|  |  |
| --- | --- |
| Download Robert H Smith School Of Business Logo Png Transparent -  University Of Maryland PNG Image with No Background - PNGkey.com | **BUDT 758T**  **Final Project Report**  **Spring 2024** |

Group 21: AUC Competition

Saurabh Badkas, Pranav Bhushan,

I-Hung Ko, Amlan Mohanty, Anwesha Mohanty

Professor Eaman Jahani

BUDT758T-0506

12 May 2024

**Section 1:** Team Member Names & Contributions

|  |  |
| --- | --- |
| **Member Name:** | **Project Contributions:** |
| Saurabh Badkas | Spearheaded the group in setting up zoom meetings and worked on the competition/game for increasing the AUC. Also, worked on Section 3 of the Report with I-Hung. |
| Pranav Bhushan | Worked on some of the modeling, though mostly worked on writing the Final Project Report. Regarding the Report, Pranav specifically wrote Sections 1, 2, and 5 and focused on proofreading, formatting, and editing the report. |
| I-Hung Ko | Worked on models for the competition/game to increase AUC , the EDA component of the project, and worked on Section 3 of the Report with Saurabh. |
| Amlan Mohanty | Worked on feature engineering, model building and evaluation and optimizing AUC score for the competition.Worked on Section 4 of the Report with Anwesha. |
| Anwesha Mohanty | Worked on model selection,building and evaluation, to increase AUC for the competition. Also worked on Section 4 of the Report with Amlan. |

**Section 2:** Business Understanding

The travel/hospitality industry is an industry mainly marketed to families regarding traveling and vacationing. This industry has historically been stable, including services ranging from traditional hotels and motels to car rentals and tour guides. However, there is one unorthodox service that has taken the industry by storm: Peer-to-Peer Hospitality Services. This is mostly dominated by Airbnb, but includes competitors such as Vrbo and HomeAway. While Airbnb is the most dominant of the three, and actually outperforms many traditional hotel/motel firms, unfortunately, the travel/hospitality industry is on a decline, and Airbnb is taking some of the hit. Airbnb has been struggling to make a profit in 2024, and with inflation, stricter rental regulations (i.e. New York City’s “Airbnb ban”), declining travel/vacationing by consumers, and even a nationwide housing shortage, many Airbnb Hosts have raised prices on rentals and have even removed some listings from the website. Due to these factors, many hosts have been struggling to find renters, and even predict which unit(s) are more profitable/more likely to be booked.

Given these trends and concerns, this is where Group 21 Consulting comes into play. As seasoned data analysts and consultants, we can help Airbnb Hosts during this difficult economic period. Our team utilizes AUC (Area Under Curve) Modeling to predict which Airbnb listings would have high booking rates, or a high probability of being booked. This modeling method distinguishes itself from other models, as for this model, scores above 0.5 helps distinguish opposing variables, in this case, High Booking Rate and Low Booking Rate, and unlike metrics such as RMSE or FPR, higher scores are better. This helps not only with simplicity, but also transparency. With our modeling and solutions, hosts would be able to better strategize, price, and list their rental unit(s), and can see an upwards of 20% increase in rental revenue.

**Section 3:** Data Understanding and Data Preparation

**Section 3.1:**

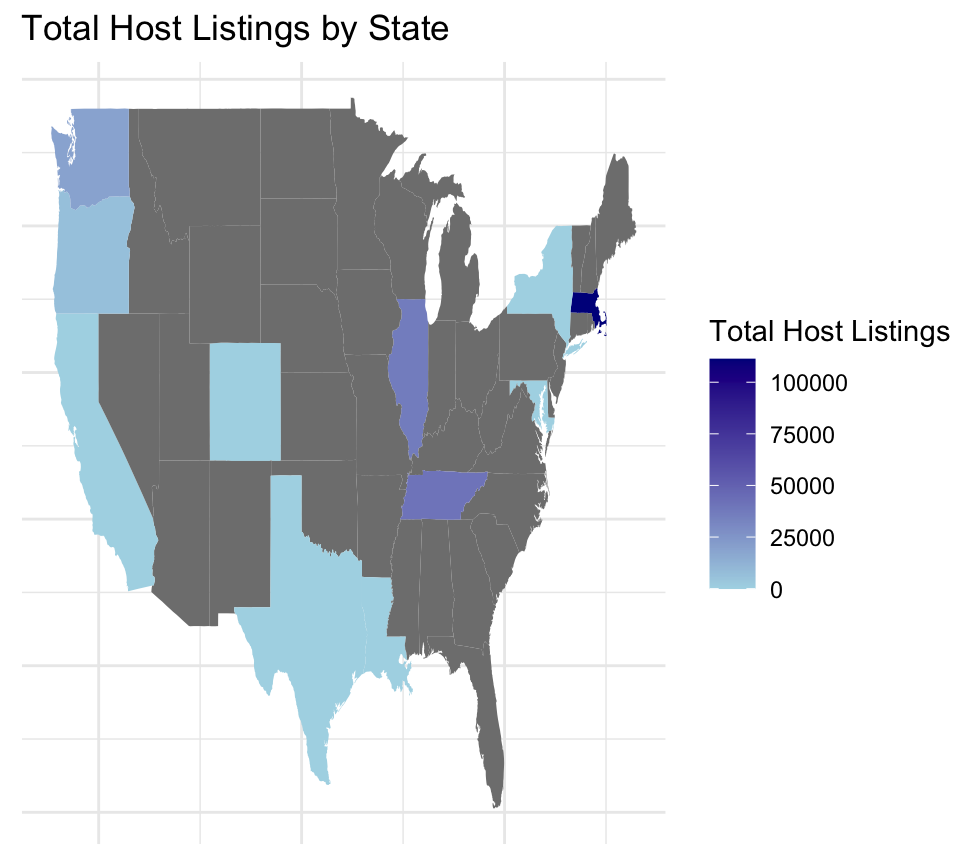
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Feature Name** | **Brief Description** | **R Code Line Numbers** |
| 1 | accommodates | Original | 38 |
| 2 | amenities | Amenities are represented as data frame | 211 - 240 |
| 3 | bath\_per\_bedroom | Number of bathrooms per bedroom | 188 |
| 4 | bathrooms | Original | 55 |
| 5 | bed\_category | Type of bed (“Real, Bed”, “bed”, “other”) | 96 -97 |
| 6 | bedrooms | Original | 51 |
| 7 | beds | Original | 52 |
| 8 | cancellation\_policy | Original | 47 - 48 |
| 9 | charges\_for\_extra | Charges extra for extra people | 109 -110 |
| 10 | cleaning fee | Original | 41 |
| 11 | first\_review | Number of days since the first review | 85 |
| 12 | has\_cleaning\_fee | Has a cleaning fee | 93 |
| 13 | has\_min\_nights | Has a minimum number of nights a renter needs to book to rent the unit | 134 -135 |
| 14 | has\_security\_deposit | Whether there is a mandatory security deposit or not | 179 -182 |
| 15 | host total listings count | Original | 58 - 60 |
| 16 | host\_acceptance | Percentage of bookings a host has accepted | 120 -124 |
| 17 | host\_has\_profile\_pic | Shows whether the Airbnb host has a profile picture | 148 -151 |
| 18 | host\_identity\_verified | Shows whether the Airbnb host’s identity is verified | 161 -164 |
| 19 | extra\_people | Original | 109-110 |
| 20 | availability\_30 | Original | 322 |
| 21 | availability\_60 | Original | 325 |
| 22 | availability\_90 | Original | 324 |
| 23 | availability\_365 | Original | 326 |
| 24 | minimum\_nights | Original | 134-135 |
| 24 | host\_is\_superhost | Shows whether the Airbnb host is a “super host” | 138 -140 |
| 25 | host\_response | Shows whether the Airbnb host has responded to users | 127 -131 |
| 26 | host\_response\_time | Original | 127 -131 |
| 27 | host\_since | Number of days since they became hosts | 80 -83 |
| 28 | host\_verifications | Shows if the Airbnb host has verifications/is verified | 246 - 267 |
| 29 | house\_rules | Shows if the Airbnb host has house rules | 270 -305 |
| 30 | instant\_bookable | Shows if the Airbnb listing can be instantly booked | 143 - 145 |
| 31 | is\_location\_exact | Shows if the Airbnb listing’s exact location is showing | 155 -158 |
| 32 | latitude | Original | 328 |
| 33 | longitude | Original | 329 |
| 34 | guests\_included | Original | 110 |
| 35 | is\_monthly\_price | Shows if the Airbnb listing has a monthly price | 194 |
| 36 | is\_weekly\_price | Shows if the Airbnb listing has a weekly price | 192 |
| 37 | long\_stay | Shows if the Airbnb listing allows stays of at least 28 days | 202 |
| 38 | market | Original | 70 -72 |
| 39 | maximum\_nights | Original | 44 |
| 40 | num\_amenities | Counts the Airbnb listing’s number of amenities | 175 -176 |
| 41 | num\_of\_features | Counts the Airbnb listing’s number of features | 167 -168 |
| 42 | num\_of\_verif | Counts the Airbnb host’s number of verifications | 171 -172 |
| 43 | ppp\_ind | Indicates whether the price per person is higher than median price\_per\_person. | 113 - 117 |
| 44 | price | Original | 35 |
| 45 | price\_per\_person | Shows the Airbnb listing’s price per person | 90 |
| 46 | price\_per\_night | Shows the Airbnb listing’s price per night | 185 |
| 47 | property\_category | Shows the Airbnb listing’s property type (i.e. Apartment, Hotel, Condo, House) | 100 -106 |
| 48 | room\_type | Original | 67 |
| 49 | same\_nhood | Shows whether the Airbnb Host and listing are in the same neighborhood | 197 -199 |
| 50 | security\_deposit | Original | 75 -77 |

**Section 3.2**

**Word Cloud for ‘transit’:**

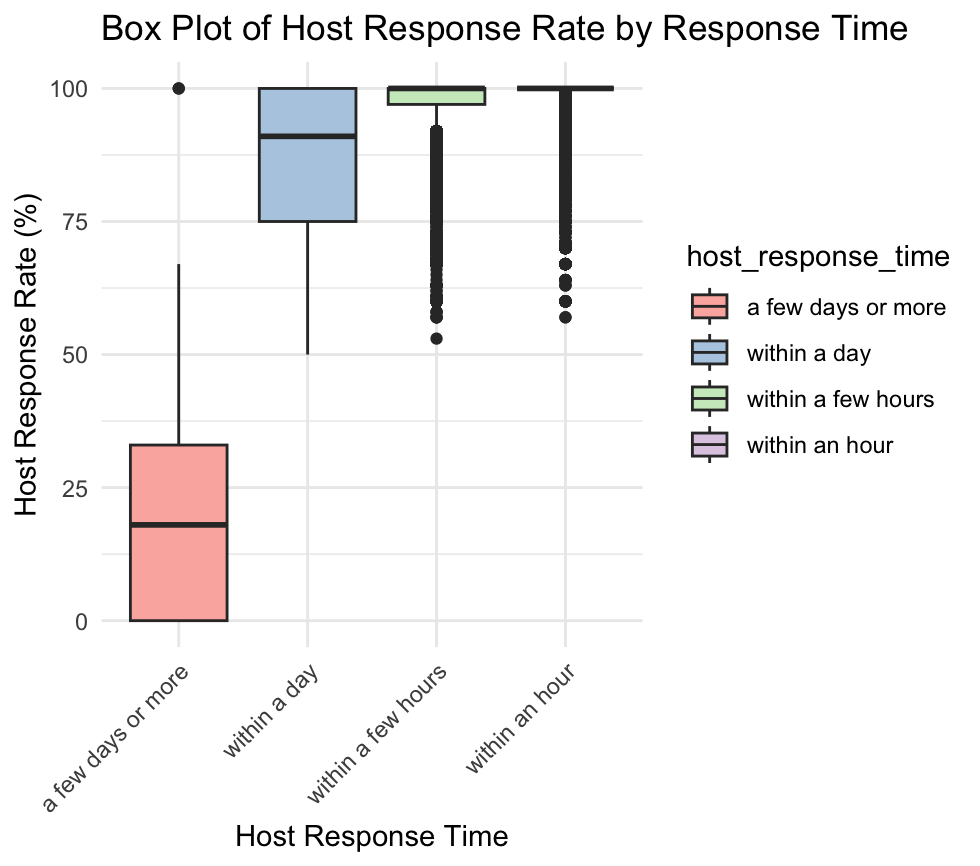


The word cloud above visualizes the frequency of terms in the "transit" column of the dataset. Words like "walk," "subway," and "bus" are prominently featured, indicating that these are common modes of transportation mentioned in proximity to the properties. The emphasis on terms like "downtown," "minutes," and "station" suggests that many properties are conveniently located near major transportation hubs and city centers, highlighting walkability and easy access to public transit as key selling points. The size of each word indicates its prevalence; the larger the word, the more frequent it is in the data. Therefore, a glimpse of the word cloud can convey the trends in the particular column.

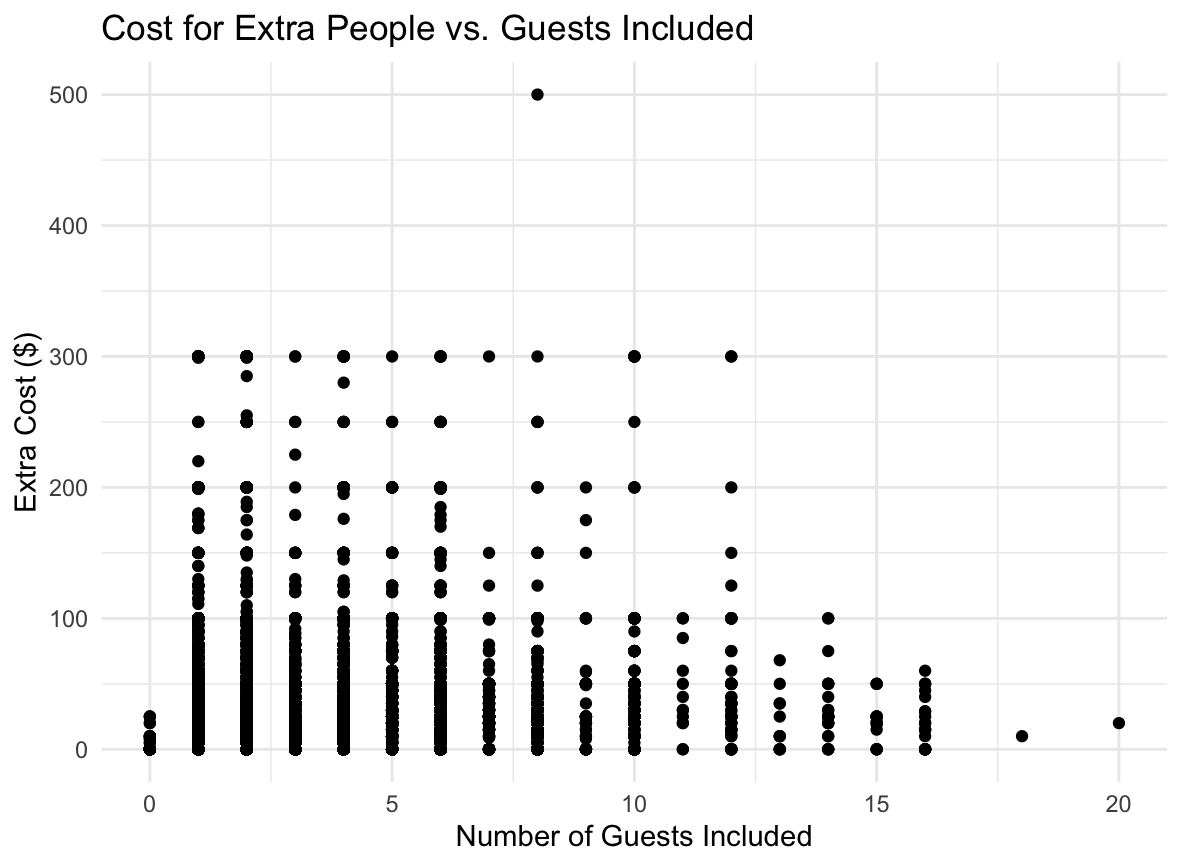
**Geographical representation of Total Host Listings by State:**

The map displays the total host listings by state i.e., the total of all listings made by a host in a particular state; across the United States, using shades of blue to indicate the density of listings. Darker shades represent a higher number of listings, highlighting states like California or New York as likely hotspots with over 75,000 listings. It is important to note that Massachusetts has the highest total listings among all the states. Medium shades suggest moderate listing counts between 25,000 to 75,000, while lighter shades indicate fewer listings, less than 25,000. This visual representation helps identify the popularity of states where hosts list their Airbnb.

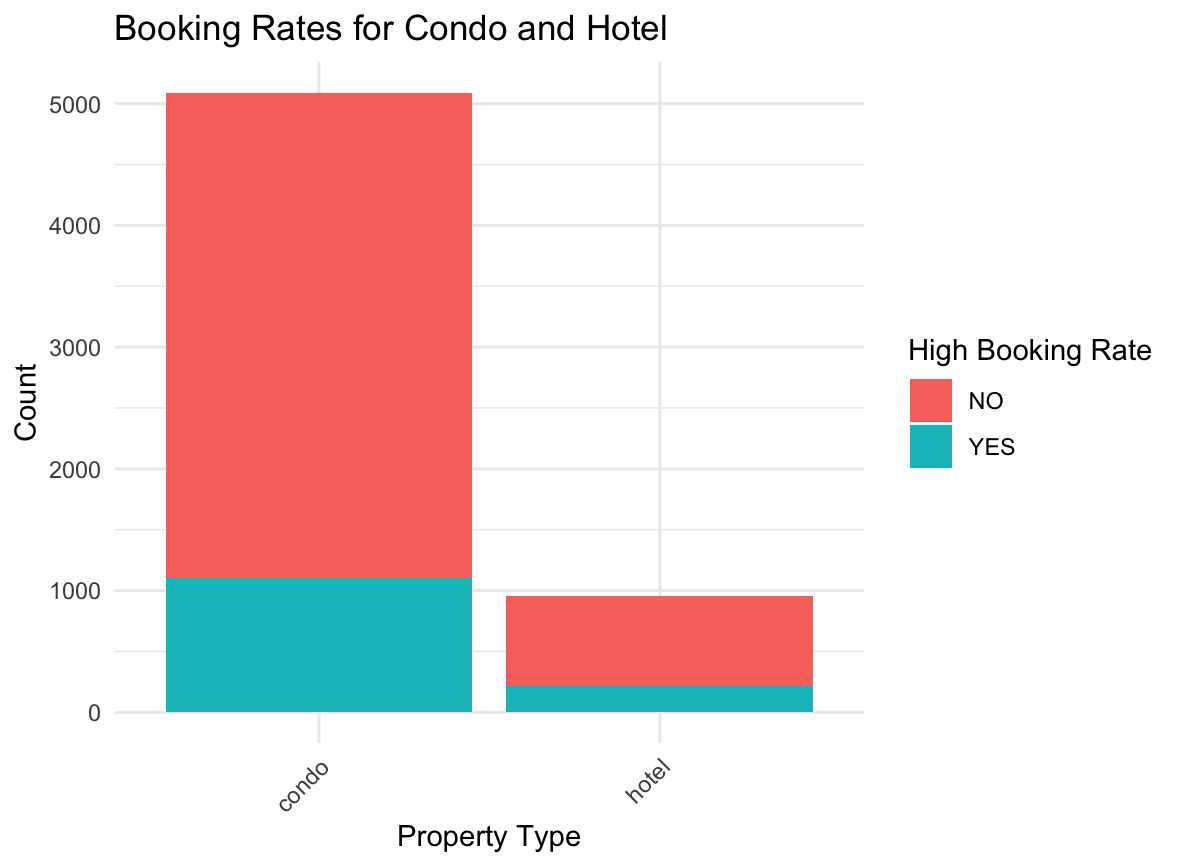
**Boxplot of Host Response Rate(%) & Host Response Time:**



This box plot illustrates the distribution of host response rates across different response times, like "a few days or more" and "within an hour." Hosts who respond "within an hour" typically exhibit higher response rates, while those who respond within "a few days or more" show the lowest and most varied rates. Notably, there are outliers in the faster response categories, indicating that some hosts may not respond occasionally. In summary, hosts who reply "within an hour" or "within a few hours" are more likely to maintain consistent communication. Following these, hosts who respond "within a day" generally have response rates slightly above 85%. For those responding "a few days or more," there is a marked drop, with an average response rate falling below 25%.

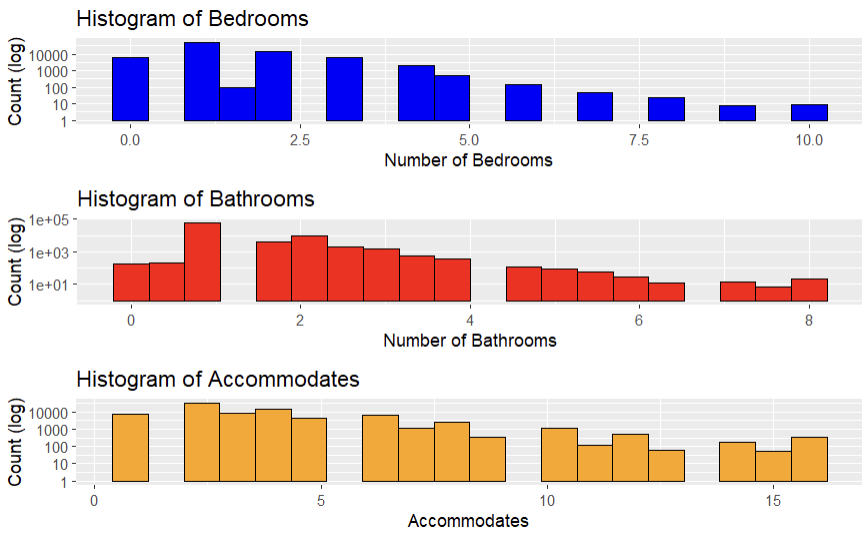
**Scatterplot Comparing Cost for Extra people and the Number of Guests Included:**

This scatter plot illustrates a very interesting relationship between the number of guests included in a booking and the extra cost charged for additional guests. Data points show considerable variation in extra costs, but we see a general increase in extra costs as the number of included guests rises. Despite this trend, there's a wide range of costs at each level of included guests, highlighting inconsistencies or varying pricing strategies among listings. The plot also shows that while some listings have no extra cost for additional guests, others charge significantly, especially as the number of included guests increases.

**Bar chart representing High Booking Rate with Property Type:**

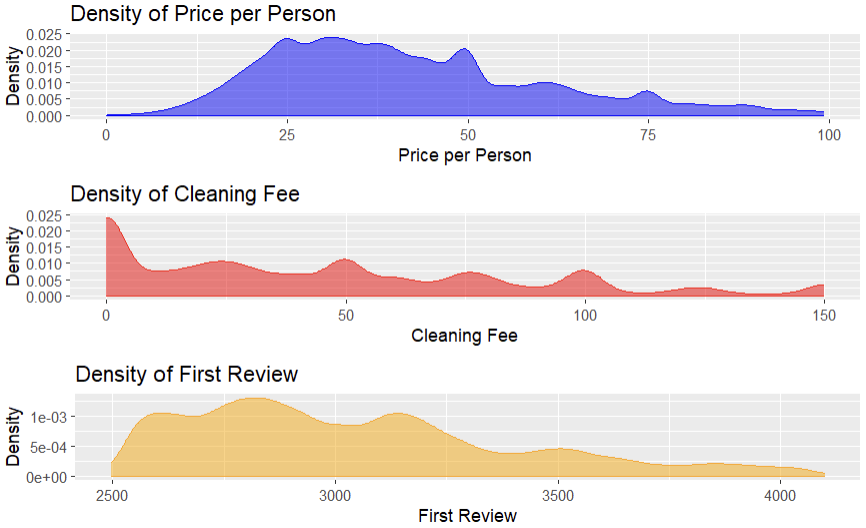
The two bar charts compare the high booking rate for different property types. The first chart shows property types like "apartment" and "house", which are the most prevalent, with a significant number of listings with high bookings rate ("NO"). The second chart zooms in on the "condo" and "hotel" categories specifically, highlighting that even here the majority of listings are ("NO"), with a small proportion of them having a high booking rate ("YES"). This indicates that although there is a difference in the number of records for each of the categories, proportionally for a high booking rate to a low booking rate, "apartment," "house," and “condo” are one fifth. The barchart is intentionally separated owing to the number of records for each category & to make the chart visually appealing.

**Histograms comparing Bedrooms, Bathrooms, and Accommodates (3 Graphs):**



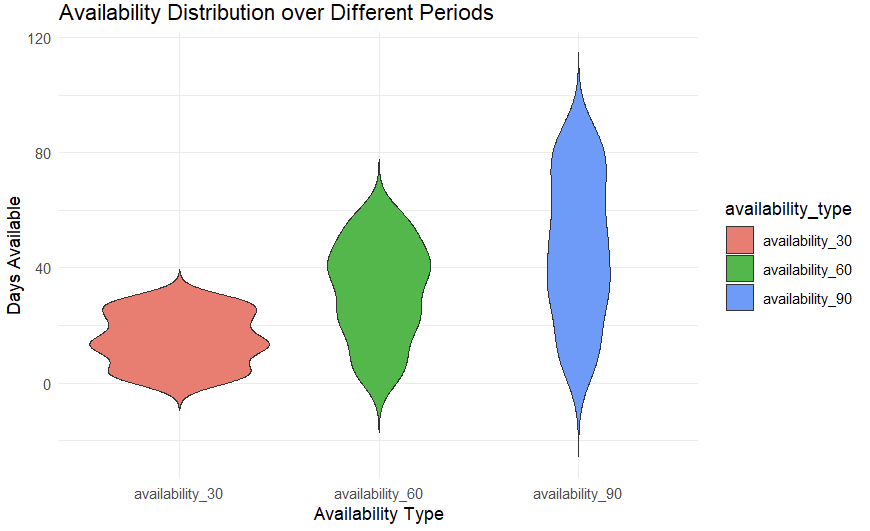
The comparison of these histograms highlights that the most common listings have between 1 or 2 bedrooms and there are fewer listings with 2 or more bedrooms. Listings with one bathroom are in the majority and fewer listings have more bathrooms than 1. Therefore, properties with 1 or 2 bedrooms and 1 bathroom dominate the market. In terms of accommodation capacity, options for 2 to 4 people are more popular, indicating that couples and small families are the primary market segment. Fewer people opt for larger accommodations in the data.

**Density Graphs comparing Price per Person, Cleaning Fee, and First Review (3 Graphs):**



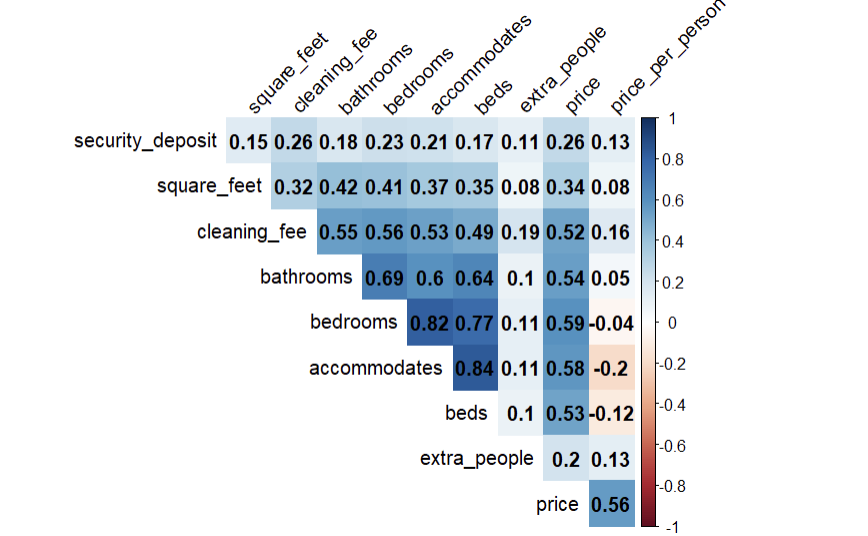
From the three density plots above, there are noticeable peaks in the density of price per person. Firstly, the density of price per person is heavily skewed towards the right , implying that it is quite affordable per person. We can also observe that the density of cleaning fee is more common under $50 and declines as the cleaning fee increases. This indicates that there are moderate fees for cleaning. Lastly, the density of the first review data, shows that most listings are newer since the first review that they got came a few years ago.

**Violin Graphs of Availability Distribution over Different Periods:**

****

The three violin plots show the number of days listings are available over different time periods. For the short-term (30 days), the chart shows a wide range of availability, meaning some listings are available most of the month, while others are not. The medium-term (60 days) chart shows less variation, with more listings being available throughout 60 days. For the long-term (90 days), most listings have most days available.

**Correlation Matrix:**



From this correlation matrix, there is a strong positive correlation of 0.82 between the number of bedrooms and the accommodation capacity, indicating that the number of bedrooms is a good predictor of how many people the Airbnb can host. Another strong positive correlation of 0.69 between bedrooms and bathrooms suggests that as the number of bedrooms increases, so does the number of bathrooms. On the other hand, the extremely low correlation of 0.1 between the number of beds and extra people and bathrooms suggest that these features do not significantly impact the allowance for extra people. The slightly negative correlation of -0.2 indicates that the number of people an accommodation can host does not increase the price per person.

**Section 4:** Evaluation and Modeling

**Section 4.1:** We tried six different models for evaluating the AUC score but our team’s final model was XGBoost. After extensive hyperparameter tuning and grid search cross validation, the XGBoost model consistently outperformed other algorithms in terms of AUC score. Its estimated training performance was 0.8574, with a generalization performance of around 0.8248 on validation data. Finally, we also checked its performance on the 5% data we had kept aside and its AUC score came out to be 0.822. XGBoost is an ensemble model that employs boosting techniques to average multiple weak classifiers. A state-of-the art model , XGBoost has superior performance and the ability to effectively capture the complexity of the dataset while avoiding overfitting. We found that our target variable had class imbalance, with negative to positive classes in the ratio 4:1. We tuned the XGBoost model to assign 4 times higher weightage to the positive class by setting the parameter scale\_pos\_weight = 3.9. This solved the problem of class imbalance and our model started showing improvements in AUC score.

The final decision to use XGBoost was due to the fact that it showed better results on the unseen test data. While some of the other models had a higher validation AUC score, the XGBoost model consistently performed the best on the test data over the course of our multiple contest submissions.

**Section 4.2:**

**Set of features used:** accommodates,bedrooms,beds,cancellation\_policy,cleaning\_fee, host\_total\_listings\_count, price, ppp\_ind, price\_per\_person, property\_category, bed\_category,bathrooms,extra\_people,host\_acceptance,host\_response, host\_response\_time,availability\_30,availability\_60,availability\_90,availability\_365,num\_of\_features,minimum\_nights,market,host\_since,first\_review,host\_is\_superhost,instant\_bookable, latitude,longitude, guests\_included

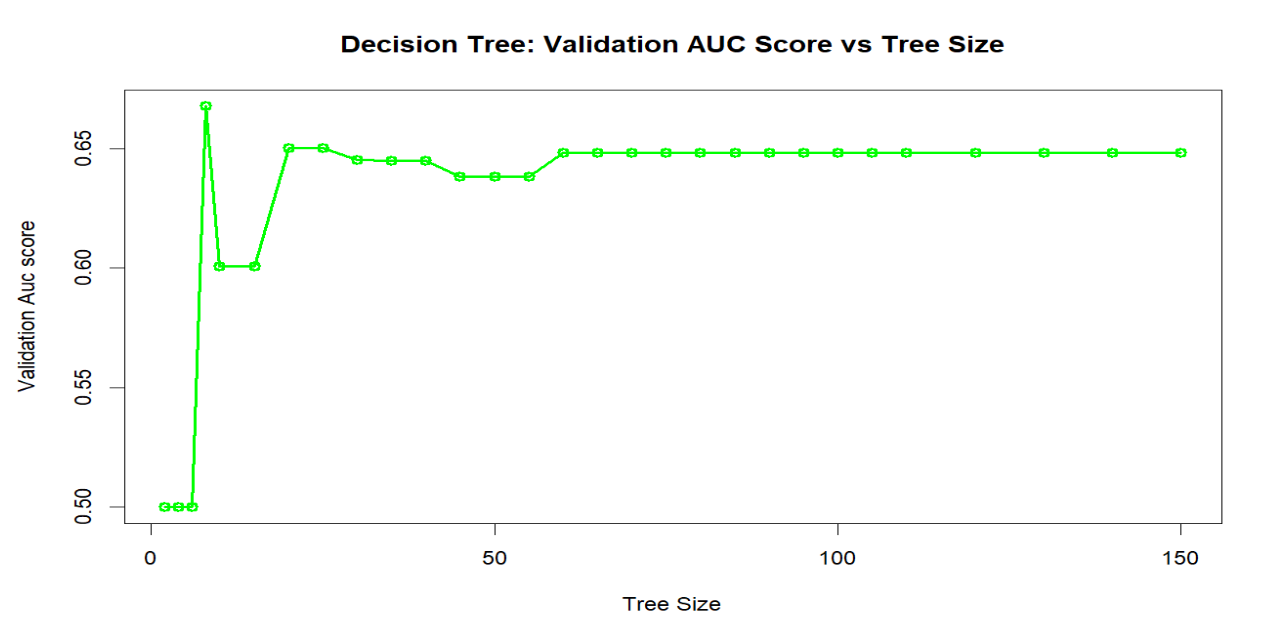
**Extra features tried:** log\_price, log\_accomodates, log\_max\_nights, has\_min\_nights, has\_cleaning\_fee, has\_security\_deposit, is\_location\_exact, room\_type, host\_identity\_verified, host\_has\_profile\_pic, charges\_for\_extra, maximum\_nights, long\_stay, transit, neighborhood\_overview, access, is\_weekly\_price, is\_monthly\_price, weekly\_price, monthly\_price, num\_amenities, num\_of\_verif, price\_per\_night, bath\_per\_bedroom, same\_nhood

Model 1: Logistic Regression

* Type: Unregularized logistic regression
* Library: glm
* Performance:
  + Training AUC: 0.8448
  + Validation AUC: 0.8465
  + Test AUC: 0.827
* Train (70%) ,Validation (25%) , Model checking (5%)
* Line numbers: 363-370

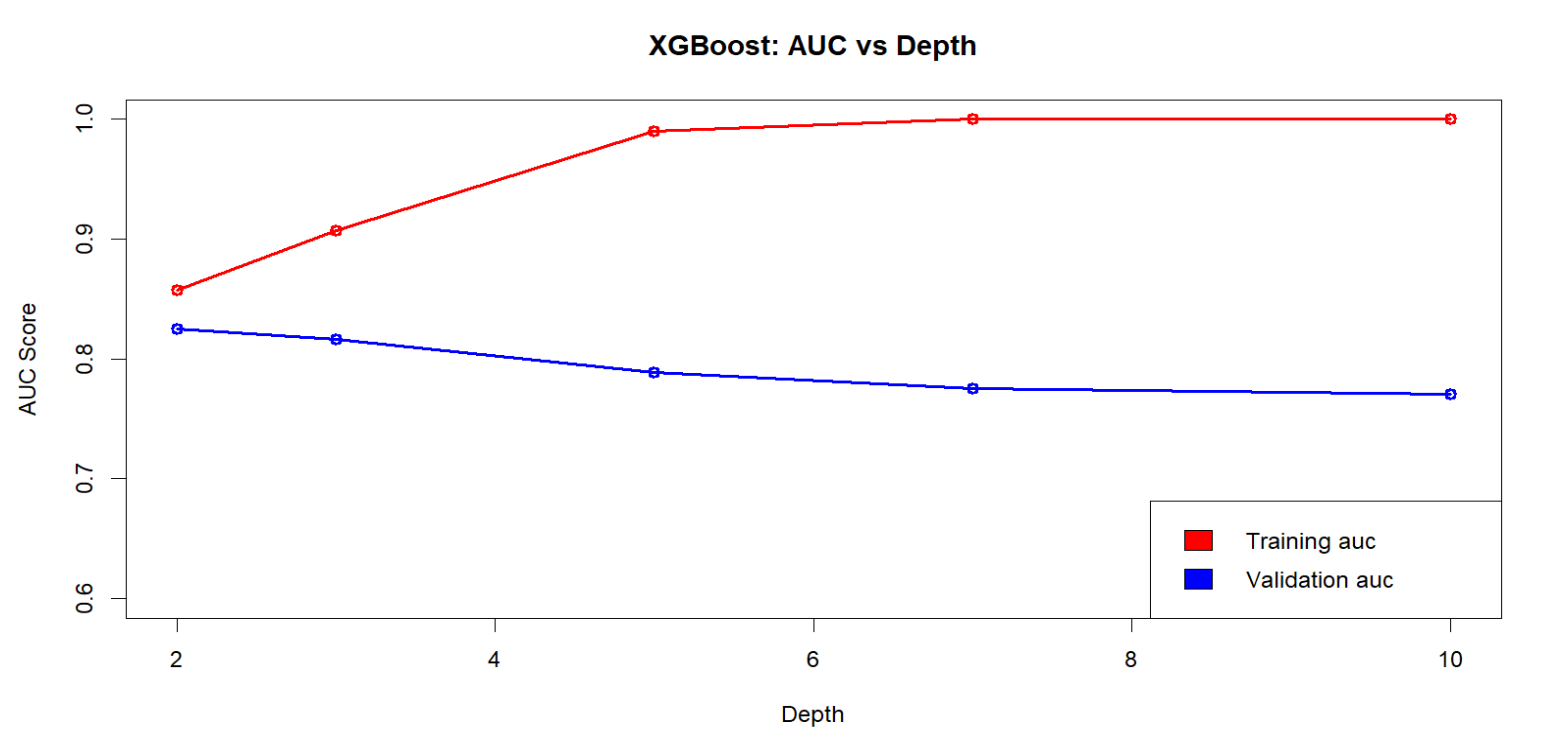
Model 2: Decision Tree

* Type: Decision Tree Classifier
* Library: tree
* Performance:
  + Training AUC: 0.6667
  + Validation AUC: 0.6678
  + Test AUC: 0.734
* Train (70%) ,Validation (25%) , Model checking (5%)
* Line numbers: 375-419
* Hyperparameters tuned:
  + mincut = [1,2,3,4,5,10]
  + minsize = [10,20,30]
  + mindev = [0.0005, 0.0001, 0.0002, 0.001]
  + tree\_size = [2, 4, 6, 8, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100, 105, 110, 120, 130, 140, 150]
* Fitting Curve: AUC score vs Tree Size



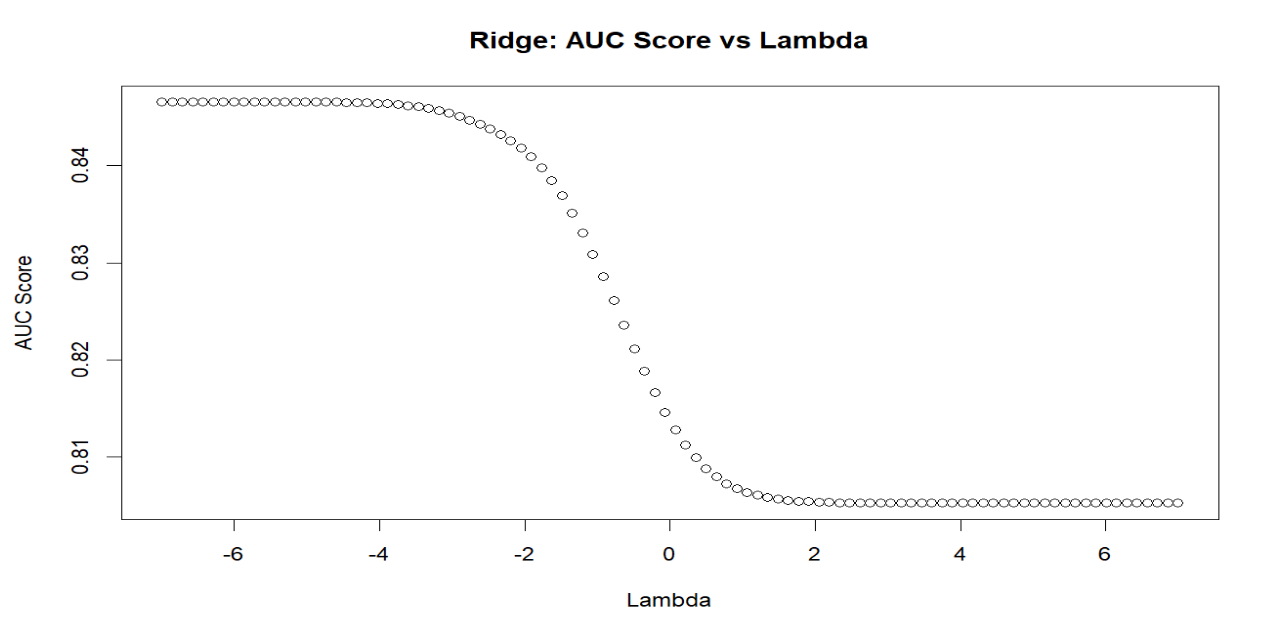
Model 3: XGBoost

* Type: Boosting model
* Library: xgboost
* Performance:
  + Training AUC: 0.8574
  + Validation AUC: 0.8248
  + Test AUC: 0.902
* Train (70%) ,Validation (25%) , Model checking (5%); grid search cross validation
* Line numbers: 508-521
* Hyperparameters tuned:
  + eta = [0.1,0.3,0.5,0.7,1]
  + nrounds = [400,500,600,800,1000]
  + max.depth = [2,3,5,7,10]
  + scale\_pos\_weight = 3.9
  + early\_stopping\_rounds = 5
* Fitting Curve: AUC score vs Depth



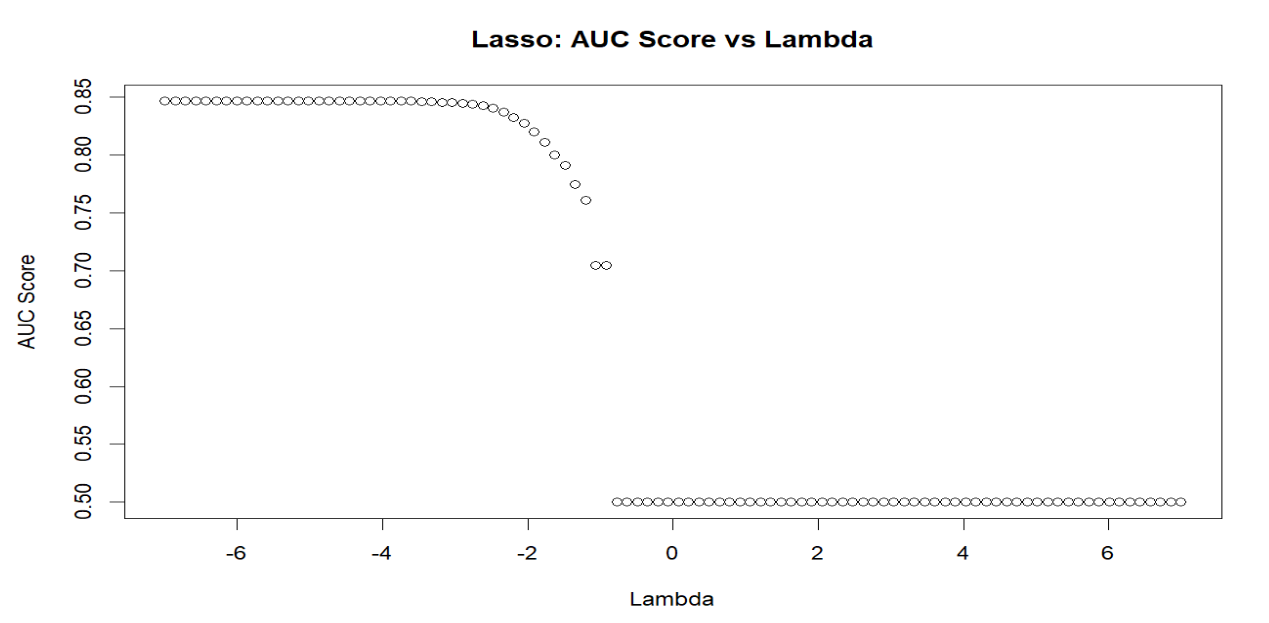
Model 4: Ridge

* Type: Regularized Logistic regression
* Library: glmnet
* Performance:
  + Training AUC: 0.8451
  + Validation AUC: 0.8466
* Train (70%) ,Validation (25%) , Model checking (5%); grid search cross validation (lambda ranging from 10-7 to 107 )
* Line numbers: 562-567
* Hyperparameters tuned:
  + lambda = [10-7 - 107]
  + alpha = 0
* Fitting Curve: AUC Score vs Lambda



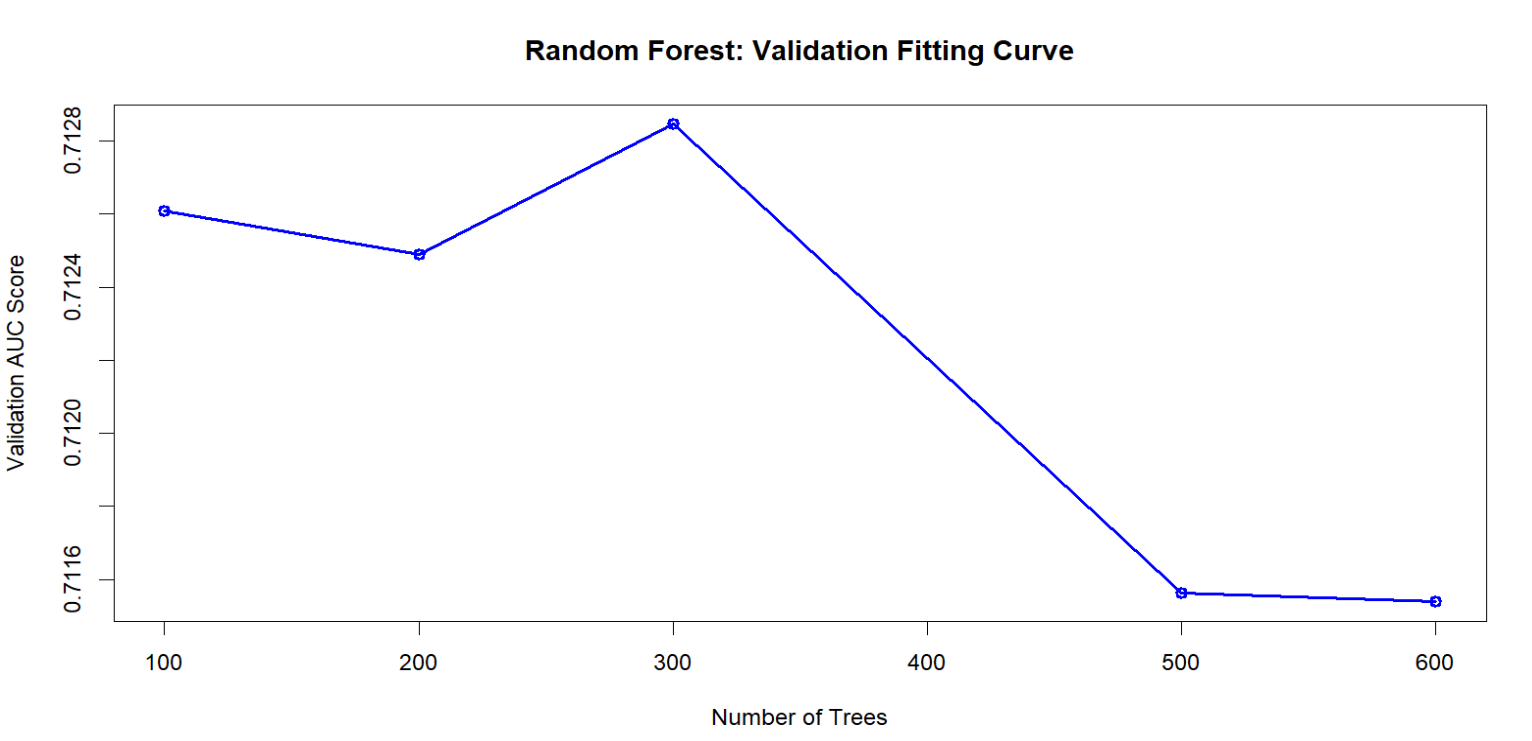
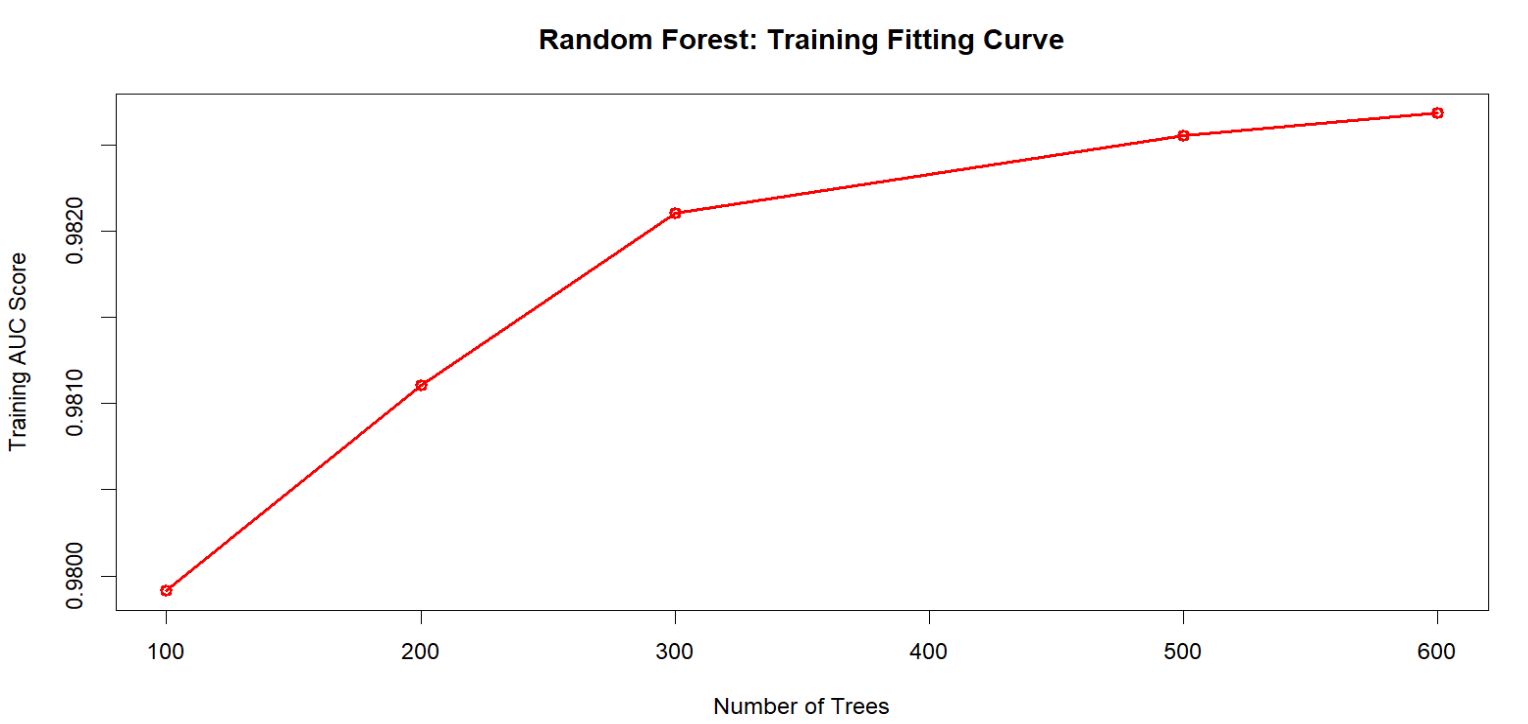
Model 5: Lasso

* Type: Regularized Logistic regression
* Library: glmnet
* Performance:
  + Training AUC: 0.845
  + Validation AUC: 0.8463
* Train (70%) ,Validation (25%) , Model checking (5%); grid search cross validation
* Line numbers: 596-601
* Hyperparameters tuned:
  + lambda = [10-7 - 107]
  + alpha = 1
* Fitting Curve: AUC Score vs Lambda

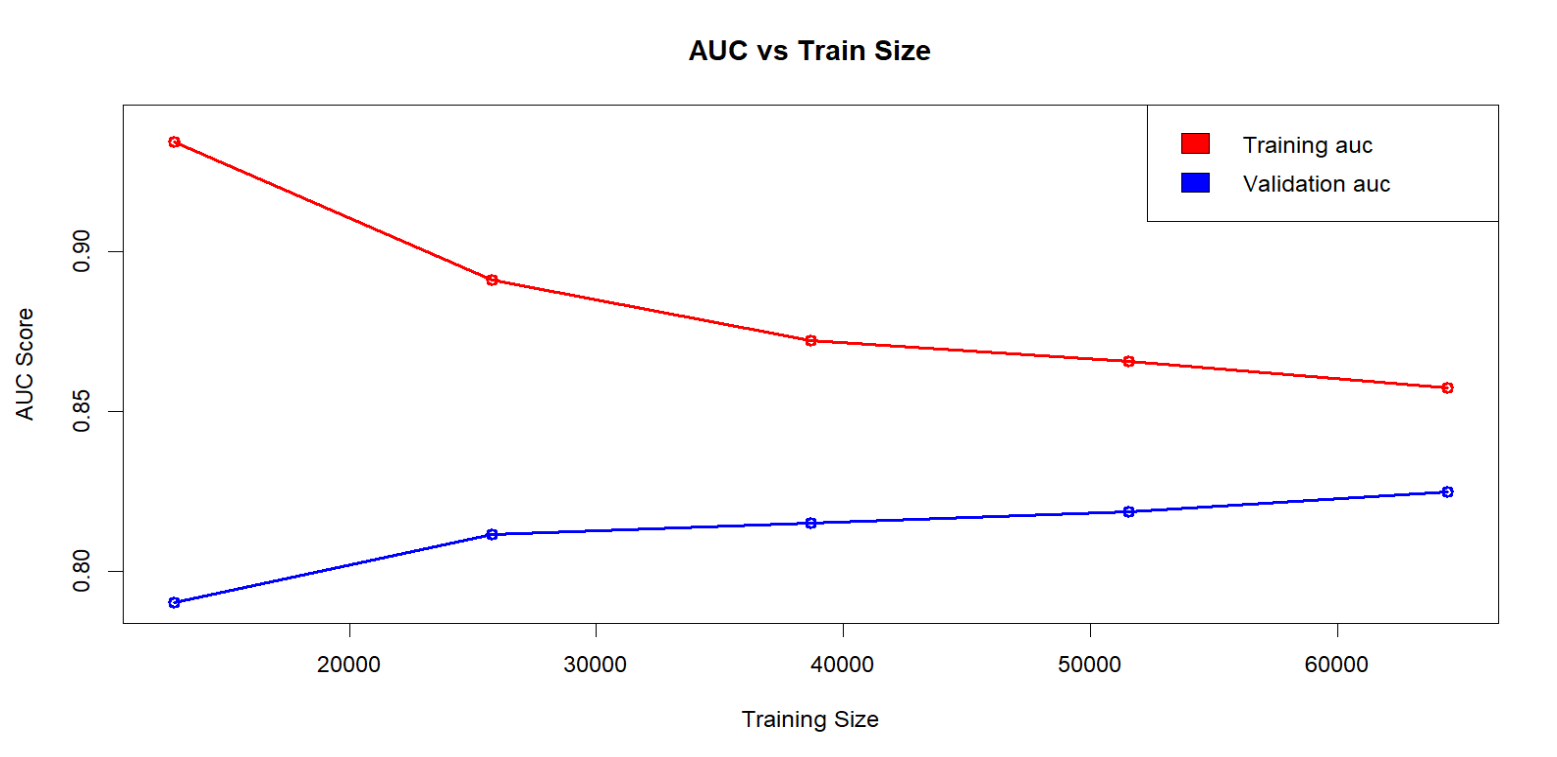


Model 6: Random Forest

* Type: Ensemble method: Bagging
* Library: ranger, randomForest
* Performance:
  + Training AUC: 0.982
  + Validation AUC: 0.712
* Train (70%) ,Validation (25%) , Model checking (5%); grid search cross validation .
* Line numbers: 666 -676
* Hyperparameters tuned:
  + mtry= [10,15,20,30,40,50]
  + num.trees=[100,200,300,500,600]
  + importance=TRUE
* Fitting Curve: AUC Score vs num.trees



**Section 4.3: Learning curve**

For creating the learning curve, we took batches of the training data and evaluated the generalization performance. As evident from the graph, the training performance decreased and the validation performance increased with training size. We noticed that as the training size increased, there was a tradeoff between complexity and performance, which was evident from the fact that the training and validation performances came closer to each other.

**Section 5:** Reflection/Takeaways

Our team, Group 21, worked very diligently to receive first place for the AUC competition. Despite receiving the highest score, this was no easy feat, as our team had to apply each individual member’s diverse strengths in order to maintain team chemistry and create optimal outputs. Each member had vastly different strengths, ranging from R coding and creating visualizations, to even “liberal arts” skills such as essay writing. This project/competition helped our team strengthen our team-building and managerial strategies, and showed us how to work with people with diverse backgrounds and perspectives. Our team’s commitment to strengthening team chemistry was pivotal in reaching our goal of being first place, despite encountering challenges throughout the competition.

Our team has a plethora of strengths which allowed us to work well together and eventually reach first place in the AUC competition. The main two strengths include fostering an inclusive environment and optimizing members’ diverse skills/strengths. Regarding the first strength, our team created an environment where members feel included and are allowed to be vulnerable. In each meeting, every member is allowed to speak their mind and have healthy disagreements, and everyone’s needs are continually met. With this strength, group members were able to be honest and vulnerable about their skills, strengths, weaknesses, and time conflicts if any, without fear of judgement. This vulnerability and honesty made it easier to not only connect with members on a personal level, but also divert roles and strategize for future deadlines, and especially, utilize each members’ individual skills/strengths.

When it comes to utilizing each members’ individual skills/strengths, our team utilizes each member’s strengths in order to create optimal outputs and optimize efficiency. This also connects with our first strength, as every member is to be included, even if he/she is not strong in a particular skill. With this, a pivotal example comes to mind. One of the members opened up about their struggles with R coding, but explained that they could still make attempts at coding and even writing many parts of the report if necessary. All of the members were understanding of this, and allowed this member to utilize their potential, even if said potential disregarded majority coding-related tasks. Due to this, our team made the necessary role diversions to make sure that this member’s tasks were more oriented toward their skills. Utilizing each member’s diverse skills and fostering an inclusive environment helped us reach our goal of being first place

While our group had solid strengths which allowed us to reach first place, this was no easy feat, as two challenges arose in this process. The first challenge was our members not connecting quickly enough, but this was only an issue for the first two weeks. This was mostly due to the fact that only three out of five members knew each other, and it took some time to meet the other members and understand their strengths and weaknesses. The second challenge we faced were the issues with the dataset. The data had many missing values, collinearity among variables, and class imbalance.We employed extensive feature engineering, hyperparameter tuning and modeling techniques to deal with these issues.

If our team could start the project all over again, we would communicate more in-person in order to increase team bonding and connection. Our team was close to perfect, as we all accommodated each other well and made sure everyone was included in major decisions. If we had more months for this project, we would have taken more time to connect with each other and also not rush as much with the deliverables. Now, when it comes to advice for future groups, specifically regarding the deliverables, our advice is to take some time to understand the features in the dataset before jumping into building models. Also, make sure to strengthen connections and utilize member strengths for optimal outputs and results.