UNIVERSITY OF MARYLAND
ROBERT H. SMITH SCHOOL OF BUSINESS

BUDT 758T
Final Project Report
Spring 2023

Section 1: Team member names and contributions

List all team members and summarize what they did to contribute to the project.

Ankeet Singh – Code refinement, data interpretation, feature engineering, running validations on test data

Anwesha Gupta - Reporting, formatting, data interpretation, running validations on test data

Priyanka Khot – Calibrating logistic regression model, feature engineering, running validations

Riya Sharma – feature extraction, Text mining, removing majority NA's, Calibrating Random Forest Model

Soham Dasgupta - Calibrating XgBoost model, Model Selection, Data Interpretation, removing majority NA's

Section 2: Business Understanding

This section should comprise a 1-2page executive summary laying out a business case for your model. Think about what type of person and/or business could use the model, what business actions could be taken based on the output of the model, or what value your predictions could generate.

Answer: The Airbnb platform is a disruptive innovation of our time, in a race to ace the hospitality and management business. Every parameter that affects the customer's sway counts and with the XGBoost we have created, with around 50 variables, could be a game changer. Our model predicts a high booking rate and uses a variety of variables that can affect how customers view properties to rent. The first aspect of booking is looking at the availability and price, if the rent is to be shared, what is the precise price per person, cleaning fee, property category, etc. For avid travel planners, more information is a goldmine that they are willing to explore and therefore we have taken a considerable number of features to keep them invested.

Churning out a high booking rate can help with price regulation and price optimization of the property. This model can traverse through historical booking data, and look into seasonality and other factors to optimize prices for different properties, considering its location and amenities. This would help Airbnb to maximize its revenue and put up competitive prices considering the demand and influx of tourists at that time.

With our model, we can also dive into demand forecasting where availability, host acceptance, and easy transit, i.e. features of the locations heavily influence a customer's decision of booking the property. Locations at certain times of the year will drive the prices up and one can also charge

a preference fee for booking those locations, this boosts occupancy rate and lowers the vacancy periods. This will result in a surge in profits during peak visiting season.

As a platform that pioneers booking, optimization of booking rates is irrevocably the single most crucial factor for revenue growth. An XGBoost model can help hosts determine the appropriate minimum and maximum length of stay, cancellation policies, and rules for their listings. Such booking policies enhance the guest experience, attract more bookings, and reduce the likelihood of cancellations.

While renting out properties, customers prefer their space and are often upset(which results in negative online reviews) when they feel otherwise with features like host_total_listings_count, host_is_superhost, host_identity_verified, host_response, host_has_profile_pic, and requires_license, our model will be able to identify high performing hosts, promote super hosts and further cement the consumers trust in the platform.

Recommendations in the search bar are usually the first place a customer looks into when trying to search for the best fit for their stay, with accommodates, beds, bedrooms, bathrooms, bed_category, and property_category, as features our XGBoost model can recommend properties to potential guests within budget, and desired amenities. This will help increase the booking conversion rates.

In the end, no one likes a wolf in sheep's clothing. Our model can identify patterns associated with fraudulent listings or hosts, flagging them for further investigation, due to some of the features that we are using: like, host_total_listings_count, host_identity_verified, and requires_license.

Section 3: Data Understanding and Data Preparation

This section should include:

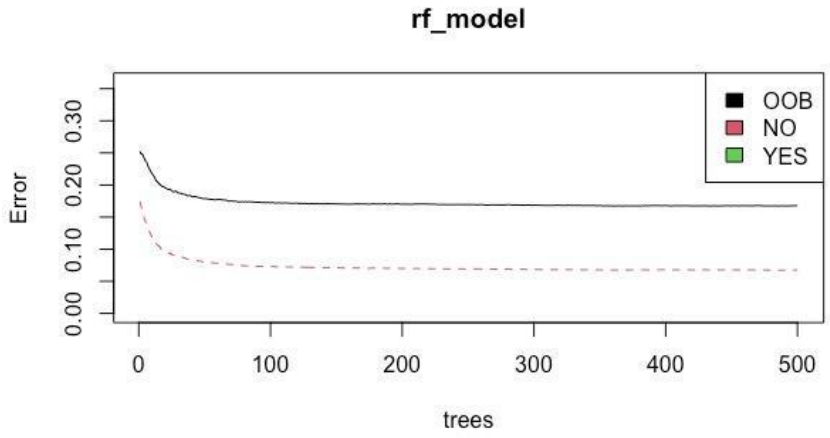
1) *A table of the following format (for example):*

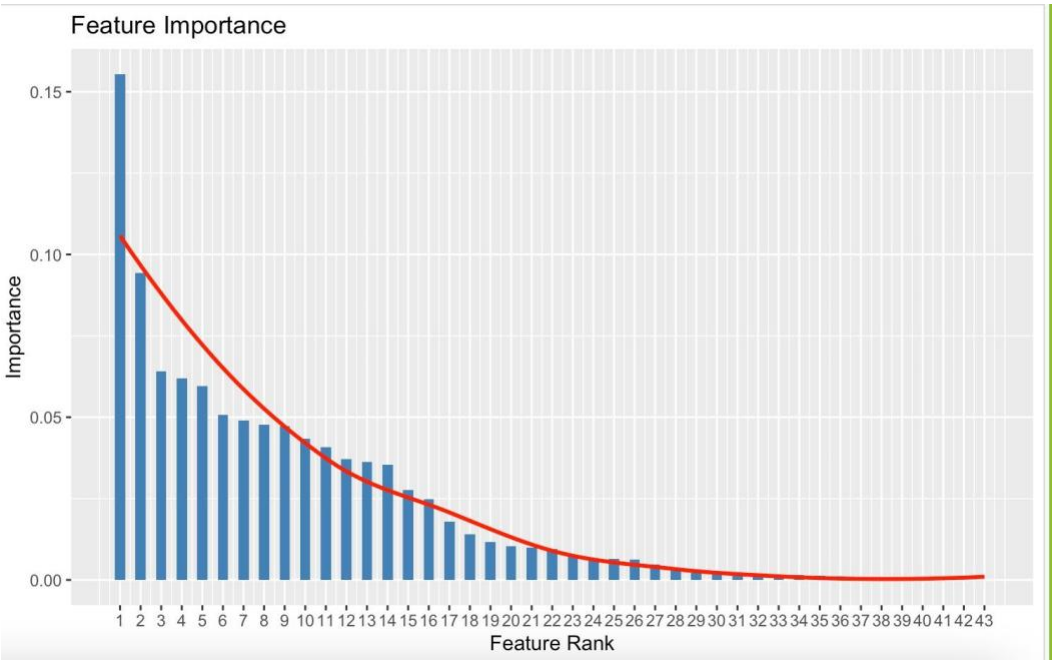
ID	Feature Name	Brief Description	R Code Line Numbers
	accommodates	Original feature from dataset	27
	bedrooms	Original feature from dataset	29
	beds	Original feature from dataset	32
	has_ cleaning_fee	Original feature from dataset	36
	host_total_listings_count	Original feature from dataset	37
	price	Original feature from dataset	38
	ppp_ind	Created	39
	bathrooms	Original feature from dataset	40
	extra_people	Original feature from dataset	43
	host_acceptance_rate	Original feature from dataset	45
	host_response_rate	Original feature from dataset	48
	has_min_nights	Created from has_minimum_nights (True/False)	52
	host_is_superhost	Original feature from dataset	51
	cleaning_fee	Original feature from dataset	53

	price_per_person	Created price/ accommodates	55
	maximum_nights	Original feature from dataset	60
	minimum_nights	Original feature from dataset	63
	availability_60	Original feature from dataset	72
	availability_90	Original feature from dataset	86
	availability_30	Original feature from dataset	96
	availability_365	Original feature from dataset	111
	zipcode	Original feature from dataset	112
	instant_bookable	Original feature from dataset	113
	is_location_exact	Original feature from dataset	117
	requires_license	Original feature from dataset	118
	license	Original feature from dataset	119
	host_has_profile_pic	Original feature from dataset	120
	host_identity_verified	Original feature from dataset	121
	security_deposit	Original feature from dataset	122
	has_rules	Created from house_rules (True/False)	123
	easy_transit	Created from transit for NA is False	124
	notes	Created from notes from text to binary field	125
	interaction	Transformed from string to binary field	126
	host_response_time	Converted to binary 0 or 1	127
	host_about	Original feature from dataset	128
	apartment	Dummy variable of product_category	129
	hotel	Dummy variable of product_category	130
	condo	Dummy variable of product_category	131
	house	Dummy variable of product_category	132
	other	Dummy variable of product_category	133
	realBed	Dummy variable of bed_category based of bed_type	134
	otherBed	Dummy variable of bed_category based of bed_type	133
	high_booking_rate	Original feature from dataset	135
	access	Converted to binary	136

	cancellation_strict	Dummy variables for cancellation_policy	137
	cancellation_moderate	Dummy variables for cancellation_policy	138
	cancellation_flexible	Dummy variables for cancellation_policy	139
	cancellation_no_refunds	Dummy variables for cancellation_policy	140
	cancellation_super_strict_30	Dummy variables for cancellation_policy	141
	cancellation_super_strict_60	Dummy variables for cancellation_policy	142
	cancellation_no_policy	Dummy variables for cancellation_policy	143
	total_verifications	Derived from unstructured text field	145
	Sentiment_intensity	Score from sentiment library	144
	Description_Length		146
	total_amenities	Derived from unstructured text field	147

2) *Graphs or tables demonstrating useful or interesting insights regarding features in the dataset.*
For XGBoost





3) *(optional) Any additional comments about the data or the steps you undertook for data preparation.*

Section 4: Evaluation and Modeling

1) *Include a short (one-paragraph) description of the winning model, the variables included in the model, your estimated training and generalization performance, and how you decided that it was the winning model. Also, list the line numbers in your R code where you generated the final predictions that you submitted for the contest.*

Answer:

Winning Model and Why: Xgboost, out of all the models we exceuted the complexity and accuracy given by Xgboost was the highest and we went one step further and added a layer of text mining, we added a sentiment intensity score that was computed for each description using the sentiment intensity analyzer from the sentiment library. To normalize the sentiment intensity, the score is divided by the total number of words in the description.

Additionally, the description length was determined by performing a word count. This engineered feature aims to investigate the potential correlation between longer descriptions and higher ratings.

Variables Included: accommodates, bedrooms, beds, has_cleaning_fee, host_total_listings_count, price, ppp_ind, bathrooms ,extra_people , host_acceptance_rate , host_response_rate , has_min_nights , host_is_superhost , cleaning_fee , price_per_person , maximum_nights , minimum_nights , availability_30 , availability_60 , availability_90 , availability_365 , zipcode , instant_bookable , is_location_exact , requires_license , license , host_has_profile_pic , host_identity_verified , security_deposit , has_rules , easy_transit , notes , interaction

, host_response_time , host_about , apartment , hotel , condo , house , other , realBed , otherBed , high_booking_rate , access , cancellation_strict , cancellation_moderate , cancellation_flexible , cancellation_no_refunds , cancellation_super_strict_30 , cancellation_super_strict_60 , cancellation_no_policy , total_verifications , total_amenities, sentiment_intensity, description length

Estimated Training and Generalization Performance:

Line Numbers R code:

2) *For each type of model that you try, please list:*
   a) *The type of model (i.e. model family – for instance: logistic regression, decision trees, etc.).*

      Answer: Logistic Regression – Regression Model
      Random Forest – Ensemble Method
      XGBoost - Ensemble Method

   b) *The R function and/or library used.*

      Answer: R libraries: caret, tree, class, randomForest, tidyverse, ggplot2, ggthemes, scales, pROC, ROCR, xgboost

      R functions: read_csv, cbind, mutate, as.factor, as.numeric, parse_number, is.na, replce_na, mean, na.rm, group_by, ungroup, median, select, as.character, as.logical, sample, nrow, **glm (family = binomial)**, predict, sum, length, **randomForest**, classifier.predict_prob, plot, legend, importance, data.frame, resample, ggplot, geom_bar, geom_text, labs, coord_flip, theme_few, table, performance, createDataPartition, data.matrix, xgb.DMatrix, summary, xgb.train, list, roc, write.csv, write.table

   c) *Estimated training and generalization performance for this model.*

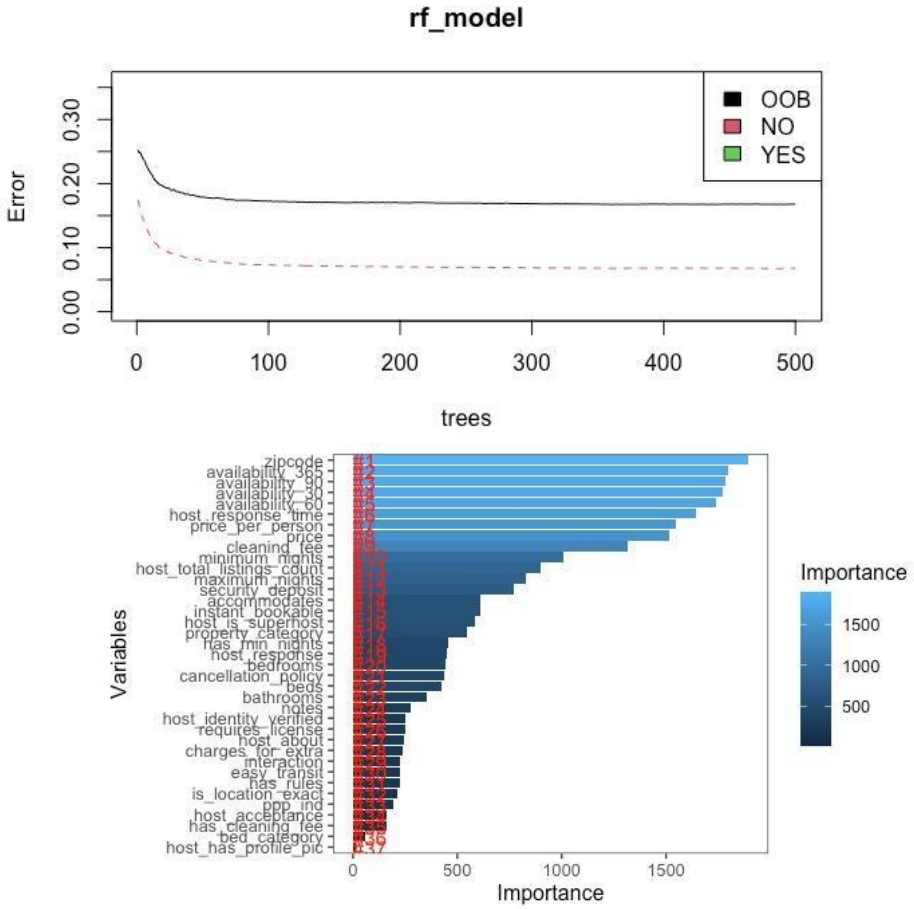
      Answer: Logistic Regression – Regression Model
      Accuracy: 78.995
      Random Forest – Ensemble Model
      AUC: 86.92
      XGBoost - Ensemble Method
      AUC: 93.83

   d) *How you estimated the generalization performance for this model. This should include the methodology used (i.e. simple train/validation split, cross validation, nested holdout) as well as the specific parameters of the validation setup (i.e. how much data, how many folds, etc.).*

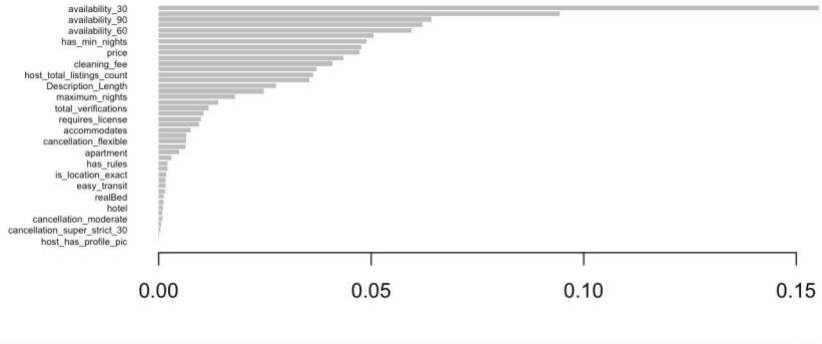
      Answer: For all models, we have done 30 % test data and 70 % training data split, a simple train/validation split.

   e) *The best-performing set of features that you used in the model (you may also optionally note other features that you tried but didn't include in the final feature set).*

For Random Forest



We see the top variables available_365, available_90, avaailble_60,avaailble_30, host_response_person, price, and zip code For XGBoost



```
> xgb.importance(model = xgb_model)
                         Feature         Gain        Cover    Frequency
 1:              availability_30 1.554197e-01 0.0555187562 0.0402470612
 2:                      zipcode 9.432942e-02 0.1409604517 0.1153616258
 3:              availability_90 6.415959e-02 0.0447094051 0.0454273760
 4:               minimum_nights 6.204708e-02 0.0380740597 0.0213189878
 5:              availability_60 5.951412e-02 0.0327407060 0.0334728033
 6:             instant_bookable 5.062144e-02 0.0220241093 0.0129507870
 7:               has_min_nights 4.893445e-02 0.0165018056 0.0099621439
 8:             availability_365 4.767054e-02 0.0552524863 0.0651524208
 9:                        price 4.716192e-02 0.0456811047 0.0615660490
10:            host_is_superhost 4.338311e-02 0.0219150938 0.0131500299
11:                 cleaning_fee 4.089092e-02 0.0618307587 0.0462243475
12:              price_per_person 3.713846e-02 0.0578968273 0.0623630205
13:    host_total_listings_count 3.628290e-02 0.0355510487 0.0442319187
14:              total_amenities 3.539080e-02 0.0502698004 0.0667463638
15:           Description_Length 2.755767e-02 0.0412466376 0.0587766487
16:          Sentiment_intensity 2.475105e-02 0.0398890552 0.0735206216
17:               maximum_nights 1.786660e-02 0.0406067040 0.0312811317
18:             security_deposit 1.397633e-02 0.0239116225 0.0276947599
19:          total_verifications 1.168924e-02 0.0186106477 0.0284917314
20:                 extra_people 1.046271e-02 0.0168965385 0.0276947599
21:             requires_license 9.993930e-03 0.0156924927 0.0073719865
22:                     bedrooms 9.513387e-03 0.0157032654 0.0103606296
23:                 accommodates 7.582721e-03 0.0145495028 0.0145447300
24:                    bathrooms 6.520923e-03 0.0143209279 0.0117553297
25:        cancellation_flexible 6.395487e-03 0.0057600137 0.0055788006
26:                         beds 6.281280e-03 0.0124188952 0.0107591154
27:                    apartment 4.788939e-03 0.0035502556 0.0045825862
28:          cancellation_strict 2.901734e-03 0.0032244975 0.0071727436
29:                    has_rules 2.057749e-03 0.0044084634 0.0051803148
30:                        house 2.036040e-03 0.0013432583 0.0045825862
```

We see a similar set of variables for XGBoost as well, here we have used zipcode as a numeric value since smaller zip codes are associated with older cities. This was one of the unique features of our model.

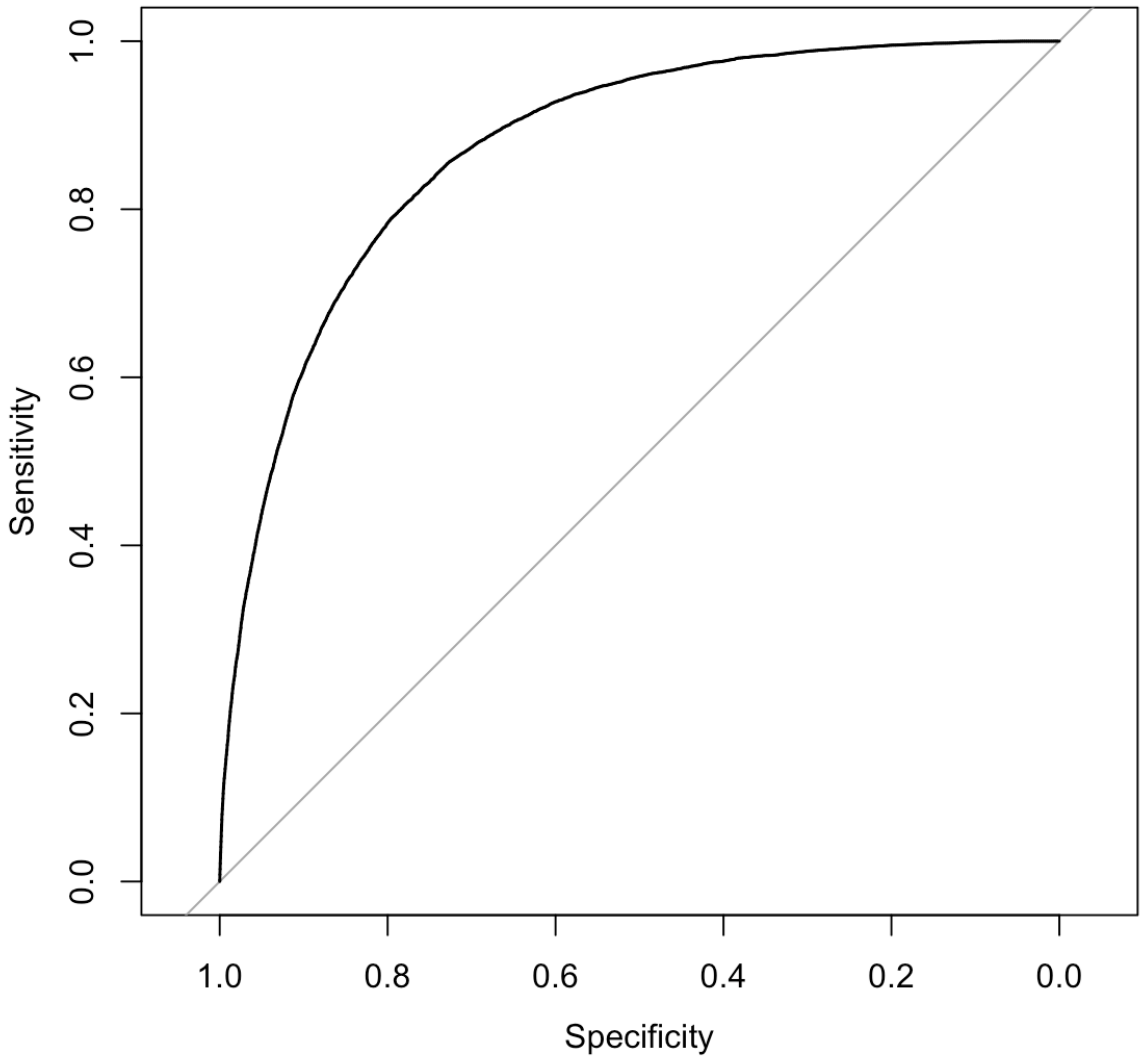
f) The line numbers in your R code where you trained the model and estimated its generalization performance., lef: 267 - 271 line number

g) (Required to receive at least 80 points): Any hyperparameters that you tuned and a list of the values that you tried.

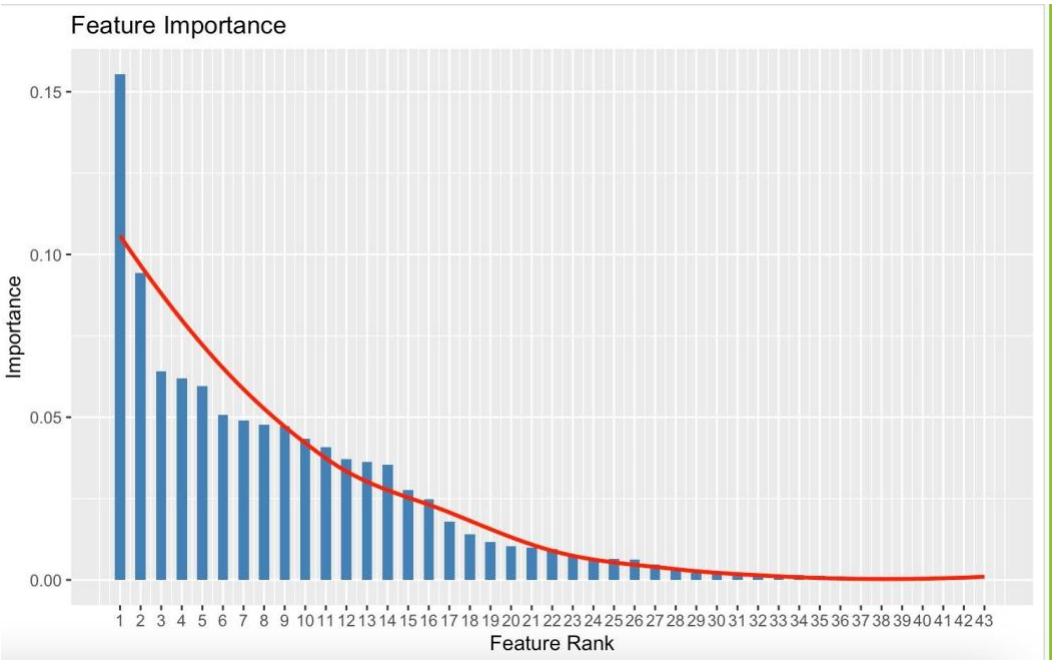
Answer: We have used XGBoost and Random Forest for hyperparameter tuning.

h) *(Required to receive at least 80 points): Any fitting curves that you created for this model.*

Cross validation predictions:



3) *(Required to receive at least 85 points) Also include any learning curves that your team created as well as any insights generated by these curves.*



Section 5: Reflection/takeaways

1-2 pages reflecting on the project. Possible questions that could be answered in this section include (but are not limited to):

1) *What did your group do well?*

Answer Better model selection - At first we struggled with model selection, opting for trial and error instead of perusing through the dataset to understand what each column represents and how it could correlate to the high booking rate. Once we set our sights on featuring engineering, we were able to come up with variables that contribute significantly towards all our models and were finally able to generate competitive predictions.

Interpretations of features - This took a considerable amount of time and effort. Without extensive domain knowledge in this regard finalize the features, mutate them to make NA's non-existent, and run complex models.

Efficient Code Writing - With a complex dataset like this one it was imperative to write clean code so that parts of our code arrange themselves in a coherent compact way. This helped team members pick up very quickly on the updated versions of the code.

2) *What were the main challenges?*

Answer: Data Cleaning - Since we did not know which features were more likely to contribute significantly to the models we held up data cleaning for quite some time, also all the factors needed to be mutated and we ended up spending more time that we had hoped for.

Handling of NA's - Till the very last submission we coped with the troublesome NA's and we hope with the final submission there would not be any need to replace NA's with 0

Feature selection - There were many combinations of features that we tried and spent a significant portion of our time tailoring the right features to get the maximum output.

3) *What would your group have done differently if you could start the project over again?*

   Answer: We would have preferred to get our hands on an external dataset so that our understanding of features would have been precise and significantly improve our predictions and accuracy.

4) *What would you do if you had another few months to work on the project?*
   Answer: Work extensively on feature engineering, model selection, text mining, sentiment analysis, and digging deeper into data wrangling.

5) *What advice do you have for a group starting this project next year?*
   Answer: A Dog is a man's best friend, Boosting is yours.