THE PRICE IS RIGHT:

MODELING DYNAMIC PRICING STRATEGIES IN UNITED STATES HOTELS

By: Aditya Khera

Applied Machine Learning
Professor Gabriel Young
May 2023

1. Introduction

1.1 Objective

A quick Google search with the terms “pricing model” will yield over half a billion results in less than a second. What these sites, charts, models, and papers all show effectively boils down to one thing: finding the right price to charge for a good or service. Historically, price models have been a core tenant of any firm, as they try to maximize profits given a product’s demand. From cost-plus, value-based, or performance-based pricing the list of hyphenated and complex methods goes on and on. The problem with many of these models, however, is that they are largely static. Whether it is the nature of the product or the reliance on slow trickling demand data, firms have historically been slow with how they price themselves. Miscalculations in pricing can have dire consequences, as inefficiencies can cost firms revenue or clients and be catastrophic to their bottom line.

Thanks to a surge in e-commerce and the rapid pace of technological development, minute-by-minute data on a product’s demand is easily accessible. With this has come the birth of dynamic pricing, a new model that uses real-time demand data to assign prices based on a plethora of factors. Time of year, the customer's location, and even past transactions can now be combined to find the perfect price for a product, maximizing the firm's revenue and the customer's satisfaction. This pricing model relies on the massive information available to firms and assigns values to products based on market conditions, modifying the price of products according to what each customer is willing to pay.

Perhaps no industry has benefited from the emergence of dynamic pricing quite like the travel sector. Rideshare apps, Airlines, and museums have all readily adopted the technology as a way of minimizing market inefficiencies and ensuring firms match consumer demand. This study examines another user of dynamic pricing in the travel industry: hotels. By using a variety of machine learning models, the aim of this paper is to describe the various means of predicting price points for different hotel locations. By engineering a predictive model for the various hotel rooms and reservations, the hope is to then be able to effectively price future hotel rooms to match demand under certain market conditions.

1.2 Dynamic Pricing

In environments where firms must compete with each other for the same products and market conditions are constantly changing, the need for dynamic pricing is obvious. Hotels, which often offer similar products and exist in close proximity to each other, therefore have much to gain from pricing their products, rooms and reservations in this case, below their competitors. Additionally, market conditions and target audiences are constantly changing, warm weather, low inflation, and school holidays all play major factors in the overall demand for hotel rooms.

Dynamic pricing can essentially be seen as the combination of a variety of predictors into one predictive model. For the purposes of hotel dynamic pricing, these predictors can be seen as three groups: condition-based pricing, competitor-based pricing, and time-based pricing. This paper will concern itself primarily with condition-based pricing where data on the customer or their reservation and their previous payment history can be used to predict what they may be willing to pay in the future. Competitor-based pricing is often seen in gas stations, where the price of a product is determined by what competitors are charging at that given moment. Finally, time-based pricing uses time windows to update prices throughout the day, week, or season. The regression models included in this paper will factor in the week of the year in an effort to combine time-based pricing with condition-based pricing.

2. Methodology

2.1 Process

The process for accomplishing the goals established in the objective section is shown below in Figure 1. Essentially, after obtaining the larger data set, a subset of “United States City Hotels” was selected. Exploratory data analysis was performed on this subset to see the way single variables interacted with the average daily rates. Cleaning and Exploratory data analysis will both be discussed later in this section. After EDA, 3 types of prediction models were built: Linear Regression, K-means Regression, and Random Forest. All three of these models were initially built using randomized parameters but were then cross-validated to find ideal fits for the training data. In the following section, the shortcomings and benefits of each of the three models will be discussed. Finally, loss functions were used to calculate both the training and test errors for all three models. The results from each prediction model were compared and the final Results section reflects on these models.

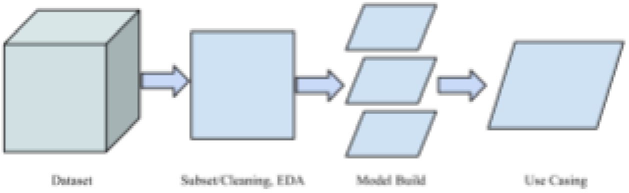


Figure 1: Methodology

2.2 Data

The data set used in this study was titled “Hotel booking demand” from Kaggle and was the “brainchild” of Nuno Antonio, Ana Almeida, and Luis Nunes three researchers at the University Institute of Lisbon. As such, the data primarily consisted of hotel bookings in Portugal, though a sizeable number of observations were pulled from the United States and France. The data set spanned over two years, tracking reservations from June 2015 to August 2017. Though obvious factors like inflation may call into question the validity of the research and models made in this paper, the underlying principle of combining multiple market conditions for predicting the price of a hotel reservation stands nonetheless.

The dataset initially contained around 120,000 observations from over 170 countries. With over 32 columns representing data from the number of weeknights reserved to the total number of special requests made, the dataset represented a massive undertaking by the research team at the University Institute of Lisbon and their efforts to provide an open-source dataset for others to experiment with.

2.3 Cleaning

As mentioned in the previous subsection, the dataset was a fantastic endeavor by the University Institute of Lisbon. To make this dataset more manageable and applicable to specific regions, a subset was taken. In this case, the dataset was cut to represent only United States hotels located in cities.

The second step in cleaning the subset was the removal of variables based on three sets of criteria: homogeneity, privacy concerns, and extraneity. Homogeneity was used as a basis for cutting when a categorical variable had more than 97% of the same value, such as whether or not a parking spot was required in a city hotel. Privacy concerns meant that information on the actual room or hotel had been scrubbed by the data set and represented by an alphanumeric code, which would be unhelpful for a regression problem. Finally, extraneity was used when data were considered irrelevant to the data set, not necessarily statistically insignificant, but rather unrelated to the study. After cutting variables, the dataset was 1618 observations of 13 variables, the variables cut are as follows.

- Homogeneity: number_babies, is_repeated_guest, previous_cancellations, required_car_parking_spaces, deposit_type, days_in_waiting_list, and customer_type,
- Privacy: reserved_room_type, assigned_room_type, company, and agent
- Extraneity: market_segment, distribution_channel, reservation status, arrival_date_month, and arrival_date_day_of_month (final two were conveyed in arrival_date_week_number)

2.4 Exploratory Data Analysis

The goal of the Exploratory Data Analysis was to map individual variables to the Average Daily Rate (ADR) and see what type of relationship existed between the two. This step provides additional information about certain variables and can help in model selection in the coming section. EDA in this paper is limited to 4 variables for brevity but further studies could help map the univariate relationship between all factors and the adr.

Arrival Date Week Number

Figure 2 shows the relationship between the arrival date week number and the adr of the corresponding reservation. From the figure, it can be seen that the distribution is neither linear nor uniform. Higher prices correspond to summer months and show evidence of an on and an off-season for the industry. The data shows perhaps a non-parametric or non-linear model may be best for representing the effects of the arrival date week number on adr.

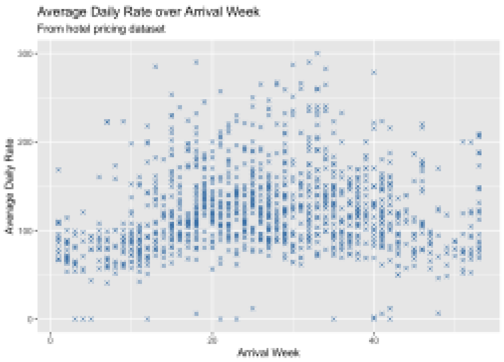


Figure 2: Arrival Week on adr

Number of Children and Adults

Figure 3 shows the linear relationship between the number of children in a reservation and the adr. As the figure shows, an increase in one child corresponds to a near-constant increase in the adr. This makes sense for hotels, as an additional child represents a higher-priced product for the consumer.

Figure 4 offers similar insight into the pricing for adding one adult to the reservation. In this figure, there is a more or less linear relationship between 0 to 3 adults in the reservation, but at 4 the mean adr drops. This could be because the 4th adult may represent a couple, or simply because 4-person accommodations are cheaper than 3 people. What this indicates, for the most part, is that as the size of the reservation grows for either children or adults, the adr increases linearly.

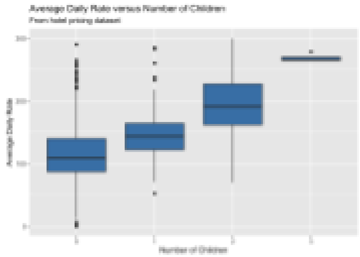


Figure 3: Children on adr

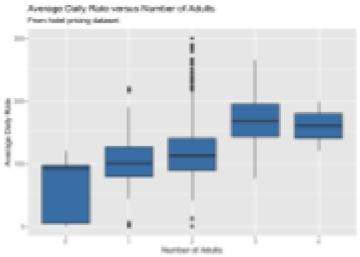


Figure 4: Adults on adr

Number of Accommodation Requests

Lastly, figure 5 represents the uniform distribution between the total number of special requests made and the adr. This relationship implies that special requests have little to no effect on the adr. A linear regression ought to have an approximately zero beta value for this variable.

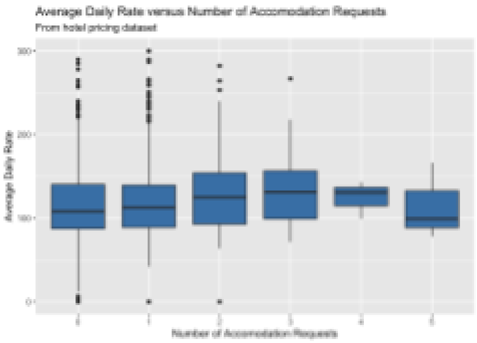


Figure 5: Number of Special Requests on adr

3. Modeling

After EDA, three different models were built, all with similar approaches. All models were built with 5-fold cross-validation and then evaluated using a squared loss function and RMSE.

3.1 Linear Regression

The first model built was a linear regression of all the variables in the data set. Using cross-validation, a formula for adr was developed with tuned beta values that performed well given the model's simplicity. Appendix A contains graphs of the linear function to showcase how well the model fits individual variables to the adr. For the four variables examined in the EDA, the results were mixed. The model assumed linearity between arrival date week number and adr despite the EDA showing other evidence. As the appendix shows, this relationship is quite inaccurate. The model did successfully predict the relationship between both children and adults on adr. Finally, the relationship between total special requests and adr was weakly positive which is in line with the EDA findings.

The shortcomings of linear regression are rather obvious, the assumption of linearity on variables that are not linear leads to erroneous predictions and contributes to the MSE of the model. Despite the inaccuracies of non-linear relationships, linear regression was by far the most interpretable and simplest of the three models. Though long, the equation is still easily understood by humans and the firms that may use such a model. This can not be said about the next two models.

$$adr \ = \ 69.08 \ + \ 17.81(canceled) \ -.13(leadtime) \ +.46(arrival\ week) \ - \ 3.52(weekend\ nights)$$
$$- \ 1.2(week\ nights) \ + \ 24.21(adults) \ + \ 31.3(children) \ + \ 36.6(HB\ Meal) \ - \ 23.05(SC\ Meal)$$
$$3.19(booking\ change) \ + \ 6.58(special\ requests)$$

3.2 K-means Regression

For the first non-parametric model, a K-means Regression was built to predict the adr in relation to our factors. Cross-validation showed that the best model was built using 7 neighbors. As such, the underlying shape of each variable's distribution was better preserved versus the linear regression model.

Appendix B contains visualizations of both the training and test data predictions by the model. As the graphs show, the overall structure of the data is preserved, though much less spread out. As the k is fairly large, the variance of the model is rather low, leading to a much more compressed spread for predictions. As Appendix B shows, both arrival week and total request predictions fit average adrs much better than the outliers, since there is less data or neighbors to draw from for high and lost price adrs.

This represents a large shortcoming for k-means regressions - the curse of dimensionality. Since there are a large number of predictors being combined into the model, neighbors are farther away from one another, leading to strong coalescence around the average adr. To overcome this, reducing the number of dimensions or increasing the amount of data would assist in building stronger predictions. While the shape of the data was preserved in the model, a massive amount of data would be required to build less biased estimators. According to some literature, overcoming the curse of dimensionality would require 2^d*10 observations, which in this case would be over 40000, much more than the 1600 being analyzed.

3.3 Random Forests

The final model built was Random Forests. Random Forest is seen as one of the most accurate and highly developed models in machine learning, aside from neural networks and boosted forests. The ensemble method uses the average of multiple smaller trees to help form a prediction, using the "wisdom of the crowd" to be more accurate than individual tree predictors. With this model, the cross-validation attempted to find the right tuning parameters to build the individual trees in a random forest.

With over 56 combinations of tree parameters trained in the cross-validation set, the best tune was then selected on the basis of its R-squared value, which was judged as the best metric to compare different models. Appendix C shows the R-squared performances for all 56 combinations. In the end, the model with the "variance" split rule, 3 variables to possibly split each node, and a minimum of 6 entries to split a node was selected. The model also built 300 trees, averaging them to minimize the variance of predictions. Building 300 independent trees would seem inefficient given the size of the data subset, but by using bagging and resampling methods the model was able to build 300 independent trees that formed in parallel.

As the later results section will show, the model performed better than both linear regression and k-means regression. The shortcoming, however, is the lack of interpretability. While linear regression could be shown as a formula and k-means could be shown graphically, actually breaking down the Random Forest model into something firms could understand proves difficult. While feature importance metrics can be informative, these “black box” modeling methods are hard for company executives to understand and adopt into their own business strategy.

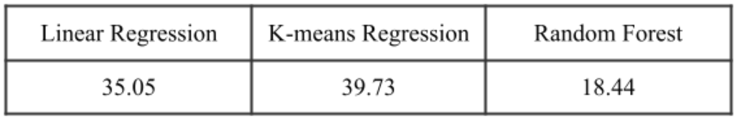
4. Results

For the three models, root mean squared error was used as a comparative metric. Within each model, the training and test errors were calculated and are reported below.

4.1 Training Errors

After building the models using cross-validation, the tuning parameters were noted down. Taking these tuning parameters, the root means squared errors were calculated. The errors were as follows: Linear Regression training RMSE 35.05, K-means Regression training RMSE 39.73, and Random Forest training RMSE 18.44.

These findings are rather mixed. Unsurprisingly, random forest appeared to perform the best on the training data. The parametric model, linear regression, surprisingly outperformed the non-parametric k-means regression. This can perhaps be explained by the high bias of the k-means regressor on account of the curse of dimensionality. With more data, k-means would likely outperform linear regression, but as it stands the k-means performed the worst of the three models.

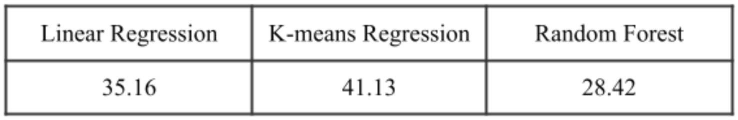


Linear Regression	K-means Regression	Random Forest
35.05	39.73	18.44

4.2 Test Errors

Prior to building each model the data subset was split 80-20 into training and testing sets. These testing sets were used to evaluate the model and estimate the generalization error of each model. The results mirrored the rankings of the training RMSEs with the following errors: Linear Regression testing RMSE 35.16, K-means Regression testing RMSE 41.13, and Random Forest testing RMSE 28.42.

The results match the training RMSEs but do also offer some insight into the generalizability of the models. The change in test and training RMSE for Linear Regression and K-means Regression were rather close, indicating a low chance of overfitting. The Random forests had a much larger difference between the training and test RMSE. While both measures were the lowest compared to the other two models, the discrepancy does raise concerns about the model’s ability to handle new and unseen data.



Linear Regression	K-means Regression	Random Forest
35.16	41.13	28.42

4.3 Applications and Future Work

The body of this work shows important findings (1) hotel pricing follows a series of patterns based on market conditions that can be modeled using machine learning and (2) those models all have their own advantages and shortcomings.

The generally low RMSE values for each of these models indicate that there is an underlying pattern to the way hotels are priced and that consumer demand follows certain patterns. These patterns can in turn be predicted using a variety of methodologies including but not limited to the ones observed in this paper.

The applications of these three models are quite apparent, they can all potentially be used to help hotels accurately price the reservations given a variety of conditions. As the previous section discussed, however, the models do come with their own unique shortcomings. The general trend is the more interpretable the method is, the less accurate, which means firms will have to decide if they want to have a pricing system that is less efficient but more interpretable or vice versa.

Future work in this project includes the following workarounds for existing limitations

- Extended EDA: As mentioned in the EDA subsection, only 4 variables were modeled in relation to the adr. For the sake of the paper's brevity, the other variables were simply added to the model formulas, without mapping them before inclusion. In the future, graphic depictions of all variables would assist in prior knowledge and reference.
- Mode observations: Since the subset only contained 1600 observations, k-means regression could not successfully be carried out due to the curse of dimensionality. Perhaps with a larger subset or more observations of United States city hotels, the k-means regression would outperform the linear regression model.
- Greater computational resources: This study was carried out using a Macbook Pro with limited computational power and RAM. Numerous times the small data subset was enough to crash the computer making larger observations practically impossible. For future work, devices with greater computational power would be ideal.

Appendix A Linear Regression Model

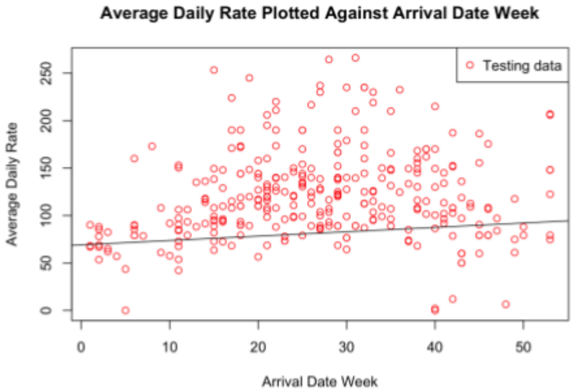


Figure A1: ADR vs Arrival Date Week

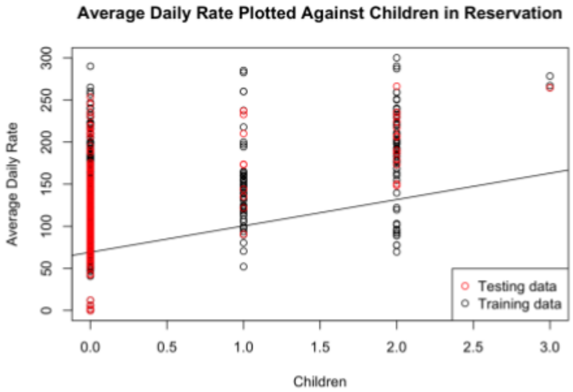


Figure A2: ADR vs Children in Reservation

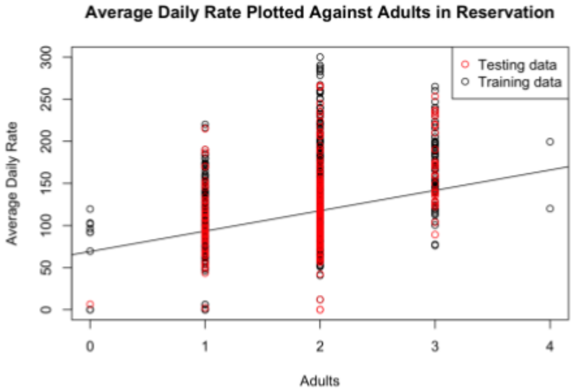


Figure A3: ADR vs Adults in Reservation

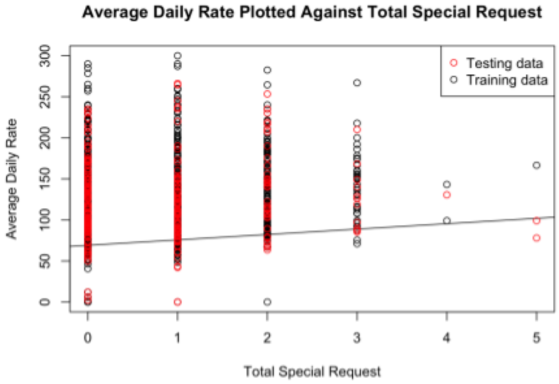


Figure A4: ADR vs Total Special Requests

Appendix B K-means Regression

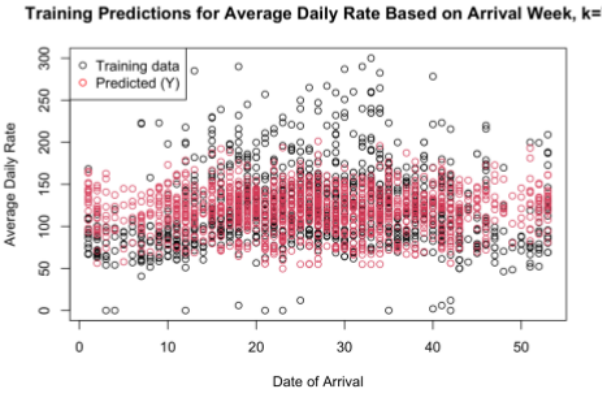


Figure B1: Training Predictions of Arrival Date Week

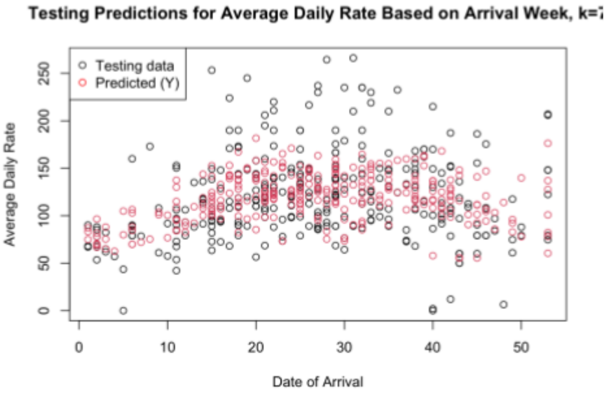


Figure B2: Test Predictions of Arrival Date Week

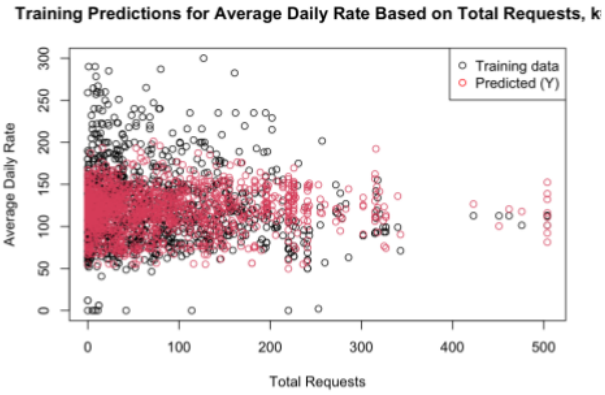


Figure B3: Training Predictions of Total Requests

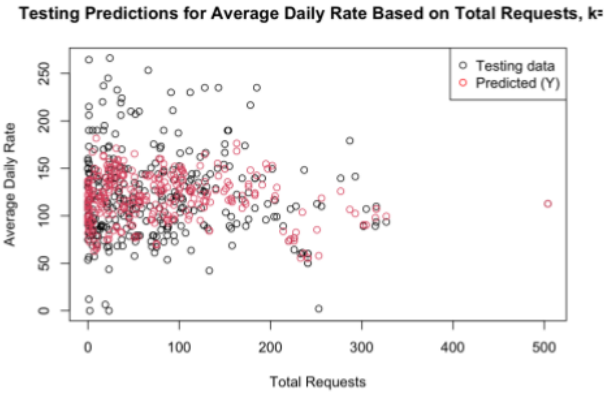
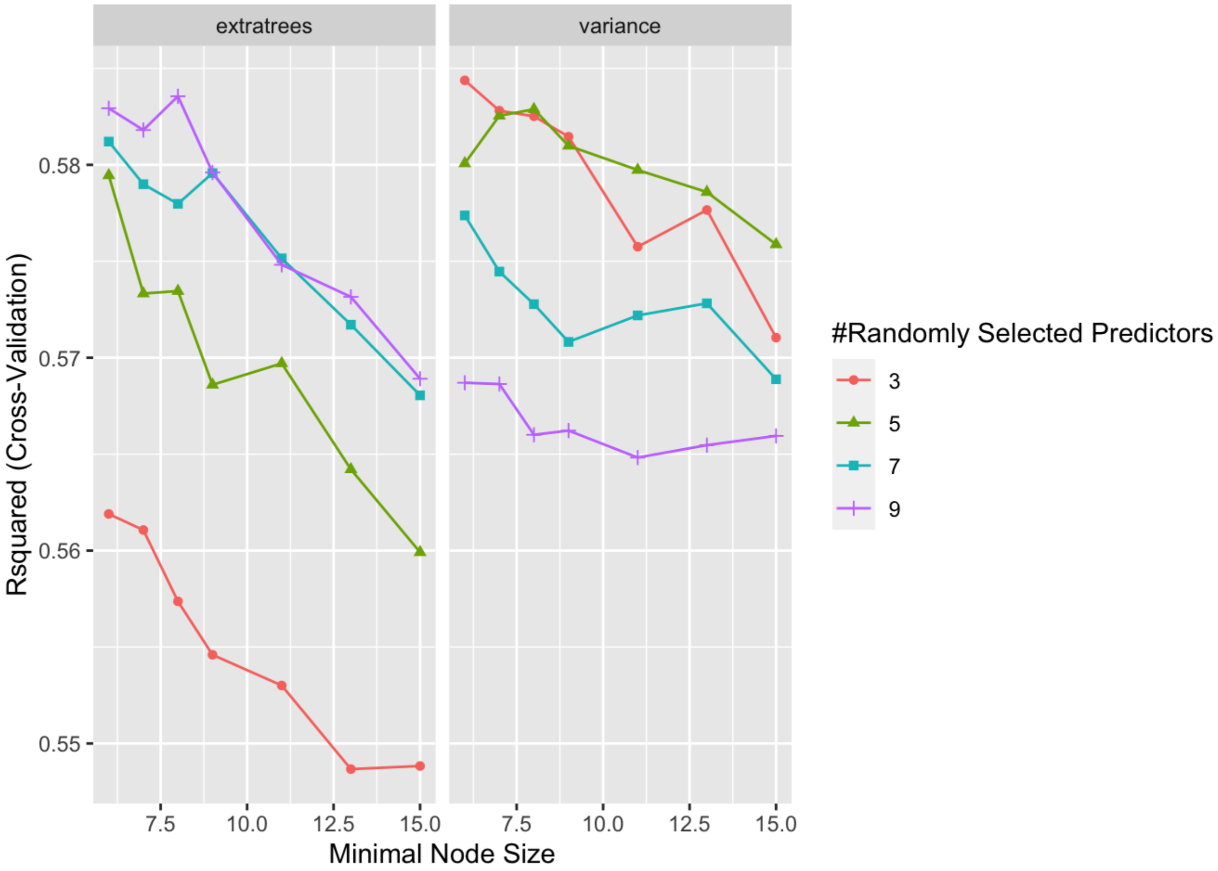


Figure B4: Test Predictions of Total Requests

Appendix C Random Forest Tuning


extratrees
variance
Rsquared (Cross-Validation)
Minimal Node Size
#Randomly Selected Predictors
● 3
▲ 5
■ 7