lognormal distribution where each $\varepsilon_t^* = \log(1+\varepsilon_t)$ is distributed normally with mean zero and variance σ^2. With multiplicative errors, the transitions for the growth states are guaranteed to be positive and thus well-defined when taking b_{t-1} to the power ϕ.

Note that the multiplicative errors in our model are modified versions of those in typical exponential smoothing models (Hyndman et al. 2008, Chapter 15.2.2). In our transition equations, we use $(1+\varepsilon_t)^\alpha$ and $(1+\varepsilon_t)^\beta$ rather than $1+\alpha\varepsilon_t$ and $1+\beta\varepsilon_t$, respectively. The reason we use modified multiplicative errors is that any exponential smoothing models with standard multiplicative errors suffer from a problem where all simulated sample paths converge to zero (Hyndman et al. 2008, 15.1.2). This is a problem for any standard multiplicative-error model's prediction intervals because they are not in closed form and require simulation to produce them.

To derive the prediction distribution from the model, we apply a logarithmic transformation to each side of each equation in (3.2). This transformed model becomes a convenient exponential smoothing model with an additive trend:

$$\begin{aligned}
\text{Measurement Equation:} \quad & y_t^* = \ell_{t-1}^* + \phi b_{t-1}^* + \log(\tau) + \varepsilon_t^* \\
\text{Transition Equation for the Level:} \quad & \ell_t^* = \ell_{t-1}^* + \phi b_{t-1}^* + \log(\tau) + \alpha\varepsilon_t^* \\
\text{Transition Equation for the Growth:} \quad & b_t^* = \phi b_{t-1}^* + \log(\tau) + \beta\varepsilon_t^*,
\end{aligned} \tag{3.3}$$

where $y_t^* = \log(y_t)$, $\ell_t^* = \log(\ell_t)$, $b_t^* = \log(b_t)$, and ε_t^* is as defined above. This transformed state-space formulation can be viewed as an extension of the exponential smoothing model with additive errors and a damped additive trend, by adding a drift term $\log(\tau)$ to its measurement and transition equations. Note that the drift

term $\log(\tau)$ is negative since $0 < \tau < 1$. Adding drift into exponential smoothing models has been discussed in Hyndman and Billah (2003). In their paper, however, they only consider the simple exponential smoothing model that is equivalent to the ARIMA(0,1,0) model.

3.2.2 Forecasting Method and Prediction Distribution

Next we show how to generate forecasts from our exponential smoothing model with a life-cycle trend. We denote an h-step ahead point forecast for y^*_{t+h} by $\hat{y}^*_{t+h}$. Our forecasting method below is comprised of three equations: a point forecasting equation and two equations for updating the transformed states. Given the transformed observations $y^*_1, \ldots, y^*_t$, we update beliefs about the transformed states recursively from periods 1 to t. Then, given the transformed states as of time t, we use the point forecasting equation to make forecasts h-step ahead.

For the transformed model in (3.3), the forecasting method is given by the following:

$$\begin{aligned}
\text{Point Forecasting Equation:} \quad & \hat{y}^*_{t+h} = \ell^*_t + \sum_{i=1}^{h} \phi^i b^*_t + \sum_{i=1}^{h} (1 + \phi + \cdots + \phi^{i-1}) \log(\tau) \\
\text{Level Updating Equation:} \quad & \ell^*_t = \alpha y^*_t + (1-\alpha)(\ell^*_{t-1} + \phi b^*_{t-1} + \log(\tau)) \\
\text{Growth Updating Equation:} \quad & b^*_t = \beta^*(\ell^*_t - \ell^*_{t-1}) + (1-\beta^*)(\phi b^*_{t-1} + \log(\tau)),
\end{aligned} \tag{3.4}$$

where $\beta^* = \beta/\alpha$ with $\beta \leq \alpha$. See Section 3.7.1 in the Appendix for a derivation of this forecasting method and the prediction distribution below.

With the forecasting method above, we can see the intuitive way in which the transformed model updates beliefs about its states. The current level is a weighted average of the most recent observation of the time series and the previous period's one-step-ahead forecast (which is based, in part, on the previous period's level). The current growth is a weighted average of the most recent growth estimate (based on the difference of the two most recent levels) and a function of the previous period's growth. Like other exponential smoothing models, if we increase either of the smoothing parameters α or β^*, beliefs about the current trend become more

reactive to recent (or local) changes in the environment.

For the untransformed time series, the h-step ahead prediction distribution, made at time t, is described by the equation:

$$y_{t+h} = \ell_t b_t^{\phi+\phi^2+\cdots+\phi^h} \tau^{\sum_{i=1}^{h}(1+\phi+\cdots+\phi^{i-1})} e^{\eta_h} \tag{3.5}$$

where $\eta_h = \sum_{i=1}^{h-1}(\alpha+\beta(\phi+\cdots+\phi^i))\varepsilon^*_{t+h-i} + \varepsilon^*_{t+h}$ is normally distributed with mean zero and variance $Var[\eta_h] = \left(\sum_{i=1}^{h-1}(\alpha+\beta(\phi+\cdots+\phi^i))^2+1\right)\sigma^2$. Consequently, y_{t+h} is lognormally distributed with the following statistics:

$$\begin{aligned}
\text{Median:} \quad & \hat{y}_{t+h} = \ell_t b_t^{\phi+\phi^2+\cdots+\phi^h} \tau^{\sum_{i=1}^{h}(1+\phi+\cdots+\phi^{i-1})} \\
\text{Mean:} \quad & \hat{y}_{t+h} e^{Var[\eta_h]/2} \\
p \text{ Quantile:} \quad & Q_p = \hat{y}_{t+h} e^{Var[\eta_h]^{1/2}\Phi^{-1}(p)},
\end{aligned}$$

where Φ is the cdf of a standard normal random variable. The $0 < u < 1$ central prediction interval is given by $[Q_{(1-u)/2}, Q_{1-(1-u)/2}]$.

Throughout the rest of the chapter, we take the median of y_{t+h} to be the point forecast of y_{t+h}:

$$\text{Point Forecasting Equation:} \quad \hat{y}_{t+h} = \ell_t b_t^{\phi+\phi^2+\cdots+\phi^h} \tau^{\sum_{i=1}^{h}(1+\phi+\cdots+\phi^{i-1})}. \tag{3.6}$$

Note that $\hat{y}_{t+h} = e^{\hat{y}^*_{t+h}}$, which provides a connection between the transformed model's point forecast and the original model's point forecast. There are two reasons why we choose the median to be the model's point forecast. First, the median, in this case, is the zero-error point forecast. That is, it follows from the state-space model in (3.2) with the errors $\varepsilon_{t+1}, \ldots, \varepsilon_{t+h}$ all set to zero. This method is often applied to derive the point forecasting equation for exponential smoothing models with multiplicative trends and multiplicative errors (Hyndman et al. 2008). Second, the median, in any case, is the optimal point forecast under the mean absolute error (Gneiting 2011).

An important version of our model is the model with time-invariant parameters.

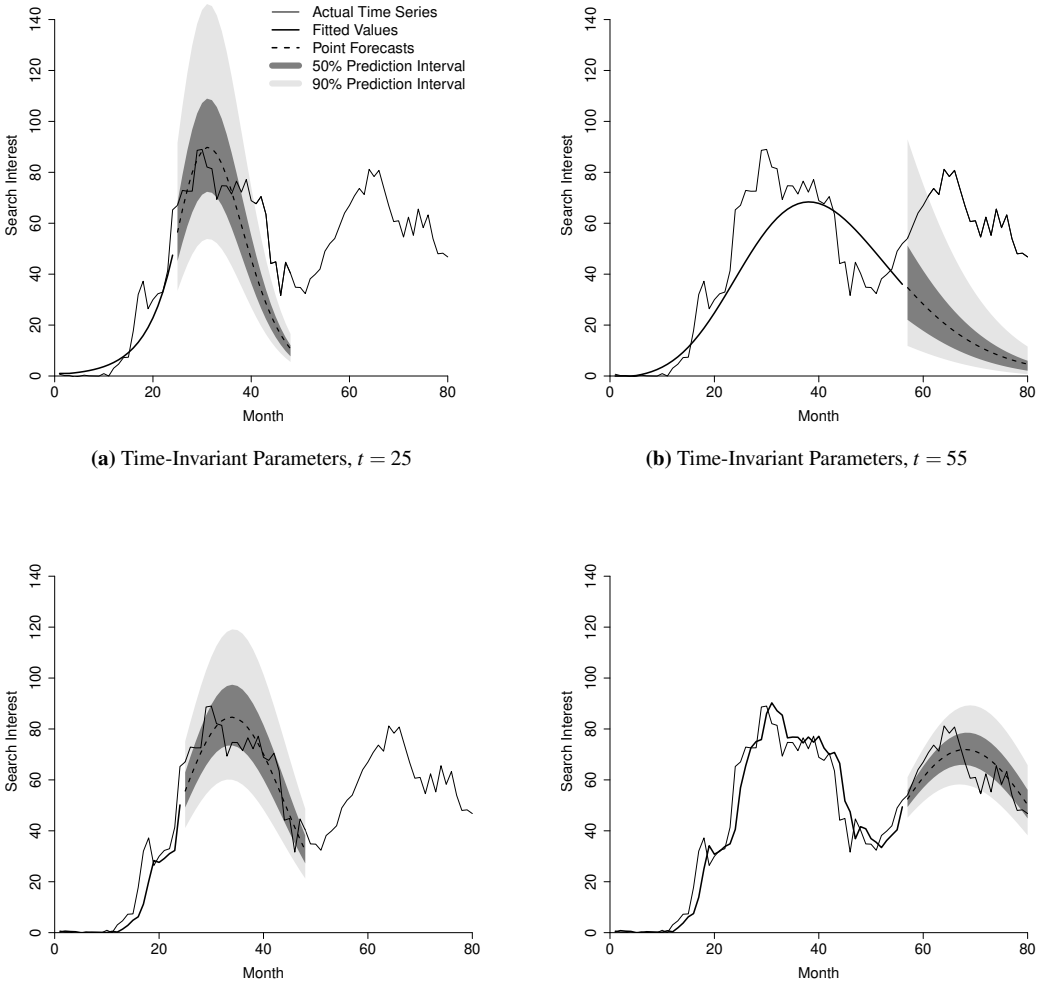
In this version, both smoothing parameters α and β are set to zero, and the state-space model reduces to the model with time-invariant parameters

$$y_t = \ell_0 b_0^{\phi+\phi^2+\cdots+\phi^t} \tau^{\Sigma_{i=1}^{t}(1+\phi+\cdots+\phi^{i-1})} e^{\varepsilon_t^*}.$$

With non-zero smoothing parameters, our state-space model has level and growth parameters ℓ_t and b_t that are stochastically time-varying, so we call this version of the model the tilted-Gompertz model with time-varying parameters.

In Figure 3.3, we compare the fitted values, point forecasts, and prediction intervals from the two versions of our model, with time-invariant and time-varying parameters. The time series is monthly search interest from Google Trends in a social gaming network Itsmy, which is analyzed later in our second empirical study. The time series of the search interest in Itsmy has 126 observations in total. We fit each model twice here—once to the first 25 months of data ($t = 25$) and then a second time to the 55 months of data collected ($t = 55$). From these two time points, we forecast two-years ahead using each version of the model. After the first 25 months, the two models in Figures 3.3(a) and 3.3(c) both forecast the first peak. We can also see from Figures 3.3(b) and 3.3(d), after the first 55 months, that only the model with time-varying parameters is able to forecast a second peak. This flexibility should be attractive, for example, to a manager looking to adjust server capacity to handle the next peak in demand. Such double-peaked life cycles are common and apply to product diffusions that are known to have a "chasm", or dip after a large number of early adoptions, later followed by another peak (Van den Bulte and Joshi 2007). Double-peaked life cycles are also common in customer out-of-service curves. As seen in Figure 3.1(d), the number of customers experiencing power outages may increase again after the first peak due to post-storm impacts.

Next we provide the continuous-time version of the discrete-time point forecasting equation in (3.6) and show that it is proportional to the density of the tilted-Gompertz (*TiGo*) distribution proposed in Chapter 2. We denote the resulting point forecasting function by $\hat{y}_t(h)$, a function that can be evaluated for any real number $h \geq 0$. This function follows from re-expressing the summations in the point



(a) Time-Invariant Parameters, $t = 25$

(b) Time-Invariant Parameters, $t = 55$

(c) Time-Varying Parameters, $t = 25$

(d) Time-Varying Parameters, $t = 55$

Figure 3.3: Our Exponential Smoothing Model, with Time-invariant and Time-varying Parameters, Fit to Search Interest in Itsmy.

forecasting equation in (3.6) so that we can evaluate $\hat{y}_{t+h}$ for any real number $h > 0$.

From the point forecasting equation for $\hat{y}_{t+h}$ in (3.6), we have

$$\text{Point Forecasting Function:} \quad \hat{y}_t(h) = m_t \frac{\lambda(\rho_t e^{-\lambda h})^{\delta} e^{-\rho_t e^{-\lambda t}}}{\gamma(\delta, \rho_t) - I_{(-\infty,0)}(\lambda)\Gamma(\delta)}, \tag{3.7}$$

where $\lambda = -\log(\phi)$, $\delta = \frac{\log(\tau)}{\log(\phi)(1-\phi)}$, $\rho_t = \frac{\phi}{1-\phi}\big(\log(b_t) - \frac{\log(\tau)}{1-\phi}\big)$, and $m_t = \ell_t e^{\rho_t}(\gamma(\delta,\rho_t) - I_{(-\infty,0)}(\lambda)\Gamma(\delta))/\lambda\rho_t^{\delta}$. In m_t, $\gamma(\delta, s) = \int_0^s z^{\delta-1}e^{-z}\,dz$ is the lower incomplete gamma function, $\Gamma(\delta) = \int_0^{\infty} z^{\delta-1}e^{-z}\,dz$ is the gamma function, and the

indicator $I_A(z) = 1$ if $z \in A$ and equals zero otherwise. For details on this derivation, see the Appendix.

The point forecasting function above has the same expression as the rate of adoption in Eq. (2.2) from Chapter 2, except that m_t and ρ_t now depend on the value of the state variables ℓ_t and b_t. In other words, m_t and ρ_t are not fixed but changing over time t. Therefore, the exponential smoothing model we propose in this chapter can also be considered as a tilted-Gompertz model with time varying parameters m_t and ρ_t. Later in the empirical studies, we will compare the performance of both the time-invariant and time-varying versions of the tilted-Gompertz model.

3.3 Model Estimation

In this section, we introduce a method for estimating our exponential smoothing model with a life cycle trend.We incorporate prior information about the model's parameters into our estimation procedure. In the empirical studies below, the prior information is learned from past similar life cycles. With the prior information incorporated, we describe our method as a regularized maximum likelihood (RML) approach, which is largely based on the maximum a posteriori (MAP) method discussed in Chapter 2.

Incorporating prior information about model parameters has received some attention in the literature. Xie et al. (1997) introduce a Bayesian approach to esti mating life-cycle models with time-varying parameters. Their approach is an extended Kalman filter with continuous state and discrete observations (EKF-CD). Xie et al. (1997, p. 379) state one of their motivation for their approach as follows: "By incorporating prior estimates of unknown parameters and updating initial estimates as new data become available, time-varying estimation procedures often can provide better early forecasts." This approach, however, can be difficult to implement for some diffusion models. For instance, the gamma/shifted-Gompertz model estimated by the EKF-CD often produces negative parameter values (which break the model) and negative point forecasts (which make little sense), so we omit it from the empirical studies below. The extended Kalman filter can also be difficult

to implement because it is computationally expensive. Similar to routines such as Markov Chain Monte Carlo (MCMC) simulation, the extended Kalman filter can be slow to run on even hundreds of time series. In our second empirical study below, the extended Kalman filter runs slower than our estimation approach by an order of magnitude. Moreover, the forecasts from the extended Kalman filter in this study are far less accurate than any other competing model.

In the first subsection below, we describe how we apply MAP to estimate the time-invariant tilted-Gompertz model (the exponential smoothing model with α and β set to zero). In the next subsection, we extend this procedure so that it applies to the time-varying tilted-Gompertz model. In other words, we apply RML to estimate our exponential smoothing model with a life-cycle trend. In the third subsection, we describe how we estimate the benchmark diffusion models. Our approach to estimating these other models is also based on MAP.

3.3.1 Estimating the Time-invariant Tilted-Gompertz Model with $\alpha, \beta = 0$

To estimate the model when $\alpha, \beta = 0$, we start with a prior distribution of the tilted-Gompertz distribution's parameters. We let parameters $(\lambda, \log(\delta), \log(\rho_0), \log(m_0))$ be jointly normally distributed with density denoted by f_N. In addition, we let each error's precision σ^{-2} be independent of $(\lambda, \log(\delta), \log(\rho_0), \log(m_0))$ and distributed according to a gamma distribution with shape $a_\sigma > 1$ and rate $b_\sigma > 0$. For the time-invariant version of the model, we choose the parameters λ, δ, ρ_0, m_0, and σ^{-2} to maximize the logarithm of these parameters' posterior distribution

$$\max_{\lambda,\delta,\rho_0,m_0,\sigma^{-2}} \Bigg\{ \log\left(f_N(\lambda, \log(\delta), \log(\rho_0), \log(m_0))\right) + \log\left|\frac{1}{\delta\rho_0 m_0}\right| + (a_\sigma - 1)\log(\sigma^{-2}) - b_\sigma\sigma^{-2} + \frac{1}{2}\sum_{i=1}^{t}\left(\log(\sigma^{-2}) - \sigma^{-2}(y_i^* - \hat{y}_i^*)^2\right)\Bigg\},$$

where y_i^* is the logarithm of the ith observation in the time series and $\hat{y}_i^*$ is equal to the point forecasting function from Eq. (3.4).

Each fitted value $\log(\hat{y}_i)$ is a function of $(\phi, \tau, b_0^*, \ell_0^*)$, which we can rewrite as a function of $(\lambda, \delta, \rho_0, m_0)$. To do this, we apply the one-to-one transformation from $(\phi, \tau, b_0^*, \ell_0^*)$ to $(\lambda, \delta, \rho_0, m_0)$ given by $\lambda = -\log(\phi)$, $\delta = \frac{\log(\tau)}{\log(\phi)(1-\phi)}$, $\rho_0 = \frac{\phi}{1-\phi}\left(b_0^* - \frac{\log(\tau)}{1-\phi}\right)$, $\log(m_0) = \ell_0^* + \rho_0 - \log(c_0)$, and $c_0 = \lambda\rho_0^\delta/(\gamma(\delta,\rho_0) - I_{(-\infty,0)}(\lambda)\Gamma(\delta))$. The first two lines in the objective above represent the logarithm of the prior distribution (up to a constant). The second term in the first line is the Jacobian associated with the transformation from $(\lambda, \log(\delta), \log(\rho_0), \log(m_0))$ to $(\lambda, \delta, \rho_0, m_0)$. The third line in the objective represents the logarithm of the likelihood of the observed data according the model (up to a constant).

3.3.2 Estimating the Time-Varying Tilted-Gompertz Model Using RML

For the time-varying version of the tilted-Gompertz model, we include smoothing parameters α and β^* into the formulation. Since α and β^* range from zero to one, we let them be distributed according to beta distributions with parameters (a_α, b_α) and $(a_{\beta^*}, b_{\beta^*})$, respectively. Similar to the MAP method described in Section 2.4, we let parameters $(\lambda, \log(\delta), \log(\rho_0), \log(m_0))$ be jointly normally distributed with density denoted by f_N. The error's precision σ^{-2} is again distributed according to a gamma distribution. We then choose the parameters $(\alpha, \beta^*, \phi, \tau, b_0^*, \ell_0^*, \sigma^{-2})$ to maximize the objective

$$\begin{aligned}\max_{\alpha,\beta^*,\phi,\tau,b_0^*,\ell_0^*,\sigma^{-2}} \Bigg\{ & (a_\alpha - 1)\log(\alpha) + (b_\alpha - 1)\log(1-\alpha) \\ & + (a_{\beta^*} - 1)\log(\beta^*) + (b_{\beta^*} - 1)\log(1-\beta^*) \\ & + \log\left(f_N(\lambda, \log(\delta), \log(\rho_0), \log(m_0))\right) + \log\left|\frac{1}{\delta\rho_0 m_0}\right| \\ & + (a_\sigma - 1)\log(\sigma^{-2}) - b_\sigma\sigma^{-2} \\ & + \frac{1}{2}\sum_{i=1}^{t}\left(\log(\sigma^{-2}) - \sigma^{-2}(y_i^* - \hat{y}_i^*)^2\right)\Bigg\},\end{aligned} \tag{3.8}$$

where y_i^* is the logarithm of the ith observation in the time series and $\hat{y}_i^*$ is the one-step-ahead forecast at time $i-1$ according to the forecasting method in Eq. (3.4). The first four lines in the objective above represent the logarithm of the prior distributions. The fifth line in the objective represents the logarithm of the likelihood of the observed data according the model.

Each fitted value $\hat{y}_i^*$ in this version of the model can be written as a function of $(\phi, \tau, b_t^*, \ell_t^*)$ using the updating equations in the forecasting method in (3.4). We can then rewrite each fitted value as a function of $(\lambda, \delta, \rho_0, m_0)$. To do this, we apply the one-to-one transformation from $(\phi, \tau, b_t^*, \ell_t^*)$ to $(\lambda, \delta, \rho_0, m_0)$ given by $\lambda = -\log(\phi)$, $\delta = \frac{\log(\tau)}{\log(\phi)(1-\phi)}$, $\rho_0 = \frac{\phi}{1-\phi}\left(b_{0|t}^* - \frac{\log(\tau)}{1-\phi}\right)$, $\log(m_0) = \ell_{0|t}^* + \rho_0 - \log(c_0)$, and $c_0 = \lambda\rho_0^\delta/(\gamma(\delta,\rho_0) - I_{(-\infty,0)}(\lambda)\Gamma(\delta))$. This transformation looks similar to the transformation for the time-invariant version of the model, but it is different because $\ell_{0|t}^*$ and $b_{0|t}^*$ are *retrospective* estimates of the initial level and growth. They are as if the transitions from ℓ_0^* to ℓ_t^* and from b_0^* to b_t^* were time-invariant. When we run the transitions in (3.3) in reverse, with $\alpha = 0$ and $\beta = 0$, we have $b_{0|t}^* = \phi^{-t}\left(b_t^* - (1 + \phi + \cdots + \phi^{t-1})\log(\tau)\right)$ and $\ell_{0|t}^* = \ell_t^* - \sum_{i=1}^t \phi^i b_{0|t}^* - \sum_{i=1}^t (1 + \phi + \cdots + \phi^{i-1})\log(\tau)$. See the Appendix for an explanation of how these two functions are derived.

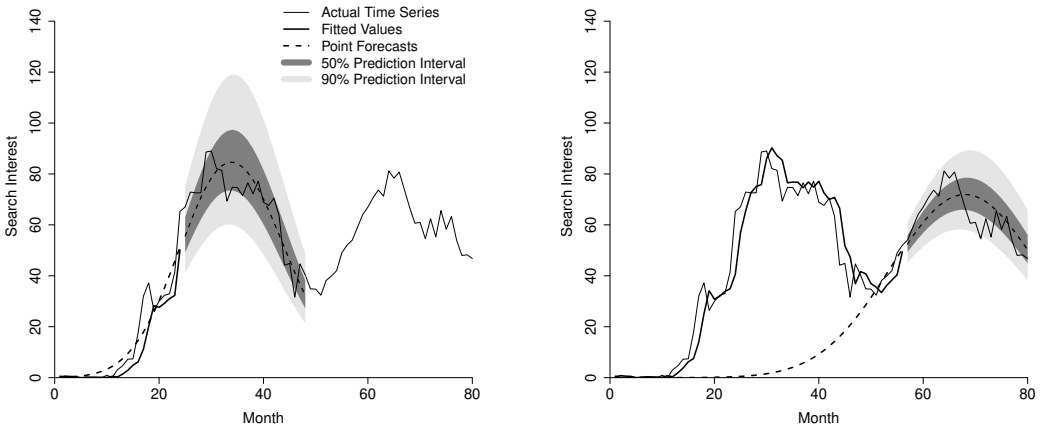
Because the prior distribution in the time-varying model is over the retrospective initial estimates $\ell_{0|t}^*$ and $b_{0|t}^*$, rather than the actual initial estimates ℓ_0^* and b_0^*, we say the time-varying model is estimated with a regularized maximum likelihood approach. To our knowledge, this regularization approach is new to the forecasting literature. When α and β are both equal to zero, our RML approach is equivalent to the MAP approach. The two terms in the third line of (3.8) are regularization terms which make sure any time-varying life cycle does not stray too far from the time-invariant life cycle at the mode of the prior distribution. We tried to estimate our time-varying model with MAP, but many time-varying life cycles reacted too much to noisy data and produced poor forecasts. Consequently, we developed the idea of retrospective life cycles and constrained them to not stray too far from the time-invariant life cycle at the mode of the prior distribution.

To illustrate what we put the prior distribution over through the transformation

in our RML approach, we show two retrospective life cycles for Itsmy's search interest. The dashed lines in Figures 3.4(a) and (b), after time t, are the 1-24 steps ahead forecasts, which are the same forecasts shown in Figures 3.3(c) and (d), respectively. The dashed lines in Figures 3.4(a) and (b), before time t, are the retrospective life cycles. These retrospective life cycles follow from the transitions in Eq. (3.3) run in reverse, with $\alpha = 0$ and $\beta = 0$. In other words, these life cycles are the hypothetical time-invariant life cycles that go with the estimates of ℓ_t^* and b_t^*. Running the transitions in Eq. (3.3) in reverse, with $\alpha = 0$ and $\beta = 0$, yield $\ell_{0|t}^*$ and $b_{0|t}^*$, which are what we put the prior distribution over through the transformation from $(\phi, \tau, b_{0|t}^*, \ell_{0|t}^*)$ to $(\lambda, \delta, \rho_0, m_0)$ above.

3.3.3 Estimating Benchmark Models Using MAP

In Chapter 2, the benchmark models are estimated using $N(t_i) - N(t_{i-1}) = mF(t_i) - mF(t_{i-1}) + \varepsilon_i$, where $N(t)$ is the number of cumulative adoptions by time t, $N(t_i) - N(t_{i-1})$ is the number of adoptions during $(t_i, t_{i-1}]$, m is the market potential, and F is the cdf of a diffusion distribution. In this procedure, the error term ε_i is assumed to be normally distributed. Since the exponential smoothing model we propose in this chapter assumes multiplicative errors that follow lognormal distribu-



(a) $t = 25$ **(b)** $t = 55$

Figure 3.4: Retrospective Life Cycles and Forecasts of Search Interest in Itsmy.

tions, in the empirical studies below we also test the benchmark models estimated using $N(t_i) - N(t_{i-1}) = (mF(t_i) - mF(t_{i-1}))(1+\varepsilon_i)$, where $\log(1+\varepsilon_i)$ is normally distributed with mean zero and variance σ^2. The parameters of the benchmark models are again estimated using the MAP method. Take the Bass model as an example, we assume $\log(p)$, $\log(q)$, and $\log(m)$ are jointly normally distributed, and σ^{-2} is gamma distributed. (The properties of the Bass model can be found in Table 2.5) Thus, we choose the parameters p, q, m, and σ^{-2} to maximize the logarithm of these parameters' posterior distribution

$$\max_{p,q,m,\sigma^{-2}} \left\{ \log\left(f_N(\log(p),\log(q),\log(m))\right) + \log\left|\frac{1}{pqm}\right| + (a_\sigma - 1)\log(\sigma^{-2}) - b_\sigma\sigma^{-2} + \frac{1}{2}\sum_{i=1}^{t}\left(\log(\sigma^{-2}) - \sigma^{-2}(y_i^* - \hat{y}_i^*)^2\right)\right\},$$

where y_i^* is the ith observation in the time series and $\hat{y}_i^*$ is equal to $\log(mF(t_i) - mF(t_{i-1}))$.

A limitation of assuming lognormally distributed errors is that the time series has to be always positive. Demand and sales, however, can sometimes drop to zero. In this case, we can either add a small amount to the zeros, or transform the entire time series by adding a constant. If we apply the second method, we need to subtract the constant from the quantile forecasts to transform them back to the original scale.

3.4 Empirical Studies

In this section, we revisit the two empirical studies in the previous chapter, and include the models developed in this chapter as well as more benchmark models. In the first study, we forecast the complete life cycles of 170 Dell new computers from their very beginning. In the second study, we generate rolling forecasts for the search interests of 122 social networking websites.

We test two versions of the tilted-Gompertz model with time-invariant parameters. The first version of the model is estimated using equation (2.4), which is based on the cdf of the tilted-Gompertz distribution and the additive assumption of the error terms that follow normal distributions. Therefore, we use tilted-Gompertz

(CDFN) to denote this model in this section. The second version of the model is the exponential smoothing model proposed in this chapter with α and β^* set to zero. The point forecasting function is proportional to the pdf of the tilted-Gompertz distribution, and the errors are multiplicative and follow the lognormal distribution. We denote this version of the model by tilted-Gompertz (PDFLN). We also report the accuracy of the tilted-Gompertz model with time-varying parameters (or the exponential smoothing model with a life cycle trend) developed in this chapter. This version of the model is denoted by tilted-Gompertz (ETS).[1]

As discussed in Chapter 2, we use the pinball loss and hit rate to evaluate the probabilistic forecasts generated from the models. The pinball loss corresponds to the payoffs in a newsvendor problem, which will be applied in Section 3.5 to study the implications of forecasting on decision making. Lower pinball loss means higher payoff can be obtained by using the forecasts in a newsvendor problem. Hit rate is an intuitive measure that shows whether the prediction intervals have the correct width. The mean absolute errors (MAE) and root mean squared errors (RMSE) are used to evaluate the accuracy of point forecasts. To specify the prior distributions for the model parameters, we use the same approach as described in Section 2.5.2 and Figure 2.6. For the time-varying tilted-Gompertz model developed in this chapter, we also search for the hyperparameters of the prior distributions for α and β^*, which have significant impact on the performance of the quantile forecasts.

3.4.1 Study 1: Forecasting New Computer Sales

In this study, we use the Dell new computer sales data from Hu et al. (2017), and forecast for the complete life cycle before the product is launched on the market. See Section 2.5.3 for details on the data set. In Chapter 2, we compared the accuracy of the tilted-Gompertz model, Bass, gamma/shifter-Gompertz, and trapezoid model with additive errors. Below in Table 3.1 and Table 3.2 , we add the accuracy results of the exponential smoothing model with α and β set to zero (the tilted-Gompertz (PDFLN) model), and the other three benchmark models estimated with lognormal

[1]Exponential smoothing models are often denoted by "ETS" since they can decompose time series into three components: error(E), trend(T), and seasonality(S).

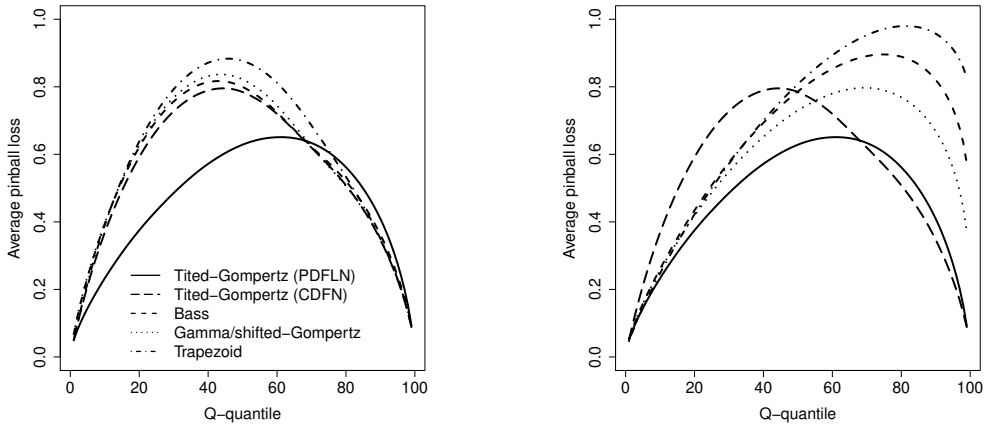
errors. We notice that there are 6 time series in this data set contain one or two zero sales at the beginning of the life cycle. Therefore, when we fit the models with lognormal errors, we add 10% of the minimum positive value in the time series to these zeros.

As shown in Table 3.1 of the models' accuracies in forecasting period sales, the tilted-Gompertz (PDFLN) model has the lowest average pinball loss and point forecasting errors, while the tilted-Gompertz (CDFN) model developed in Chapter 2 performs the best in hit rates. If we compare the benchmark models with different types of errors, we find those with normally distributed errors have lower average pinball loss but higher point forecasting errors than those with lognormally distributed errors. Additionally, the models estimated using normal errors perform much better in hit rates than those estimated using lognormal errors. Figure 3.5 provides more details of the models' performance on forecasting each of the 99 quantiles. Overall, the models with normal errors (Figure 3.5(a)) perform better in forecasting upper quantiles, and the models with lognormal errors (Figure 3.5(b)) perform better in forecasting upper quantiles. The gamma/shifted-Gompertz model with normal errors is the best model for the 0.72 to 0.85 quantiles, while the tilted-Gompertz (CDFN) model is the best for the 0.86 to 0.99 quantiles, and the tilted-

	Quantile Forecasting Errors			Point Forecasting Errors	
	100×Avg. pinball loss	Hit rate (0.5)	Hit rate (0.9)	MAE	RMSE
Tilted-Gompertz (CDFN)	0.561	**0.506**	**0.882**	0.0157	0.0200
Tilted-Gompertz (PDFLN)	**0.470**	0.393	0.758	**0.0126**	**0.0168**
Benchmark models with normal errors					
Bass	0.579	0.490	0.857	0.0163	0.0207
Gamma/shifted-Gompertz	0.577	0.409	0.858	0.0164	0.0207
Trapezoid	0.611	0.364	0.842	0.0175	0.0216
Benchmark models with lognormal errors					
Bass	0.659	0.177	0.365	0.0157	0.0204
Gamma/shifted-Gompertz	0.592	0.180	0.405	0.0146	0.0191
Trapezoid	0.702	0.156	0.326	0.0162	0.0207
Significance[a]	***	-	-	***	**

[a] Here we check whether the difference of each column's tilted-Gompertz (PDFLN) loss (except for the hit rates) and the loss of the next best model (excluding the tilted-Gompertz (CDFN) model) is significant. The symbols *** and ** indicate significance at the 0.1% and 1% level based on t-tests. Values in bold indicate the lowest errors.

Table 3.1: Errors of Forecasts for the Per-period Sales of Dell New Computers.



(a) The tilted-Gompertz models compared to the benchmark models with additive errors

(b) The tilted-Gompertz models compared to the benchmark models with lognormal errors

Figure 3.5: The Average Pinball Losses of the Period Computer Sales Forecasts for Each of the 99 Quantiles.

Gompertz (PDFLN) model performs the best in forecasting the 0.01 to 0.71 quantiles.

Table 3.2 presents the accuracy results of the forecasts for the cumulative sales in Study 1. The tilted-Gompertz (PDFLN) model outperforms the other models in all five accuracy measures. The differences between the tilted-Gompertz (PDFLN) model and the second best model in each column (excluding the tilted-Gompertz (CDFN) model) are statistically different from zero, except for the RMSE. Unlike the results for period sales, the Bass, gamma/shifted-Gompertz and the trapezoid model perform better in quantile forecasting when they are estimated using lognormal errors. We can also see evidence of this result in Figure 3.6. The tilted-Gompertz (CDFN) model performs the best at the 0.83 to 0.99 quantiles, while the tilted-Gompertz (PDFLN) model is the best model for the remaining 82 quantiles. It is also true for the benchmarks that the model estimated using normal errors performs better on the upper tail.

Using the data from this study, we also fit the tilted-Gompertz model with time-varying parameters. Its forecasts though were equivalent to the best tilted-Gompertz (PDFLN) model for almost all the time series. That is, the mode of prior distributions for α and β^* are both zero in the best tilted-Gompertz (ETS)

	Quantile Forecasting Errors			Point Forecasting Errors	
	Avg. pinball loss	Hit rate (0.5)	Hit rate (0.9)	MAE	RMSE
Tilted-Gompertz (CDFN)	0.066	0.419	0.635	0.161	0.197
Tilted-Gompertz (PDFLN)	**0.056**	**0.467**	**0.773**	**0.154**	**0.179**
Benchmark models with normal errors					
Bass	0.075	0.335	0.569	0.181	0.220
Gamma/shifted-Gompertz	0.077	0.313	0.559	0.185	0.223
Trapezoid	0.075	0.333	0.602	0.185	0.222
Benchmark models with lognormal errors					
Bass	0.065	0.396	0.659	0.165	0.194
Gamma/shifted-Gompertz	0.059	0.440	0.723	0.155	0.182
Trapezoid	0.065	0.427	0.614	0.159	0.195
Significance[a]	***	-	-	*	

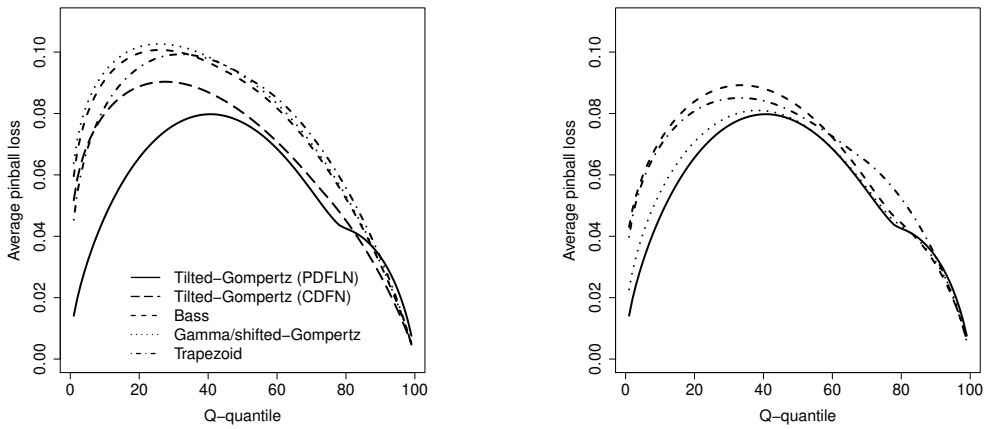
[a] Here we check whether the difference of each column's best tilted-Gompertz (PDFN) loss (except for the hit rates) and the loss of the next best model (excluding the tilted-Gompertz (CDFN) model) is significant. The symbol *** and * indicate significance at the 99.9% and 95% level based on t-tests.

Table 3.2: Errors of Forecasts for the Cumulative Sales of Dell New Computers.

model. Therefore, we do not show the results of the tilted-Gompertz model with time-varying parameters in the tables and figures.

3.4.2 Study 2: Forecasting Search Interest in Social Networks

In this study, we generate 1-step through 24-steps ahead forecasts for the search interest of social networking websites on a rolling basis. The details of the data set



(a) The tilted-Gompertz models compared to the benchmark models with additive errors

(b) The tilted-Gompertz models compared to the benchmark models with lognormal errors

Figure 3.6: The Average Pinball Losses of the Cumulative Computer Sales Forecasts for Each of the 99 Quantiles.

and the method we apply to construct prior distributions of the parameters can be found in Section 2.5.4.

Because the time series in study 2 are longer and we are making rolling forecasts, we compare the forecasting accuracy of 13 total models. These models are: (i) time series models, including the naïve model, the exponential smoothing model with additive errors (A), additive damped trend (A_d), and no seasonality (N), denoted by ETS(A,A_d,N), and the functional linear regression with dynamic updates (FLR-DU, Shang 2017), a recent model developed for forecasts updating, (ii) diffusion models with additive errors, including the Bass, gamma/shifted-Gompertz and the trapezoid model estimated using normal errors and MAP, and the Bass model estimated by EKF-CD, (iii) diffusion models with multiplicative models, which are estimated using lognormal errors and MAP, and (iv) the tilted-Gompertz models, including the tilted-Gompertz (CDFN) model, the tilted-Gompertz (PDFLN) model, and the exponential smoothing model with a life cycle trend (tilted-Gompertz (ETS) model). We notice that 48 time series in this data set contain zeros. For the models estimated using lognormal errors, we add one to the original search interests when we apply these models. Once the quantile forecasts are generated from the model, we subtract one from these quantiles to get the forecasts in the original search interests level.

We do not consider advanced machine learning models as benchmarks because of the following reasons. First, machine learning models are not yet widely applied in quantile forecasting of univariate time series. Second, although machine learning models are accurate in many cases, there exists evidence showing that they are not as accurate as traditional statistical approaches in time series forecasting (Makridakis et al. 2018). Third, machine learning models often need a relatively large training set, making it difficult to predict from the beginning of a time series, especially without enough similar past life cycles.

We define the naïve benchmark at time $t > 1$ as a random walk without drift. This model is equivalent to the ARIMA(0,1,0) model available in the popular `forecast` package in R (Hyndman 2016). Forecasts from the ETS(A,A_d,N)

model were generated using the `forecast` package in R. At $t = 0$ and $t = 1$, the naïve and ETS(A,A_d,N) models' forecasts are the quantiles of y_1 of the time series in the prior-information fold. We also tested Taylor's exponential smoothing model with additive errors (A), a multiplicative damped trend (M_d), and no seasonality (N) denoted by ETS(A,M_d,N), but its performance was much worse than the model with additive damped trend. In addition, an extension of the EKF-CD using the Chebyshev approximation method (CAM) was applied to estimate the time-varying parameters for the Bass model. Kolsarici and Vakratsas (2015) present evidence on longer range forecasts of two time series that suggests EKF-CD with CAM outperforms EKF-CD without CAM. In our study, however, we found its performance worse than the Bass model estimated by the simpler EKF-CD method proposed in Xie et al. (1995).

Table 3.3 and 3.4 present the accuracy results of the 13 models for short-term forecasts (1-12 steps ahead) and longer term forecasts (13-24 steps ahead). Both short-term and long-term forecasts can be used for handling network demand. Long-term forecasts can also be applied for planning advertising inventories (Kolsarici and Vakratsas 2015). The pinball losses in the first column are averaged over the 99 quantiles, over each time series' rolling training sets, over the 122 time series, and over the 12 different steps ahead. The hit rates of the 50% and 90% central prediction intervals are shown in the next two columns. We provide the MAE and RMSE for measuring point forecasts in the last two columns. At the bottom of each table, we report the significance level of the differences between the tilted-Gompertz (ETS) model and the next best model (excluding the other two versions of the tilted-Gompertz model). All the differences are statistically significantly different from zero.

In terms of quantile forecasting, we can see from these tables that the tilted-Gompertz model with time varying parameters (tilted-Gompertz (ETS)) performs the best in all three measures for 1-12 steps ahead forecasts. For longer-term forecasts, the titled-Gompertz model with time varying parameters has the lowest average pinball loss, while the ETS(A,A_d,N) and the Bass (EKF-CD) model perform

	Quantile Forecasting Errors			Point Forecasting Errors	
	Avg. pinball loss	Hit rate (0.5)	Hit rate (0.9)	MAE	RMSE
Tilted-Gompertz (ETS)	**4.218**	**0.524**	**0.861**	**10.603**	**12.297**
Tilted-Gompertz (CDFN)	5.057	0.527	0.815	12.879	14.478
Tilted-Gompertz (PDFLN)	5.547	0.422	0.738	13.867	15.499
Traditional time series models					
Naïve	4.784	0.568	0.778	10.983	12.749
ETS(A,A_d,N)	5.671	0.540	0.759	13.217	15.496
FLR-DU	9.455	0.414	0.820	26.206	27.622
Diffusion models with normal errors					
Bass	6.136	0.533	0.833	16.015	17.488
Bass (EKF-CD)	15.890	0.813	0.994	21.559	22.927
Gamma/shifted-Gompertz	6.415	0.441	0.744	17.124	18.818
Trapezoid	5.760	0.467	0.760	14.754	16.338
Diffusion models with lognormal errors					
Bass	6.124	0.454	0.833	16.107	17.594
Gamma/shifted-Gompertz	5.843	0.281	0.636	14.922	16.557
Trapezoid	6.943	0.324	0.658	17.979	19.513
Significance[a]	***	-	-	***	**

[a] Here we check whether the difference of each column's tilted-Gompertz loss (except for the hit rates) and the loss of the next best model (excluding the other two tilted-Gompertz models) is significant. The symbols *** and ** indicate significance at the 0.1% and 1% level based on t-tests.

Table 3.3: Errors of 1-12 Steps Ahead Rolling Forecasts of Search Interest in Social Networks.

the best in the 50% and 90% hit rates, respectively. In terms of point forecasting, the tilted-Gompertz (ETS) model outperforms the others at both forecasting horizons and both measures. When we compare the tilted-Gompertz models with time-invariant parameters, the model estimated using normal errors performs better than the model estimated using lognormal errors.

The naïve model is the second best model at the shorter-term horizon for the average pinball loss and point forecasting errors. This result is not surprising and lends some support to those who advocate for the use of simple forecasting models (Armstrong 2001, Green and Armstrong 2015). At the longer-term horizon (13-24 steps ahead), however, the tilted-Gompertz (CDFN) model is the second best model in both quantile and point forecasting. One might expect that a life-cycle model, which can predict growth or decline depending on where in the life cycle the rolling forecast is made, to dominate a no-change forecast further into the future. In the context of life cycles, which often grow and decline in dramatic and systematic

	Quantile Forecasting Errors			Point Forecasting Errors	
	Avg. pinball loss	Hit rate (0.5)	Hit rate (0.9)	MAE	RMSE
Tilted-Gompertz (ETS)	**8.014**	0.387	0.721	**19.832**	**21.120**
Tilted-Gompertz (CDFN)	8.197	0.448	0.711	20.809	22.094
Tilted-Gompertz (PDFLN)	8.786	0.277	0.542	21.315	22.626
Traditional ime series models					
Naïve	9.728	0.488	0.705	22.052	23.311
ETS(A,A_d,N)	14.641	**0.498**	0.711	31.621	33.176
FLR-DU	9.793	0.448	0.799	26.974	28.212
Diffusion models with normal errors					
Bass	9.185	0.416	0.710	23.660	24.794
Bass (EKF-CD)	16.420	0.850	**0.990**	26.725	27.855
Gamma/shifted-Gompertz	9.563	0.332	0.579	23.033	24.309
Trapezoid	9.848	0.320	0.575	24.104	25.310
Diffusion models with lognormal errors					
Bass	9.174	0.335	0.644	22.884	24.076
Gamma/shifted-Gompertz	9.933	0.393	0.636	23.220	24.458
Trapezoid	11.062	0.194	0.440	26.490	27.655
Significance[a]	***	-	-	***	***

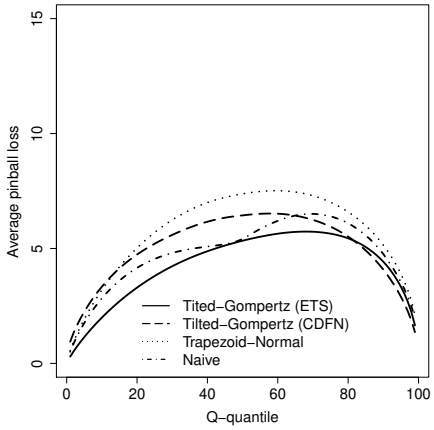
[a] Here we check whether the difference of each column's tilted-Gompertz (ETS) loss (except for the hit rates) and the loss of the next best model (excluding the other two tilted-Gompertz models) is significant. The symbol *** indicates significance at the 0.1% level based on t-tests.

Table 3.4: Errors of 13-24 Steps Ahead Rolling Forecasts of Search Interest in Social Networks.

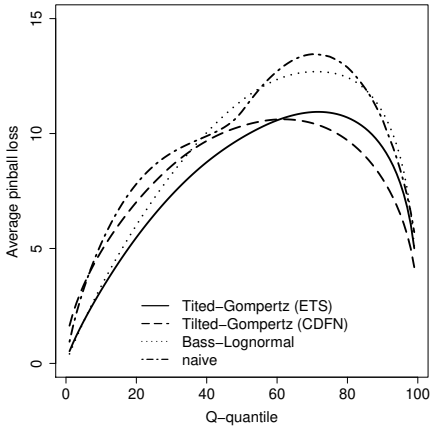
ways, a more complicated model may be more appropriate, even for point forecasting purposes.

Figure 3.7 presents the average pinball loss for each of the 99 quantiles and for each of the 24-steps ahead forecasts, respectively. In Figure 3.7(a) and (b), the pinball losses are averaged over the rolling training sets, over the 122 time seres, and over the 12-steps ahead. In Figure 3.7(c) and (d), the pinball losses are averaged over the 99 quantiles, over the rolling training sets, and then over the 122 time seres. For short-term forecasts, we show the pinball losses of the tilted-Gompertz (ETS) model, the tilted-Gompertz (CDFN) model, the best time series model (naïve), and the best diffusion model (the trapezoid model with normal errors). Fore long-term forecasts, we replace the trapezoid model with the Bass model estimated using log-normal errors as it outperforms other benchmark models in average pinball loss.

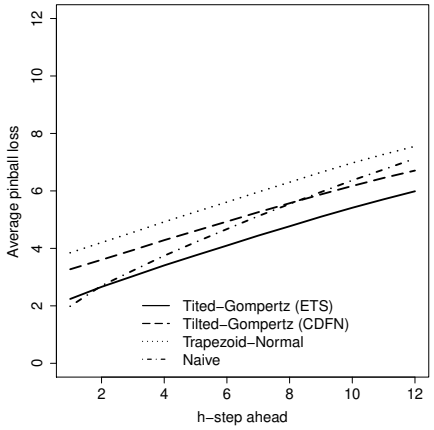
Among all these models, the tilted-Gompertz model with time varying parameters estimated using exponential smoothing performs the best on most quantiles.



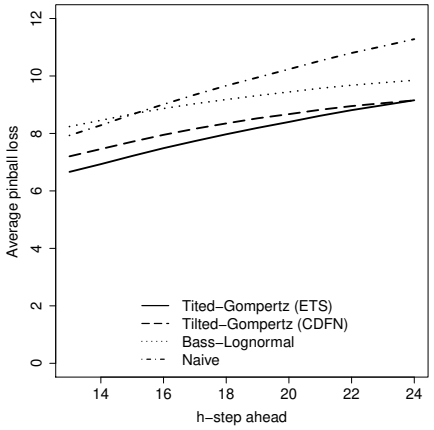
(a) Pinball losses for each of the 99 quantiles at the short-term horizon



(b) Pinball losses for each of the 99 quantiles at the long-term horizon



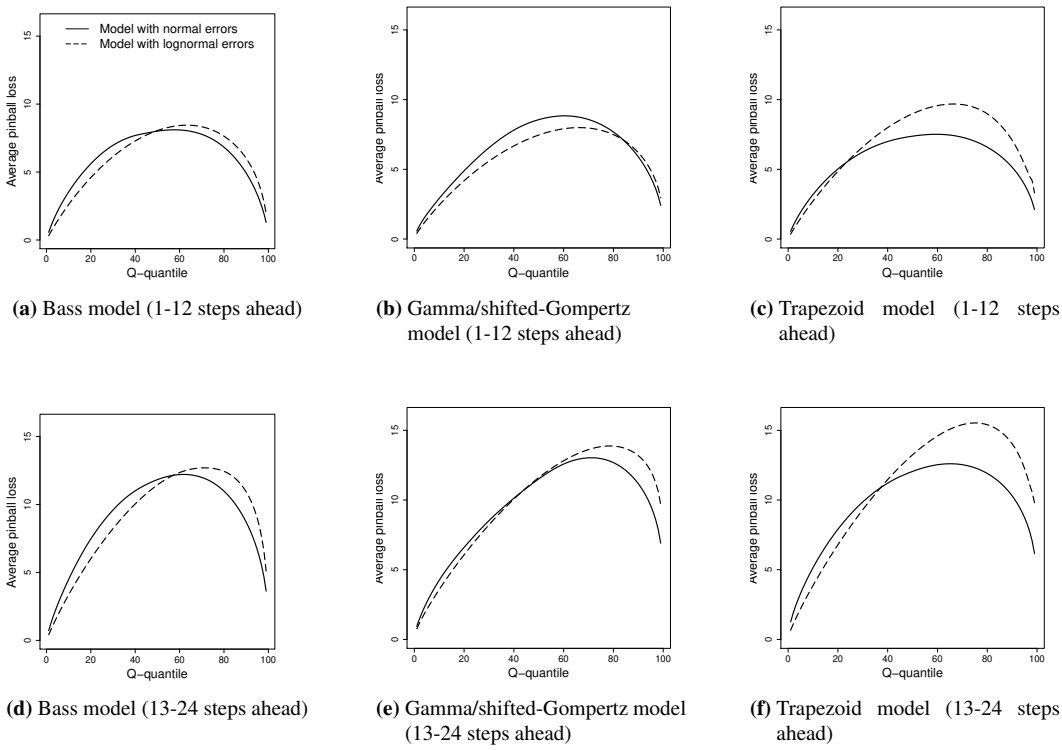
(c) Pinball losses for each of the 1-12 steps ahead forecasts



(d) Pinball losses for each of the 13-24 steps ahead forecasts

Figure 3.7: The Average Pinball Losses for the 99 Quantiles and the 1-24 Steps Ahead Forecasts of Search Interest in Social Networks.

The tilted-Gompertz (CDFN) model outperforms the others on a few upper quantiles (0.83 - 0.99 quantiles for short-term forecasts and 0.59 - 0.99 quantiles for long-term forecasts). According to Figure 3.7(c) and (d), the naïve model is the most accurate at one-step ahead, and the tilted-Gompertz (ETS) model is the most accurate from two-steps onwards. The difference between the pinball losses of the tilted-Gompertz (ETS) and the tilted-Gompertz (CDFN) model decreases with the forecasting horizon. At the 24-steps ahead, these two models have almost the same



(a) Bass model (1-12 steps ahead)

(b) Gamma/shifted-Gompertz model (1-12 steps ahead)

(c) Trapezoid model (1-12 steps ahead)

(d) Bass model (13-24 steps ahead)

(e) Gamma/shifted-Gompertz model (13-24 steps ahead)

(f) Trapezoid model (13-24 steps ahead)

Figure 3.8: The Average Pinball Loss of the 99 Quantiles Predicted by the Diffusion Models Estimated Using Normal Errors and Lognormal Errors in Study 2.

average pinball loss. The MAE and RMSE of the 24-steps ahead forecasts follow a similar pattern as shown in Figure 3.7(c) and (d).

We believe our model outperforms the other benchmarks for three reasons. The first reason is that we have time-varying parameters in our model, allowing us to react to local changes in the environment. As shown in the results above, the tilted-Gompertz model estimated using exponential smoothing techniques outperforms the other two versions of the tilted-Gompertz model without time-varying parameters.

The second reason why the model outperforms is that the estimation procedure incorporates prior information via regularization. When estimated using the maximum likelihood approach (i.e., without any regularization), all the diffusion models often wildly overshoot the actuals, especially when making forecasts before a life cycle's peak. Additionally, although the Bass model estimated by the EKF-

CD method has time-varying parameters and incorporates prior distributions over its parameters, it performs much worse than the Bass model estimated by the MAP method.

Thirdly, the tilted-Gompertz model captures "extra-Bass" skew better than the gamma/shifted-Gompertz model can, even though both of these time-invariant models have four parameters. In Study 1, we saw evidence of this effect on completed life cycles with mostly single peaks. Because the life cycles in that study were mostly single peaked, allowing our model's parameters to time-vary did not improve our model's forecasts. In this study, many time series have multiple peaks and are not yet completed, such as Itsmy. For these time series, our skewness measure does not apply because it relies on a completed life cycle with a single peak. Consequently, we do not show histograms of skewness values for this study.

We also note that in both studies, the tilted-Gompertz (ETS) model which assumes multiplicative errors performs the best on lower and middle quantiles, whereas the time-invariant tilted-Gompetz model with additive errors performs the best on upper quantiles. This finding roughly applies to the benchmark diffusion models as well (see Figure 3.8 for more detail). The benchmark models with multiplicative errors are more accurate than those with additive errors on lower quantiles, while the models with additive errors are more accurate on upper quantiles.

3.5 Numerical Evaluations of the Predictions' Impact on Decision-Making

Until now, we have shown that our new model performs well in terms of out-of-sample forecasting accuracy. However, a decision maker's primary concern might be whether such an improvement in forecasting accuracy could significantly improve their decision-making processes. In this section, we extend the evaluation by examining newsvendor related decisions made using the quantile forecasts generated from models compared in Section 3.4.

In the study of forecasting Dell computer sales, we are interested primarily in production quantities and the profit associated with each of the models. As de-

scribed in Hu et al. (2017), demand planners at Dell typically make 34 steps ahead forecasts before the product is launched on the market. Based on these forecasts, they then make decisions about order quantities from their overseas manufacturing partners. Let Q and Y denote the production level and the cumulative demand of a product in the first 34 weeks since its initial launch. Let p denote its selling price, and $c < p$ the manufacturing and shipment cost of a unit of the product. If production is greater than demand, $Q - Y$ units will be left over with a salvage value $s < c$. These items can be sold latter but also cost extra for storage. The optimal production level, Q^*, can be determined by solving a classic newsvendor problem with the following expected profit:

$$\Pi(Q) = E[p\min(Q,Y) + s(Q-Y)^+] - cQ.$$

The optimal solution for the above problem is $Q^* = F_Y^{-1}((p-c)/(p-s))$, where F_Y is the cdf of the cumulative demand Y, and therefore Q^* is the $(p-c)/(p-s)$ quantile of Y's predicted distribution. The term $(p-c)/(p-s)$ is often referred to as the critical fractile in a newsvendor setting.

Table 3.5 reports the average percentage improvements offered by our model compared to the benchmark models, for five different settings of the critical fractile. Assume all the products in the data set have the same price $p = 500$ and salvage value $s = 100$. Setting cost c to 480, 400, 300, 200 and 120 can cover a range of scenarios corresponding to the different critical fractiles shown in Table 3.5 (the 0.05, 0.25, 0.5, 0.75, and 0.95 fractile, respectively). Based on the accuracy results in Study 1, we use models with normal errors for the 0.95 fractile (or quantile) since they are more accurate on predicting quantiles on the upper tail. For the rest of the quantiles, we apply the models with lognormal errors. Across the five settings of $(p-c)/(p-s)$, the tilted-Gompertz model could improve Dell's profit by anywhere between 4% and 45.6% compared to the Bass model, and between 15.4% and 61.1% compared to the Trapezoid model. In comparison to the gamma/shifted-Gompertz model, our new model could offer improvements that range from 1.6% to 19.6%, except when the critical fractile is 0.25. In this case, the firm may receive 2% more

	Critical fractile $((p-c)/(p-s))$				
	$Q_{0.05}$	$Q_{0.25}$	$Q_{0.50}$	$Q_{0.75}$	$Q_{0.95}$
Bass	45.6	15.4	18.4	13.7	4.0
Gamma/shifted-Gompertz	19.6	-1.9	1.6	5.1	8.0
Trap	61.1	33.8	34.0	26.2	15.4

The percentage improvement is calculated as $100 * (\frac{\Pi^*_{Tilted\ \ Gompertz}}{|\Pi^*_{benchmark}|} - 1)\%$.

Table 3.5: Percentage Improvements in Profits of Dell Computers.

profit if the gamma/shifted-Gompertz model is applied.

Next, we explore the decision-making problem faced by a manager that operates a social networking website. Web service companies often need to plan server capacity well in advance. For example, based on our conversations with the forecasting team at Google, the lead time of ordering servers to meet service levels is usually around four to six months. In other words, at $t = 0$, a manager needs to predict the search volume for $t = 4$ and order servers according to the prediction. Next, at $t = 1$, the manager predicts for $t = 5$ and again needs to make decisions on order quantities.

We formulate the decision-making problem faced by a social networking website from a cost-minimization perspective. Suppose the cost of ordering a server with capacity n is c_o, and the cost of not satisfying a visitor (the penalty cost) is c_p. The optimal ordering quantity, Q^*, can be determined by solving a newsvendor problem with the following expected cost:

$$C(Q) = E[c_p(y - nQ)^+] + c_o Q,$$

where y is the per-period web traffic, approximated by search interest. The optimal solution for the above problem is $Q^* = F_y^{-1}(1 - c_o/(nc_p))/n$, where F_y is the cdf of the period demand y, and therefore nQ^* is the $1 - c_o/(nc_p)$ quantile of y's predicted distribution. Assume that $c_o = 2850$ and $n = 100$. Setting c_p to 30, 38, 57, 114 and 570 can cover the scenarios corresponding to the 0.05, 0.25, 0.5, 0.75 and 0.95 quantiles.

Table 3.6 shows the percentage of cost reductions for two planning horizons

	Critical fractile $((p-c)/(p-s))$				
	$Q_{0.05}$	$Q_{0.25}$	$Q_{0.50}$	$Q_{0.75}$	$Q_{0.95}$
	Planning four-month ahead				
Naïve	17.8	5.8	0.3	2.4	1.2
Trap	20.4	12.9	8.0	4.1	1.9
	Planning six-month ahead				
Naïve	22.2	8.6	1.9	4.2	1.8
Trap	22.1	11.9	7.1	3.3	2.5

The percentage improvement is calculated as $100*(1-\frac{\Pi^*_{Tilted\ \ Gompertz}}{\Pi^*_{benchmark}})\%$.

Table 3.6: Percentage Improvements in Costs on Ordering Servers.

when using our model compared to the best time series model (naïve) and the best diffusion model (trapezoid) according to Table 3.3. For the tilted-Gompertz model, we use the version with time-invariant parameters and normal errors for the 0.95 quantile, and use the version with time-varying parameters for the remaining quantiles. Similarly, the trapezoid model with normal errors is applied in all cases except for the 0.05 quantile, where the model with lognormal errors performs better. When the planning horizon is four months ahead, the tilted-Gompertz model could offer 0.3% to 17.8% cost reduction compared to the naïve model. When the planning horizon is six months in advance, costs are reduced by 1.8% to 22.2% if the tilted-Gompertz model is applied. Compared to the trapezoid model , the cost reduction provided by the tilted-Gompertz ranges from 1.9% to 20.4% and 2.5% to 22.1% for the four-months and six-months planning horizons, respectively.

3.6 Conclusions

In this chapter, we introduce a new life-cycle model based on the principles of exponential smoothing. In two empirical studies, we demonstrate that the model outperforms several benchmarks in out-of-sample forecasting. The trend in our exponential smoothing model follows the density of a new distribution introduced in Chapter 2, the tilted-Gompertz distribution.

The model includes multiplicative errors, instead of the usual assumption of additive errors. During the course of conducting our empirical studies, we find that the model is also more accurate than existing diffusion models with either additive or multiplicative errors. We also find that when predicting the upper tail of the dis-

tribution, the models with additive errors appear to be more accurate than the models with multiplicative errors. Another reason the model produces accurate quantile forecast is because of the prior information we include in the regularization terms of our model. Including prior information is especially critical for accurately forecasting a product life cycle from its beginning. Because our model with time-varying parameters is based on exponential smoothing, it is also computationally efficient to estimate. Unlike the Kalman filter or other Bayesian methods, it does not need to solve differential equations numerically or run simulations to get posterior distributions. In our second study, both the Bass model estimated by extended Kalman filter and the FLR-DU model run slower than our model by an order of magnitude. Overall, the model appears to be well-suited for practical use in large-scale forecasting setting where key operational decisions depend on quantile forecasts. It may also be used as one of several models in an ensemble of models, or mixed with neural networks in an advanced hybrid model (Oreshkin et al. 2019, Smyl 2020).

In this chapter, we assume that the errors in the model follow either a normal distribution or a lognormal distribution. If the time series is long enough, the empirical distribution of the residuals can be used to construct prediction intervals. The time series in our empirical studies are relatively short. As a result, once we have been able to access enough data to build the empirical distribution, the product is often close to the end of its life cycle, where demand forecasts may not be as important as prior to the peak.

3.7 Appendix

This appendix includes derivations of several results in this chapter.

3.7.1 Derivation of the Forecasting Method Equations in (3.4) and Prediction Distributions in (3.5)

Below we show that the method in (3.4) and the prediction distributions follow from the state-space model in (3.3). We show the prediction distribution holds by induction. First, for $d = 1$,

First, we derive the point forecasting equation in two steps.

(i) We show by induction that $b^*_{t+h} = \phi^h b^*_t + (1+\phi+\cdots+\phi^{h-1})\log(\tau) + \beta(\varepsilon^*_{t+h} + \phi\varepsilon^*_{t+h-1} + \cdots + \phi^{h-1}\varepsilon^*_{t+1})$ for any $h \in \{1,\ldots,n-t\}$. For $h=1$, $b^*_{t+1} = \phi b^*_t + \log(\tau) + \beta\varepsilon^*_{t+1}$ holds by the transition equation for the growth in (3.3). Next we assume the equation holds for h and show that it holds for $h+1$:

$$\begin{aligned} b^*_{t+h+1} &= \phi b^*_{t+h} + \log(\tau) + \beta\varepsilon^*_{t+h+1} \\ &= \phi[\phi^h b^*_t + (1+\phi+\cdots+\phi^{h-1})\log(\tau) + \beta(\varepsilon^*_{t+h} + \phi\varepsilon^*_{t+h-1} + \cdots + \phi^{h-1}\varepsilon^*_{t+1})] \\ &\quad + \log(\tau) + \beta\varepsilon^*_{t+h+1} \\ &= \phi^{h+1} b^*_t + (1+\phi+\cdots+\phi^h)\log(\tau) + \beta(\varepsilon^*_{t+h+1} + \phi\varepsilon^*_{t+h} + \cdots + \phi^h\varepsilon^*_{t+1}), \end{aligned} \tag{3.9}$$

where the first equality follows from the transition equation for the growth in (3.3) and the second equality from our induction hypothesis.

(ii) Similarly, we show by induction that $\ell^*_{t+h} = \ell^*_t + \sum_{i=1}^h \phi^i b^*_t + \sum_{i=1}^h (1+\phi+\cdots+\phi^{i-1})\log(\tau) + \sum_{i=1}^{h-1}(\alpha+\beta(\phi+\cdots+\phi^i))\varepsilon^*_{t+h-i} + \alpha\varepsilon^*_{t+h}$ for any $h \in \{1,\ldots,n-t\}$. For $h=1$, $\ell^*_{t+1} = \ell^*_t + \phi b^*_t + \log(\tau) + \alpha\varepsilon^*_{t+1}$ holds by the transition equation for the level in (3.3). Next we assume the equation holds for h and show that it holds for $h+1$:

$$\begin{aligned} \ell^*_{t+h+1} &= \ell^*_{t+h} + \phi b^*_{t+h} + \log(\tau) + \alpha\varepsilon^*_{t+h+1} \\ &= \ell^*_t + \sum_{i=1}^{h} \phi^i b^*_t + \sum_{i=1}^{h}(1+\phi+\cdots+\phi^{i-1})\log(\tau) + \sum_{i=1}^{h-1}(\alpha+\beta(\phi+\cdots+\phi^i))\varepsilon^*_{t+h-i} \\ &\quad + \alpha\varepsilon^*_{t+h} + \phi[\phi^h b^*_t + (1+\phi+\cdots+\phi^{h-1})\log(\tau) + \beta(\varepsilon^*_{t+h} + \phi\varepsilon^*_{t+h-1} + \cdots + \phi^{h-1}\varepsilon^*_{t+1})] \\ &\quad + \log(\tau) + \alpha\varepsilon^*_{t+h+1} \\ &= \ell^*_t + \sum_{i=1}^{h+1} \phi^i b^*_t + \sum_{i=1}^{h+1}(1+\phi+\cdots+\phi^{i-1})\log(\tau) + \sum_{i=1}^{h}(\alpha+\beta(\phi+\cdots+\phi^i))\varepsilon^*_{t+h+1-i} \\ &\quad + \alpha\varepsilon^*_{t+h+1}, \end{aligned} \tag{3.10}$$

where the first equality follows from the transition equation for the level in (3.3) and the second equality from our induction hypothesis and our previous induction

argument for b^*_{t+h}.

The expression for y_{t+h} then follows from substitution according to the expressions for ℓ^*_{t+h-1} and b^*_{t+h-1} established above into the measurement equation in (3.3):

$$\begin{aligned}
y^*_{t+h} &= \ell^*_{t+h-1} + \phi b^*_{t+h-1} + \log(\tau) + \varepsilon^*_{t+h} \\
&= \ell^*_t + \sum_{i=1}^{h-1} \phi^i b^*_t + \sum_{i=1}^{h-1} (1+\phi+\cdots+\phi^{i-1})\log(\tau) + \sum_{i=1}^{h-2} (\alpha+\beta(\phi+\cdots+\phi^i))\varepsilon^*_{t+h-1-i} \\
&\quad + \alpha\varepsilon^*_{t+h-1} + \phi[\phi^{h-1} b^*_t + (1+\phi+\cdots+\phi^{h-2})\log(\tau) \\
&\quad + \beta(\varepsilon^*_{t+h-1} + \phi\varepsilon^*_{t+h-2} + \cdots + \phi^{h-2}\varepsilon^*_{t+1})] + \log(\tau) + \varepsilon^*_{t+h} \\
&= \ell^*_t + \sum_{i=1}^{h} \phi^i b^*_t + \sum_{i=1}^{h} (1+\phi+\cdots+\phi^{i-1})\log(\tau) + \sum_{i=1}^{h-1} (\alpha+\beta(\phi+\cdots+\phi^i))\varepsilon^*_{t+h-i} + \varepsilon^*_{t+h}.
\end{aligned}$$

Based on this expression for y^*_{t+h} and the logarithmic transformations of the observations and states, we have that

$$\begin{aligned}
\log(y_{t+h}) = \log(\ell_t) &+ \sum_{i=1}^{h} \phi^i \log(b_t) + \sum_{i=1}^{h} (1+\phi+\cdots+\phi^{i-1})\log(\tau) \\
&+ \sum_{i=1}^{h-1} (\alpha+\beta(\phi+\cdots+\phi^i))\varepsilon^*_{t+h-i} + \varepsilon^*_{t+h},
\end{aligned}$$

which implies that

$$\begin{aligned}
y_{t+h} &= \ell_t b_t^{\phi+\phi^2+\cdots+\phi^h} \tau^{\sum_{i=1}^{h}(1+\phi+\cdots+\phi^{i-1})} e^{\sum_{i=1}^{h-1}(\alpha+\beta(\phi+\cdots+\phi^i))\varepsilon^*_{t+h-i}+\varepsilon^*_{t+h}} \\
&= \ell_t b_t^{\phi+\phi^2+\cdots+\phi^h} \tau^{\sum_{i=1}^{h}(1+\phi+\cdots+\phi^{i-1})} e^{\eta_h},
\end{aligned}$$

where η_h is normally distributed with mean zero and variance $Var[\eta_h] = \sum_{i=1}^{h-1}(\alpha + \beta(\phi+\cdots+\phi^i))^2\sigma^2 + \sigma^2$. Consequently, y_{t+h} is lognormally distributed as stated (Hyndman et al. 2008, p. 263).

Second, the method's level updating equation follows from eliminating ε^*_t from

the transition equations using the measurement equation:

$$\begin{aligned}\ell_t^* &= \ell_{t-1}^* + \phi b_{t-1}^* + \log(\tau) + \alpha\varepsilon_t^* \\ &= \ell_{t-1}^* + \phi b_{t-1}^* + \log(\tau) + \alpha(y_t^* - \ell_{t-1}^* - \phi b_{t-1}^* - \log(\tau)) \\ &= \alpha y_t^* + (1-\alpha)(\ell_{t-1}^* + \phi b_{t-1}^* + \log(\tau)).\end{aligned}$$

Third, the method's growth updating equation becomes

$$\begin{aligned}b_t^* &= \phi b_{t-1}^* + \log(\tau) + \beta\varepsilon_t^* \\ &= \phi b_{t-1}^* + \log(\tau) + \beta(y_t^* - \ell_{t-1}^* - \phi b_{t-1}^* - \log(\tau)) \\ &= \beta(y_t^* - \ell_{t-1}^*) + (1-\beta)(\phi b_{t-1}^* + \log(\tau)) \\ &= \beta\left(\frac{\ell_t^* - (1-\alpha)(\ell_{t-1}^* + \phi b_{t-1}^* + \log(\tau))}{\alpha} - \ell_{t-1}^*\right) + (1-\beta)(\phi b_{t-1}^* + \log(\tau)) \\ &= \beta^*\ell_t^* - \beta^*(\ell_{t-1}^* + \phi b_{t-1}^* + \log(\tau)) + \beta(\ell_{t-1}^* + \phi b_{t-1}^* + \log(\tau) - \ell_{t-1}^*) \\ &\quad + (1-\beta)(\phi b_{t-1}^* + \log(\tau)) \\ &= \beta^*(\ell_t^* - \ell_{t-1}^*) + (1-\beta^*)(\phi b_{t-1}^* + \log(\tau)),\end{aligned}$$

where $\beta^* = \beta/\alpha$. For $\alpha = 0$, the method's growth updating equation follows directly from the second transition equation in the state-space model.

3.7.2 Derivation of the Point Forecasting Function in (3.7)

In this derivation, we use a fact about the geometric series: $\sum_{i=0}^{h} r^i = 1 + \sum_{i=1}^{h} r^i = \frac{1-r^{h+1}}{1-r}$. Hence, we have that

$$\sum_{i=1}^{h} \phi^i = \frac{1-\phi^{h+1}}{1-\phi} - 1 = \frac{\phi}{1-\phi} - \frac{\phi^{h+1}}{1-\phi} = \frac{\phi}{1-\phi} - \frac{\phi}{1-\phi} e^{\log(\phi)h}$$

and

$$\sum_{i=1}^{h}(1 + \phi + \cdots + \phi^{i-1}) = \sum_{i=1}^{h}\sum_{j=0}^{i-1}\phi^j = \sum_{i=1}^{h}\frac{1-\phi^i}{1-\phi} = \frac{h}{1-\phi} - \frac{\phi}{(1-\phi)^2} + \frac{\phi}{(1-\phi)^2}e^{\log(\phi)h}.$$

In the point forecasting equation in (3.6), we substitute according to the facts above:

$$\begin{aligned}
\hat{y}_{t+h} &= \ell_t b_t^{\phi+\phi^2+\cdots+\phi^h} \tau^{\sum_{i=1}^{h}(1+\phi+\cdots+\phi^{i-1})} \\
&= \ell_t b_t^{\frac{\phi}{1-\phi}-\frac{\phi}{1-\phi}e^{\log(\phi)h}} \tau^{\left(\frac{h}{1-\phi}-\frac{\phi}{(1-\phi)^2}+\frac{\phi}{(1-\phi)^2}e^{\log(\phi)h}\right)} \\
&= \ell_t b_t^{\frac{\phi}{1-\phi}} e^{-\frac{\phi\log(b_t)}{1-\phi}e^{\log(\phi)h}} \tau^{-\frac{\phi}{(1-\phi)^2}} e^{\frac{\log(\tau)}{1-\phi}h} e^{\frac{\phi\log(\tau)}{(1-\phi)^2}e^{\log(\phi)h}} \\
&= \ell_t b_t^{\frac{\phi}{1-\phi}} \tau^{-\frac{\phi}{(1-\phi)^2}} e^{\frac{\log(\tau)}{1-\phi}h} e^{\left(\frac{\phi\log(\tau)}{(1-\phi)^2}-\frac{\phi\log(b_t)}{1-\phi}\right)e^{\log(\phi)h}} \\
&= \ell_t b_t^{\frac{\phi}{1-\phi}} \tau^{-\frac{\phi}{(1-\phi)^2}\left(e^{\log(\phi)h}\right)\frac{\log(\tau)}{\log(\phi)(1-\phi)}} e^{-\frac{\phi}{1-\phi}\left(\log(b_t)-\frac{\log(\tau)}{1-\phi}\right)e^{\log(\phi)h}} \\
&= m_t c_t e^{-\lambda\delta h} e^{-\rho_t e^{-\lambda h}}.
\end{aligned}$$

As there are no integer-dependent expressions (e.g., summations) remaining in the expression for $\hat{y}_{t+h}$, we can evaluate $\hat{y}_{t+h}$ for any real number $h > 0$ and call it $\hat{y}_t(h)$.

3.7.3 Derivation of the Mapping for Estimating the Time-Varying Tilted-Gompertz Model

The equations for $b^*_{0|t}$ and $\ell^*_{0|t}$ follow from the forecasting method equations in (3.9) and (3.10), respectively. In (3.9) and (3.10), set t, α, and β to zero and h to $t-1$. Then replace b^*_0 and ℓ^*_0 by $b^*_{0|t}$ and $\ell^*_{0|t}$, respectively, and solve for $b^*_{0|t}$ and $\ell^*_{0|t}$.

Chapter 4

Forecasting Airport Transfer Passenger Flows Using Real-Time Data and Machine Learning

4.1 Introduction

Passengers often experience delays when travelling through airports, especially at immigration and security. These delays are caused in large part by the volatility and uncertainty in arrival patterns of passengers. Airports and airlines have long invested in optimizing and controlling aircrafts' arrivals and departures (Barnhart and Cohn 2004, Lohatepanont and Barnhart 2004, Lan et al. 2006, Atkinson et al. 2016). Once passengers have disembarked, however, airports typically have little knowledge of passengers' whereabouts in the airport. Improved passenger tracking in real time would enable airports to better serve their passengers, stabilize and predict departure times, and plan resourcing needs.

Flights landing or departing from international hubs often carry a high proportion of connecting passengers. For example, flights traveling through Heathrow, the busiest airport in Europe, have at least 30% connecting passengers (Heathrow 2020). Missed connections are the third leading reason for filing a complaint (MacDonald 2016). Better predictions of the time passengers need to traverse the airport can help minimize such missed connections, or if unavoidable, can alert airlines in advance so that their schedules are not affected by unexpected late arrivals. Additionally, predictions of transfer passenger movements can help predict bottlenecks at immigration and security, allowing for preventive actions, thereby avoiding further delays.

Working with Heathrow airport, we develop a two-phased system to produce both individual level predictions, namely the distribution of passengers' connection times, and aggregate level predictions, i.e. the distribution of the number of passenger arrivals at immigration and security. Here, the connection time predicted in the first phase is defined as the time difference between a passenger's arrival at the airport, i.e. when their plane arrives at its gate, and their arrival at the airlines' conformance desk, the last check point before passengers progress to immigration and security. The conformance desk is also where the airlines check whether a passenger has sufficient time to make their connection. Therefore, we can use the predicted distribution of the connection time to calculate the likelihood of a passenger missing their connecting flight.

In the second phase of our model, the system samples from the distribution of each passenger's connection time to predict the distribution of passenger flows, or the number of arrivals at the conformance desk, which could enable managers to make robust resourcing decisions for peak and trough passenger flows (Borndörfer et al. 2007, Solak et al. 2009, Wu and Mengersen 2013) and when there are understaffing and overstaffing costs (Bassamboo et al. 2010, Zychlinski et al. 2019). The system is currently in use at Heathrow airport, supporting the resourcing of immigration desks and security lanes, informing airlines of late passengers, and facilitating improved collaborative decision-making. Although our predictive system is developed for airport decision-making, it can be generalized to other domains, such as hospitals and theme parks, in which estimates of the number of arrivals are needed for customer flow management.

Passenger delays and passenger flow management have received some attention in the literature (Barnhart et al. 2014, Wei and Hansen 2006). Lack of passenger level data, however, has made it difficult to explore passenger-centric problems. Our work is the first to study passengers' transfer journeys using airport data, and to provide decision support in real time. Decision-making using real-time information has been studied in other contexts (Bertsimas and Patterson 2000, Mukherjee and Hansen 2009, Jacquillat and Odoni 2015), but most of these studies focus on

air traffic issues. Additionally, the integration of machine learning, big data, and real-time decision making has received limited attention in the literature (Shang et al. 2017). Our study is the first to exploit large data sets of flight and passenger information using customized machine-learning algorithms.

Throughout the project, we collaborated with managers at Heathrow and followed the nine-stage approach depicted in Figure 4.1. This nine-stage framework we present here can support the development and implementation of other real-time data-driven systems. It is worth noting that Stage 8: live trial and accuracy test is the most important stage in securing the buy-in of Heathrow teams for scaling up implementation. We presented our model and shared the accuracy results generated during the day to four groups of Heathrow stakeholders right after the live trial. The insights derived from our model, such as the driving features of transfer passengers' connection times, received their enthusiastic support. In this chapter, I mainly focus on Stage 4-7 on Model Training. See Guo et al. (2020) for more detail on the nine-stage framework.

A system that can be implemented at the airport to predict transfer passengers'

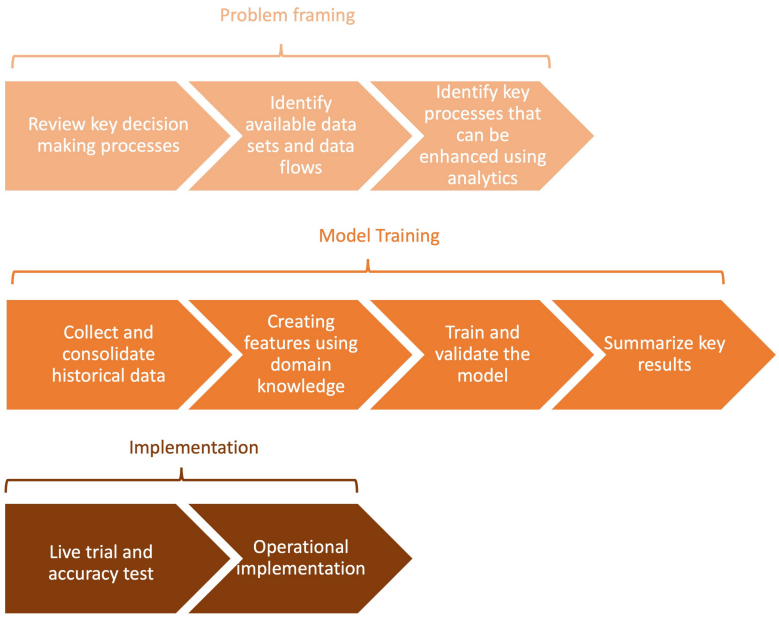


Figure 4.1: The Nine-stage Approach to Developing and Implementing the Predictive System for Improving the Connecting-passenger Flows Through Heathrow Airport.

movements should meet three requirements. First, the approach must produce both accurate point and distributional forecasts. Second, operational plans need to be updated frequently; and thus, any predictive system developed must be capable of generating forecasts rapidly. Third, the model needs to be intuitive in the sense that it enables airport managers to understand the key factors that influence passengers' connection times. This third requirement was essential in order to secure the buy-in of Heathrow teams for scaling up implementation.

Our predictive model of connection times is based on regression trees (Breiman et al. 1984). Regression trees partition the feature space and then fit a simple model (e.g. simple average) to each of the segments (Hastie et al. 2009). The interpretation of the results from a regression tree is intuitive compared to other advanced machine learning methods. Although regression trees have been widely used to make point forecasts in business (Eliashberg et al. 2007, Ferreira et al. 2015, Xue et al. 2015), few have applied it to make distributional forecasts.

In this study, we use regression trees to categorize passengers, and then generate distributional forecasts of their connection times. Using these individual distributions, we can then calculate the distribution of the number of arrivals at immigration and security within certain time intervals in closed form. However, this is only possible if passenger arrivals are assumed independent, which is not the case in practice, and would make the distribution of passenger arrivals overconfident. Therefore, we add dependences among passenger arrivals to produce well-calibrated distributional forecasts, using simulation. The degree of dependences, which become tuning parameters in our approach, are incorporated into the model using Gaussian copulas.

Several results are derived from our model. First, we compare the performance of our model in forecasting individual connection times and identifying late passengers against several benchmark methods. Given the simplicity of a regression tree model, we anticipated trading off accuracy for improved interpretability and run time. Surprisingly, however, the regression tree model performed favorably. Second, we compare the performance of our two-phased approach in pre-

dicting passenger flows with traditional time series models and Heathrow's legacy systems. Our two-phased approach outperforms the benchmarks in generating both point forecasts and distributional forecasts. Finally, we report several findings regarding passengers' connection times based on the regression tree.

To evaluate the impact of our predictive system on real-time decision-making, we simulate a newsvendor-based resourcing problem, reflecting understaffing and overstaffing costs at immigration and security. Such costs represent passengers' dissatisfaction on the one hand and resourcing costs on the other. The optimal resourcing level in this problem is a function of the quantile forecasts of passenger flows. Based on this formulation, we show that in comparison to Heathrow's legacy system, our predictive system could reduce the cost at immigration and security by 12% to 54%, depending on the ratio of staffing cost to understaffing cost. Moreover, our predictive system could offer a 11% to 29% cost reduction compared to traditional time series models.

Our work makes the following contributions. First, to our knowledge, we are the first to develop a predictive system using real-time data to forecast passenger movements at an airport. Second, our two-phased approach, generating individual and aggregate level predictions simultaneously, exhibits superior performance over the traditional approach which considers the two forecasting tasks as independent problems. The individual forecasts of connection times can be used to identify late passengers. These forecasts also serve as input to the second phase of the system where we make aggregate level forecasts. The aggregate level forecasts of passenger arrivals can be applied to adjust resourcing plans at immigration and security areas. Although various components of our model (e.g. generating aggregate level predictions using individual information or using copulas to introduce dependences in customer flows) have been used in previous studies (e.g. Hoot et al. 2008, Van Brussel 2018), the combined system, and the focus on generating and utilizing distributional forecasts, are new. As a final contribution, our system has been fully implemented at Heathrow, where it is being used to minimize missed connections and reduce congestion at immigration and security areas, resulting in improved pas-

senger satisfaction. When combined, these contributions can assist practitioners and academics looking to use machine learning to solve practical operations management problems.

4.2 Problem Description

In this section, we describe a typical transfer journey at Heathrow airport, the related decision-making processes, and the forecasting challenges involved.

The journey for connecting passengers at Heathrow airport is not dissimilar to that at other international hubs. In our study, we focused on the most complex passenger journey at Heathrow, i.e. passengers arriving on international flights, and connecting to a flight at Terminal 5. Typically, once an international flight lands and the passengers disembark, connecting passengers follow signs for flight connections. Passengers arriving at Terminals 2, 3, or 4 first transfer to Terminal 5 using a shuttle bus. Upon arrival at Terminal 5, all passengers check in at the airport conformance desks, where their boarding pass is checked to ensure that the passengers are in the right place with enough time to catch their onward flight. If a passenger arrives after a specific cut-off time (typically 30 minutes before connecting flight's departure), they are deemed unlikely to reach their outbound flight in time, and are redirected to a ticket desk for assistance. This transfer journey is depicted in Figure 4.2.

Passengers connecting to domestic destinations enrol at UK Border Control, and then progress to security screening. Enrolment at UK Border Control is not required for passengers connecting to international destinations. After passing through the conformance desk, these passengers progress directly to the security screening. Finally, after passing airport security, passengers enter the departure lounge, and proceed toward their boarding gates.

Heathrow's passenger flow manager, an individual working with terminal-based operations, is in charge of resource deployment and optimizing passengers' experience. In cases where the passenger flow manager realizes that there is likely to be passenger congestion, she informs relevant teams, such as the Border Force

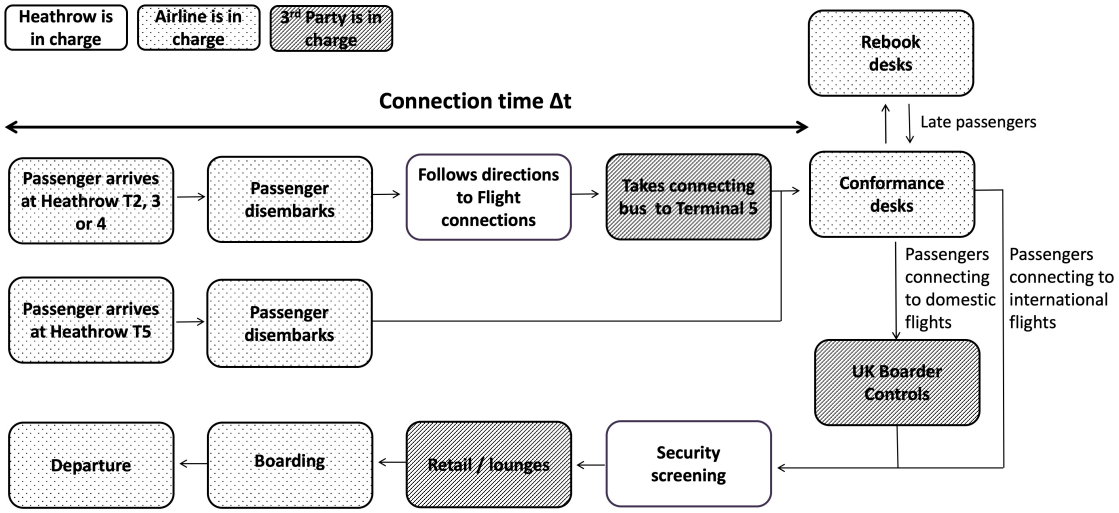


Figure 4.2: The Connecting Journey of International Arriving Passengers Departing through Terminal 5.

and airlines, and recommends actions to resolve the issue. Examples of the recommended actions by the passenger flow manager include adding more staff at the Border Force, offloading and rebooking late passengers to other flights, or delaying a flight if the majority of the passengers connecting to the flight are going to be late. Heathrow's security flow manager, an individual in charge of the security screening areas, continually assesses the flow of passengers, and makes decisions, in real time, on how many lanes are required to be opened and whether staff could be better distributed between terminals (De Reyck et al. 2016). Both the passenger flow manager and the security flow manager operate from Heathrow's Airport Operation Centre (APOC), which consolidates all airport operations including gate management, security, baggage, passenger processes and crisis management in order to improve data sharing and collaborations across teams (Eurocontrol 2010).

Prior to the current study, decisions relating to transfer passengers were not all made with complete information in the center, preventing proactive and truly collaborative decision-making. Without individual level forecasts of connection times, for instance, the passenger flow manager cannot precisely identify passengers with high chances of missing their outbound flight in order to take proactive actions.

Also, lack of real-time predictions of passenger flows into immigration and security areas limits the passenger and security flow managers' ability to adjust necessary resources. Heathrow's legacy system generated forecasts of transfer passenger flows the day before operation, based on historical patterns, e.g. historical averages of the proportion of transfer passengers and connection times between terminals, and the predicted schedule of the next day's arriving flights. As no real-time passenger information was used, significant variability in arrival times and passenger delays made this approach of limited value. Recognizing this shortcoming, the security flow manager would monitor passengers through CCTV cameras. However, it was often too late to reallocate staff by the time the manager realized there was potential congestion.

To support the decision-making processes mentioned above, two forecasting challenges needed to be solved. The first challenge was to predict the likelihood of a passenger missing her connecting flight. Given these probabilities, airlines, with the support of the passenger flow manager, would be able to help late passengers move faster through the airport or facilitate early offloading and rebooking. The second forecasting challenge was to predict the distribution of passenger flows, or the number of arrivals at the immigration and security areas within certain time intervals, to support resourcing decisions.

In the predictive model developed in the next section, forecasts are updated every 15 minutes for the next 90-minute time window, using real-time inputs of passenger and flight information. There are two major constraints to consider: the availability of real-time data and the time needed to adjust plans. For example, it usually takes a security flow manager more than half an hour to move staff between terminals. To guarantee that a manager has enough time to make adjustments, the forecasts should be provided as soon as new data comes in. This requires an effective procedure that can produce accurate forecasts ahead of time.

4.3 The Predictive Model

A typical approach for solving the two forecasting challenges described in Section 4.2 would be to produce two independent forecasts: one from a classification model that identifies late passengers, and one from a time series model that predicts passenger arrivals (Wei and Chen 2012, Milenković et al. 2018). In this chapter, we propose a two-phased approach, with the output from one model serving as input to the second. As we describe below, the two-phased approach produces better forecasts of passenger flows compared to a single-phase model, by allowing the use of real-time information, calibration between the phases using hyper parameters, and the use of dependencies between passenger movements.

In the first phase, we predict the distribution of passengers' connection times, Δt, the time between a passenger's arrival at the airport and their arrival at the conformance desk, where passengers have to show up 30 minutes before their next flight. In the second phase, using the distributions of individual connection times, we generate distributional forecasts of the number of passenger arrivals at the conformance desk within certain time intervals.

Our two-phased approach generates better forecasts of passenger flows for two reasons. First, by utilizing distributions of individual connection times, real-time passenger and flight information, such as a passenger's travel class and the inbound flight's stand type, can be incorporated when predicting passenger flows. Second, although the calibration of the distributional forecasts depends largely on the performance of the model in the first phase, these forecasts can be calibrated using a hyper-parameter. In our model, this hyper-parameter is the correlation between passengers' arrivals at the conformance desk, and is tuned using a validation set. Intuitively, arrivals of passengers travelling together, or from the same inbound flight, are likely to be highly correlated. The traditional time series models are fit to the aggregate level data, i.e. the time series of the number of arrivals at the conformance desk, making it difficult to incorporate correlations between individual passengers. In addition, a single predictive framework can better support airport collaborative decision-making. For example, managers now can easily see the impact of a deci-

sion, such as delaying an outbound flight with many delayed transfer passengers, on both missed connections and on the number of passenger arrivals at immigration and security.

In the rest of this section, we first describe the data used to train and test the predictive model. We then follow with details on how we evaluated the model's forecasts. Finally, we explain how we developed the model.

4.3.1 Data Processing and Model Accuracy

In our model, we use both flight and passenger data. Flight data includes information on departures, arrivals, and aircraft features (e.g., aircraft body type). Passenger data includes individual passenger information, such as their travel class. Historical data for all of 2015, a total of 3,762,690 records, was used to train and test the model. This data was collected from three data sets: Heathrow's Business Objective Search System (BOSS), Baggage Daily Download (BDD), and Conformance data. Figure 4.11 in the Appendix shows how we consolidated the three databases. Table 4.6 in the Appendix provides descriptions of the variables in the data set. Details of the databases used in our study can be found in De Reyck et al. (2016).

The target variable of our study — passengers' connection times Δt — was calculated as the time between the arrival time at the airport, measured by the on-chock time of the aircraft, and the arrival time at the immigration and security areas. According to Heathrow, the service time and queues at the conformance desk are negligible, so we use the time a passenger scans their boarding pass at the conformance desk to approximate their arrival time at the immigration and security areas. The data was cleansed for errors on connection times, such as rerouted passengers (0.24%), negative connection times (1%), and extremely long connection times greater than the 99% quantile. In total, we removed 84,339 records, and the resulting data set contains 3,678,351 passenger records and 32 variables. The median and mean of Δt are 27.0 minutes and 30.5 minutes, respectively. Passenger records of the first 80% of the days in the data set formed our training and validation set. The remaining 20% was used as our out-of-sample testing set.

It is well known that variable selection and creating new input features are key

parts of building an accurate machine learning model (Domingos 2012). In this study, we create eight new features relying on Heathrow experts' knowledge of the aviation domain and the connecting passengers' journey. We anticipate that other airports are likely to find most of these variable relevant as well, as procedures do not vary substantially. The new features we included were:

Inbound flight region and outbound flight region. Where a passenger is travelling from and to (Europe, East Asia, North America, rest of the world) may have an effect on the time needed to traverse the airport. Passengers travelling from or to European countries, for instance, may be more familiar with Heathrow, resulting in shorter connection times. Also, long-haul passengers may have more hand luggage and therefore need more time to disembark.

Punctuality of the arriving flight. Punctuality is defined as a flight's on-chock time minus its scheduled arrival time. This variable can be negative when a flight arrives ahead of schedule. Passengers on late flights might collectively be affected, e.g. due to a lack of gate availability.

Hour of the day the arriving flight lands at the airport. A passenger can move faster through the airport during hours when the airport is not busy.

Day of the week the arriving flight lands at the airport. Day of the week has been shown to be a significant factor on passenger delays in the literature (Barnhart et al. 2014).

Perceived connection time. This feature is calculated as the connecting flight's scheduled departure time a passenger has at the time they depart from the original airport, minus the arriving flight's actual on-chock time. The perceived connection time may influence the stress level of a passenger trying to make a connection, thereby influencing the speed with which they move through the airport.

Arriving flight load factor and connecting flight load factor. The load factor is calculated as the ratio of the actual number of passengers to the capacity of the flight. Passengers arriving on a flight that is relatively full may need more time to disembark from the aircraft.

We considered a total of 40 predictors, given in Table 4.6 in the Appendix. In

the end, we only used 17 predictors, including the six newly created features described above, because the others (1) were not available in real time, (2) were too specific or entailed too many categorical levels, or (3) did not improve the model's accuracy when included. Summary statistics for the target variable and the 17 predictive variables can be found in Table 4.7 and Table 4.8 in the Appendix.

In this study, we generate distributional forecasts of passengers' connection times and the number of passenger arrivals at the conformance desk. Thus, our objective is to minimize the error between the distributional forecasts and the realizations. When evaluating the performance of our model and the benchmarks, we measure the accuracy of both probabilistic forecasts (quantiles and probabilities) and point forecasts.

The primary scoring rule we use for evaluating distributional forecasts is the pinball loss function (Jose and Winkler 2009), computed for multiple quantiles that roughly describe the entire distribution. The pinball loss is a piecewise linear function and is negatively-oriented (lower is better). According to the pinball loss, if a realization falls above a reported quantile, say, the 0.05 quantile, the quantile's loss is its distance from the realization multiplied by its probability of 0.05. Otherwise, if the realization falls below the reported quantile, the quantile's loss is its distance from the realization multiplied by one minus its probability (0.95 in the case of the 0.05 quantile). The results in the pinball loss function penalizing low-probability quantiles more for overestimation than for underestimation and vice versa in the case of high-probability quantiles. In the example of the 0.05 quantile, we would penalize more when the realization falls below this low-probability quantile than above it.

Given the realization y_i of the ith observation, the pinball loss of the p quantile (Q_p) is

$$PL(Q_p, y_i) = \begin{cases} p(y_i - Q_p) & \text{for} \quad Q_p \leq y_i \\ (1-p)(Q_p - y_i) & \text{for} \quad Q_p > y_i \end{cases}.$$

The different values p and $1-p$ for the slope of the loss function when the quan-

tile is below and above the realization, respectively, reflect the desired imbalance in evaluating quantile forecasts (Gneiting 2011). For the median ($p = 0.5$), the loss function is symmetric with an equally weighted loss for a realization falling below or above its reported quantile. For the tails of the distribution, the function is asymmetric with a higher weighted loss when the realization falls below a reported low-probability quantile ($p < 0.5$) or above a reported high-probability quantile ($p > 0.5$).

The pinball loss has two additional appealing properties: (1) it is a strictly proper scoring rule for quantile forecasts (Grushka-Cockayne et al. 2017), i.e. it incentivizes reporting of true beliefs, and (2) the pinball loss function has an economic interpretation. The newsvendor problem, which we will use to make resourcing decisions in Section 4.6, is an affine transformation of the pinball loss (Jose and Winkler 2009). The lower the pinball loss of the quantile forecasts, the higher the payoff in a newsvendor problem.

The pinball loss is also closely related to the Continuous Ranked Probability Score (CRPS), which is used for density forecasts. As the number of quantiles goes to infinity, the limit of the pinball loss summed over all quantiles is equal to the CRPS (Grushka-Cockayne et al. 2017). Compared to the CRPS, however, the pinball loss has fewer assumptions and is easier to implement (Hong et al. 2016). The logarithmic score, an alternative scoring rule that also works in the density space, may return infinity scores when the realization is outside of the predicted distribution's support, making it difficult to evaluate the average performance of a model.

In this study, we score the median and the 0.05, 0.25, 0.75, and 0.95 quantiles as they are commonly used for calculating confidence intervals, p-values and interquartile ranges, as well as plotting standard boxplots. Moreover, these five quantiles are often considered as critical quantiles and are reasonably spaced out to represent the entire distribution.

We also check the calibration of the distributional forecast, i.e. the consistency between the distributional forecasts and the observations (Gneiting and Raftery

2007), using the probability integral transform (PIT). We check the calibration of the distributions because in the second phase, passengers' connection times are simulated from the distributions generated in the first phase. Well calibrated distributions of passengers' connection times can provide accurate simulation results in the second phase. The PIT of a distributional forecast's F is $p = F(y_i)$, the cumulative distribution function (cdf) F evaluated at the realization y_i. If the predicted distribution is consistent with observations, the PIT should follow a uniform distribution. If the predicted distribution is overconfident or underconfident, the density of its PIT is bathtub-shaped or hump-shaped, respectively. A bathtub-shaped density indicates that the predicted distribution is too narrow, with too many realizations falling farther out in the distribution's tails than expected; a hump-shaped density indicates that the predicted distribution is too wide, with too few realizations falling out far enough (Lichtendahl et al. 2013).

Once we have estimated the distribution of Δt, we can evaluate the model's ability to identify late passengers, using the logarithmic loss (log loss) and brier score to test the probability of a passenger being late. The log loss is calculated as $-(y_i \log(p_i) + (1-y_i)\log(1-p_i))$, and the brier score is calculated as $(p_i - y_i)^2$, where y_i is a binary indicator of the ith realization, and p_i is the model's predicted probability of a passenger being late.

The point forecast of connection times and passenger flows we score is the mean, and we use the root mean squared error (RMSE) to measure its accuracy, defined as $\sqrt{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2/n}$, where y_i is the realization, $\hat{y}_i$ is the prediction, and n is the number of predictions. The mean is the optimal point forecast under RMSE. If the mean absolute error is preferred as the primary accuracy measure, one could instead use the median as the point forecast (Gneiting 2011, Jose 2016).

4.3.2 Phase One: The Regression Tree Model for Connection Times

In phase one, we estimate the distribution of passengers' connection times, Δt, using a regression tree. We fit the tree by minimizing the mean squared error, while the tuning parameters (the maximum depth of the tree and the minimum number of

observations) are selected by minimizing the pinball loss. The simplicity of the regression tree approach is not only useful for rapid prediction, but can also yield intuitive explanations as to why observations are predicted in a particular manner. An overview of the steps taken in Section 4.3.2 and 4.3.3 is shown in Figure 4.12 in the Appendix.

Regression trees are widely used for generating point forecasts (James et al. 2013); however, relatively few studies have applied them to produce entire distributions. We use regression trees to generate distributional forecasts as follows. After fitting a tree to the logarithmic transform of the connection times, each of the leaves in the tree represents a passenger segment, containing a number of observations.[1] We find that the majority of the leaves have connection times, in their original scale, that follow right-skewed distributions. Fitting distributions that can capture right skewness, such as the log-normal and gamma distributions, however, produce underconfident forecasts in the test set, with a hump-shaped PIT. Pure empirical distributions, on the other hand, produce overconfident forecasts, because the distribution is limited to the range of data in the training set. Instead, we apply kernel density estimation (Wand and Jones 1994) to approximate the empirical distribution of the connection times within each leaf, which generates a continuous version of the empirical distribution as a compromise between a parametric distribution and a pure empirical distribution.

Given a connecting passenger's information, the regression tree will first determine which leaf (or segment) this passenger belongs to; the empirical distribution attached to this leaf will then provide the quantile forecasts of this passenger's connection time, and the probability of this passenger being late for his onward flight.

To avoid overfitting to the training set, the maximum depth of the tree, N_{max}, and the minimum number of observations, l_{min}, in each leaf are tuned using a two-fold cross validation approach.[2] The tree is fit — for a range of values of the two parameters ($5 <= N_{max} <= 25$ with an increment of 1; $50 <= l_{min} <= 1000$ with

[1] We found that the distribution of the connection times is highly right skewed, and therefore transforming the data may yield better accuracy results, as suggested in De'ath and Fabricius (2000).

[2] We also ran a 5-fold cross validation, with results ending up similar. With 5-fold, we obtained $N_{max} = 15$ and $l_{min} = 200$.

an increment of 50) — to half of the data in the training set, and the pinball loss of the 0.05, 0.25, 0.50, 0.75 and 0.95 quantiles are computed and then averaged in the remaining half. This is done in turn for each fold, and the two pinball scores are averaged. The results of the grid search suggest the optimal N_{max} and l_{min} should be 14 and 200, respectively. The tree with these optimal tuning parameters partitions the transfer passengers in our training set into 2,569 segments, with branches between 8 and 14 levels, and segments containing between 200 and 6,215 passengers.

4.3.3 Phase Two: Distributional Forecasts for the Number of Arrivals at Immigration and Security

In phase two, we use the leaf distributions from the regression tree to generate predictions for the number of passengers arriving at immigration and security. This will help with predicting congestion in these areas and reallocating resources in advance of such congestion in order to improve the passenger experience. A passenger's arrival at the conformance desk during the ith time interval $t_{iu} - t_{il}$, where t_{iu} and t_{il} are the upper and lower bounds of the time interval, can be considered as a Bernoulli process. Let random variable X_{ij} denote whether or not the jth passenger arrives during this ith time interval. Then, $X_{ij} \sim Bernoulli(p_{ij})$, where p_{ij} is the probability of this passenger arriving during the time interval. Each X_{ij} may have different probabilities p_{ij} due to the segment (leaf of the tree) a passenger is classified into. The number of passengers arriving during the ith interval, $S_i(n)$, can then be calculated as $X_{i1} + X_{i2} + ...X_{in}$, where n is the total number of connecting passengers. If X_{ij} are independently distributed, then $S_i(n)$ follows a Poisson Binomial distribution, or a normal distribution if n is large enough. Then, the mean $\hat{\mu}_{S_i(n)}$ and variance $\hat{\sigma}^2_{S_i(n)}$ of $S_i(n)$ are $\sum_{j=1}^{n} p_{ij}$ and $\sum_{j=1}^{n} p_{ij}(1-p_{ij})$, respectively. The probability of arriving during the ith time interval, p_{ij}, can be calculated as $F_j(t_{iu} - t_{arrj}) - F_j(t_{il} - t_{arrj})$, where t_{arrj} is passenger j's arrival time at the airport, and F_j is the cdf of the predicted distribution of passenger j's connection time.

As not all passengers travel independently but often in groups, it is important to consider dependences between passengers' connection times and arrivals at the conformance desk. In that case, the mean of the distribution for passen-

ger flows stays the same as above, and the variance of the distribution becomes $\sum_{j=1}^{n} p_{ij}(1-p_{ij}) + 2\sum_{i=1}^{n}\sum_{l=i}^{n} Cov(X_{ij}, X_{il})$, with $Cov(X_{ij}, X_{il})$ capturing the fact that the arrivals of passengers travelling together are correlated. Incorporating correlations between passenger arrivals is also a convenient way of recalibrating predictions: if predictions assuming independent arrivals are overconfident, i.e., the prediction intervals are too narrow, incorporating positive correlations among passenger arrivals will make the predictions well-calibrated. If the predictions are underconfident, i.e., the prediction intervals are too wide, negative correlations among passenger arrivals will again make the predictions well-calibrated.

Although we can still derive the mean and variance of the distribution for passenger flows, incorporating correlated arrivals makes it difficult to derive analytical expressions for the entire distribution and its quantiles. Instead, we run simulations to generate distributional forecasts or quantile forecasts, with the X_{ij} correlated through Gaussian copulas, a multivariate cdf defined as $C_R(\mathbf{U}) = \Phi_R(\Phi^{-1}(U_1), \Phi^{-1}(U_2), ..., \Phi^{-1}(U_n))$, where Φ^{-1} is the inverse cdf of a standard normal distribution, and Φ_R is the joint cdf of a multivariate normal distribution with zero mean vector and correlation matrix $\mathbf{R}$. The term $\mathbf{U} = (U_1, U_2, ..., U_n)$ is a vector of uniformly distributed random variables, with $(U_1, U_2, ..., U_d) = (B_{i1}(X_{i1}), B_{i2}(X_{i2}), ..., B_{in}(X_{in}))$, where B_{ij} is the cdf of the Bernoulli distribution with parameter p_{ij}, the probability of passenger j arriving during the ith time interval.

Unfortunately, our data does not show which passengers were travelling together. Therefore, we assume that passengers arriving on the same flight are correlated with each other in a homogenous way, using ρ. We also assume the same copula parameter ρ for all flights. As a result, the off-diagonal elements in $\mathbf{R}$ are all equal to ρ if the passengers arrive on the same flight, and zero otherwise. The copula parameter ρ can be viewed as a tuning parameter, chosen using a grid search to minimize the average pinball loss using the two-fold validation process.

To compute the distribution of passenger flows, we use the following procedure: First, in each iteration of the simulations for a time interval i, we randomly

generate a uniformly distributed vector between 0 and 1, $(U_1, U_2, ..., U_d)$, from the Gaussian copula. Next, to obtain the binary variables $(X_{i1}, X_{i2}, ..., X_{in})$, we apply the quantile function (or the inverse cdf) of their corresponding Bernoulli distributions to the vector of probabilities generated in the first step. Finally, for time interval i, we calculate the number of passenger arrivals by summing $(X_{i1}, X_{i2}, ..., X_{in})$. The distribution of the passenger flow is then approximated by the empirical distribution constructed by the number of arrivals obtained from the simulations.

Instead of modelling dependencies among passengers' arrivals X_{ij}, we also tried to include dependencies among passengers' connection times, which can be defined as continuous variables. In this case, the accuracy results are similar to those obtained by applying the procedure described above.

4.4 Results

Next we present results derived from our predictive model. We first report on the accuracy of our model on the 20% test set in predicting connection times and passenger flows compared against several benchmarks. Next, we highlight a few key findings from the regression tree model.

4.4.1 Accuracy of the Distributional Forecasts for Individuals' Connection Times

We first compare the accuracy of our regression tree method in predicting the 0.05, 0.25, 0.5, 0.75 and 0.95 quantiles of individuals' connection times. Among the methods that can generate distributional forecasts, four are widely used by the machine learning community: linear regression, quantile regression, quantile regression forests, and gradient boosting machines (Hong et al. 2016). Quantile regression and gradient boosting machines estimate different quantiles independently, possibly resulting in a lack of monotonicity in the estimated quantile function. This longstanding problem is also known as the quantile crossing problem (Bassett and Koenker 1982). Chernozhukov et al. (2010) propose a method of rearranging the curve into a monotone curve. This, however, requires the estimates of thousands of quantile regressions, making the method computationally expensive. Linear regres-

sion and quantile regression forests, on the other hand, are able to produce monotone quantiles. Linear regression is also easy to fit and usually runs fast. However, it does not perform well with nonlinear relationships and complex interactions. Quantile regression forests are a generalization of the random forests and are competitive in terms of predictive power (Meinshausen 2006). They are, however, time consuming and typically treated as a black box.

We compare our regression tree model with these four methods and a naïve forecast on the test set. The naïve model predicts passengers' connection times based on their arrival terminals, the most important predictor given by our regression tree. Specifically, given a new passenger's arrival terminal, the quantiles of her connection time are predicted as the quantiles of the connection times of all passengers arriving at the same terminal in the training set. The linear regression and the quantile regression are fit to all 17 variables that were selected as predictors in Section 4.3.1 and their interactions. We fit these two models to the data, with the goal of predicting the logarithmic transformation of the connection times. For the linear regression, the distribution of a passenger's connection time is predicted as a log-normal distribution with the mean and standard deviation set to the point forecast, and the standard error of the forecast from the linear regression, respectively. The standard error of the forecast generated from a linear regression is typically calculated as the square root of the residuals' variance plus the variance contributed by the regression coefficients. For the quantile regression, the exponential of the predictions generated from the model are the quantile forecasts of connection times. We also found that the LASSO penalized quantile regression (Friedman et al. 2010) produces more accurate forecasts than the quantile regression without penalty terms. The LASSO linear regression is not considered here because it is not typically used for producing prediction intervals due to the difficulties in estimating standard errors of the coefficients, which are part of the uncertainties captured by prediction intervals (Goeman 2010, Goeman et al. 2018).

The other two methods, quantile regression forests and gradient boosting machines are tree-based models. A quantile regression forest fits independent trees

and constructs conditional distributions from these trees. Gradient boosting is an ensemble technique in which the regression trees are not fit independently, but sequentially, learning from one tree to the next. Since the quantile regression forest is extremely time consuming and requires processing with a large amount of memory, we fit the model to the seven key factors identified by our regression tree to balance accuracy and computational cost. Details of these factors will be discussed in Section 4.4.3. For the gradient boosting machine, we use all 17 variables as predictors. We tune the number of trees for both models to avoid overfitting. The quantile regression forest model is trained using the log transform of connection times. The gradient boosting machine is the only model that fits to the original connection times, as we found the log transform of the target variable makes the model performs worse than without the transformation.

As shown in Table 4.1, among all six models, our regression tree has the lowest out-of-sample average pinball loss (2.74) and is best on four of the five quantiles in the test set. Not surprisingly, the naïve model performs the worst with an average pinball loss of 3.29. The quantile regression forest, which is often considered as an advanced machine learning method with high accuracy, performs worse than the third best model, the LASSO penalized quantile regression. The performance of the quantile regression forest could be due to the fact that only seven variables were used to train the model to reduce computing time.

The predictions generated from our regression tree model are also well-calibrated, evidenced by a roughly uniform empirical density of the PIT (Fig-

	RMSE	Pinball losses					
	of mean	$Q_{0.05}$	$Q_{0.25}$	$Q_{0.50}$	$Q_{0.75}$	$Q_{0.95}$	Avg.
Naïve model	15.93	0.92	3.40	4.90	4.73	2.48	3.29
Linear regression	14.12	0.82	2.79	4.12	4.24	2.24	2.84
LASSO quantile regression	14.10	0.79	2.80	4.09	4.03	2.10	2.76
Quantile regression forest	14.13	0.81	2.83	4.12	4.07	2.16	2.80
Gradient boosting machine	14.20	0.79	2.82	4.08	**4.00**	2.09	2.76
Regression tree	**14.00**	**0.77**	**2.77**	**4.07**	4.02	**2.08**	**2.74**
Significance[a]	*	***	***	*			***

[a] The symbol *** and * indicate the difference between the regression tree model and the second best model in each column is significant based on a t-test at the 1% and 10% level, respectively. Values in bold indicate the lowest errors.

Table 4.1: Accuracy of Forecasts on Connection Times in the Test Set.

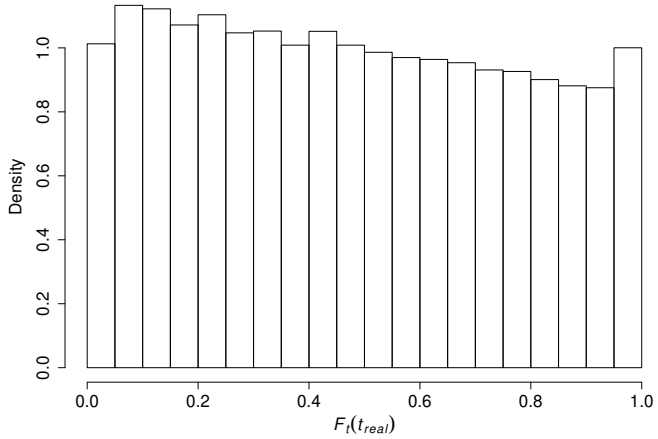


Figure 4.3: Empirical PIT Densities from the Distributional Forecasts of Connection Times Generated by the Regression Tree.

ure 4.3), the density of the predicted cdf F_t for passengers' connection times evaluated at the realizations t_{real}. The accuracies of the predicted distributions evaluated by other performance measures (CRPS and log scores) can be found in Table 4.9 in the Appendix. In terms of point forecasting, our regression tree model has the lowest RMSE. The mean is used as point forecast for the naïve model, linear regression, quantile regression forest and our regression tree model. The 0.5 quantile is used as point forecasts for LASSO quantile regression and gradient boosting machine as these methods are designed to predict quantiles.

Among all six models in Table 4.1, only linear regression and our regression tree model can generate forecasts of the entire distribution and therefore can be used in our two-phased approach. The current implementation of quantile regression forests in leading machine learning tools does not provide complete leaf-distributions as outputs. For the gradient boosting machine and the quantile regression, thousands of models need to be fit and stored to construct an empirical distribution and simulate from it, making the model-training process inefficient. The crossing problem inherent to these two models may also make it difficult to construct empirical distributions.

Since we use the predictions of passengers' connection times to identify late transfer passengers, we also compared the accuracy of our regression tree model

with several benchmarks in the context of a classification problem. We first calculate the time difference between a passenger's scheduled departure time and their arrival time at the conformance desk. A passenger is considered to be late if this time difference is less than 30 minutes. Table 4.2 shows the out-of-sample average log loss and brier score of our regression tree model and benchmark models on predicting probabilities of being late. The benchmark models considered include the naïve model, linear regression, LASSO Logistic regression, random forest with binary target variable, gradient boosting machine with binary target variable, and classification tree. The naïve forecast is defined as the proportion of late passengers in the training set.

Among all seven models in Table 4.2, the classification tree is the most accurate in log loss, while the regression tree that predicts the distribution of passengers' connection times is the most accurate according to the brier score. We note that the differences between our method and the classification tree are not statistically significant. Given the results above, our regression tree model performs favourably in both identifying late passengers and predicting distributions of connection times, making it the preferred model to be used in the predictive system's first phase.

4.4.2 Accuracy of Aggregate Forecasts for Arrivals at Immigration and Security

Based on the predicted number of passenger arrivals, airports can decide how many immigration desks and security lanes to open throughout the day. These two decisions are based on the passenger flows into the immigration and security areas,

	Log Loss	Brier Score
Naïve model	0.3030	0.07092
Linear regression	0.0396	0.00971
LASSO logistic regression	0.0436	0.01143
Random forest	0.0585	0.00965
Gradient boosting machine	0.0311	0.00760
Classification tree	**0.0295**	0.00742
Regression tree	0.0304	**0.00736**

The difference between the best and second best model in each column is not statistically significant.

Table 4.2: Accuracy of Forecasts of the Probability of Being Late in the Test Set.

respectively. As discussed in Section 4.2, only passengers connecting to domestic destinations need to go though immigration desks. Therefore, we generate two sets of forecasts of the passenger flows, one for those connecting to domestic destinations, and the other for those connecting to international destinations. Since Heathrow's security resources are planned in 15 minutes intervals, we use same windows to construct passenger flow patterns. In Section 4.6 we will examine the implication of the quality of the forecasts on the resourcing decisions. To generate forecasts from our two-phased approach, we use the two-fold cross validation approach to search for the copula parameter ρ of the Gaussian copula,[3] with the goal of of minimizing the average pinball loss over five quantiles during business hours of 5:00 am to 10:00 pm daily.

We construct a naïve model as a benchmark to predict passenger flows. For each day in the training set, we calculate the number of passengers connecting to domestic (international) destinations that arrive at the immigration (security) areas during each 15-minute interval. These numbers of passengers across all days in the training set are then used to construct the naïve distribution of the domestic (international) passenger flows.

Univariate time series models have long been used to forecast short-term passenger or customer arrivals (Taylor 2008, Wei and Chen 2012). Among these models, the Seasonal Autoregressive Integrated Moving Average Model (SARIMA) emerges as a benchmark since it can handle complex seasonality and is often accurate in forecasting arrivals (Taylor 2012, Hyndman and Athanasopoulos 2018).[4] We fit a SARIMA model to the time series of the number of arrivals at the conformance desk, with a frequency of 15 minutes, i.e. with 96 periods for a 24-hour day. We consider the following variables calculated for the current period (t), and for the previous three 15-minute periods ($t-1$, $t-2$ and $t-3$), as covariates in the SARIMA model: (1) the number of passengers arriving at the airport and traveling

[3] We also tested 5-fold approach, and the optimal ρ ends up the same.

[4] We also experimented with other time series models, including the Exponential Smoothing State Space Model with Trigonometric Seasonality, Box-Cox Transformation, ARMA Errors, and Trend And Seasonal Components (TBATS). All these models underperformed compared to the SARIMA model.

to domestic destinations (n_{dom_t}, $n_{dom_{t-1}}$, $n_{dom_{t-2}}$, $n_{dom_{t-3}}$), (2) number of passengers arriving at the airport and traveling to international destinations (n_{int_t}, $n_{int_{t-1}}$, $n_{int_{t-2}}$, $n_{int_{t-3}}$), (3) number of passengers arriving at Terminal 5 (n_{T5_t}, $n_{T5_{t-1}}$, $n_{T5_{t-2}}$, $n_{T5_{t-3}}$), and (4) the number of passengers arriving in economy class (n_{econ_t}, $n_{econ_{t-1}}$, $n_{econ_{t-2}}$, $n_{econ_{t-3}}$). We consider only the previous three 15-minute periods as 90% of the passengers in the data set spent less than 45 minutes to make their connections.

Logarithmic and square root transformations are often applied to time series to stabilize count data and enforce positive predictions (Taylor 2012). We apply square root transformation in our study as the data contains many zeros. To identify seasonality in the data, we inspected the autocorrelation function (ACF, Venables and Ripley 2002) and the periodogram (Bloomfied 2004). Only daily seasonality (hour of the day) can be easily observed from the results. Although we did not find evidence of weekly seasonality (day of the week), we also tested models with weekly seasonality since it was reported to be an important factor on passenger delays in Barnhart et al. (2014).

The SARIMA model is fit to the first 80% of the days of data, and used thereafter to produce one-period-ahead forecasts for the remaining 20%. When we predict the number of passengers arriving at the conformance desk during the interval $(t,t+1]$, we use the actual number of arrivals at the conformance desk up to time t, and assume the information of passengers arriving at the airport during $(t,t+1]$ is also available so we can calculate the covariates listed above. The final SARIMA model, SARIMA$(5,0,1)(0,1,0)^{96}$, can be written as $S_t = S_{t-96} + \beta_1(S_{t-1} - S_{t-97}) + \beta_2(S_{t-2} - S_{t-98}) + \beta_3(S_{t-3} - S_{t-99}) + \beta_4(S_{t-4} - S_{t-100}) + \beta_5(S_{t-5} - S_{t-101}) + \beta_6\varepsilon_{t-1} + \varepsilon_t + \mathbf{bV}$, where S_t is the transformed time series of passengers' arrivals at the conformance desk, ε_{t-1} and ε_t are the error terms, $\mathbf{V}$ is a vector of the covariates listed above, and $\mathbf{b}$ are their coefficients. Here, for example, when generating the prediction for S_t, $S_{t-101}, S_{t-100} \ldots S_{t-1}$ are the actual number of arrivals at the conformance desk during these periods, ε_{t-1} is set to zero, ε_t follows $N(0,\sigma)$, where σ is estimated using the training data. All the covariates captured in $\mathbf{V}$ are calculated using the actual information of passengers arriving at

the airport in the current period and in the last 45 minutes. Finally, we transform the prediction of S_t back to its original scale.

Two benchmark models are included to simulate Heathrow's legacy systems. In the legacy systems, individual transfer passengers' information was not used. Rather, the number of passengers transferring to domestic (or international) destinations on an arrival flight was estimated as the total number of passengers on the flight, which is known by the airport, multiplied by the estimated percentage of passengers transferring to domestic (or international) destinations on similar flights in the training data. The estimated percentage for an arrival flight was calculated as the percentage of passengers transferring to domestic (or international) destinations on the flights in the training set that arrived from the same region, at the same terminal, and run by the same carrier. Passengers arriving on the same flight were assumed to have the same connection time. The distribution of their connection time was estimated as the empirical distribution of the connection times of passengers arriving at the same terminal, and during the same hour of the day in the training set.

To compute distributions of passenger flows, we use the same simulation approach described in Section 4.3.3. We test two versions of Heathrow's legacy systems: a static version, which uses flights' scheduled arrival times to approximate passengers' arrival times at the airport, and a dynamic version, which uses their actual arrival times at the airport. It should also be noted that Heathrow's legacy systems only generated point forecasts. Here, however, we generalize their methods to produce also distributional forecasts.

Next we measure the accuracy of the models in predicting passenger flows between 5:00 am and 10:00 pm. The point forecasting errors and pinball losses are presented in Table 4.3 and Table 4.4. The results in Table 4.3 are for domestic destinations, and the results in Table 4.4 are for international destinations. The last three models in the table apply the two-phased approach using linear regression and regression tree as their first phase model to predict connection times. These models are much more accurate in both quantile forecasting and point forecasting than the other models. Our regression tree model with copula-based simulations

	RMSE of mean	Pinball losses					
		$Q_{0.05}$	$Q_{0.25}$	$Q_{0.50}$	$Q_{0.75}$	$Q_{0.95}$	Avg.
Naïve model	14.97	1.07	3.82	5.45	4.91	1.86	3.42
SARIMA with covariates	13.46	1.06	3.51	4.72	4.15	1.57	3.00
Static legacy system	15.50	1.03	3.65	5.19	4.94	2.58	3.48
Dynamic legacy system	12.25	0.82	2.90	4.01	3.63	1.56	2.59
Linear regression with copula	7.66	0.62	2.02	2.58	2.10	0.72	1.61
Regression tree without copula	7.42	0.97	2.06	2.45	2.11	1.07	1.73
Regression tree with copula	**7.42**	**0.60**	**1.91**	**2.44**	**2.01**	**0.71**	**1.53**
Significance[a]	*	**	***	***	**		***

[a] The symbol ***, **, and * indicate the difference between the regression tree model with copula and the second best model in each column (excluding the regression tree model without copula) is significant based on a t-test at the 1% , 5% and 10% level, respectively.

Table 4.3: Accuracy of the Predicted Flow of Passengers Connecting to Domestic Destinations in the Test Set.

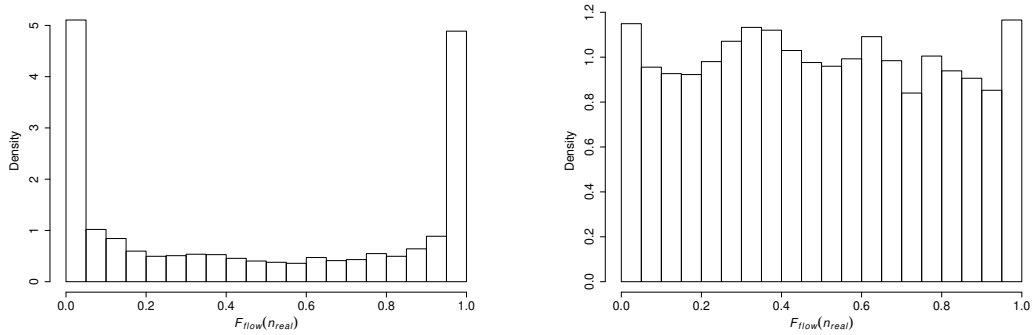
outperforms the other models on all five quantiles. It is also the most accurate in generating point forecasts for the domestic passenger flow. The two-phased approach with linear regression as the first phase model performs slightly better in point forecasting of the international passenger flow. The difference between the two RMSEs, however, is not statistically significant.

The passenger flows predicted by the regression tree using independent conformance desk arrivals (i.e. without using copulas) are overconfident. This is also the case when connection times are predicted by the linear regression. Figure 4.4(a) shows the PIT density for the forecasts generated from our model without using copulas for the flow of passengers connecting to international destinations. As shown in

	RMSE of mean	Pinball losses					
		$Q_{0.05}$	$Q_{0.25}$	$Q_{0.50}$	$Q_{0.75}$	$Q_{0.95}$	Avg.
Naïve model	53.09	4.77	15.63	19.83	15.69	5.16	12.22
SARIMA with covariates	39.40	3.42	11.16	14.82	12.56	4.56	9.30
Static legacy system	53.37	5.17	16.16	20.81	17.28	5.84	13.05
Dynamic legacy system	41.66	4.61	13.45	16.43	13.29	4.58	10.47
Linear regression with copula	**22.33**	2.13	6.43	8.05	6.70	2.38	5.14
Regression tree without copula	22.41	3.79	6.83	7.94	7.02	4.14	5.94
Regression tree with copula	22.41	**2.04**	**6.21**	**7.93**	**6.60**	**2.35**	**5.03**
Significance[a]		**	**	*			*

[a] The symbol ** and * indicate the difference between the regression tree model with copula and the second best model in each column (excluding the regression tree model without copula) is significant based on a t-test at the 5% and 10% level, respectively.

Table 4.4: Accuracy of the Predicted Flow of Passengers Connecting to International Destinations in the Test Set.



(a) Regression tree with independent simulations

(b) Regression tree with copula-based simulations

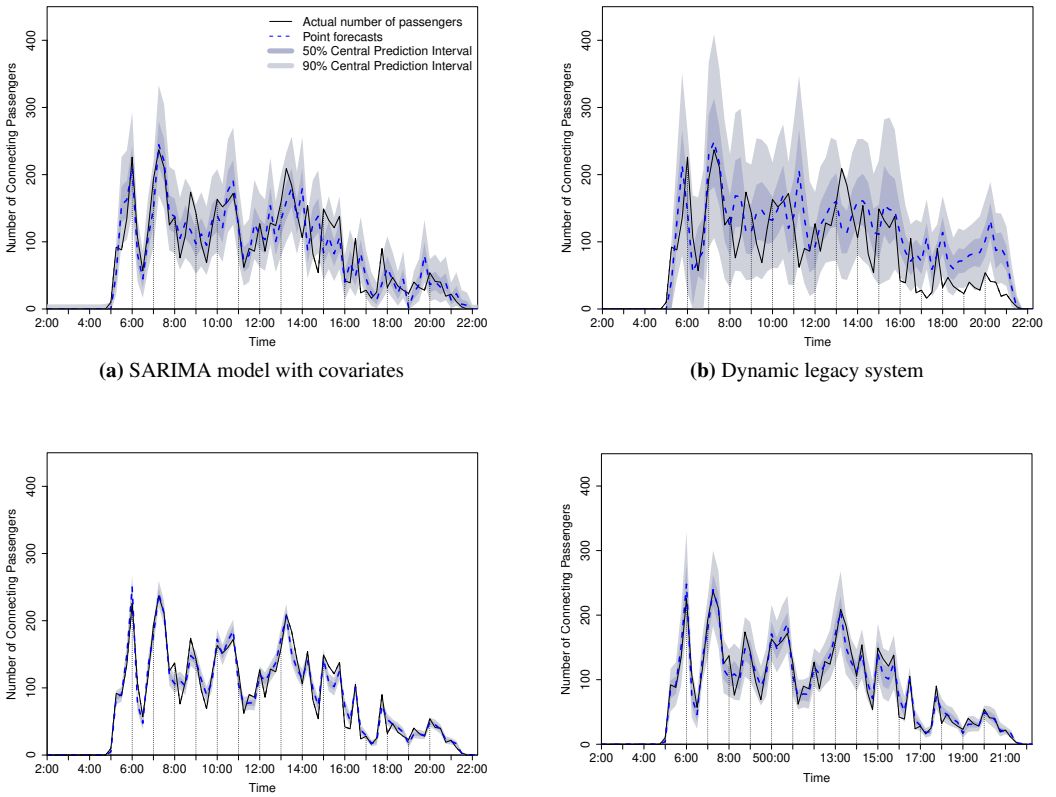
Figure 4.4: Empirical PIT Densities from the Distributional Forecasts of the Flow of Passengers Connecting to International Destinations Generated by Our Model.

the figure, too many realizations fall in the tails of the predicted cdf. Incorporating dependences between passengers' arrivals using Gaussian copula results in a more uniform PIT density (Figure 4.4(b)). The PIT densities for the flow of passengers connecting to domestic destinations exhibit similar patterns.

Figure 4.5 presents the prediction intervals for the flow of passengers connecting to international destinations on a randomly selected day from our test set. These prediction intervals are generated by the SARIMA model with covariates (Figure 4.5(a)), the dynamic legacy model with copula-based simulations (Figure 4.5(b)), and the regression tree model using independent and copula-based simulations, Figure 4.5(c) and Figure 4.5(d), respectively. It is easy to observe the accuracy of the forecasts from the two-phased models. The prediction intervals produced when assuming independence (Figure 4.5(c)) are clearly much narrower than when using copula-based simulations (Figure 4.5(d)). Such narrow intervals result in overconfidence and misjudged decisions regarding peak activity. Similar patterns hold when using alternative scoring rules. The CRPS and log scores of the models' distributional forecasts are shown in Table 4.10 in the Appendix.

4.4.3 Key Findings from the Model

As the tuned regression tree model has more than 2,000 leaves, we use a pruned tree for deriving insights. The pruned version of the tree partitions the passengers into



(a) SARIMA model with covariates

(b) Dynamic legacy system

(c) Regression tree with independent simulations (PIT shown in Figure 4.4(a))

(d) Regression tree with Copula-based simulations (PIT shown in Figure 4.4(b))

Figure 4.5: Forecasts and Actuals of the Flow of Passengers Connecting to International Destinations at the Conformance desk, On a Random Test Day.

only a few segments, making it easier to explain why a passenger's connection time is predicted in a particular manner.

We find when the tree is fit to different subsamples of the data, and with different settings of the tuning parameters, the pruned tree, which is essentially the first four levels of the full tree, always relies on the same features, and the sequences of these features chosen to be split are the same. We also find that the coefficients of variation of the variables' cutting points that appear in the first four levels are within a reasonable level (5%). In addition, the pruned tree is visually manageable so that we can easily derive insights from it. When generating forecasts from our predictive system, we use the full tree because of its accuracy. The pruned tree was only applied to generate insights.[5]

[5]The average pinball loss of the reduced version of the tree on the test set increases from 2.74 to

The first four levels of the tree have partitioned the passengers into 16 segments. Figure 4.6 visualizes the pruned tree trained to the entire training set. Each node of the tree presented in the figure gives information on the name of the feature split in this node, percentage of passengers falling in this node, and the average connection time of these passengers. A summary of the 16 segments, represented by the 16 leaf nodes in Figure 4.6, is provided in Table 4.11 in the Appendix. The empirical distributions estimated by kernel smoothing for leaves are shown in Figure 4.7. Using the results summarized in the table and the figures, we highlight three key findings regarding passengers' connection times.

Key Finding 1: Key Factors The key factors that impact passengers' connection times are (1) whether or not the passenger arrives at Terminal 5, (2) whether or not the passenger arrives from an European Union (EU) country, (3) perceived connection time, (4) whether or not the passenger is in economy class, (5) hour of the day the arriving flight lands at the airport, (6) arriving flight's stand type (pier served or remote stand), and (7) the total number of passengers on the arriving flight. Although our predictive model contains 17 predictors, the first four levels of the tree only use seven of them. Therefore, we treat only these seven features listed above as the key factors that impact passengers' connection times.

Passengers arriving at Terminal 5 need less time to make the connection since they do not need to take a transfer bus. European airports usually have similar layouts, and therefore passengers arriving from EU countries may be more familiar with the transfer journey at Heathrow. Perceived connection time indicates a passenger's stress level when he or she transfers through the airport. If the perceived connection time is short, a passenger may move faster to the conformance desk. Business and first class passengers sit in front of the aircraft so that they can disembark from the aircraft faster than those in economy class. Moreover, these passengers usually carry less hand luggage, thereby also moving faster through the airport. Hour of the day the arriving flight lands at the airport is related to the level of busyness of the airport. A passenger can move faster during hours when the air-

2.86. The RMSE for the point forecasts increases from 14.0 to 14.4.

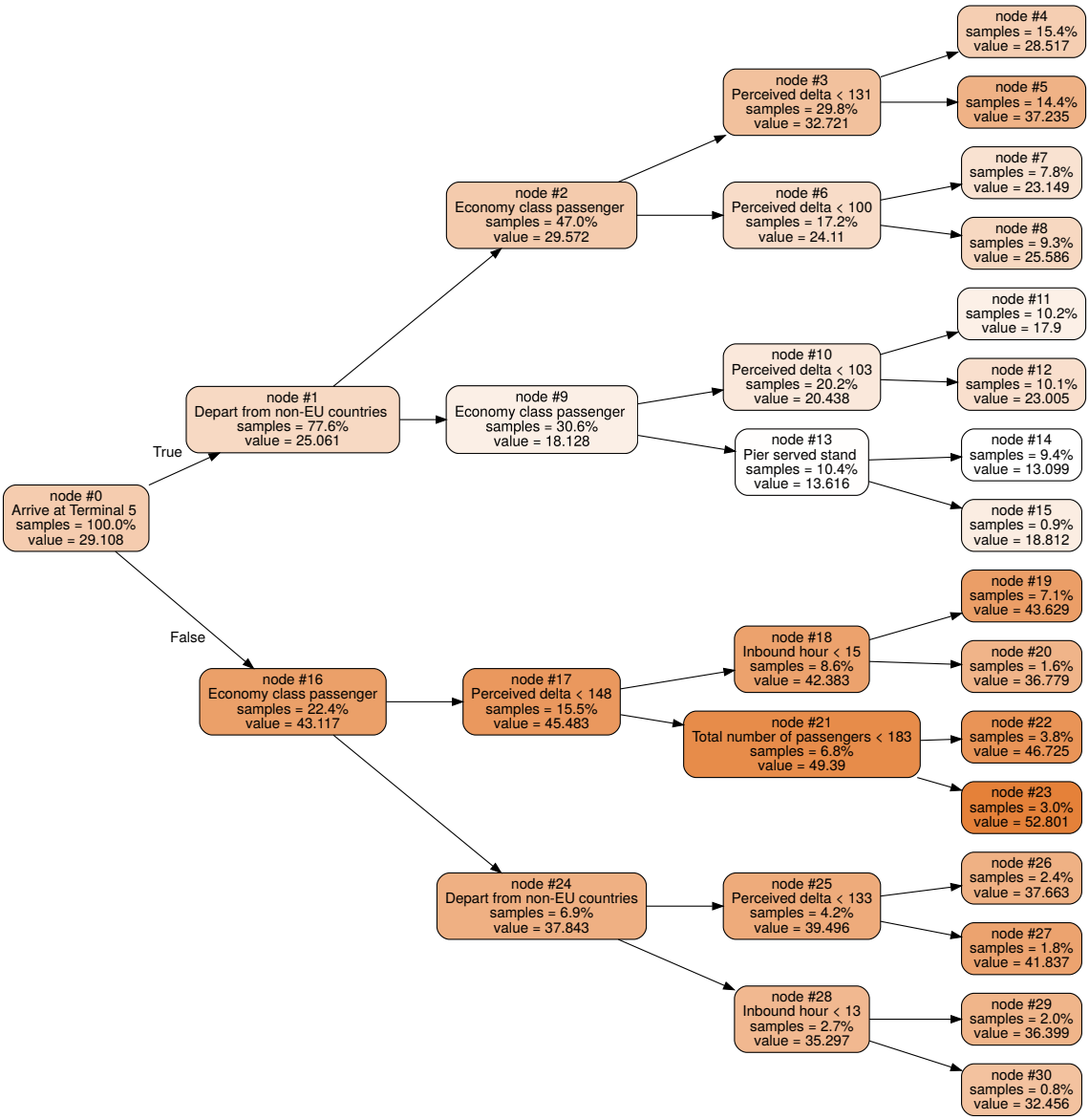


Figure 4.6: The First Four Levels of the Tree Trained to the Entire Training Set (The value shown in each node (or segment) is the mean of the connection times).

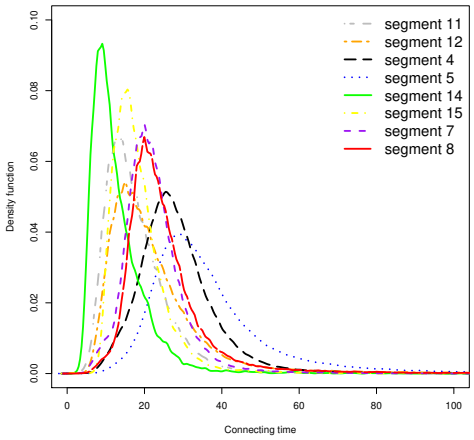
port is not busy. Passengers arriving from an aircraft parked at a remote stand need to first take a shuttle bus to the terminal, making their connection times longer than those arrived at pier served stand. Finally, if the arriving flight has a large number of passengers, it may take longer to disembark from the aircraft.

Key Finding 2: Expected Connection Times Passengers arriving at Terminal 5, from EU countries, in business or first class, whose arriving flight parks at a pier-served stand, take the shortest amount of time to make their connections (Segment

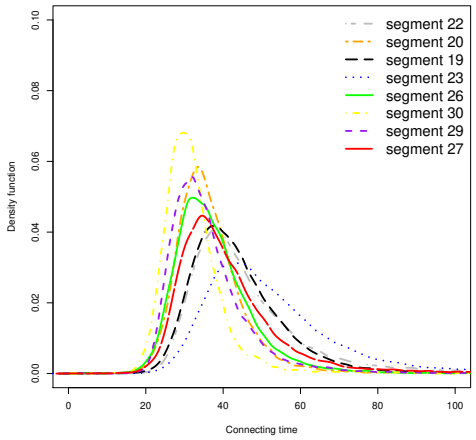
14 in Figure 4.6 and 4.7). We also find that passengers arriving at Terminal 5 and departing from EU countries have shorter connection times compared with other passengers. This is regardless of the value of the rest of the predictors. According to Table 4.11, these passengers form the first four segments that have the lowest connection times.

Passengers arriving at Terminal 2, 3 or 4, in economy class, with perceived connection time greater than 148 minutes, whose arriving flight has more than 183 passengers, take the longest time to make their connections (Segment 23 in Figure 4.6 and 4.7).

Key Finding 3: Uncertainty of Connection Times In general, the uncertainty and average value of passengers' connection times within a segment are positively correlated. For most segments shown in Figure 4.7(b), not only are the average connection times longer, but the variances among the passengers are also larger compared to those in Figure 4.7(a). The uncertainty, however, does not increase by as much as the average, which can be seen from the coefficient of variation, defined as the ratio of the standard deviation to the average connection time. The coefficient of variation compares the degree of variation of the connection times in different segments, where the means are drastically different from one another.



(a) Segments with low average connection times



(b) Segments with high average connection times

Figure 4.7: Smoothed Empirical Distributions of the Connection Times in the 16 Segments (The numbered nodes in the figures are the leaf nodes shown in Figure 4.6).

Based on the results shown in Table 4.11, the higher the connection time, the lower the coefficient of variation. Therefore, connection times of passengers falling into the segments with longer connection times are more predictable. The variations of connection times in these segments are smaller than those with shorter connection times.

4.5 Real-Time Implementation

With the predictive model at hand, we worked with Heathrow's APOC to develop an application for forecasting individual connection times, and the flow of transfer passengers into the immigration and security areas in real time. We use these predictions to also calculate the probability of a passenger missing the connection, and the expected number of late passengers for each outbound flight. The application generates forecasts on a rolling basis and updated every 15 minutes. A prototype of the predictive system was first tested at Heathrow in 2016. In order to get buy in from the various APOC stakeholders, the project team spent a day presenting the model to each group and discussing the implications and new possibilities made available by the improved predictions.

Real-time flight-level information was exported from the Airport Flight Operations System, also known as IDAHO. Real-time passenger level information was obtained from Passenger Transfer Message (PTM) files. These files are sent by the airline when a flight takes off from an origin airport. PTM files are currently the only real-time data source that contains passenger level information. How far in advance of the flight's arrival these files get sent by the airline has a significant impact on the accuracy of our predictions; for instance, missing PTMs can cause forecasts to underestimate the passenger flow. According to Heathrow, PTMs for 88% of the passengers are received more than 90 minutes prior to arrival. Therefore, to ensure that there is sufficient data to enable the model to provide accurate forecasts, the forecasting window chosen for the application was 90 minutes. If a flight is arriving in the next 90 minutes but has not sent the PTM message, the system will use historical data to estimate the number of transfer passengers on the flight and their

connection times. Specifically, the average historical number of transfer passengers on a flight and the distribution of historical connection times between terminals are used to simulate passenger arrivals in the second phase of the model.

It should also be noted that the actual on-chock time, the time when an aircraft is parked at gate, will not be available if an aircraft is en-route. In this case, we used estimated on-chock time. If we had neither of the actual or estimated on-chock time, we used the actual time of ground handling, the estimated time of ground handling, or the scheduled time of ground handling. If none of these five fields was presented in IDAHO, we dropped the record.

A few variables in our historical data set were not available in real-time. Some of these variables, such as stand number and runway number which may indicate the distance between gate and transfer bus station, could be significant predictors in predicting connection times. Once these variables become available in the future, the predictive model should be retrained and reassessed. Other variables that are currently important, such as stand type, may become less significant predictors.

A prototype for real-time forecasting of passenger connection times was developed using a Python GUI scripting interface. Forecasts were generated every 15 minutes, on a rolling basis, for the next 90-minute time window. At the start of each iteration, the application collected real-time information of passengers who have arrived in the previous 150 minutes or will arrive in the next 90 minutes. The application ran for approximately three minutes on a typical Heathrow machine, and generated as output several CSV files containing individual level and aggregate level forecasts. Plots were also generated to visualize the forecasts of transfer passenger flows.

In the application, users can set the granularity of the passenger flow and the forecasted time-horizon. Forecasts with different granularities are generated for different purposes. For example, the forecasts of passengers arriving in every 5 minutes provide detailed flow profiles, while the forecasts of passengers arriving in every 15 minutes can be used to adjust resourcing plans.

Figure 4.8 shows an example of the output for the forecasts of individual con-

nection times generated at 12:00 on July 1, 2016. Each row in the file represents the forecast for one passenger. In addition to the quantile forecasts of passengers' arrival time at immigration and security areas, this file also contains probabilities of these passengers being late for their connecting flights. Here, a passenger is considered to be late if they arrive at the conformance desk later than 30 minutes before the scheduled departure time of the connecting flight. Based on the predicted probabilities, Heathrow and airlines can easily identify which passengers are at risk of missing their onward flight. Given this information, they would be able to help late passengers move faster through the airport and facilitate early rebooking. The threshold time of rebooking late passengers may vary among airlines. In those cases, managers can easily set different thresholds in the system.

In the output shown in Figure 4.9, we grouped passengers by their outbound flights, and calculated how many of them are expected to be late based on the simulation results. We also calculated the expected number of passengers that would be still at risk if the airline delayed the departure time by 5, 10, 20, and 30 minutes. These forecasts can help improve the predictability and stability of outbound flights' scheduled departure time. During the aircraft turn around, its scheduled departure time can be adjusted, but only a few times. Many airlines consider passenger delays when they amend flights' scheduled departure times. In this case, predictions of

	A	B	C	D	E	F	G	H	I	J	K
1	passenger_id	on_chock_time	q0.05	q0.25	median	q0.75	q0.95	ib_flight_no	ob_flight_no	P(missing connecting flight)	
2	323698	01/07/2016 12:46	01/07/2016 13:09	01/07/2016 13:16	01/07/2016 13:22	01/07/2016 13:29	01/07/2016 13:46	BA847	BA293	0	
3	323723	01/07/2016 12:19	01/07/2016 12:43	01/07/2016 12:51	01/07/2016 12:56	01/07/2016 13:04	01/07/2016 13:19	BA479	BA069	0	
4	324028	01/07/2016 11:42	01/07/2016 11:52	01/07/2016 11:57	01/07/2016 12:01	01/07/2016 12:06	01/07/2016 12:14	BA309	BA115	0	
5	324213	01/07/2016 13:23	01/07/2016 13:34	01/07/2016 13:38	01/07/2016 13:41	01/07/2016 13:46	01/07/2016 13:57	BA763	BA279	0.11	
6	323846	01/07/2016 11:45	01/07/2016 11:56	01/07/2016 12:00	01/07/2016 12:04	01/07/2016 12:09	01/07/2016 12:18	BA565	BA287	0.01	
7	322652	01/07/2016 12:35	01/07/2016 12:58	01/07/2016 13:05	01/07/2016 13:11	01/07/2016 13:18	01/07/2016 13:31	MS777	BA269	0	
8	323561	01/07/2016 12:06	01/07/2016 12:18	01/07/2016 12:21	01/07/2016 12:25	01/07/2016 12:29	01/07/2016 12:38	BA757	BA1394	0.80	
9	323767	01/07/2016 12:28	01/07/2016 12:39	01/07/2016 12:43	01/07/2016 12:47	01/07/2016 12:51	01/07/2016 13:01	BA343	BA049	0	
10	323759	01/07/2016 12:28	01/07/2016 12:39	01/07/2016 12:43	01/07/2016 12:47	01/07/2016 12:52	01/07/2016 13:03	BA343	BA227	0	
11	323493	01/07/2016 11:31	01/07/2016 11:41	01/07/2016 11:46	01/07/2016 11:49	01/07/2016 11:54	01/07/2016 12:03	BA805	BA1484	0.96	
12	324030	01/07/2016 11:42	01/07/2016 11:59	01/07/2016 12:07	01/07/2016 12:14	01/07/2016 12:22	01/07/2016 12:36	BA309	BA113	0	
13	323657	01/07/2016 12:11	01/07/2016 12:28	01/07/2016 12:32	01/07/2016 12:38	01/07/2016 12:44	01/07/2016 12:57	BA573	BA197	0.22	
14	323967	01/07/2016 11:34	01/07/2016 11:45	01/07/2016 11:49	01/07/2016 11:52	01/07/2016 11:57	01/07/2016 12:07	BA431	BA227	0	
15	324058	01/07/2016 12:10	01/07/2016 12:34	01/07/2016 12:39	01/07/2016 12:45	01/07/2016 12:51	01/07/2016 13:04	EI712	BA279	0	
16	321912	01/07/2016 12:28	01/07/2016 12:51	01/07/2016 12:58	01/07/2016 13:05	01/07/2016 13:12	01/07/2016 13:25	QR003	BA203	0	
17											

ID of the passenger

Inbound flight estimated on-chock time

0.05, 0.25, 0.50, 0.75 and 0.95 quantiles of a passenger's arrival time at immigration and security areas

Inbound and outbound flight no.

probability of a passenger being late for her connection flight

Figure 4.8: Output from the Application: Individual Connection Times.

passenger delays ahead of time can help airlines make accurate adjustments. Moreover, if our predictions are accurate, no further changes are to be expected because of late transfer passengers; and therefore, the stability of the scheduled departure time can be improved.

In the output file of passenger flow forecasts (Figure 4.10), each row represents the forecast for a 15-minute time interval. The 90% prediction intervals and the median of the 5-minute and 15-minute passenger flows are also visualized in the output figures. These real-time predictions allow the dynamic planning of immigration and transfer security resourcing, by applying the real-time demand forecasts to

	A	B	C	D	E	F
1	ob_flight_no	current SDT	+5 min	+10 min	+20 min	+30 min
2	BA005	3	2	2	1	0
3	BA007	22	20	17	11	6
4	BA009	1	1	0	0	0
5	BA011	0	0	0	0	0
6	BA017	2	2	1	1	1
7	BA031	0	0	0	0	0
8	BA033	0	0	0	0	0

Outbound flight no.

Expected number of late passengers with current scheduled time for departure, and current scheduled time for departure + 5 min(C), +10 min(D), +20 min(E), and +30 min(F)

Figure 4.9: Output from the Application: Expected Number of Late Passengers for Each Outbound Flight.

	A	B	C	D	E	F	G
1	inter_begin	inter_end	q0.05	q0.25	median	q0.75	q0.95
2	01/07/2016 12:00	01/07/2016 12:15	233	244	253	260	271
3	01/07/2016 12:15	01/07/2016 12:30	217	227	235	244	257
4	01/07/2016 12:30	01/07/2016 12:45	172	183	190	197	207
5	01/07/2016 12:45	01/07/2016 13:00	214	225	233	240	252
6	01/07/2016 13:00	01/07/2016 13:15	198	207	216	224	234
7	01/07/2016 13:15	01/07/2016 13:30	163	173	179	189	199

Beginning and ending time of the 15 minutes time interval

0.05, 0.25, 0.50, 0.75 and 0.95 quantiles of the number of passengers arriving at immigration and security areas in windows of 15 minutes intervals

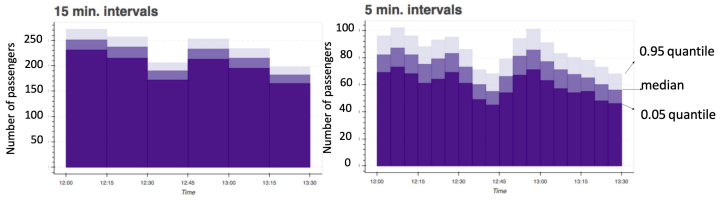


Figure 4.10: Output from the Application: Arrivals at Immigration and Security.

Heathrow's existing lane planning tool.

Our predictive system has been implemented at Heathrow since 2017, integrated into the APOC's Dynamic Model of Operations system. The probability of a passenger missing her connection appears in a "Connection at Risk" table, and the passenger flow at the Conformance desks is shown in a "Transfer Security Flow" table. The forecasts of the number of late passengers for each departing flight are provided to APOC's connections team who liaise with the airline.

The predictive system is based on the Azure Machine Learning platform. The regression tree is retrained daily with rolling five years of historical data. On the day of operation, the real-time data is read from Heathrow's SQL server into Azure platform in every 15 minutes. The forecasts for the next 90 minutes are then saved in CSV format to an Azure Blob storage, which is designed to store excessively large quantities of data files.

The Flow Managers in the APOC are currently using the predictive system to optimize the operation for smoother and more predictable service. As commented by Florian Guillermet (Guillermet 2017), the executive director of the Single European Sky Air Traffic Management Research (SESAR) project, "These predictions allow stakeholders – the airport and airlines – to take strategic decisions, such as holding an aircraft at the gate so that delayed passengers can board and the consequences that this may have on traffic." Passenger experience has been improved through reduced queuing, as capacity and resourcing along the journey more closely match with the dynamic demand. In addition, the managers are able to identify passengers who are at risk of missing their connection, and work with Heathrow and airline teams to assist and expedite their journeys.

An accuracy test of the expected passenger flows has been conducted over July and August 2017. The RMSE of the forecasts generated from the new predictive system is 25% lower in comparison to Heathrow's previous system. Encouraged by the good results, Heathrow's APOC has expanded the usage of the predictive system to enhance other airport services. Demand forecasts for bags and passengers with restricted mobility have now all been developed using predictive techniques. In

addition, real-time prediction of the direct departure flows has been developed to support the optimization of direct security operation. The team is now looking at how these techniques could be applied to other parts of the passenger journey such as surface access and trolley operations.

The encouraging results of our study have also changed the way that the major European airports think about data and machine learning techniques. Robert Graham, the head of airport research at Eurocontrol, stated that "This study has become a reference in SESAR Total Airport Management and is used by a number of major European airports. The study is: (i) groundbreaking, changing the way that the European Air Traffic Management players think about data, data science, and collaborating on sharing data; (ii) demonstrates how decision making can be better informed by the flow of data and the use of predictive algorithms, and (iii) brings state-of-the-art thinking in machine learning, applied to a problem of crucial importance to airports around the world into the airport operations domain" (Graham 2018).

4.6 A Numeric Evaluation of the Improved Predictions' Impact

To evaluate the benefits of having more accurate forecasts, we conduct a backtesting study over the 20% test set, examining the optimal staffing decision made when relying on forecasts from the various models. We first formulate the resourcing decision-making problem for the immigration desks.

For simplicity, we assume managers need only one period (or 15 minutes) to reallocate resource, and therefore, decisions on resourcing plans are made dynamically at the beginning of each time period for the next time period. For example, the staffing plan for interval $(t, t+1]$ is made at time $t-1$, based on the actual number of passengers who have not been served until $t-1$, and the predicted passenger arrivals in $(t-1, t]$ and $(t, t+1]$. Results are similar when we assume managers need more time to reallocate resource, and decisions need to be made at time $t-2$, $t-3$, etc.

Staffing the immigration desk carries a cost, and understaffing also carries a cost in terms of dissatisfied passengers and breach of targeted maximum queuing time goals. Therefore, the resourcing problem can be formulated as a newsvendor problem. A similar formulation was applied in Bassamboo et al. (2010) in the context of call centers. Suppose we are at time $t-1$, the optimization problem for the staffing plan in time interval $(t,t+1]$ is as follows: find the number of immigration desks to be opened, $b^*_{t+1} \geq 0$, that minimizes:

$$\Pi(b_{t+1}) = c_{IU}E(y_t + N_{t+1} - \mu b_{t+1})^+ + c_{IS}b_{t+1}, \tag{4.1}$$

where c_{IU} is the immigration understaffing cost per passenger per 15 minutes, y_t is the number of passengers in line in the beginning of $(t,t+1]$, N_{t+1} is the number of passengers connecting to domestic flights that arrive in $(t,t+1]$, μ is the service rate at each immigration desk, and c_{IS} is the cost of opening a desk per 15 minutes. We assume term y_t equals $\left(x_{t-1} + E(N_t) - \mu b^*_t\right)^+$, where x_{t-1} is the realized number of passengers in line in the beginning of $(t-1,t]$ under optimal decisions $b^*_1, b^*_2, ...b^*_{t-1}$, and $E(N_t)$ is the expected number of passengers who connect to domestic flights and arrive at the immigration area during $(t-1,t]$. The mean value of our prediction on the passenger flow described in Section 4.3.3 is used as $E(N_t)$.

The optimization problem in Eq. (4.1) is an instance of the familiar newsvendor problem: optimize the production quantity μb_{t+1} with a unit production cost of c_{IS}/μ and a unit understocking cost c_{IU}. Thus, we obtain the newsvendor-based staffing level at the immigration area as the standard critical quantile solution

$$b^*_{t+1} = \frac{1}{\mu}\left(y_t + F^{-1}_{N_{t+1}}(1 - \frac{c_{IS}}{\mu c_{IU}})\right), \tag{4.2}$$

where $F_{N_{t+1}}$ is the cdf of N_{t+1}, and therefore $F^{-1}_{N_{t+1}}(1 - c_{IS}/(\mu c_{IU}))$ is the $(1 - c_{IS}/(\mu c_{IU}))$ quantile forecast of N_{t+1} provided by the predictive system. The term $(1 - c_{IS}/(\mu c_{IU}))$ is often referred to as the critical fractile in a newsvendor setting. Note that passengers that are left over from interval t must be processed during interval $t+1$. Since b^*_{t+1} is directly linked to the remaining passengers from the

previous period (y_t), this solution guarantees in theory that passengers do not need to wait for more than two periods, i.e. 30 minutes. The decision variable b_{t+1} in the above problem is continuous; however, the number of lanes to be opened should be an integer. Therefore, we round b^*_{t+1} up and down, and retain the option that achieves lower expected cost.

Similarly, the optimization problem in time interval $(t,t+1]$ at the transfer security area is as follows: find the number of scurity lanes to be opened, $q^*_{t+1} \geq 0$, that minimizes:

$$\Pi(q_{t+1}) = c_{SU}E(z_t + \mu b^*_{t+1} + G_{t+1} - \eta q_{t+1})^+ + c_{SS}q_{t+1}, \tag{4.3}$$

where c_{SU} is the understaffing cost per passenger per 15 minutes in the security area, z_t is the number of passengers left over from previous time periods, μb^*_{t+1} denotes the number of passengers who join the queue after passing through immigration during $(t,t+1]$, η is the service rate of each security lane, and c_{SS} is the cost of a lane per 15 minutes. Variable G_{t+1} is the number of passengers connecting to international flights that arrive during $(t,t+1]$. These passengers do not need to go through immigration desks. We assume $z_t = (m_{t-1} + \mu b^*_t + E(G_t) - \eta q^*_t)^+$, where m_{t-1} is the realized number of passengers left over at time $t-1$ under optimal decisions $q^*_1, q^*_2, ...q^*_{t-1}$ and $b^*_1, b^*_2, ...b^*_{t-1}$, μb^*_t is the number of passengers joining the queue from immigration in $(t-1,t]$, and $E(G_t)$ is the expected number of passengers connecting to international flights and arriving at the security area during $(t-1,t]$. The term $E(G_t)$ is the mean value of our prediction on the flow of passengers connecting to international destinations. The solution for the problem in Eq. (4.3) is

$$q^*_{t+1} = \frac{1}{\eta}\big(z_t + \mu b^*_{t+1} + F^{-1}_{G_{t+1}}(1 - \frac{c_{SS}}{\eta c_{SU}})\big), \tag{4.4}$$

where $F_{G_{t+1}}$ is the cdf of G_{t+1}, and therefore $F^{-1}_{G_{t+1}}(1 - c_{SS}/(\eta c_{SU}))$ is the $(1 - c_{SS}/(\eta c_{SU}))$ quantile of the distribution for G_{t+1}. The decision variable q^*_{t+1} is rounded here as well.

For our empirical evaluation, service rates are assumed to be deterministic. We

set μ and η to 12 and 35 passengers per 15 minutes, respectively, based on the information provided by Heathrow. Since we do not want to make any judgement in advance about the staffing and understaffing cost, we allow flexibility and examine the performance under different scenarios. We set the critical fractile levels in (4.2) and (4.4) to 0.05, 0.25, 0.5, 0.75, and 0.95 to cover a wide range of scenarios corresponding to different ratios of the cost of staffing to the cost of understaffing. When the 0.05 fractile (or quantile) is used, the unit cost of staffing, c_{IS}/μ, is comparable to the unit cost of understaffing, c_{IU}; when the 0.95 fractile is used, the unit cost of staffing is much lower than the unit cost of understaffing. For each of the critical fractiles, we then add up the staffing cost and understaffing cost based on the realized passenger flows and the decisions made under the predictive models.

For the immigration staffing problem, across the five settings of $(1 - c_{IS}/(\mu c_{IU}))$ described above, our two-phased approach reduces cost by anywhere between 20% and 54% as shown in Table 4.5, when compared to Heathrow's static and dynamic legacy system. In comparison to the SARIMA model with covariates, the two-phased approach reduces cost by 11% to 29%. When testing on the staffing problem at the transfer security area, terms z_t and μb^*_{t+1} in the optimal solution q^*_{t+1} (Eq. 4.4) depend on the setting of $(1 - c_{IS}/(\mu c_{IU}))$. Therefore, we test the staffing plan at the security area for all 25 different combinations of $(1 - c_{IS}/(\mu c_{IU}))$ and $(1 - c_{SS}/(\eta c_{SU}))$. Percentage improvements in costs using our new predictive system, averaged over the five staffing and understaffing cost ratios at the immigration area, are also presented in Table 4.5. These results show that, compared to Heathrow's static and dynamic legacy systems, costs are reduced by 12% to 26% when applying our two-phased predictive system. In addition, the cost reduction offered by utilizing our prediction approach compared to the SARIMA model ranges from 10% to 16%.

Capacity, or the maximum number of immigration desks and security lanes that can be opened, is not constrained in the formulation above. In fact, the optimal solutions never exceed the actual capacity at Heathrow. However, for robustness purposes, we also run simulation with a capacity constraint equal to 80% of the

maximum number of immigration desks and security lanes to be opened in the study without constraints. Similar conclusions are drawn as in Table 4.5.

4.7 Generalized Application of the Two-Phased Predictive System

According to the airport layout and the description of the passenger transfer journey at Amsterdam Schiphol (Schiphol Airport 2019), Brussels (Brussels Airport 2019), Frankfurt (Frankfurt Airport 2019), and Charles de Gaulle (Paris Aéroports 2019), passengers connecting through European airports seem to have very similar transfer journeys as that described in Figure 4.2. Therefore, the approach developed in this chapter could easily be applied to other airports. Implementation of the approach would require collecting data, training, validating, and testing the model to account for the unique characteristics of each airport.

Most European international hubs are in Schengen countries. All passengers arriving from non-Schengen counties and connecting to Schengen destinations need to go through immigration desks. Instead of predicting two separate passenger flows for those connecting to domestic destinations and international destinations as at Heathrow, these airports could generate forecasts of passenger flows for Schengen destinations and non-Schengen destinations using the two-phased approach, whereby the Schengen region appears as a predictor in the model (similar to the origin or destination region in our current model). In addition, at some of the airports, such as Charles de Gaulle, passengers need to go through border force before taking shuttle service to other terminals. As a result, for these passengers, the time

	Improvements at immigration[a]					Improvements at security				
	1 - staffing/understaffing cost					1 - staffing/understaffing cost				
	0.05	0.25	0.50	0.75	0.95	0.05	0.25	0.50	0.75	0.95
SARIMA with covariates	11	15	20	23	29	10	11	12	14	16
Static legacy system	22	27	34	41	54	21	22	22	24	26
Dynamic legacy system	20	23	26	27	31	17	15	15	15	12

[a] The percentage improvement is calculated as $100 * (1 - \frac{\Pi^*_{two-phased}}{\Pi^*_{benchmark}})\%$.

Table 4.5: Percentage Improvements in Costs[a] at Immigration and Security.

between their departures from the immigration area to their arrivals at the security checking area may also need to be predicted by a separate model. Again, this could be captured by the first phase prediction in our system.

Transfer journeys at international hubs outside Europe are also similar to the journey depicted in Figure 4.2. For example, based on conversations with representatives from LAX (Los Angeles international airport), international arrivals (with the exception of Qantas passengers) go through immigration and customs before connecting to their domestic flights. These facilities only operate from a few terminals, implying that some passengers have to walk to an adjacent terminal after arrival (similar to the need at Heathrow to travel to Terminal 5). In some US airports, passengers arriving from international flights also need to pick up their bags from baggage claim and check in them again to their next domestic flight before going through the security check. Therefore, when modelling passengers' connection times in these airports, variables related to the baggage process, such as the number of checked-in baggage, should also be considered as predictors, and the entire transfer journey might need to be broken up into more segments to improve prediction accuracy. Moreover, change in airport layout and passengers' transfer journey would also affect the dependence between passengers' arrivals. The optimal value of ρ used to simulate arrivals in the second phase would need to be reestimated using airport specific data.

Our two-phased approach of predicting arrival times and the number of arrivals can also be generalized to other operations management domains, such as hospitals or theme parks. In a hospital setting, a similar framework can be applied to predict the number of patients in the Emergency Department (ED). Hoot et al. (2008) describes a simulation system for an ED that first fits parametric distributions to patient arrival rates, individual patients' treatment times, wait time for hospital bed, etc. When estimating treatment times, the authors fit separate log-normal distributions for patients who have the same acuity level. Finally, they run simulations based on all the distributions, and generate outputs such as the expected number of patients in the ED.

Our two-phased predictive system can similarly generate predictions of the number of patients in the ED. In the first phase, the system can predict the distribution of time spent in the ED for individual patients, based on their acuity level, their age, the number of patients with higher level of acuity in real-time, etc. Next in the second phase, from these individual distributions, the system can compute the probability of a patient that is still in the ED during different time intervals, and calculate the number of patients as the summation of several Bernoulli variables, with each of them denote whether or not a patient is still in the ED. In this context, distributions and quantiles for the number of patients in the ED can help with resource allocations, and help balance underage and overage costs related to the optimal number of beds (Zychlinski et al. 2019).

A theme park could also potentially utilize a similar predictive system to manage flows of visitors. These visitors' walking times from one attraction to another might be affected by many factors, such as previous location in the park and the number of restaurants along the way. Unlike passengers' connecting journey at airport, however, the next attraction to visit is decided by the visitor. Therefore, if we apply the two-phased approach to predict visitor flows at attraction a, the first-phase output may contain two sets of predictions: distribution of the walking time to attraction a, and the probability of attraction a is the next one to be visited. Based on these predictions, the probability of each visitor j arriving at an attraction during time interval i can be calculated. This probability is similar to the p_{ij} described in Section 4.3.3, which is the probability of a passenger j arriving at the immigration and security area during time interval i. Next, by aggregating individual arrivals, the theme park can predict visitor flows at an attraction. These predictions of visitor flows can be applied to set up ride capacities to optimize the number of rides (Ahmadi 1997) or retail profits (Rajaram and Ahmadi 2003). The newsvendor-based staffing decision described in Section 4.6 can also be applied here to set ride capacities.

Although our predictive system can be adapted to solve other operations problems, there are a few challenges when implementing a similar predictive system in

real-time. First, the quality of real-time data, such as how quickly the manager can receive the data and the format of the data, has a significant impact on the quality of the predictions. Second, individual level information and personal data are always difficult to collect and store. Third, the usefulness of the predictive system depends on the decision maker's flexibility of reacting to the real-time predictions. To implement a similar system in other operations domains, firms or organizations may need to build a centralized data system to collect and store data, create potential factors that may affect their customers' behavior using existing data, and train their managers to quickly adjust their decisions according to the real-time predictions.

4.8 Conclusions

In this chapter, we offer a first study of passengers' transfer journeys using data provided by Heathrow airport. We develop a two-phased predictive system to provide real-time information about transfer passengers' journeys through the airport. This information is vital for the airport in order to best serve the passengers, the airlines, and their employees. Compared to Heathrow's legacy systems and other benchmarks, our two-phased predictive system performs favorably in both identifying late passengers and predicting number of passenger arrivals at the immigration and security areas.

Our study makes advanced machine learning accessible to managers with no data science background working in data science at the Airport Operations Centre (APOC). It also enables the APOC to move from fragmented data and Excel based reporting to a more sophisticated data science environment. The capabilities demonstrated by this study inspired institutional investment in improving data science skills. Given the fact that APOCs are becoming standard in Europe (Eurocontrol 2010), the impact of this work extends beyond Heathrow, with interest expressed by Aéroports de Paris.

The work here demonstrates the usefulness of the regression tree method. Although simple in nature, it provides accurate predictions and easy interpretations. While other models served only as benchmarks here, we encourage continuous ex-

ploration of alternative models for improvement. Developing superior models in forecasting entire probability distributions would be an interesting future direction of research. A key part of our predictive system is the calibration of the distributions. We make assumptions that passenger arrivals are correlated and use copula-based simulation to make the forecasts well-calibrated. Developing other ways to calibrate the forecasts would be a potential topic for future work. In addition, in this chapter, we formulate a relatively simple staffing problem based on a newsvendor objective. For future work, we encourage researchers to formulate other decision making problems that make use of distributional forecasts.

4.9 Appendix

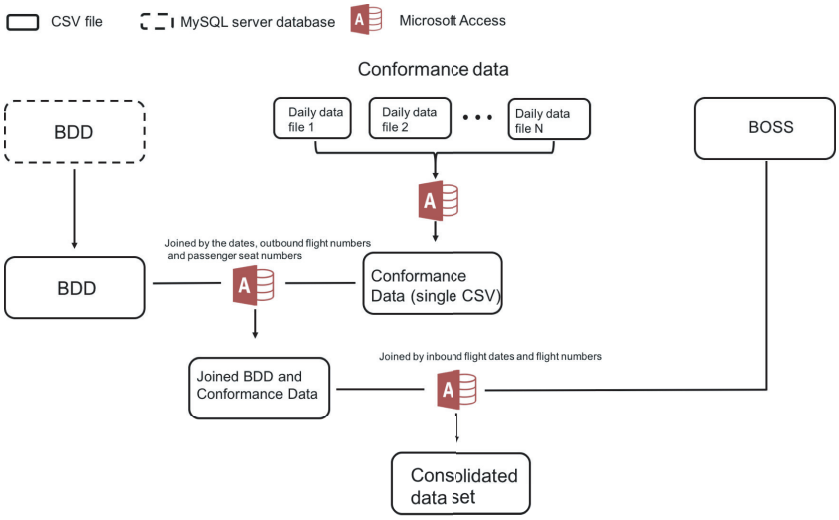


Figure 4.11: Consolidation of the Existing Databases Carrying Information about Transferring Passengers.

Data set	Variable name[a]	Description
BOSS	on chocks time[e]	The time when an aircraft is parked at gate.
	aircraft body	A flight's aircraft body type: W (wide) or N (narrow).
	aircraft type[c]	A flight's aircraft type. There are 23 types in total.
	passenger capacity	The capacity of the flight.
	passenger total	Total number of passengers on the flight.
	passenger transfer	Number of transfer passengers on the flight.
	runway no.[b]	Runway number of the flight. There are four runway numbers in the dataset: 27L, 27R, 09L, and 09R.
	scheduled time[c]	Scheduled arrival/departure time of the flight.
	stand no.[b]	Stand number of the flight; 213 numbers in total.
	inbound date[c]	The date of a flight arrives at the airport.
	flight no.[c]	There are 692 and 399 unique flight numbers for arriving and connecting flights, respectively.
	origin/destination airport[c]	There are 165 unique origin airports for international arrival flight with passengers connecting through T5, and and 143 unique destination airports of flights that have transfer passengers and depart from T5.
BDD	**passenger travel class**	Passengers' travel class on the arriving flight.There are five classes in the data set. We grouped them into two categories: economics or business and first class.
	inbound terminal	A passenger's arriving terminal.
	inbound stand type	Stand type of the arriving flight: P (Pier served stand) or R (Remote stand).
	outbound stand type	Stand type of the connecting flight.
	passenger outbound seat[d]	A passenger's seat number on the outbound flight.
Conformance	local conform time[e]	Time of when a passenger arrives at conformance desk.
	conform location code[d]	Code of the conformance desk.
	conform location descrp[d]	Terminal number, conformance desk number, and international or domestic connecting flight.
Created variables	**inbound region**	The region of the departure airport for the arriving flight. There are four regions: UK, Europe, North America, and the rest of world.
	outbound region	The region of the connecting flight's destination airport.
	inbound punct	Punctuality of the arriving flight.
	inbound hour	Hour of the day when the arriving flight lands.
	perceived delta	Time difference between the inbound flight's on-chock time and the outbound flight's scheduled departure time.
	Inbound load[d]	Load factor of the arriving flight. Defined as the ratio of the actual number of passengers to the capacity of the flight for inbound flight.
	outbound load[d]	The ratio of the actual number of passengers to the capacityof the flight for the outbound flight.
	day of the week[d]	Day of the week when the passenger arrives at airport.

[a] Variables in bold represent the 17 predictors used to train the model. Note eight predictors are from the BOSS data as each row in the table under BOSS are for both arriving and connecting flights.

[b, c, d, e] These variables were excluded because they were not available in real-time (b), had too many levels (c), did not help improve model accuracy (d), or were used only to calculate connection times (e).

Table 4.6: Descriptions of the Variables in the Data Set Excluded Variables Either Did Not Provide useful Information, Did Not Improve Accuracy of the Model, Was Not Available in Real Time or Dad Too Many Levels.

	mean	median	standard deviation
inbound flight passenger capacity	221	189	84
outbound flight passenger capacity	228	205	84
inbound passenger total	176	151	86
outbound passenger total	187	161	90
inbound passenger transfer	62	48	50
outbound passenger transfer	91	76	61
inbound punctuality	42	49	55
perceived delta	132	113	108
connection time	30.5	27	17.4

Table 4.7: Summary Statistics of the Numerical Predictors.

	Summary
inbound aircraft body	55% of the flights' body type was "Narrow", the others were "Wide"
outbound aircraft body	51% of the flights' body type was "Narrow", the others were "Wide"
inbound hour	The busiest hours in 2015 were 6:00 - 7:00 and 12:00 - 13:00, both with an average of 28 international flights landing at the airport.
passenger travel class	73% of the passengers traveled in economy class
inbound terminal	65% of the passengers arrived at T5, the others arrived at other terminals
inbound stand type	91% were pier served, the others were remote stand
outbound stand type	90% were pier served, the others were remote stand
inbound region	39%, 36%, 6%, and 19% of the passengers were from EU countries, North American, Asia, and rest of the world
outbound region	49%, 30%, 5%, and 16% of the passengers traveled to EU countries, North American, Asia, and rest of the world

Table 4.8: Summary Statistics of the Categorical Predictors.

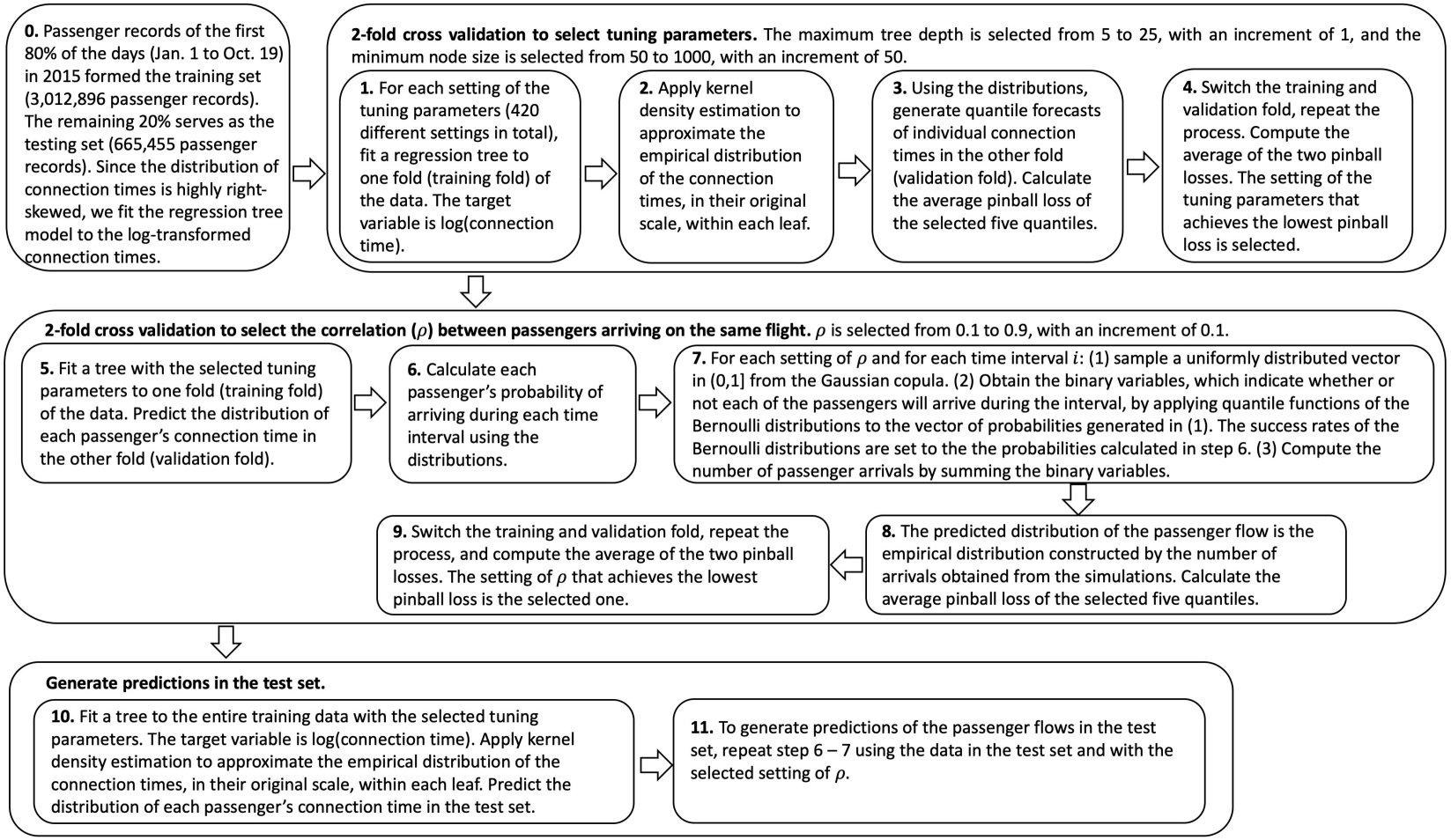


Figure 4.12: Steps Taken in Section 4.3.2 and 4.3.3 to Train the Two-phased Model and Generate Predictions from the Model.

Accuracy Results Based On the Continuous Ranked Probability Score (CRPS) and Logarithmic Score

Although we focus on quantiles when we evaluate the accuracy of distributional forecasts, here we also report the CRPS and logarithmic scores which evaluate the accuracy of probabilities predicted by the distributions. For those methods that only generate independent quantiles, we follow Lichtendahl et al. (2013) and approximate the entire distribution by fitting piecewise-linear cdfs to the five reported quantiles and the minimum and maximum values predicted by the models.

The CRPS of a realization, x, and a predicted cdf, F, is defined as $CRPS(F,x) =$

	CRPS	Log score[a]
Naïve model	7.30	3.80
Linear regression	6.29	3.62
LASSO quantile regression	6.21	3.68
Quantile regression forest	6.26	3.63
Gradient boosting machine	6.19	3.71
Regression tree	**6.10**	**3.58**
Significance[b]	***	***

[a] Excludes 1.7% passenger records that are outside of one of these models' predicted distributions' support.
[b] The symbol *** indicates the difference between the regression tree model and the second best model in each column is significant at the 1% level.

Table 4.9: CRPS and Log Scores of the Forecasts on Connection Times.

	Domestic		International	
	CRPS	Log score[a]	CRPS	Log score[a]
Naïve model	7.76	4.00	27.83	5.24
SARIMA with covariates	4.86	4.01	15.06	5.09
Static legacy system	7.72	5.72	29.65	5.68
Dynamic legacy system	5.82	4.02	23.70	5.27
Linear regression with copula	3.65	3.23	11.62	4.39
Regression tree without copula	3.79	4.10	12.77	6.38
Regression tree with copula	**3.48**	**3.23**	**11.36**	**4.37**
Significance[b]	**		*	

[a] Excludes 4.8% and 12.0% time intervals that are outside of one of these models' predicted distributions' support for the domestic and international flows, respectively.
[b] The symbol ** and * indicate the difference between the regression tree model with copula and the linear regression model with copula in each column is significant based on a t-test at the 5% and 10% level.

Table 4.10: CRPS and Log Scores of the Forecasts on Passenger Flows Connecting to Domestic and International Destinations at the Conformance Desk.

$\int_{-\infty}^{\infty}\big(F(y)-\mathbb{1}(y-x)\big)^2 dy$, where $\mathbb{1}$ is the indicator function that equals 1 if $y-x\geq 0$ and 0 otherwise. The logarithmic score is defined as $LogS(F,x) = -\log(f(x))$, where f is the probability density function of the predicted distribution. More details regarding these two accuracy measures can be found in Gneiting and Raftery (2007). As shown in Table 4.9 and 4.10, the regression tree model performs the best in predicting the distribution of connection times, and the two-phased approach using regression tree in the first phase outperforms the others in predicting passenger flows.

node number[a]	Average conecting time	Std. of the connecting times	Arriving terminal	Departing region of the arriving flight	Travel class of the arriving flight	Perceived connection time	Stand type of the arriving flight	Hour of the day the arriving flight lands at the airport	Total numbber of passengers on the arriving flight
14	13.1	8.8	Terminal 5	EU	Business/First		Pier served		
11	17.9	8.8	Terminal 5	EU	Economy	< 103			
15	18.8	9.5	Terminal 5	EU	Business/First		Remote		
12	23.0	15.8	Terminal 5	EU	Economy	≥ 103			
7	23.1	9.1	Terminal 5	Non-EU	Business/First	< 100			
8	25.6	13.1	Terminal 5	Non-EU	Business/First	≥ 100			
4	28.5	10.8	Terminal 5	Non-EU	Economy	< 131			
30	32.5	9.3	Terminal 2/3/4	EU	Business/First			after 13:00	
29	36.4	11.6	Terminal 2/3/4	EU	Business/First			before 13:00	
20	36.8	10.0	Terminal 2/3/4		Economy	< 148		after 15:00	
5	37.2	18.0	Terminal 5	Non-EU	Economy	≥ 131			
26	37.7	10.9	Terminal 2/3/4	Non-EU	Business/First	<133			
27	41.8	15.7	Terminal 2/3/4	Non-EU	Business/First	≥ 133			
19	43.6	12.5	Terminal 2/3/4		Economy	<148		before 15:00	
22	46.7	17.6	Terminal 2/3/4		Economy	≥ 148			< 183
23	52.8	18.6	Terminal 2/3/4		Economy	≥ 148			≥ 183

[a] The numbered nodes are the leaf nodes shown in Figure 5.

Table 4.11: Descriptions of the 16 Passenger Segments.

Chapter 5

Conclusions

Throughout the past decade, the explosion of big data and the rapid growth of computational power have created great opportunities for the development of predictive models, for both researchers and practitioners. Given its potential to play an important role in many business areas, predictive analysis and forecasting are becoming an integral part of most decision-making processes. Point forecasting using statistical and machine learning models has been widely studied in literature. Probabilistic forecasting which is often more important in decision making, however, has received relatively limited attention.

This thesis contributes to the advancement of probabilistic forecasting models in both methodological and practical aspects. In Chapter 2 and 3, I study the problem of forecasting a time series that evolves according to a dynamically changing, skewed life cycle. A tilted-Gompertz model and a new exponential smoothing model are developed to generate accurate distributional forecasts of product life cycles. The forecasts from these models can be made prior to launch, updated frequently thereafter, and generated at scale. In Chapter 4, I develop a predictive system to forecast transfer passenger flows in an airport, using passenger level and flight level data. The regression tree model, which is often applied to produce point estimates, is generalized to predict distributions. Although simple in nature, it provides accurate predictions and easy interpretations. In this chapter, I also show that the predictive system that aggregates the forecasts of individual passenger arrivals is more accurate than traditional time series models in forecasting passenger flows at the immigration and security areas.

There are various avenues for further research related to this thesis. First, Chapter 2 and 3 have developed two plausible life cycle models. Since product or innovation diffusions have been a popular research area since the 1960s, there

are many well-established models to consider when selecting a model to predict product or service life cycles. In this case, an ensemble model based on the established life cycle models should be developed to better fit the shape of the curve, and generate more accurate out-of-sample distributional forecasts. Second, Chapter 4 demonstrates the usefulness of the regression tree method in predicting distributions. Although some advanced machine learning methods can generate accurate point forecasts and even quantile forecasts, it is not easy to construct an entire distribution from these model. Therefore, developing superior models in forecasting entire probability distributions would be an interesting direction of future research. Finally, in Chapter 3 and 4, I formulate a few simple newsvendor-related problems to illustrate the impact of more accurate probabilistic forecasts on decision making. For future work, I encourage researchers and practitioners to combine predictive and prescriptive analysis in their studies, and formulate more advanced decision-making problems that make use of probabilistic forecasts.

Bibliography

Acimovic J, Erize F, Hu K, Thomas DJ, Van Mieghem JA. 2018. Product life cycle data set: Raw and cleaned data of weekly orders for personal computers. *Manufacturing & Service Operations Management* Forthcoming.

Ahmadi R. 1997. Managing capacity and flow at theme parks. *Operations research* **45**(1) 1–13.

Armstrong JS. 2001. Standards and practices for forecasting. *Principles of Forecasting*. Springer, 679–732.

Arnold BC, Groeneveld RA. 1995. Measuring skewness with respect to the mode. *The American Statistician* **49** 34–38.

Atkinson SE, Ramdas K, Williams JW. 2016. Robust scheduling practices in the us airline industry: Costs, returns, and inefficiencies. *Management Science* **62**(11) 3372–3391.

Baardman L, Perakis G, Levin I, Singhvi D. 2017. Leveraging comparables for new product sales forecasting Working Paper, available at `https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3086237`. Accessed: August 5, 2018.

Ban GY, Gallien J, Mersereau A. 2019. Dynamic procurement of new products with covariate information: The residual tree method. *Manufacturing & Service Operations Management* **21**(4) 798–815.

Barnhart C, Cohn A. 2004. Airline schedule planning: Accomplishments and opportunities. *Manufacturing & Service Operations Management* **6**(1) 3–22.

Barnhart C, Fearing D, Vaze V. 2014. Modeling passenger travel and delays in the national air transportation system. *Operations Research* **62**(3) 580–601.

Bass FM. 1969. A new product growth for model consumer durables. *Management Science* **15** 215–227.

Bassamboo A, Randhawa RS, Zeevi A. 2010. Capacity sizing under parameter uncertainty: Safety staffing principles revisited. *Management Science* **56**(10) 1668–1686.

Bassett Jr G, Koenker R. 1982. An empirical quantile function for linear models with iid errors. *Journal of the American Statistical Association* **77**(378) 407–415.

Bauckhage C, Kersting K, Rastegarpanah B. 2014. Collective attention to social media evolves according to diffusion models. *Proceedings of the 23rd international conference on World Wide Web*. 223–224.

Bemmaor AC. 1994. Modeling the diffusion of new durable goods: Word-of-mouth effect versus consumer heterogeneity. Pras B Laurent G, Lilien GL, ed., *Research Traditions in Marketing*. Kluwer, 201–229.

Bemmaor AC, Lee J. 2002. The impact of heterogeneity and ill-conditioning on diffusion model parameter estimates. *Marketing Science* **21** 209–220.

Bertsimas D, Patterson SS. 2000. The traffic flow management rerouting problem in air traffic control: A dynamic network flow approach. *Transportation Science* **34**(3) 239–255.

Bloomfield P. 2004. *Fourier Analysis of Time Series: an Introduction*. John Wiley & Sons.

Borndörfer R, Grötschel M, Pfetsch ME. 2007. A column-generation approach to line planning in public transport. *Transportation Science* **41**(1) 123–132.

Breiman L, Stone CJ, Friedman J, Olshen RA. 1984. *Classification and Regression Trees*. CRC press.

Brown RG. 1959. *Statistical Forecasting for Inventory Control*. McGraw-Hill.

Brown RG. 1963. *Smoothing, Forecasting and Prediction of Discrete Time Series*. Prentice Hall.

Brussels Airport. 2019. Connecting flights. `https://www.brusselsairport.be/en/passengers/your-travel-planner/connections`. Accessed: January 11, 2020.

Butler RW. 2007. *Saddlepoint approximations with applications*, vol. 22. Cambridge University Press.

Chernozhukov V, Fernández-Val I, Galichon A. 2010. Quantile and probability curves without crossing. *Econometrica* **78**(3) 1093–1125.

Chouakria AD, Nagabhushan PN. 2007. Adaptive dissimilarity index for measuring time series proximity. *Advances in Data Analysis and Classification* **1**(1) 5–21.

Corless, Gonnet GH Hare DEG Jeffrey DJ, RM, DE Knuth. 1996. On the LambertW function. *Advances in Computational mathematics* **5** 329–359.

De Reyck B, Guo X, Grushka-Cockayne Y, Lichtendahl Jr. KC, Karasev A, Garside T, Coss N, Tasker F. 2016. APOC business process reengineering big data study. Tech. rep.

De'ath G, Fabricius KE. 2000. Classification and regression trees: a powerful yet simple technique for ecological data analysis. *Ecology* **81**(11) 3178–3192.

Dixon R. 1980. Hybrid corn revisited. *Econometrica* **48** 1451–1461.

Domingos P. 2012. A few useful things to know about machine learning. *Communications of the ACM* **55**(10) 78–87.

Easingwood CJ, Muller E, Mahajan V. 1983. A nonuniform influence innovation diffusion model of new product acceptance. *Marketing Science* **2** 273–295.

Eliashberg J, Hui SK, Zhang ZJ. 2007. From story line to box office: A new approach for green-lighting movie scripts. *Management Science* **53**(6) 881–893.

Eurocontrol. 2010. The potential role of the Airport Operations Centre (APOC) in the SESAR airport concept. Tech. rep. Accessed: April 10, 2016.

Ferreira KJ, Lee BHA, Simchi-Levi D. 2015. Analytics for an online retailer: Demand forecasting and price optimization. *Manufacturing & Service Operations Management* **18**(1) 69–88.

Frankfurt Airport. 2019. Transferring at FRA. `https://www.frankfurt-airport.com/en/travel/transfer.detail.suffix.html/article/travel/services-a-z/easy-travel/transfer-at-fra.html`. Accessed: June 26, 2019.

Friedman J, Hastie T, Tibshirani R. 2010. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software* **33**(1) 1.

Gardner Jr. ES, McKenzie E. 1985. Forecasting trends in time series. *Management Science* **31** 1237–1246.

Geman S, Geman D. 1984. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **6**(6) 721–741.

Gneiting T. 2011. Quantiles as optimal point forecasts. *International Journal of Forecasting* **27** 197—-207.

Gneiting T, Raftery E. 2007. Strictly proper scoring rules, prediction, and estimation. *American Statistical Association* **102**(477) 359–378.

Goeman JJ. 2010. L1 penalized estimation in the cox proportional hazards model. *Biometrical Journal* **52**(1) 70–84.

Goeman JJ, Meijer R, Chaturvedi N. 2018. L1 and l2 penalized regression models. `https://cran.r-project.org/web/packages/penalized/vignettes/penalized.pdf`. Accessed: March 3, 2019.

Golder PN, Tellis GJ. 1997. Will it ever fly? Modeling the takeoff of really new consumer durables. *Marketing Science* **16** 256–270.

Goldman A. 1982. Short product life cycles: implications for the marketing activities of small high-technology companies. *R&D Management* **12** 81–90.

Gompertz B. 1825. On the nature of the function expressive of the law of human mortality, and on a new mode of determining the value of life contingencies. *Philosophical Transactions of the Royal Society of London* **115** 513–583.

Google Trends. 2016. `https://www.google.com/trends/`. Accessed: June 29-30, 2016.

Graham R. 2018. Letter of endorsement, August 24, 2018.

Green KC, Armstrong JS. 2015. Simple versus complex forecasting: The evidence. *Journal of Business Research* **68**(8) 1678–1685.

Grushka-Cockayne Y, Jose VRR, Lichtendahl Jr. KC, Winkler RL. 2017. Quantile evaluation, sensitivity to bracketing, and sharing business payoffs. *Operations Research* **65**(3) 712–728.

Guillermet F. 2017. Towards total airport management. `http://www.airport-business.com/2017/06/towards-total-airport-management/`. Accessed: June 22, 2017.

Guo X, Grushka-Cockayne Y, De Reyck B. 2020. London heathrow airport uses real-time analytics for improving operations. *INFORMS Journal on Applied Analytics* **50**(5) 325–339.

Hastie T, Tibshirani R, Friedman JH. 2009. *The elements of statistical learning: data mining, inference, and prediction*. New York, NY: Springer.

Heathrow Airport. 2020. Heathrow airport facts and figures. `https://www.heathrow.com/company/about-heathrow/company-information/facts-and-figures`. Accessed: June 22, 2020.

Hoerl AE, Kennard RW. 1970. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* **12**(1) 55–67.

Holt CC. 1957. Forecasting trends and seasonal by exponentially weighted averages. ONR Memorandum 52, Carnegie Institute of Technology.

Holt CC. 2004. Forecasting seasonals and trends by exponentially weighted moving averages. *International Journal of Forecasting* **20** 5–10.

Hong T, Fan S, Pinson P, Zareipour H, Troccoli A, Hyndman RJ. 2016. Probabilistic energy forecasting: Global energy forecasting competition 2014 and beyond. *International Journal of Forecasting* **32**(3) 896–913.

Hoot NR, Jones I Levin SR Zhou C Gadd CS, LeBlanc LJ, Aronsky D. 2008. Forecasting emergency department crowding: a discrete event simulation. *Annals of emergency medicine* **52**(2) 116–125.

Hu K, Acimovic J, Erize F, Thomas DJ, Van Mieghem JA. 2017. Forecasting product life cycle curves: Practical approach and empirical analysis. *Manufacturing & Service Operations Management* Forthcoming.

Hyndman, RJ. 2016. forecast: Forecasting functions for time series and linear models. `http://github.com/robjhyndman/forecast`. Accessed: July 10, 2016.

Hyndman RJ. 2015. R vs Autobox vs ForecastPro vs ... `http://robjhyndman.com/hyndsight/show-me-the-evidence/`. Accessed: June 15, 2016.

Hyndman RJ, Athanasopoulos G. 2018. *Forecasting: Principles and Practice*. OTexts.

Hyndman RJ, Billah B. 2003. Unmasking the theta method. *International Journal of Forecasting* **19**(2) 287–290.

Hyndman RJ, Ord JK, Koehler AB, Snyder RD. 2008. *Forecasting with Exponential Smoothing*. Springer-Verlag, Berlin, Germany.

Jacquillat A, Odoni AR. 2015. An integrated scheduling and operations approach to airport congestion mitigation. *Operations Research* **63**(6) 1390–1410.

James G, Hastie T, Witten D, Tibshirani R. 2013. *An Introduction to Statistical Learning*. New York, NY: Springer.

Johnson NL, Balakrishnan N, Kotz S. 1995. *Continuous Univariate Distributions, Volume 2*. John Wiley & Sons, New York, NY.

Jose VRR. 2016. Percentage and relative error measures in forecast evaluation. *Operations Research* **65**(1) 200–211.

Jose VRR, Winkler RL. 2009. Evaluating quantile assessments. *Operations Research* **57** 1287–1297.

Koenker R, Bassett Jr. G. 1978. Regression quantiles. *Econometrica: journal of the Econometric Society* 33–50.

Kolsarici C, Vakratsas D. 2015. Correcting for misspecification in parameter dynamics to improve forecast accuracy with adaptively estimated models. *Management Science* **61**(10) 2495–2513.

Lan S, Clarke JP, Barnhart C. 2006. Planning for robust airline operations: Optimizing aircraft routings and flight departure times to minimize passenger disruptions. *Transportation Science* **40**(1) 15–28.

Lichtendahl Jr. KC, Grushka-Cockayne Y, Winkler RL. 2013. Is it better to average probabilities or quantiles? *Management Science* **59**(7) 1594–1611.

Lohatepanont M, Barnhart C. 2004. Airline schedule planning: Integrated models and algorithms for schedule design and fleet assignment. *Transportation Science* **38**(1) 19–32.

MacDonald M. 2016. Annual analyses of the EU air transport market. Tech. rep. Accessed: October 10, 2017.

Mahajan V, Bass FM, Muller E. 1990. New product diffusion models in marketing: A review and directions for research. *Journal of Marketing* **54** 1–26.

Mahajan V, Srinivasan V, Mason CH. 1986. An evaluation of estimation procedures for new product diffusion models. Wind J Mahajan V, ed., *Innovation Diffusion Models of New Product Acceptance*. Ballinger Publishing Co., Cambridge, MA, 203–232.

Makridakis S, Assimakopoulos V, Spiliotis E. 2018. The M4 competition: Results, findings, conclusion and way forward. *International Journal of Forecasting* **34** 802–808.

Marshall AW, Olkin I. 2007. *Life Distributions*. Springer, New York, NY.

Meade N, Islam T. 2006. Modelling and forecasting the diffusion of innovation–a 25-year review. *International Journal of Forecasting* **22** 519–545.

Meinshausen N. 2006. Quantile regression forests. *Journal of Machine Learning Research* **7**(Jun) 983–999.

Milenković M, Melichar V Bojović N, Švadlenka L, Avramović Z. 2018. Sarima modelling approach for railway passenger flow forecasting. *Transport* **33**(5) 1113–1120.

Mukherjee A, Hansen M. 2009. A dynamic rerouting model for air traffic flow management. *Transportation Research Part B: Methodological* **43**(1) 159–171.

Oreshkin BN, Chapados N, Carpov D, Bengio Y. 2019. N-BEATS: Neural basis expansion analysis for interpretable time series forecasting. *arXiv preprint arXiv:1905.10437* .

Paris Aéroports. 2019. Connecting flights. `https://www.parisaeroport.fr/en/passengers/flights/connecting-flights`. Accessed: June 26, 2017.

Pearl R, Reed LJ. 1925. Skew-growth curves. *Proceedings of the National Academy of Sciences* **11** 16–22.

Pegels CC. 1969. Exponential forecasting: Some new variations. *Management Science* **12** 311–315.

Petruzzi NC, Dada M. 1999. Pricing and the newsvendor problem: A review with extensions. *Operations Research* **47** 183–194.

Rajaram K, Ahmadi R. 2003. Flow management to optimize retail profits at theme parks. *Operations Research* **51**(2) 175–184.

Rogers EM. 2010. *Diffusion of innovations*. Simon and Schuster.

Schema. 2019. The lifespan of news stories. `https://www.newslifespan.com`. Accessed: February 4, 2019.

Schiphol Airport. 2019. Schiphol airport map. `https://www.schiphol.nl/en/airport-maps`. Accessed: June 26, 2017.

Schmittlein DC, Mahajan V. 1982. Maximum likelihood estimation for an innovation diffusion model of new product acceptance. *Marketing Science* **1** 57–78.

Shang H. 2017. Functional time series forecasting with dynamic updating: An application to intraday particulate matter concentration. *Econometrics and statistics* **1** 184–200.

Shang Y, Dunson D, Song JS. 2017. Exploiting big data in logistics risk assessment via bayesian nonparametrics. *Operations Research* **65**(6) 1574–1588.

Siegmund D. 1976. Importance sampling in the monte carlo study of sequential tests. *The Annals of Statistics* 673–684.

Smyl S. 2020. A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting. *International Journal of Forecasting* **36**(1) 75–85.

Solak S, Clarke JB, Johnson EL. 2009. Airport terminal capacity planning. *Transportation Research Part B: Methodological* **43**(6) 659–676.

Srinivasan V, Mason CH. 1986. Nonlinear least squares estimation of new product diffusion models. *Marketing Science* **5** 169–178.

Tassone E, Rohani F. 2017. Our quest for robust time series forecasting at scale. `http://www.unofficialgoogledatascience.com/2017/04/our-quest-for-robust-time-series.html`. Accessed: October 10, 2017.

Taylor JW. 2003. Exponential smoothing with a damped multiplicative trend. *International Journal of Forecasting* **19** 715–725.

Taylor JW. 2008. A comparison of univariate time series methods for forecasting intraday arrivals at a call center. *Management Science* **54**(2) 253–265.

Taylor JW. 2012. Density forecasting of intraday call center arrivals using models based on exponential smoothing. *Management Science* **58**(3) 534–549.

Tibshirani R. 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)* **58**(1) 267–288.

Van Brussel B. 2018. *Simulating the patient flow on the short stay unit*. Master thesis, University of Amsterdam.

Van den Bulte C, Joshi YV. 2007. New product diffusion with influentials and imitators. *Marketing Science* **26** 400–421.

Van den Bulte C, Lilien GL. 1997. Bias and systematic change in the parameter estimates of macro-level diffusion models. *Marketing Science* **16** 338–353.

Venables WN, Ripley BD. 2002. *Modern Applied Statistics with S*. Springer-Verlag.

Venkatesan R, Kumar V, Krishnan TV. 2004. Evolutionary estimation of macro-level diffusion models using genetic algorithms: An alternative to nonlinear least squares. *Marketing Science* **23** 451–464.

Wand MP, Jones MC. 1994. *Kernel Smoothing*. Chapman and Hall/CRC.

Wei W, Hansen M. 2006. An aggregate demand model for air passenger traffic in the hub-and-spoke network. *Transportation Research Part A: Policy and Practice* **40**(10) 841–851.

Wei Y, Chen MC. 2012. Forecasting the short-term metro passenger flow with empirical mode decomposition and neural networks. *Transportation Research Part C: Emerging Technologies* **21**(1) 148–162.

Wikipedia. 2016. List of social networking websites. `https://en.wikipedia.org/w/index.php?title=List_of_social_networking_websites&oldid=719695760`. Accessed: June 10, 2016.

Winsor CP. 1932. The Gompertz curve as a growth curve. *Proceedings of the National Academy of Sciences* **18** 1–8.

Winters PR. 1960. Forecasting sales by exponentially weighted moving averages. *Management Science* **6** 324–342.

Wu PP, Mengersen K. 2013. A review of models and model usage scenarios for an airport complex system. *Transportation Research Part A: Policy and Practice* **47** 124–140.

Xie J, Sirbu M, Song XM, Wang Q. 1997. Kalman filter estimation of new product diffusion models. *Journal of Marketing Research* **34** 378–393.

Xue Z, Wang Z, Ettl M. 2015. Pricing personalized bundles: A new approach and an empirical study. *Manufacturing & Service Operations Management* **18**(1) 51–68.

Zaveri M. 2019. Google disruptions affect Gmail, YouTube and other sites. `https://www.nytimes.com/2019/06/02/technology/google-gmail-snapchat-outage.html?smid=nytcore-ios-share`. Accessed: June 28, 2019.

Zychlinski N, Mandelbaum A, Momčilović P, Cohen I. 2019. Bed blocking in hospitals owing to scarce capacity in geriatric institutions—cost minimization via fluid models. *Manufacturing & Service Operations Management* .