Revenue Optimization for a Hotel Property with Different Market Segments: Demand Prediction, Price Selection and Capacity Allocation

by

Eduardo Candela Garza

B.Sc. Industrial Engineering, Instituto Tecnológico Autónomo de México, 2016
B.Sc. Mechatronics Engineering, Instituto Tecnológico Autónomo de México, 2015

Submitted to the Sloan School of Management
in partial fulfillment of the requirements for the degree of

Master of Science in Operations Research

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2017

Author . . . .
Sloan School of Management
August 11, 2017

Certified by . . . .
David Simchi-Levi
Professor of Engineering Systems, Institute for Data, Systems, and Society
Thesis Supervisor

Accepted by . . . .
Dimitris Bertsimas
Boeing Professor of Operations Research
Co-Director, Operations Research Center

Revenue Optimization for a Hotel Property with Different Market Segments: Demand Prediction, Price Selection and Capacity Allocation

by

Eduardo Candela Garza

Submitted to the Sloan School of Management
on August 11, 2017, in partial fulfillment of the
requirements for the degree of
Master of Science in Operations Research

Abstract

We present our work with a hotel company as an example of how machine learning techniques can be used to improve the demand predictions of a hotel property, as well as its pricing and capacity allocation decisions.

First, we build a price-sensitive random forest model to predict the number of daily bookings for each customer market segment. We feed these predictions into a mixed integer linear program (MILP) to optimize prices and capacity allocations at the same time. We prove that the MILP can be equivalently solved as a linear program, and then show that it produces upper and lower bounds for the expected revenue maximization Dynamic Program (DP), and that the gap between the bounds depends on the probabilistic distribution of the demand. Thus, for high prediction accuracies, the optimal value of the DP can be closely approximated by the MILP solution.

Finally, numerical results show that the optimized decisions are able to generate an increase in revenue compared to the historical policies, and that the fast running time achieved permits real time policy updates.

Thesis Supervisor: David Simchi-Levi
Title: Professor of Engineering Systems, Institute for Data, Systems, and Society

Acknowledgments

Thank you, MIT. Such a magnificent and incredible community that brings the most brilliant people from all around the planet together, and allows them to learn and grow together. But what drives this evolution is the common mission of making the world a better a better place through engineering and science. I will always be thankful for having the opportunity of being an MIT engineer, as I can certainly say my life changed after this. Now, I will work very hard to make a meaningful impact.

I would like to thank my life-long friends for being my family, as well as my new friends that became a part of it: for supporting me unconditionally, giving me strength when I needed it, and celebrating with me.

My gratitude to Rui Sun, not only for collaborating with me on this project, but for his friendship. It was a pleasure working and traveling with you. I would also like to thank our industry collaborators for their time, help and support throughout the project.

Finally and most importantly, thank you, Lore, for being my biggest source of motivation. You are always there for me. You make the good times simply the best, and you do whatever it takes to solve the bad ones. Thank you for helping me with this project, for coming from the other side of the country when I needed you, for being my best teammate, and for making me believe in destiny.

Contents

List of Figures

List of Tables

We present our work with a hotel company as an example of how machine learning techniques can be used to improve the demand predictions of a hotel property, as well as its pricing and capacity allocation decisions.

First, we build a price-sensitive random forest model to predict the number of daily bookings for each customer market segment. We feed these predictions into a mixed integer linear program (MILP) to optimize prices and capacity allocations at the same time. We prove that the MILP can be equivalently solved as a linear program, and then show that it produces upper and lower bounds for the expected revenue maximization Dynamic Program (DP), and that the gap between the bounds depends on the probabilistic distribution of the demand. Thus, for high prediction accuracies, the optimal value of the DP can be closely approximated by the MILP solution.

Finally, numerical results show that the optimized decisions are able to generate an increase in revenue compared to the historical policies, and that the fast running time achieved permits real time policy updates.

Chapter 1

Introduction

In this chapter, the current setting of the global hotel industry is presented, as well as the motivation that led to the creation of this research thesis. At the end, a brief description of the structure of each of the chapters of this document is displayed.

1.1 Motivation

Nowadays, hotel companies face a great challenge: surviving. The hotel industry is such a competitive and price-sensitive market, with most customers using the Internet and countless browsers to find the best prices, deals and alternatives with just a couple of clicks. That is why big hotel holdings keep buying their struggling competitors, as the world witnessed when Marriott acquired Starwood Hotels in September of 2016 [20], making them the largest hotel group in the planet.

The global hotel industry is enormous, and continuously augmenting. Its total global revenue in 2016 was $490.06 billion USD, and the company that contributed the most to this was InterContinental Hotels Group with over 4,800 hotels worldwide. Furthermore, global revenue is expected to grow to $553.8 billion in 2018 (see Figure 1-1) [25].

However, an important factor to consider when analyzing this industry is the re-

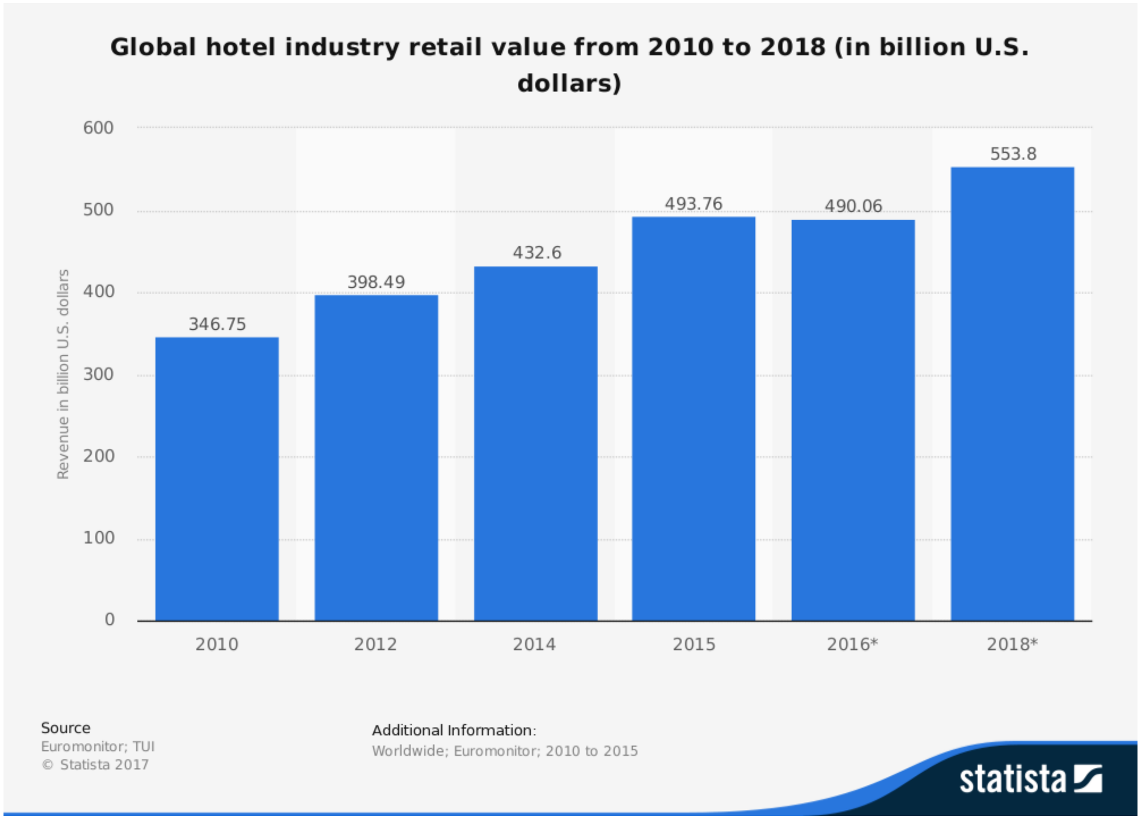

Figure 1-1: Global hotel industry revenue in 2010-2018. [25]

cent appearance and impressive growth of shared-economy lodging alternatives, with Airbnb as the main exponent. Its market valuation in 2015 was larger than Marriott's; Airbnb also had an industry leading 90% annual growth rate, as shown in Figure1-2. The success Airbnb is experiencing in greatly due to the advanced analytics methods applied to their gigantic data sets, allowing them to know demand patterns better. This posses considerable pressure on traditional hotel companies to develop and incorporate state-of-the-art analytics, in order to stay competitive.

The main objective of this thesis is to develop and implement a method that using real data, constitutes an advanced and reliable way to improve revenue for hotel companies, and that it is also scalable to use in other Revenue Management applications. In order to meet this challenge, close-to-optimal decisions are suggested by taking a two-step-approach: 1) demand forecasting using machine learning techniques; 2) price selection and capacity allocation by solving a Mixed Integer Linear Program

COMPANY	VALUATION ($B)	2015 REVENUE* ($B)	VALUATION TO 2015 REVENUE	ANNUAL GROWTH RATE**
Priceline	$59.5	$9.3	6.4	17%
HomeAway	$3	$0.5	6	20%
Hilton	$27.6	$11.5	2.4	9%
Airbnb	$25.5	$0.9	28.3	90%
Marriott	$20.9	$14.8	1.4	8%
Starwood	$14	$5.9	2.4	-2%
Expedia	$13.8	$6.5	2.1	17%

SHOWING 1 TO 7 OF 7 ROWS

*Expected **Compounded 2013 to 2015
FactSet, The Wall Street Journal

Figure 1-2: Economic comparison of diverse lodging companies. [24]

(MILP).

We attained data from a main hotel company, and were asked by it to develop a solution using advanced analytics that would meet their needs, respecting their business constraints. The enterprise is a holding of various hotel brands, with hundreds of properties around the world. They have categorized their customers in 26 different market segments (see Figure 1-3), and depending on the market segment, clients can book rooms through separate channels, allowing the company to offer different prices and booking policies for the same physical rooms. Every hour, the company suggests prices to every property, for each of the 26 different market segments, as well as how many rooms – and of which type – to allocate to them. This increases significantly the decision space, making the pricing and allocation decisions more complex. Due to the need to calculate and update their prices hourly for each property, and for hundreds of future stay days, it is crucial to achieve optimal running time on any solution implementation.

The main contributions of this thesis, which was done in collaboration with Rui Sun [22], are: 1) presenting a machine learning technique to accurately forecast demand for various days incorporating different market segments, and price sensitivity; 2) developing a Mixed Integer Linear Program that closely approximates the solution

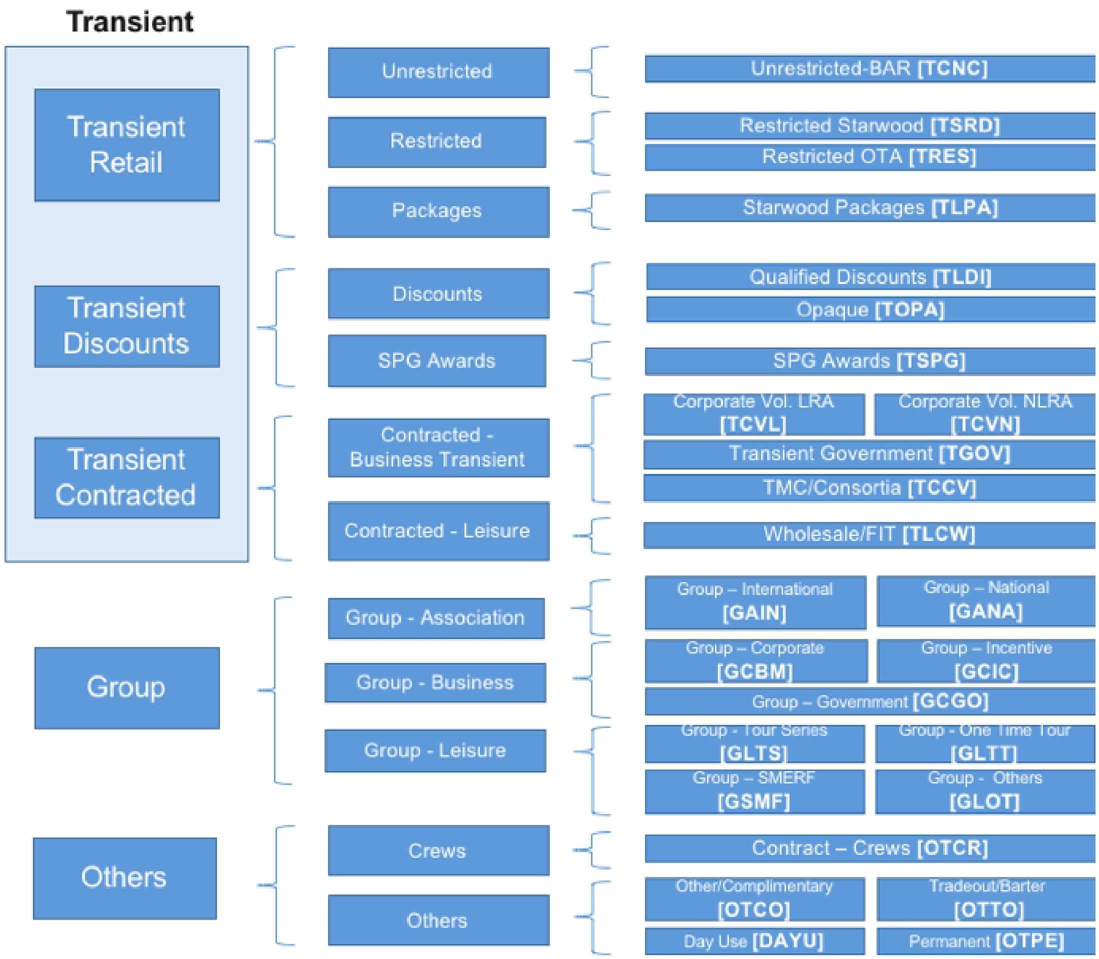

Figure 1-3: The 26 market segments for the hotel company data set.

that yields the optimal revenue for a single stay day and hotel property (which is intractable to exactly solve), and proving that the approximation is close for good forecast accuracies; 3) both the demand forecast and the optimization steps are solved in a rapid fashion, allowing a real-time implementation of the whole method; 4) a generalizable method with a product upgrade alternative following a hierarchy, that can be used not only for hotels, but for basically any problem involving uncertain demand and optimization decisions based on it.

1.2 Thesis Structure

The thesis is organized as follows:

- Chapter 1: The setting of the challenge that led to the development of this thesis is presented, as well as a brief description of the structure of each of its chapters.

- Chapter 2: There is a general Revenue Management overview, followed by a literature review of relevant papers in the field that were consulted, in order to know the current state of published knowledge.

- Chapter 3: The used data is described; based on it, the mathematical model and assumptions of the developed solution are introduced.

- Chapter 4: The random forest machine learning technique utilized for forecasting demand in shown in detail, and numerical results of its implementation are discussed.

- Chapter 5: The optimization model developed is explained, and some of its theoretical properties and theorems that permit to approximate very closely the solution of the intractable problem using a Mixed Integer Linear Program (MILP).

- Chapter 6: The conclusions of this project are presented, as well as suggestions for future research that would continue and expand this thesis.

Chapter 2

Literature Review

The purpose of this chapter is to have a brief introduction to the topic of Revenue Management, followed by a literature review of relevant papers in the field and related to this thesis, so that the reader can have a good sense of the state of the art for Hotel Revenue Management.

2.1 Revenue Management Overview

Revenue Management is the application of disciplined analytics to demand-management decisions, and the methodology and systems required to make them in order to maximize revenue. Its starting point can be traced back to the Airline Deregulation Act of 1978, causing the U.S. Civil Aviation Board to loose control of previously strictly regulated airline prices. This led to the launch of a new pricing scheme by American Airlines, called *American Super-Saver Fares.* Robert Crandall, American Airlines' vice president of marketing, was in charge of this. Seven years later, in 1985, a more sophisticated capacity-controlled approach was born: *DINAMO* (Dynamic Inventory Allocation and Maintenance Optimizer), the first large-scale Revenue Management system in the industry. [23]

Revenue Management can be thought of as the complement of Supply-Chain Management, which is related to cost minimization when improving supply, production and delivery processes. RM addresses three basic categories of decisions [23] :

1. Structural decisions: Selling formats, segmentation, terms of trade, bundling, etc.

2. Price decisions: Posted prices, reserve prices, pricing over time, discounts, etc.

3. Quantity decisions: Accepting or rejecting an offer, capacity and product allocation to segments and channels, etc.

2.2 Demand Forecasting

There is significant research available involving demand forecasting, where a common assumption is that the underlying distributions of demand are known a priori, which might not be the case for practical problems. See den Boer [10] for a recent survey on demand learning and dynamic pricing.

Considerable research has also been produced specifically in the Hotel Revenue Management field, where the objective of forecasting in the hotel industry is to get an estimate of the future demand for rooms, based on historical data on bookings. Forecasting applied to this problem should take two time-related variables: the time of booking (or reservation date) and the time to consumption [7]. The room demand forecasting phase, considered as a pre-processing step in Revenue Management, should be studied in detail since it determines the reliability of the optimization phase.

In addition to the method selection, other considerations include the type of forecast (room nights or arrivals), level of aggregation (total, by rate category, by length of stay, or some combination), the type of data (constrained or unconstrained), the number of periods to include in the forecast, which data to use, the amount of data, the treatment of outliers (due to holidays or special events), and the measurement of accuracy.

1. Historical	A. Same day, last year
	B. Moving average
	C. Exponential smoothing
	D. Other time series (ARIMA, etc.)
2. Advanced Booking	A. Additive
	i. Classical pickup
	ii. Advanced pickup
	B. Multiplicative
	i. Synthetic booking curve
	C. Other time series
3. Combined	A. Weighted average of historical and advanced booking forecasts
	B. Regression
	C. Full information model

Table 2.1: Revenue management forecasting methods. [18]

An example in variety of forecasting methods and in determining the most accurate is Weatherford and Kimes' paper [26], in which they use data from Choice Hotels and Marriott Hotels. They cite Lee [18], who classified revenue management methods into three main types: historical booking models, advanced booking models and combined models (Table 2.1).

Unconstrained demand is hard to observe since hotels usually impose certain constraints such as rate controls, stay pattern controls and capacity limitations. Liu *et al.* [19] develop parametric regression models that consider the demand distribution and the conditions under which data were collected, in order to estimate the unconstrained hotel demand based on the censored booking data.

2.3 Revenue optimization

There is also abundant literature devoted to solving the optimization problem involving the pricing or capacity allocation for the booking problem. In the hotel Revenue Management case, the problem of managing room capacity on consecutive days when customers stay multiple nights, and where a mix of customers with different lengths of stay share the capacity on any given day, can be formulated as a network problem. However, Talluri and Ryzin [23] show that for any network of realistic size, solving the dynamic program formulation is computationally intractable due to the large dimensionality of the state space, and instead propose approximations via simplified network models by posing the problem as a static mathematical programming problem. Additionally, Badinelli [1] presents a dynamical model for finding optimal booking policies for the hotel yield management problem. Although only one room type is assumed and the policy is based on a single, critical booking date, the model represents a framework for extensions like the introduction of price constraints or multiple room types. Chen and Kachani [7] study the room allocation optimization process. Since deterministic and stochastic models are too large to be solved efficiently for the hotel industry, they formulate the problem as a network flow model.

Another approach to solving revenue management optimization problems is to use approximate dynamic programming to solve a Markov decision model. Markov models are useful and easy to understand and develop, and can provide closed-loop optimal control strategies. However, as the size of the problem increases, they become affected by dimensionality. A way to address this problem is approximating state-cost-values with parametric functions [21]. The reader can find more information on Approximate Dynamic Programming (ADP) theory in Dimitri Bertseka's books: [2] and [3]. For literature on ADP applications in revenue management, Farias and Van Roy [12], and Zhang and Adelman [28] [29] can be consulted.

Outside of the hotel industry, Johnson, Lee and Simchi-Levi [17] used historical data to optimize pricing decisions for an online fashion retailer by using regression trees to forecast demand. Then they formulated a price optimization model to maximize revenue from first exposure styles, using demand predictions from the regression trees as inputs. They developed an algorithm that efficiently solves the multi-product price optimization and addresses the issues posed by the non-parametric structure of regression trees. Their method is taken as a guide for the one presented in this paper. However, due to the long time it takes them to solve the Integer Program they present, they instead work with a linear relaxation, and attain bounds for it. When comparing bounds, they only exactly solve a subset of all the optimization programs. On the other hand, this paper proposes an alternative to solve in a faster fashion the Mixed Integer Linear Program.

Chapter 3

Data and Model

This chapter first presents in detail the data provided by the hotel company, which was used to develop and implement the solution produced in this thesis. Then, there is a thorough description of the mathematical model and its business constraints, which is used as framework for both the demand prediction and decision optimization steps shown in chapters 4 and 5, respectively.

3.1 Data

The data set provided by the hotel company and utilized in this thesis consists of the daily aggregate booking history of one of the most important hotels in Times Square, in New York City. The data is from a 4-year time period, ranging from 01/01/2012 to 12/31/2015. The company categorized the customers in the 26 different market segments seen in Figure 1-3, and the data is aggregated daily for each separate market segment. The data is also divided by 5 room classes, which follow a vertical hierarchy, i.e. each room type has at most one type with a higher hierarchy, and at most one type with a lower hierarchy; a room type can be upgraded to any of its superior types. Another dimension of the data is the number of days the booking was made prior to the stay date, which is called *booking window*. The standard way of visualizing the booking information through time is using a *booking curve*, like the one shown as example in Figure 3-1. The vertical axis is the percentage of sold rooms, called

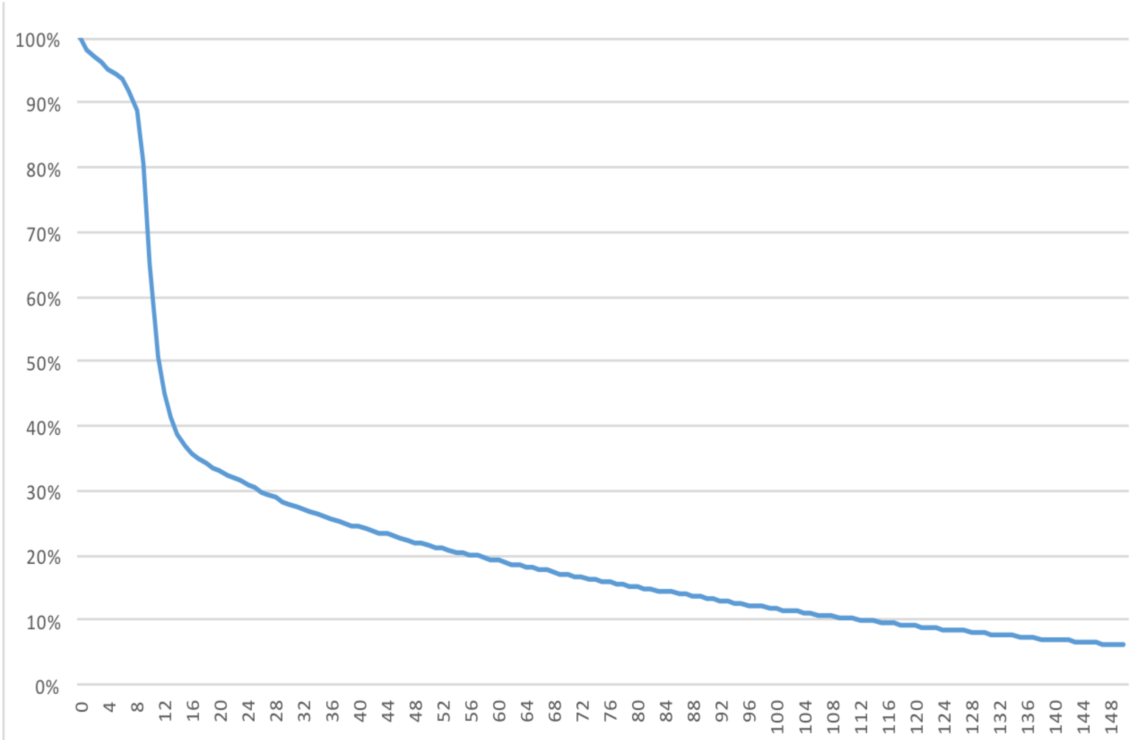

Figure 3-1: Booking window example.

occupancy; the horizontal axis is the booking window. It is possible to see that usually as the stay day is closer, the booking window is therefore smaller, and occupancy increases. Figure 3-2 shows booking curve plots for different market segments.

The data is grouped by stay day, market segment, room type, and booking window; for each group there is a column with the number of bookings, cancellations (both are often 0), charged price, and Average Daily Rate (ADR) – an internal base rate the company generates from their own demand forecast system, without taking market segments into account. It is important to mention that a stay day is different to an arrival day, because customers arrive on the first stay day, but can (and often do) stay more consecutive stay days.

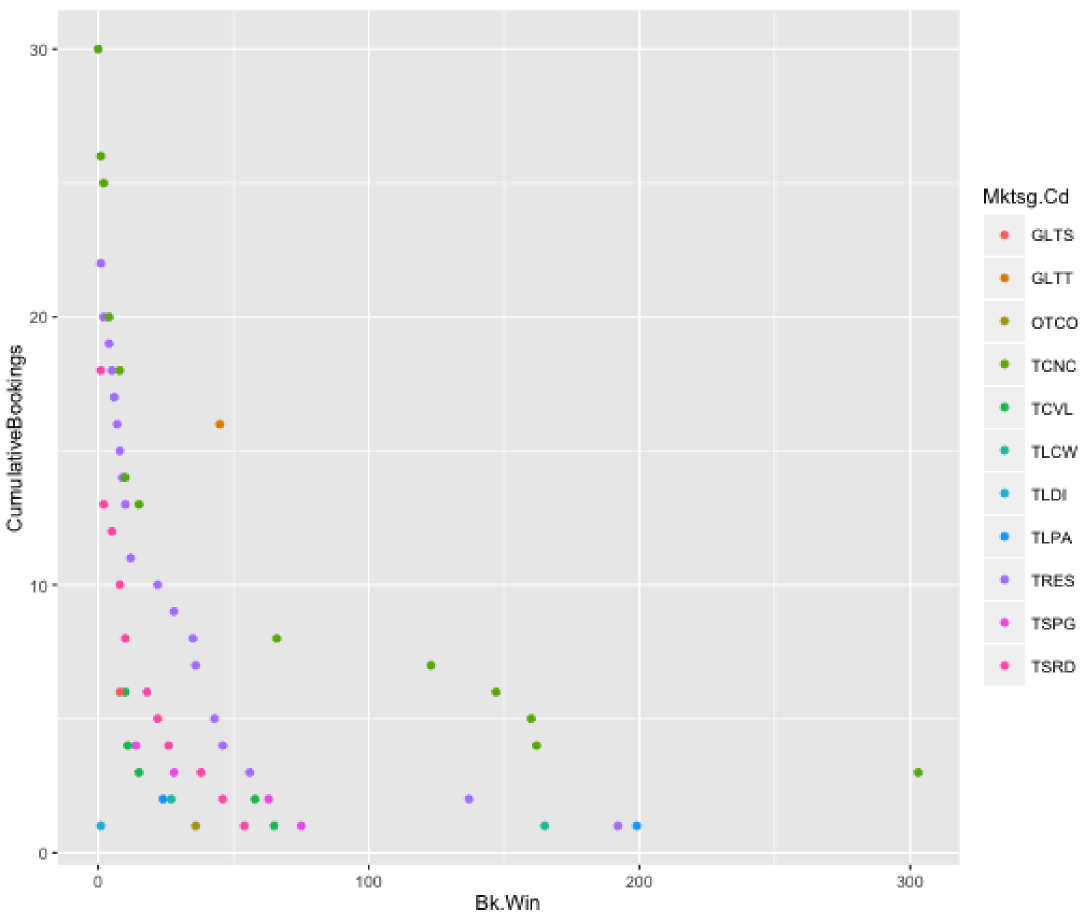

Figure 3-2: Booking window for different market segments.

3.2 Model

$\mathcal{S}$ is defined as the indexed set of market segments, e.g. $k = 3$ means "the third market segment". Hotels have various types of rooms, as well as different capacities of them, and consequently prices depend on this. $\mathcal{L}$ is the set of room types, which is also indexed.

Moreover, clients can book rooms at different time periods, ranging between the same day of their stay day, and many months in advance. The number of days the booking was realized before the stay day is denoted as the *booking window*. However, when grouping by booking window, market segment, room type and stay day, booking data is very sparse (many entries are 0's). Analyzing every single booking window is

Booking window	Days prior to stay day	Average number of bookings
1	0 (arrive on stay day)	16
2	1	18
3	2	17
4	3 - 4	32
5	5 - 6	35
6	7 - 10	51
7	11 - 15	49
8	16 - 23	62
9	24 - 30	43
10	31 - 50	68
11	51 - 70	30
12	71 and more	55

Table 3.1: Booking window aggregation and average of number of bookings.

not necessarily the best approach when for most high booking windows, there are no bookings. Also, it makes the model more complex than it needs to be. That is why, bookings are grouped in *booking window buckets*: time periods determined by the distribution of bookings. The used booking window buckets, and the average number of bookings for each are described in detail in Table 3.1. For simplicity, from now a booking window refers to a booking window bucket; let $\mathcal{T} = \{1, ..., T\}$ be the finite set of booking windows, where if $j \in \mathcal{T}$, $j = 10$ signifies that the booking was made between 31 and 50 days ahead.

Given a hotel and stay day, the goal is to maximize the total revenue by selecting the optimal price for each booking window, market segment, and room type combination (jkl), as well as the optimal number of rooms to allocate to each. We let $\mathcal{P}_{jkl}$ be the set of possible prices for each combination. These decisions will certainly depend on the demand of every combination, which we estimate in section 4.

However, the pricing and allocation of rooms at the hotel company have business constraints that must be inputed into the optimization model, in order to get a valid solution that can be implemented. They are the following:

- The set of possible prices $\mathcal{P}_{jkl}$ will be determined by the company, since it depends on each hotel, brand, country, currency, pricing strategy (e.g., \$199.99 instead of \$200), etc.

- Only one price can be selected for each jkl combination of market segment k, booking window j and room type l, i.e. an indicator variable I_{ijkl} is defined, and it has value 1 if price $p_i \in \mathcal{P}_{jkl}$ is selected, and 0 otherwise.

- Some market segments have fixed prices. For example, rates negotiated beforehand with consulting firms. Therefore, $\mathcal{S}_0 \subset \mathcal{S}$ is defined as the set of market segments with fixed prices; if $k \in \mathcal{S}_0$, then the price will not be optimized, but its stochastic demand should be incorporated into the model to account for the capacity loss it generates.

- There is a room type hierarchy, which is determined by the company. Customers of a room type may be complimentary upgraded to any of the higher hierarchies if needed.

- Sold rooms must not exceed the total hotel capacity, following the previous hierarchical upgrade rule.

Chapter 4

Demand Prediction

This chapter contains a report of the random forest machine learning technique utilized to predict demand. First, an explanation of the parameters selected for the model is provided. Then, the training and testing of the model is described. At the end, numerical results of the performance of the model on the data set granted by the hotel company are presented.

4.1 Prediction model

The objective of this model is to predict the daily aggregated demand for each market segment, room type and booking window (jkl combination), which is unknown a priori. Demand can take nonnegative integer values, and thus it is modeled as discrete. Since we the optimization is done for independent stay days and properties, demand is assumed to be independent across stay days and jkl booking possibilities. Moreover, bookings are usually higher when made closer to the stay day, which poses a problem for demand modeling: time cannot be split in periods with the same length, because there are considerably less bookings many days in advance to the stay day. That is why booking window buckets with different time lengths are chosen. Based on the distribution of bookings across the days in advance they were made, the new booking window periods are selected as presented in Table 3.1. It is important to mention that when there is no booking information for a certain jkl booking possibility, a 0 is introduced for those parameters, and then aggregated into the according booking window period.

The dependent variable of the model is the number of bookings. The independent variables used as predictors are:

- Booking window (j)
- Average Daily Rate (ADR)
- Market segment (k)
- Room type (l)
- Month of stay day
- Day of the week of stay day

In order to predict the daily demands, various machine learning techniques were tried: linear regression, quadratic regression, spline approximation, and random forests. The last was selected, since it returns the best results, mainly because of the advantage random forest models present when dealing with a mix of numerical and categorical variables, as is the case here.

A random forest is an ensemble machine learning method that can be used for classification and also regression. It is constructed by the generation of a determined number of individual decision trees, and the random forest outputs a value that is a function of all the independent trees. The usual is the mode or the mean of the values returned by the single decision trees. The advantage of a random forest over a single decision tree model is that it improves accuracy, and also reduces prediction variance, because it corrects the overfitting an individual tree can cause. There are two types of decision trees as well: classification (categorical values) and regression (numerical values). A decision tree is a tree-like graph decision model, for which the returned value is obtained by following the branches indicated by each split evaluation. The benefit of using an individual decision tree compared to a random forest is that it is easier to interpret and follow [15]. For an example of a simple regression tree, see Figure 4-1.

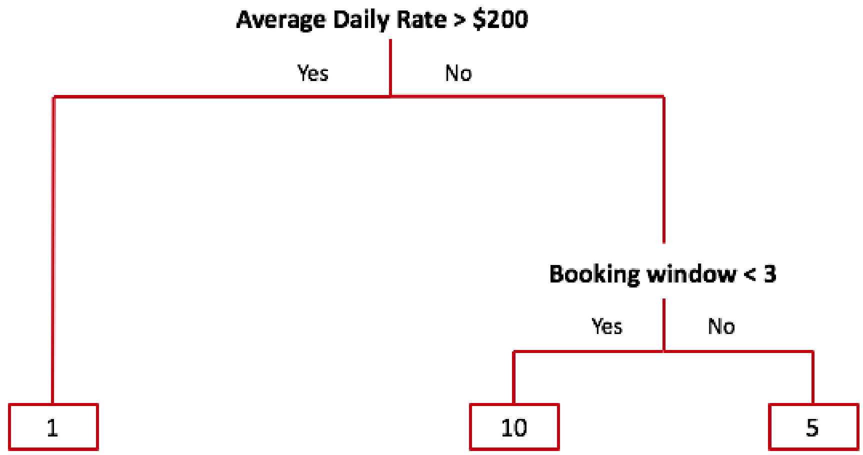

Figure 4-1: Example of a regression tree.

When constructing a random forest, the individual trees are trained with different random samples, that are subsets of the full training set. This introduces variability in the trees, and the different outputs of the individual regression trees can be used to estimate the probability density function of the true demand. The average of the returned demand values of the trees is used as the final demand prediction, which will be utilized in the revenue-maximizing Mixed Integer Linear Programming model detailed in section 5.

4.2 Numerical results

The first step is to select training and testing data sets to create the random forest model. Two-thirds of the data for 4 years of bookings (see section 3.1 for more details about the data) are randomly assigned to the training set, and the remaining one-third is the testing set. Since booking between days are assumed to be independent, time sequences are not considered for the splitting.

100 regression trees (CART) are generated for each jkl booking possibility (recall that there are 1,020) and trained by employing a bootstrap aggregation ("bagging") technique that reduces the variance caused by a single regression tree, which consists of randomly sampling (with replacement) a fixed-size subset of the training data repeated times. The sample for each tree is taken with replacement, and has a size of N data points (total number of observations in the training data). Also, 4 out of 6 input variables are randomly selected to split at each node. The minimum number of observations in a terminal node is set as 25. Smaller sets are constructed from the data in the training set, and cross-validation is used to select the random forest model's parameters.

Once the parameters are selected and the random forests are trained, the accuracy of their predictions is checked on the testing set. To do so, two error metrics are used:

1. Mean Absolute Error: For every prediction and its true value, the absolute difference is taken, and then the mean of these absolute errors. It is affected by the magnitude of the values to predict.

2. Mean Percentage Error: For every prediction and its true value, the difference is taken, divided by the true value, and then its absolute value. This returns the percentage error of each prediction. The mean of all these is computed.[1]

These two metrics are shown for the hotel company's own demand forecast, and for the random forest model presented, called RF. Adding data for booking cancellations proved to improve the predictions, thus "RF-w/o cancellations" shows the errors when not using cancellations data, and "RF-Net" when taking that data into account. Observe in Table 4.1 that both errors were significantly reduced with the random forest model developed in this thesis, yielding a 9.96% Mean Percentage Error for *RF* with cancellations data.

[1]In the case where $d_i = 0$, the percentage error is replaced with its corresponding absolute error to avoid involving infinite values.

Measure	Mean Absolute Error mean($\lvert\hat{d}_i - d_i\rvert$)	Mean Percentage Error mean($\lvert\frac{\hat{d}_i - d_i}{d_i}\rvert$)
Hotel's forecast	4.53	322%
RF-w/o cancellations	0.366	14.3%
RF-Net	0.232	9.96%

Table 4.1: Performance metrics used to compare regression models. [22]

All the model, and its training and testing is done in R, using the package *ranger*, which provides a fast implementation as well as tools for variable importance analysis [27]. Note in Figure 4-2 that by far the most important variable in the model is Average Daily Price (ADR), followed by Segment. This indicates that demand is very price sensitive, and that taking different market segments into account is important.

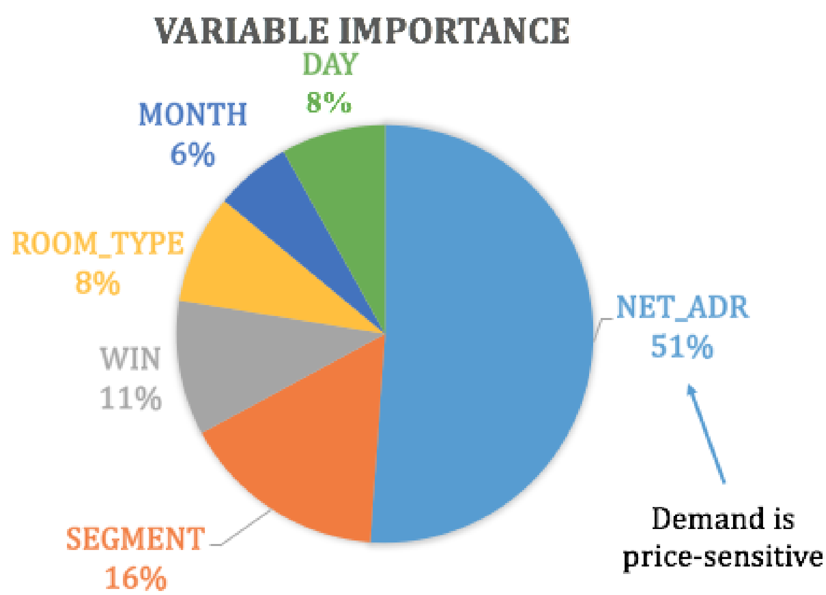

Figure 4-2: Variable importance. [22]

For more details about the specific parameter selection, tuning, pruning, and more thorough metrics for this data set, please see Rui Sun's Master's thesis [22], who collaborated in this project.

Chapter 5

Decision Optimization

This chapter presents in detail optimization mathematical model, as well as the developed solution for it, and numerical results when applied to the used data set from the hotel company. First, the model's variables, constants, objective function and constraints are described. Second, the optimization program is formulated and modified to the final Mixed Integer Linear Program to be used. Third, theoretical guarantees and theorems and provided, showing that the presented MILP returns an upper bound for the optimal revenue, and that the closeness of this bound depends on the demand forecast accuracy. At the end, numerical results for the data set are displayed and discussed.

5.1 Mixed Integer Linear Programming formulation

Solving the dynamic program that yields the optimal expected revenue is computationally intractable when the decision space is large, as it is most of the time for complex problems. Thus, a Mixed Integer Linear Programming model (MILP) is developed, that allows the user to find a close upper bound of the optimal revenue for each stay day, as will be proved later. First, the constants, variables, objective function and constraints of the MILP are described.

5.1.1 Constants

- p_{ijkl}: Price p_i for market segment k, booking window j, and room type l, i.e. $p_i \in \mathcal{P}_{jkl}$.
- c_l: The current capacity for room type l.
- d_{ijkl}: The estimated demand given price p_i for market segment k, booking window j, and room type l.

5.1.2 Variables

- Selected price indicator $I_{ijkl} \in \{0, 1\}$: Takes value 1 if price $p_i \in \mathcal{P}_{jkl}$ is selected for market segment k, booking window j, and room type l, and 0 otherwise.
- Capacity allocation $Y_{ijkl} \in [0, \infty)$: The capacity allocated to market segment k, booking window j, and room type l for price p_i.
- True capacity allocation $Z_{ijkl} = I_{ijkl}Y_{ijkl} \in [0, \infty)$: Auxiliary variable that permits having constraints and objective function all linear.

5.1.3 Objective function

The goal is to maximize the total revenue of a given stay day. For this, each of the prices is multiplied by its corresponding indicator variable I_{ijkl} (causing the product to be 0 if the price was not selected), and also by the related allocated capacity Y_{ijkl}, returning the expected revenue collected for a specific booking window, room type, market segment and price. The sum over all these yields the total revenue.

5.1.4 Constraints

- Total allocated capacity must not be greater than the room type capacity, following the free upgrade hierarchy.
- Only one price can be selected for each $ijkl$ combination.

- Allocated capacity shall not be greater than the expected demand.

- If a segment $k \in \mathcal{S}_0$, i.e. if segment k has a a fixed price, then the demand predicted for that fixed price p_{1jkl} (note $i = 1$) is inputed, and the demands for the other prices are set to 0.

5.1.5 MILP

Putting all the previous elements together, the following Optimization Integer Program is defined:

$$
\begin{aligned}
(M1): \quad & \max J = \sum_{l=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{T}\sum_{i=1}^{N} p_{ijkl} Y_{ijkl} I_{ijkl} \\
& \text{s.t.} \\
& \sum_{l=r}^{L}\sum_{k=1}^{K}\sum_{j=1}^{T}\sum_{i=1}^{N} Y_{ijkl} I_{ijkl} \leq \sum_{l=r}^{L} c_l \quad \forall r = 1, ..., L \\
& \sum_{i=1}^{N} I_{ijkl} = 1 \quad \forall\, j, k, l \\
& 0 \leq Y_{ijkl} \leq d_{ijkl} \quad \forall\, i, j, k, l \\
& I_{ijkl} \in \{0, 1\} \quad \forall\, i, j, k, l \\
& d_{ijkl} = 0 \quad \forall i \neq 1, k \in \mathcal{S}_0
\end{aligned}
$$

where:

- N is the number of prices in the set $\mathcal{P}_{jkl}$ corresponding to each i.

- T is the number of time periods in set $\mathcal{T}$.

- K is the number of market segments in set $\mathcal{S}$

- L is the number of room types in set $\mathcal{L}$.

However, this is not a linear problem, because there are decision variables being multiplied in both the objective function and the first constraint. To overcome this, the new variable $Z_{ijkl} = Y_{ijkl}I_{ijkl}$ is defined, making the constraint and objective function linear, and thus $MILP1$ a Mixed Linear Optimization Program.

$$
\begin{aligned}
(MILP1): \quad & \max J = \sum_{l=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{T}\sum_{i=1}^{N} p_{ijkl}Z_{ijkl} \\
& \text{s.t.} \\
& \sum_{k=1}^{K}\sum_{j=1}^{T}\sum_{i=1}^{N} Z_{ijkl} \leq \sum_{l=r}^{L} c_l \quad \forall r = 1, ..., L \\
& \sum_{i=1}^{N} I_{ijkl} = 1 \quad \forall\, j, k, l \\
& 0 \leq Z_{ijkl} \leq d_{ijkl}I_{ijkl} \quad \forall\, i, j, k, l \\
& I_{ijkl} \in \{0, 1\} \quad \forall\, i, j, k, l \\
& d_{ijkl} = 0 \quad \forall i \neq 1, k \in \mathcal{S}_0
\end{aligned}
$$

5.2 Theoretical properties for the MILP

5.2.1 Equivalence to a linear program

When formulating *MILP1*, the variables I_{ijkl} are binary, making it an *mixed integer linear* optimization problem, which is computationally speaking considerably harder to solve than a linear program. A common approach for this is relaxing the binary integer constraints to linear constraints $0 \leq I_{ijkl} \leq 1$, and then finding appropriate ways to approximate the true solutions.

Let *LP* be the linear program attained by relaxing the binary constraints in *MILP1*. Sun [22] shows that *LP* returns an equivalent optimal revenue than the one attained when solving *MILP1*.

5.2.2 The MILP yields an upper bound for the optimal revenue

Note that $\mathbf{d}$ is the vector of expected demands, meaning that *MILP1* solves a deterministic problem, and not the real stochastic revenue maximization problem

$$(OP) \quad J^*(x) := \max \ \mathbb{E}[\sum_{l=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{T}\sum_{i=1}^{N} p_{ijkl}U_{ijkl}]$$

with U_{ijkl} being the random variable denoting the number of rooms of type l sold at price i, booking window j, to market segment k. Every U_{ijkl} follows the business constraints in subsection 3.2.

The maximization problem presented at the end of subsection 5.1, which is called *MILP1*, can be represented in a more simple and general way as

$$\begin{aligned} (UB) \quad \tilde{J}(x) := \max \ & \mathbf{p^T W} \\ \text{s.t.} \quad & \\ & A\mathbf{W} \leq \mathbf{x} \\ & 0 \leq \mathbf{W} \leq \mathbf{d} \end{aligned}$$

with vectors: $\mathbf{W}$ for decision variables, $\mathbf{x}$ for capacities, $\mathbf{d}$ for demands, and $\mathbf{p}$ for prices. Matrix A corresponds the coefficients that multiply the variables in the first two constraints of *MILP1*. Using this theoretical framework, we prove the following theorem.

Theorem 1. *$\tilde{J}(x) \geq J^*(x)$, i.e. $MILP1$ returns an upper bound for the optimal revenue.*

Proof. Since $f(x) = max(x)$ is a convex function, by Jensen's inequality:
$\mathbb{E}[max(X)] \geq max(\mathbb{E}[X])$.

$\square$

It can also be proven using duality theory, by taking the dual of UB, which is a minimization problem. The dual's objective function turns out to be the lagrangean relaxation of OP. For more about this proof, see [23].

5.2.3 Closeness of the upper bound to the optimal revenue

Letting D_t be the random variable that describes the demand of the $ijkl$ combination (having a unique t for each $ijkl$ parameter combination), and y_t the allocated capacity to that demand, we can formulate

$$
\begin{aligned}
(LB) \quad J^L(x) :=& \max \sum_{t=1}^{C*K*T*N} p_t \mathbb{E}[\min D_t, y_t] \\
=& \max \sum_{t=1}^{C*K*T*N} p_t \left(\sum_{a=0}^{y_t - 1} a\mathbb{P}(D_t = a) + y_t \mathbb{P}(D_t \geq y_t) \right) \\
\text{s.t.}& \\
& A\mathbf{W} \leq \mathbf{x}
\end{aligned}
$$

It is easy to see that by taking the expectation of the minimum between both, and multiplying it by its corresponding price, a lower bound on the value of $J^*(x)$ is attained, since y_t is a partitioned allocation, and it is a feasible policy for the network problem [23]. Therefore, $J^L(x) \leq J^*(x)$.

Now, having both an upper and a lower bound for the optimal revenue $J^*(x)$, it is possible to analyze the gap between them, and thus the closeness of the revenue that

MILP1 returns to the optimal one. When assuming the probabilistic distribution of the demand is the one attained with the random forest bagging technique described in section 4, the gap between the upper and lower bounds depends on the demand forecast accuracy.

Theorem 2. $\tilde{J}(x) - J^L(x) \leq C \max_t \mathbb{P}(D_t \neq d_t)$.

Proof. Using the distribution obtain with the random forest, if we have perfect forecasts for the demand, then $\mathbb{P}(D_t = d_t) = 1 \quad \forall t$, and $\tilde{J}(x) = J^*(x) = J^L(x)$ (they are all deterministic). $J^L(x)$ approaches $J^*(x)$ as the $\mathbb{P}(D_t = d_t)$ increase, and thus $J^*(x) - J^L(x) \leq B \max_t \mathbb{P}(D_t \neq d_t)$. When fixing $d_t = \mathbb{E}[D_t]$, without loss of generality, $\tilde{J}(x) - J^L(x) \leq C \max_t \mathbb{P}(D_t \neq d_t)$ with $C \geq B$. □

Therefore, as the accuracy of demand predictions increases, the gap between the upper and lower bounds for the value of the optimal revenue $J^*(x)$ decreases, causing the value returned by *MILP1* to be closer to the optimal revenue.

5.3 Numerical Results

In order to test the performance of the optimization program *MILP1* developed throughout this chapter, it was implemented on the data set granted by the hotel company.

Before doing the two-step approach, the sets of possible prices $\mathcal{P}_{jkl}$ were determined as follows: Let $\overline{p}_{jkl}$ be the historical price for the parameters jkl. Then, $\mathcal{P}_{jkl} = \{\overline{p}_{jkl} - 50,\ \overline{p}_{jkl} - 40,\ \ldots,\ \overline{p}_{jkl},\ \overline{p}_{jkl} + 10,\ \overline{p}_{jkl} + 20,\ \ldots,\ \overline{p}_{jkl} + 50\}$, an evenly-spaced sequence ranged from $\overline{p}_{jkl} - 50$ to $\overline{p}_{jkl} + 50$ with an increment of 10, giving 11 different prices in each set. This allows to make a fair comparison to historical prices, and it also simplifies the price selection.

The reader can recall from chapter 3 that there are 5 room types and 12 booking windows (see Table 3.1); from the 26 defined market segments, 17 are used. This gives 1020 booking possibilities. Furthermore, there are 11 different prices for each booking possibility, and 2 variables related to each (I_{ijkl} and Z_{ijkl}). Thus, there are 22,440 decision variables (2 types of variables both with 4 dimensions) in the optimization model.

For the first step of the approach presented in this thesis, demand predictions are calculated for each of the 11,220 $ijkl$ parameter combinations using the trained random forest model RF described in chapter 4. As for the second step (the optimization), the number of variables poses a computational challenge, especially because of the need to run the whole method every hour. The MILP was implemented and solved using the package JuMP (Julia for Mathematical Optimization) [11], which is an optimization package for the programming language Julia [4].

To analyze the results, different revenue metrics are defined and compared, as illustrated in Figure 5-1:

- J^{opt}: optimal revenue obtained when using the prediction results of RF under all prices from $\mathcal{P}_{jkl}$, and then finding the optimal price and capacity allocation with *MILP1*.

- J^{opt_rf}: optimal revenue when using only historical prices, but getting the demand predictions for those prices with *RF*, and optimizing capacity allocation with *MILP1*.

- J^{opt_hist}: optimal revenue when using historical demands and prices, and only optimizing capacity allocation with *MILP1*.

- J^{hist}: the historical revenue, thus the prediction and the optimization models were not used.

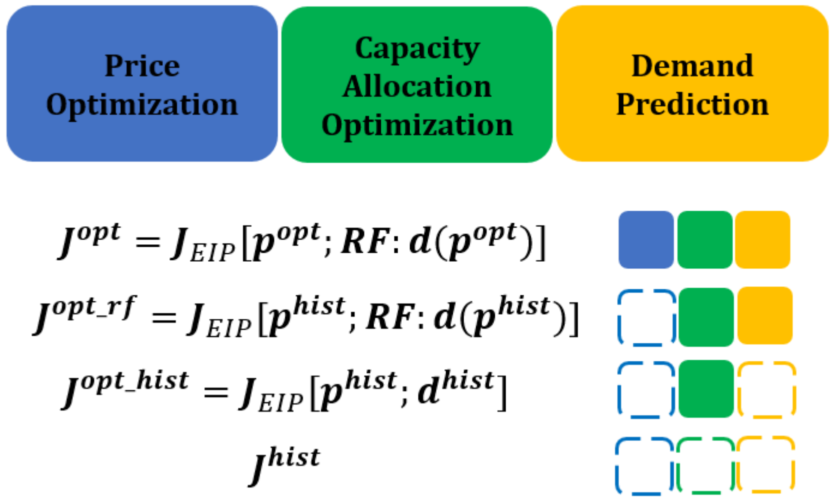

Figure 5-1: Comparison of different revenue metrics. [22]

With these metrics, it is possible to study the intermediate revenue improvement when only using some parts of the whole proposed 2-step approach.

When doing the demand forecasts and decision optimization for the property in Times Square, for a sample day the hotel company recommended, the revenue improvements are encouraging, as it can be observed in Figure 5-2. When using the complete two-step approach, there is a revenue increase of 14.5%. The difference between J^{opt_hist} and J^{hist} is null, which means that for this data, the capacity allocation decisions were the optimal. When also incorporating demand predictions, revenue grows by 3.2% in J^{opt_pred}. Finally, most of the total improvement is obtained when doing price optimization, because revenue increases by 11% in J^{opt}.

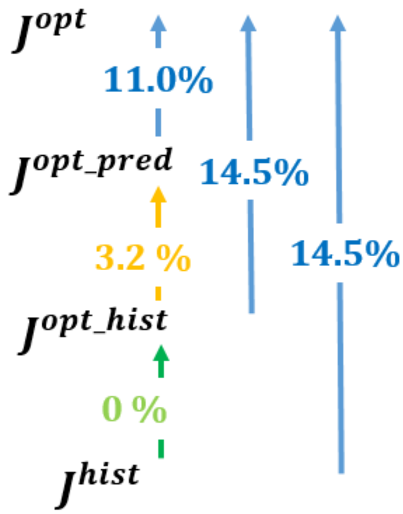

Figure 5-2: Revenue improvement under different metrics [22].

Chapter 6

Conclusions and Future Research

Almost every industry is highly dependent on demand, which is uncertain. The ability to predict demand and adapt to its changes is vital for any company that wants to stay in business. This poses a great challenge to corporations, because competitors are using Data Science to optimize their pricing decisions. Thus, being able to understand customer behavior better and faster is a very valuable advantage. Machine learning techniques are proving to be extremely useful in predicting unknown information that is required to making fast decisions, that sometimes have to be made automatically and in real time.

However, predicting demand is only the first step for successful Revenue Management solutions. Usually, there is an enormous decision space, because there is a big set of joint pricing and capacity allocation decision variables, and it grows when doing optimization simultaneously for many time periods, market segments, and even different products. The dynamic program to solve the optimization model is most of the time computationally intractable, as is the case of the problem discussed in this thesis.

In this paper a powerful and fast way to tackle this challenge is presented by using a 2-step approach: 1) demand prediction using random forests with bagging; 2) a Mixed Integer Linear Program that closely approximates the optimal revenues, and returns pricing pricing and capacity allocation decisions.

Data of one of the most important properties of a main hotel chain in Times Square, in New York City, is used to train and test the before mentioned solution, returning very positive results compared to the similar literature. Future demand is predicted with only a 9.96% mean percentage error, and the reason of this is the incorporation of price and market segment information to the random forest model, which are both highly significant in determining demand in many industries.

Then, in order to maximize revenue for a stay day, an optimization formulation is presented and developed. By using variable substitution the problem is converted into a Mixed Integer Linear Program (MILP), which is an upper bound to the optimal revenue, and returns close-to-optimal decision variables. It is shown that the MILP can be equivalently solved as a linear program, and it's also proven that it is an upper bounds for the expected revenue maximization Dynamic Program (DP), and that the closeness to the optimal depends on the probabilistic distribution of the demand. Thus, for high prediction accuracies, the optimal value of the DP can be closely approximated by the MILP solution.

Finally, the presented solution yields a 15% revenue increase compared to the historical one; and the entire process is computed in under 4 seconds for each stay day, which allows the hotel company to run it and update their room prices and allocations virtually on real time.

As a future work suggestion, a logical expansion of the presented method is to develop it for revenue maximization of larger periods of time, opposed to only single days, since a significant percentage of customers stay 2 or more consecutive nights. The Mixed Integer Linear Program can be augmented to account for aggregated demand, but demand forecasting for these multiple-day stays poses a complex challenge. Another interesting improvement to consider is incorporating multiple hotels and their interactions in both demand forecasting and revenue optimization, instead of a single hotel as is the case in this thesis.

Appendix A

MILP code

```
using JuMP
using DataFrames

#Repeat for the number of days for which there are predictions (in this case 3).
days=3;
revs=zeros(days,1);

for a=1:days

    tic();
    #Read forecast data (demand)
    data = readtable("myforecast"*string(a)*".csv");
    #Read model parameters = N,T,S,L: # of prices; # of booking windows; # of market segments; # of room classes
    par = readtable("mypar"*string(a)*".csv");
    #Read capacities
    cap = readtable("mycap"*string(a)*".csv");
    c=convert(Array, cap);
    parameters=convert(Array,par[:1]);
    N=parameters[1];
    T=parameters[2];
    S=parameters[3];
    L=parameters[4];
    p=data[:1];
    d=data[:2];

    #Define IP optimization model
    mod = Model()
    @defVar(mod, Z[i=1:N*T*S*L] >= 0)
    @defVar(mod, I[i=1:N*T*S*L], Bin)
    @setObjective(mod, Max, sum{p[i]*Z[i],i=1:N*T*S*L})
    @addConstraint(mod, xyconstr[j=1:L], sum{Z[i],i=(1+(j-1)*N*T*S):N*T*S*L} <= sum{c[i],i=j:L})
    @addConstraint(mod, xyconstr[j=0:T-1,k=0:S-1,l=0:L-1], sum{I[j*N+k*N*T+l*N*T*S+i],i=1:N} == 1)
    @addConstraint(mod, xyconstr[i=1:N*T*S*L], Z[i] <= d[i]*I[i])

    solve(mod)
    revenue=getObjectiveValue(mod);
    println(a)
    println("Revenue: ", revenue)
    println("Max of Z (capacity allocation): ", maximum(getValue(Z)))
    println("Mean of Z (capacity allocation): ", mean(getValue(Z)))
    println("Mean of I (1 if price allocated): ", mean(getValue(I)))
    toc();
    out = [getValue(Z) getValue(I)];
    df = DataFrame(out);

end

rf=DataFrame(revs);
rename!(rf, :x1, :Revenues);
writetable("revenues.csv", rf);
```

Figure A-1: MILP code in Julia.

Bibliography

[1] R. D. Badinelli. An optimal, dynamic policy for hotel yield management. *European Journal of Operational Research*, 121:476–503, 2000.

[2] Dimitri P. Bertsekas and et al. *Dynamic programming and optimal control Vol. 1*. Athena Scientific, 1995.

[3] Dimitri P. Bertsekas and et al. *Dynamic programming and optimal control Vol. 2*. Athena Scientific, 1999.

[4] Jeff Bezanzon, Stefan Karpinski, Viral Shah, and Alan Edelman. Julia: A fast dynamic language for technical computing. In *Lang.NEXT*, April 2012.

[5] Eduardo Candela, David Simchi-Levi, and Rui Sun. Analytics for a Hotel Property: Demand Prediction and Optimization of Pricing and Capacity Allocation Decisions. *Working paper*, 2017.

[6] Boxiao Chen, Xiuli Chao, and Cong Shi. Nonparametric algorithms for joint pricing and inventory control with lost-sales and censored demand. 2015.

[7] Christopher Chen and Soulaymane Kachani. Forecasting and optimisation for hotel revenue management. *Journal of Revenue and Pricing Management*, 6:163–174, 2007.

[8] Wang Chi Cheung, David Simchi-Levi, and He Wang. Dynamic pricing and demand learning with limited price experimentation. *Operations Research*, 2017.

[9] George Dantzig. *Linear programming and extensions*. Princeton university press, 2016.

[10] Arnoud V. den Boer. Dynamic pricing and learning: historical origins, current research, and new directions. *Surveys in operations research and management science*, 20(1):1–18, 2015.

[11] Iain Dunning, Joey Huchette, and Miles Lubin. Jump: A modeling language for mathematical optimization. *SIAM Review*, 59(2):295–320, 2017.

[12] Vivek F. Farias and Benjamin Van Roy. An approximate dynamic programming approach to network revenue management. 2007.

[13] Guillermo Gallego and Garrett Van Ryzin. Optimal dynamic pricing of inventories with stochastic demand over finite horizons. *Management science*, 40(8):999–1020, 1994.

[14] Guillermo Gallego and Garrett Van Ryzin. A multiproduct dynamic pricing problem and its applications to network yield management. *Operations research*, 45(1):24–41, 1997.

[15] Tin Kam Ho. Random decision forests. In *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on*, volume 1, pages 278–282. IEEE, 1995.

[16] Kris Johnson. *Analytics for online markets.* PhD thesis, Massachusetts Institute of Technology, 2015.

[17] Kris Johnson, Bing Hong Alex Lee, and David Simchi-Levi. Analytics for an Online Retailer: Demand Forecasting and Price Optimization. *Manufacturing & Service Operations Management*, 18:69–88, 2015.

[18] A. O. Lee. *Airline Reservations Forecasting: Probabilistic and Statistical Models of the Booking Process.* PhD thesis, Massachusetts Institute of Technology, 1990.

[19] P. Liu, S. Smith, E. Orkin, and George Carey. Estimating unconstrained hotel demand based on censored booking data. *Journal of Revenue and Pricing Managmement*, 1(2):1476–6930, 2002.

[20] Los Angeles Times. Marriott closes $13-billion purchase of Starwood to become world's largest hotel chain. `http://www.latimes.com/business/la-fi-marriott-starwood-20160923-snap-story.html`. Accessed: August 5, 2017.

[21] Warren B. Powell. *Approximate Dynamic Programming: Solving the curses of dimensionality.* John Wiley & Sons, 2007.

[22] Rui Sun. Analytics for Hotels: Demand Prediction and Decision OPtimization. Master's thesis, Massachusetts Institute of Technology, 2017.

[23] Kalyan Talluri and Garrett Ryzin. The Theory and Practice of Revenue Management. 2009.

[24] The Wall Street Journal. Airbnb vs. Hospitality Industry. `http://graphics.wsj.com/table/Airbnb_0615`. Accessed: August 5, 2017.

[25] TUI. Global hotel industry retail value from 2010 to 2018 (in billion U.S. dollars). `https://www.statista.com/statistics/247264/total-revenue-of-the-global-hotel-industry/`. Accessed: August 5, 2017.

[26] L. Weatherford and S. E. Kimes. A comparison of forecasting methods for hotel revenue management. *International Journal of Forecasting*, 19:401–415, 2003.

[27] Marvin N. Wright and Andreas Ziegler. ranger: A fast implementation of random forests for high dimensional data in c++ and r. *arXiv preprint arXiv:1508.04409*, 2015.

[28] Dan Zhang and Daniel Adelman. An approximate dynamic programming approach to network revenue management with customer choice. *Transportation Science*, 43(3):381–394, 2009.

[29] Dan Zhang and Daniel Adelman. An improved dynamic programming decomposition approach for network revenue management. *Manufacturing & Service Operations Management*, 13(1):35–52, 2011.