Predict Hotel Cancellation Based on Logistic Regression, Decision Tree, Random Forest and Gradient Boosting Algorithm

STATISTICAL DATA SCIENCE II

Final Report

Group 9

[Directory]

I. Introduction

With the development of communication technology and mobile phone technology, more and more people use the booking method to book their accommodation in the travel. However, we can also think that due to various reasons (such as sudden situations, finding more cost-effective housing, etc.), the cancellation rate of people after booking is not low.

After canceling the reservation, although it does not have much impact on the tourists, but the hotel or the person who provides the listing must re-list the house, and the act of canceling the reservation will cause them some trouble. Therefore, for hotels or institutions, a demand that can predict the cancellation of bookings will also be born, and this forecast can analyze for them the reasons why people cancel their bookings in order to make better improvement to create higher residual value.

i. Team introduction

In order to best solve the above problems, our team decided to divide the labor as follows:

Composition	Name	Student id
Leader	최인표	2015150214
Analysist	이은표	2014100086
Analysist	백승윤	2015150050
Organizer	이대광	2016150470
Organizer	이수현	2017320117

ii. Research Question

Finding the best model that most accurately predicts whether a reservation is to be cancelled with the data set: “Hotel booking demand”. And searching after which explanatory variables significantly affects explanatory power.

II. Description of Data

i. General introduction

The data set used in this project is "Hotel booking demand" from Kaggle. The source URL of the data set is: https://www.kaggle.com/jessemostipak/hotel-booking-demand.

This data contains data from guest stayed in two different hotels in Portugal, one is in Lisbon and the other locates in rural area. The data contains 119390 observations(*n=119390*), and each observation record represents a hotel reservation, (check-in time is in from July 1, 2015 to August 31, 2017), which include paid reservations and cancelled reservations. And this data has 32 variables (*p = 32*) that contains information about guests such as whether the reservation is canceled or not(*y*), which time of the year they stayed, through whom they booked hotel etc.(*x1,x2,…*).

We can take a look at the data in the first five lines and get a rough idea of how the data is structured.

```
  hotel is_canceled lead_time arrival_date_ye~ arrival_date_mo~ arrival_date_we~ arrival_date_da~
  <chr>       <dbl>     <dbl>            <dbl> <chr>                      <dbl>            <dbl>
1 Reso~           0       342             2015 July                          27                1
2 Reso~           0       737             2015 July                          27                1
3 Reso~           0         7             2015 July                          27                1
4 Reso~           0        13             2015 July                          27                1
5 Reso~           0        14             2015 July                          27                1
6 Reso~           0        14             2015 July                          27                1
# ... with 25 more variables: stays_in_weekend_nights <dbl>, stays_in_week_nights <dbl>, adults <dbl>,
#   children <dbl>, babies <dbl>, meal <chr>, country <chr>, market_segment <chr>, distribution_channel <chr>,
#   is_repeated_guest <dbl>, previous_cancellations <dbl>, previous_bookings_not_canceled <dbl>,
#   reserved_room_type <chr>, assigned_room_type <chr>, booking_changes <dbl>, deposit_type <chr>, agent <chr>,
#   company <chr>, days_in_waiting_list <dbl>, customer_type <chr>, adr <dbl>,
#   required_car_parking_spaces <dbl>, total_of_special_requests <dbl>, reservation_status <chr>,
#   reservation_status_date <date>
```

ii . Exploratory Data Analysis (EDA)

(1) Data Wrangling / Processing

In order to perform better data analysis, we wrangled some variables.

−Delete unnecessary variable:

First, we deleted the ‘country’ variable because it has little effect on cancellation rate. Then we deleted the company variable, because there were too many kinds of

company, and the information is mixed (the distribution of a single value is too much), so that the variable ‘company’ is expected to have very limited effect on cancellation rate. In the same way, we eliminated some components of categorical variable ‘agent’ which have 334 unique value, so that it contains only 11 components.

－Processing Missing value:

Since there were enough sample size and very small missing values, we just excluded the observations that contain missing values.

－Add variables:

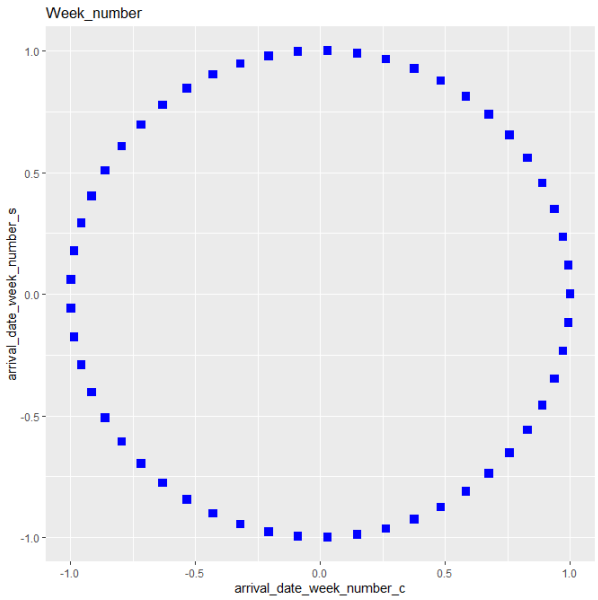
We added some variable to eliminate unnecessary numeric, categorical variables, and components of variable. In original data, there were too many unnecessary components in some variables, so that this may reduce accuracy of prediction and makes it difficult to explain the model. For example, there 4types of customer type but most of them are ‘transient’. So, we added new variables that can explain the original variable in more concise way.

First, we added ‘is_transient’ variable, that makes original variable ‘customer_type’ to binary variable that shows whether the customer is ‘Transient’ or not.

Then, we created a binary variable ‘reserved_is_assigned’, that distinguish whether the room customer get is same as the room they booked.

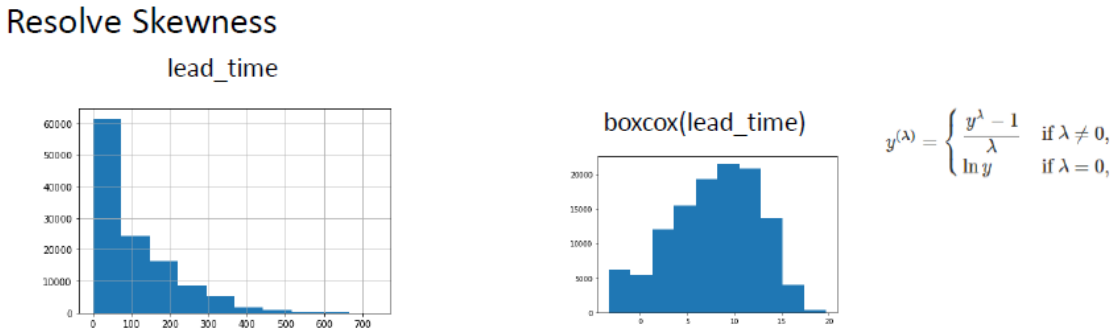
In the similar way we made numeric variables ‘children’, ‘adult’, ‘baby’ which show the number of children, adults, babies in customer group into new variable ‘customer_type’ that shows whether customer is family or couple or single.

We also made ‘week’ variable from the original data into ‘arrival_date_week_number_s’ and ‘arrival_date_week_number_c’ variables, using sin and cos transformation. One year is about 52.1429 weeks and the values of ‘week’ variable are composed of 1 to 53 weeks. Week 20 and week 21 are practically the same meaning. For the same reason week 53 and week 1 have similar meaning. So, by using sin, cos transformation we made variable ‘week’ have periodicity.



-Resolve Skewness:

As we can see from the histogram below, the ‘adr’ and ‘lead_time’ variables are right-biased. This can provoke distortion in correlation of that variables and the other variables, and as a result the prediction accuracy will be reduced. To prevent this, we normalized these variables using box cox transformation. The result is as followed.



(2) Variable introduction

After a series of variable processing, the variables of the data set and their meanings are as follows:

Name	Type	Implication

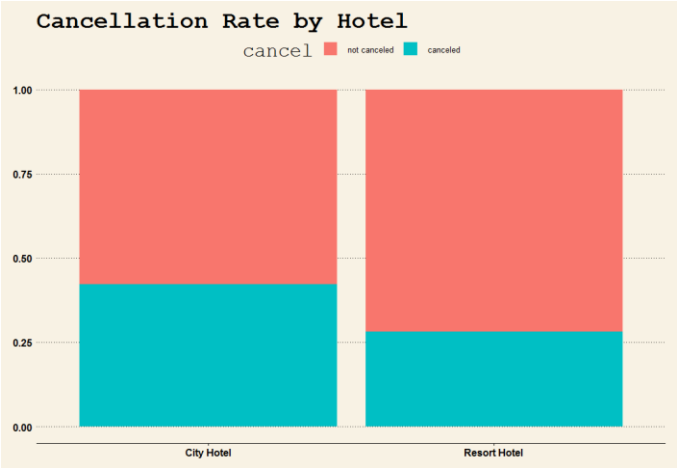
	(numerical or category)	
is_canceled *(predict)*	cat	Value indicating if the booking was canceled (1) or not (0)
hotel	cat	Type of two hotels: resort hotel and city hotel
arrival_date_year	cat	Year of arrival date
stays_in_weekend_nights	num	Number of weekend nights (Saturday or Sunday) the guest stayed or booked to stay at the hotel
stays_in_week_nights	num	Number of weeknights (Monday to Friday) the guest stayed or booked to stay at the hotel
meal	cat	Type of meal booked. Categories are presented in standard hospitality meal packages: BO, BL and ML Undefined/SC – no meal package; BB – Bed & Breakfast; HB – Half board (breakfast and one other meal – usually dinner); FB – Full board (breakfast, lunch and dinner)
market_segment	cat	Market segment designation. In categories, the term "TA" means "Travel Agents" and "TO" means "Tour Operators"
distribution_channel	cat	Booking distribution channel. The term "TA" means "Travel Agents" and "TO" means "Tour Operators"
is_repeated_guest	cat	Value indicating if the booking name was from a repeated guest (1) or not (0)
previous_cancellations	num	Number of previous bookings that were cancelled by the customer prior to the current booking
previous_bookings_not_canceled	num	Number of previous bookings not cancelled by the customer prior to the current booking
booking_changes	num	Number of changes/amendments made to the booking from the moment the booking was entered on the PMS until the moment of check-in or cancellation
deposit_type	cat	Indication on if the customer made a deposit to guarantee the booking.

agent	cat	ID of the travel agency that made the booking (After OHE)
customer_type	cat	Type of booking, Type of booking, assuming one of four categories: Contract, Group, Transient and Transient-party
required_car_parking_spaces	num	Number of car parking spaces required by the customer
total_of_special_requests	num	Number of special requests made by the customer (e.g. twin bed or high floor)
is_transient	cat	Type of booking, generated by customer type
reserved_is_assigned	cat	Indication on if the reserved room type same as assigned room type
lead_time_bc	num	Number of days that elapsed between the entering date of the booking into the PMS and the arrival date (Resolve Skewness)
adr_bc	num	Average Daily Rate (Resolve Skewness)
arrival_date_week_number_s	cat	Week number of the arrival date (sin)
arrival_date_week_number_c	cat	Week number of the arrival date (cos)
weekday	cat	Number pf weekday
days_in_waiting_list_tr	cat	Number of days in waiting list
		Total: p = 24(explanatory variables) + 1(target)

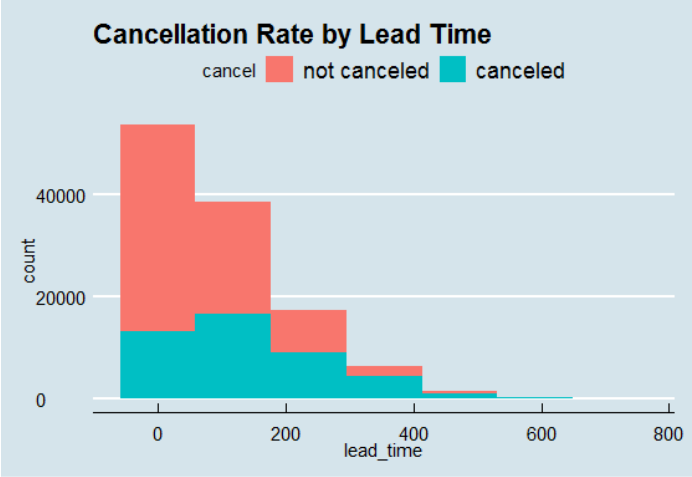
⑶ Visualization

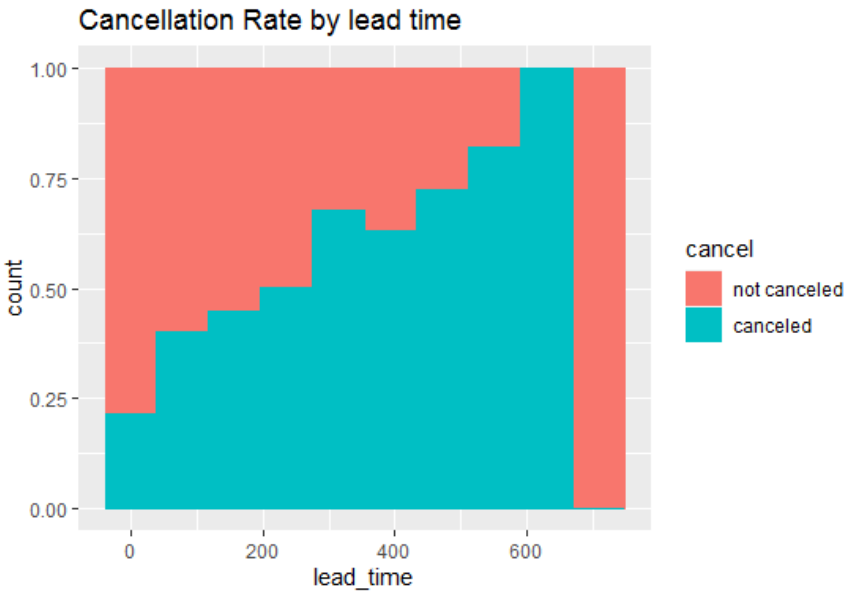
Next, we will use data visualization to further explore the relationship between some explanatory variables (x) and our target variable (y):

First of all, we sought the relationship between the cancellation rate of hotels and hotels. We can see that the cancellation rate of city hotel is higher than that of resort Hotel, which is in line with the actual situation. Because there are more people staying in the city hotel due to work, and with sudden changes, people are more likely to cancel their reservations than resort hotels where more people stay because of travel:

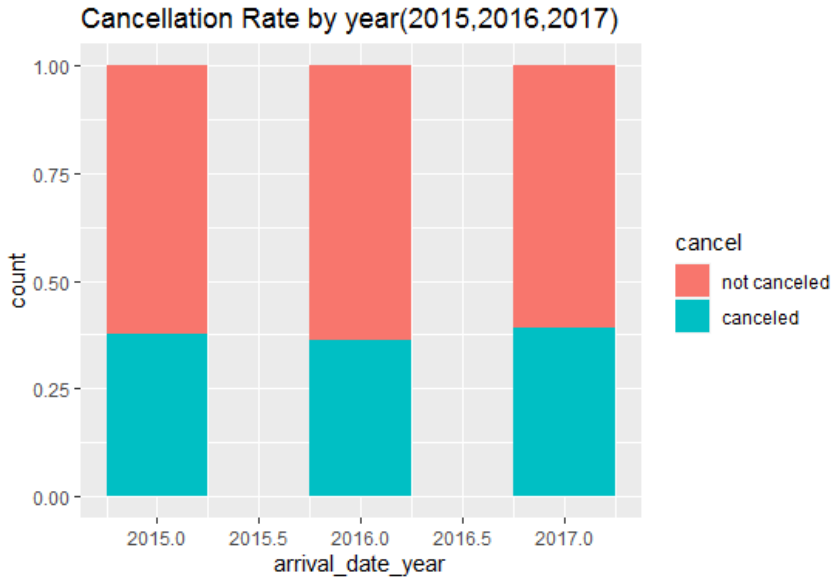


Then, we sought the relationship between the cancellation rate and the number of days reserved in advance. We can see that as the number of days reserved in advance increases, the rate of cancellations also increases, which is in line with the actual situation. Because the earlier you book in advance, the more likely it is that various unexpected situations will occur, leading people to cancel the reservation:

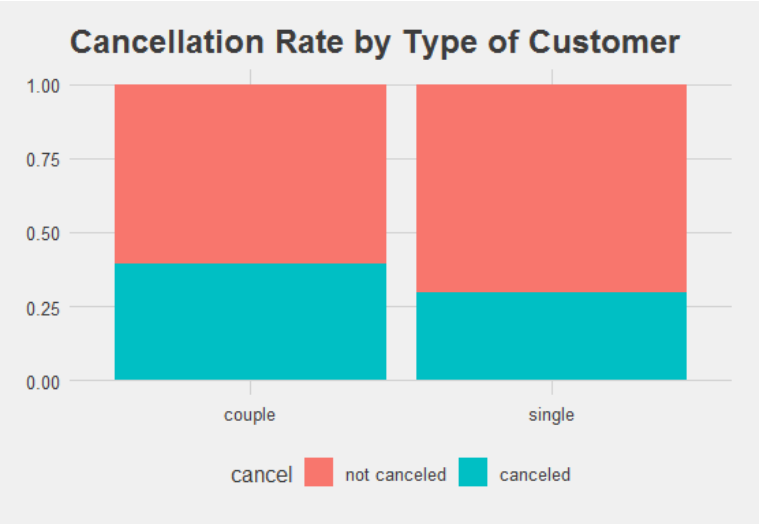




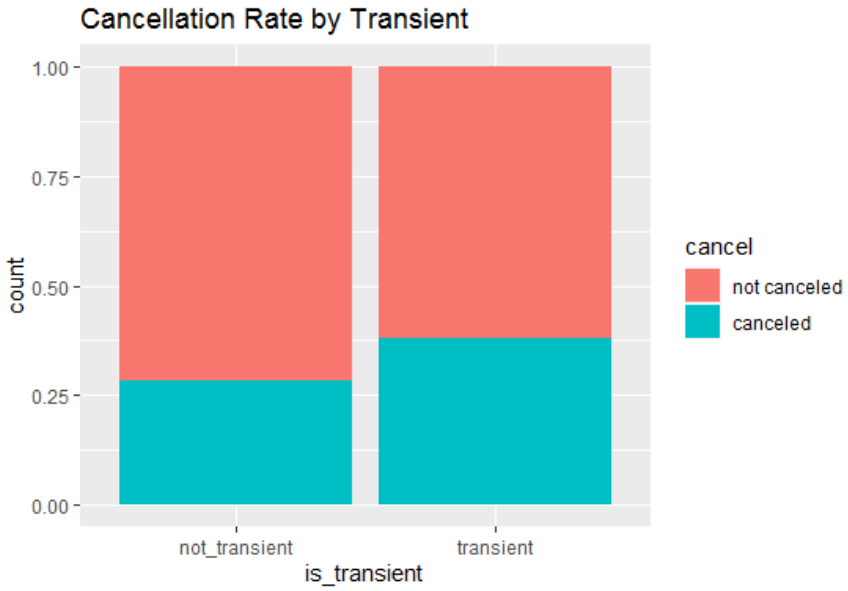
Next, we observed the relationship between the cancellation rate and the three years(2015, 2016, 2017), we can see that there is no obvious difference between them:



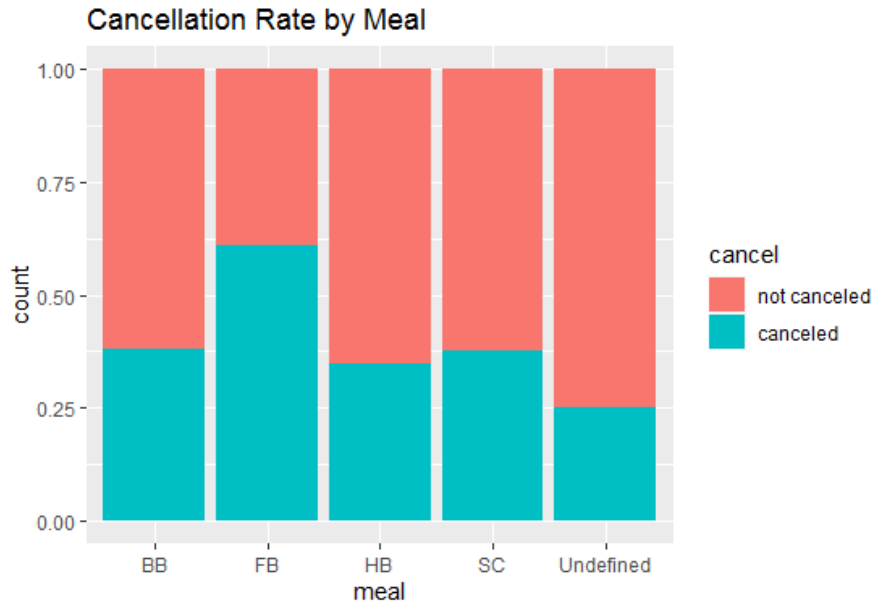
Then observed the relationship between the cancellation rate and the type of occupant, we can see that the cancellation rate of the couple is slightly higher than the cancellation rate of single:



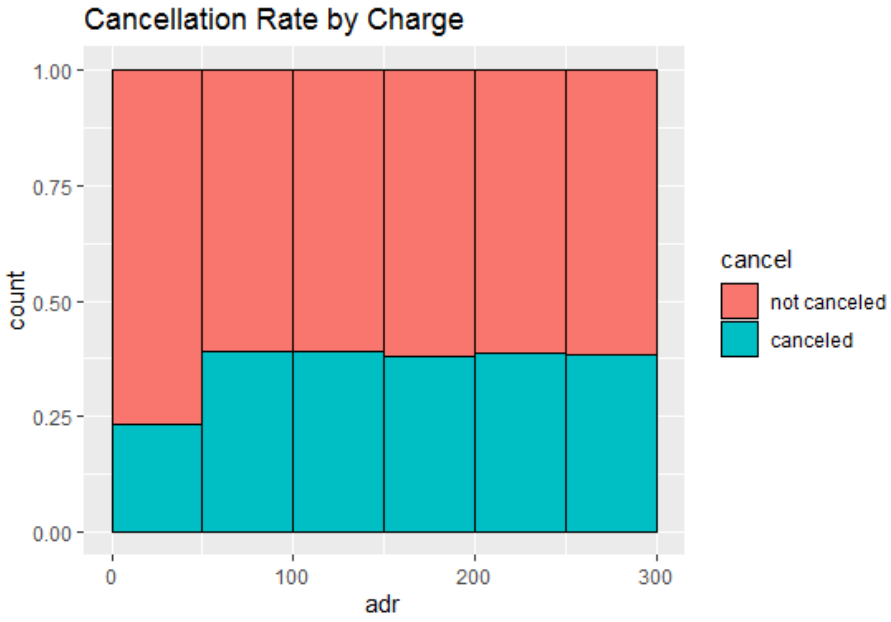
Next, we sought the relationship between the cancellation rate and the number of days of residence. We can trim the cancellation rate for short stays to be higher than the cancellation rate for longer stays, which is in line with the actual situation. Because people with long stays may consider more comprehensively, and they will not cancel the reservation easily:



Next, explore the relationship between the cancellation rate and the type of meal booked. We can see that the cancellation rate for the reservation of full board is higher than the cancellation rate of other types:



Then we observed the relationship between the cancellation rate and the price of the booked room. We can see that the cancellation rate between 0 and 50 is lower than the cancellation rate above 50, which is very realistic:



After the above data preprocessing and data visualization, we can better find effective features and lay a solid foundation for the next step of data modeling.

III. Method

In this project, our ultimate goal is to find a good enough model to predict whether the guest will eventually cancel the reservation, and use this model to find the most influential features to help the hotel better cope with the situation of the customer's cancellation.

In essence, it is a project to find a most suitable two-class classification algorithm. Next, we will first introduce several classification algorithms: logistic regression algorithm, decision tree algorithm, random forest algorithm and gradient boosting algorithm, and then divide our data into two parts: training set (75%) and test set (25%), then use these algorithms to model our data. Finally, compare these algorithms to find the model that best fits our purpose and adjust parameters of the model and optimize the model.

i. Algorithm Introduction

(1) Logistic Regression

Although Logistic Regression is called regression, it is actually a classification model and is often used for binary classification. The essence of logistic regression is: assuming that the data obey this distribution, and then use maximum likelihood estimation to estimate the parameters.

Logistic distribution is a continuous probability distribution, its distribution function and density function are:

$$F(x) = P(X \leq x) = \frac{1}{1 + e^{-(x-\mu)/\gamma}}$$

$$f(x) = F'(X \leq x) = \frac{e^{-(x-\mu)/\gamma}}{\gamma(1 + e^{-(x-\mu)/\gamma})^2}$$

As mentioned earlier, Logistic regression is mainly used for classification problems. We take binary classification as an example. For the given data set, assuming that there is such a straight line, the data can be linearly separable.

The decision boundary can be expressed as:

$$w_1 x_1 + w_2 x_2 + b = 0$$

Assuming a certain sample point

$$h_w(x) = w_1 x_1 + w_2 x_2 + b > 0$$

then it can be judged that its category is 1, this process is actually a perceptron.

Cost function:

After the mathematical form of the logistic regression model is determined, the rest is how to solve the parameters in the model. In statistics, the maximum likelihood estimation method is often used to solve, that is, to find a set of parameters, so that under this set of parameters, the likelihood (probability) of our data is the largest.

The loss function of logistic regression is:

$$J(w) = -\frac{1}{n}(\sum_{i=1}^{n}(y_i lnp(x_i) + (1-y_i)ln(1-p(x_i)))$$

We can use Stochastic gradient descent and Newton's method to solve it.

Regularization:

Regularization is a general algorithm and idea, so all algorithms that produce overfitting can use regularization to avoid overfitting.

On the basis of minimizing training errors, using a simple model as much as possible can effectively improve the accuracy of generalized prediction. If the model is too complicated and the value of the variable changes a little, it will cause prediction accuracy problems. Regularization is effective because it reduces the weight of features, making the model simpler. Regularization generally adopts L1 normal form or L2 normal form.

L1 regularization: LASSO regression

Objective function:

$$-\ln L(w) = -\sum_i [y_i \ln p(x_i) + (1-y_i)ln(1-p(x_i))] + \frac{1}{2b^2}\sum_j |w_j|$$

L2 regularization: Ridge regression

Objective function:

$$-\ln L(w) = -\sum_i [y_i \ln p(x_i) + (1-y_i)ln(1-p(x_i))] + \frac{1}{2\sigma^2} w^T w$$

Adding the regularization term allows us to choose a solution that is simpler (tends to 0) while minimizing empirical error.

(2) Decision Tree

Decision tree is a greedy algorithm, which represents a mapping relationship between object attributes and object values. The decision tree uses the entropy gain the rate or the Gini coefficient decrease range to group the object attributes. The ideal grouping should try to maximize the Gini coefficients of the two sets of output variable values.

The Gini coefficient expression is:

$$Gini = 1 - \sum p_i^2$$

The information entropy expression is:

$$info = \sum p_i \log_2(p_i)$$

Where p is the frequency of the category under the current classification sample.

(3) Random Forest

Random Forest, build a forest in a random way. The RF algorithm consists of many decision trees, and there is no correlation between each decision tree. After building the forest, when a new sample is entered, each decision tree will be judged separately, and then the classification results will be given based on the voting method.

Random Forest is an extended variant of Bagging. It builds on Bagging integration based on the decision tree as the learner, and further introduces random feature selection in the training process of the decision tree. Therefore, it can be summarized that RF includes four parts:

Randomly select samples (replace sampling);
Randomly select features;
Build a decision tree;
Random forest voting (average).

The random selection sample is the same as Bagging, using Bootstrap self-service sampling method; random selection feature means that each node randomly selects features during the splitting process (the difference is that each tree randomly selects a batch of features).

This randomness leads to a slight increase in the deviation of the random forest (compared to a single non-random tree), but due to the "average" nature of the random forest, its variance is reduced, and the reduction in variance compensates The deviation increases, so overall it is a better model.

Random sampling Since two sampling methods are introduced to ensure randomness, every tree grows as much as possible, even if it is not pruned, there will be no overfitting.

(4) Gradient Boosting

Gradient Boosting is an iterative algorithm. During the iteration, the direction of gradient descent is selected to ensure the best result.

The loss function is used to describe the "reliable" degree of the model. Assuming that the model is not over-fitted, the larger the loss function, the higher the error rate of the model

If our model can make the loss function continue to decline, it means that our model is constantly improving, and the best way is to let the loss function decrease in its gradient direction.

The following is a flowchart of Gradient Boosting:

Algorithm 1: Gradient_Boost

1. $F_0(\mathbf{x}) = \arg\min_\rho \sum_{i=1}^{N} L(y_i, \rho)$
2. For $m = 1$ to M do:
3. $\quad \tilde{y}_i = -\left[\frac{\partial L(y_i, F(\mathbf{x}_i))}{\partial F(\mathbf{x}_i)}\right]_{F(\mathbf{x})=F_{m-1}(\mathbf{x})}, \; i = 1, N$
4. $\quad \mathbf{a}_m = \arg\min_{\mathbf{a},\beta} \sum_{i=1}^{N} [\tilde{y}_i - \beta h(\mathbf{x}_i; \mathbf{a})]^2$
5. $\quad \rho_m = \arg\min_\rho \sum_{i=1}^{N} L(y_i, F_{m-1}(\mathbf{x}_i) + \rho h(\mathbf{x}_i; \mathbf{a}_m))$
6. $\quad F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \rho_m h(\mathbf{x}; \mathbf{a}_m)$
7. endFor

end Algorithm

IV. Analysis Results

First of all, we used logistic regression, PCA + logistic regression and LASSO to model our data, and obtained their respective Accuracy Scor, as shown in the following table:

Model	Accuracy Score
LR	0.818
PCA -> LR	0.736
Lasso	0.779

We can see that the logistic regression model has better accuracy in these three models.

And through LASSO preliminary obtained some important feature variables, as shown below:

Variable	Coefficients
previous_cancellations	0.013
booking_changes	-0.040
required_car_parking_spaces	-0.064
total_of_special_requests	-0.10
lead_time_bc	-0.019
adr_bc	0.010
deposit_type (non refund)	0.5

Then, we used logistic regression algorithm, decision tree algorithm, random forest algorithm and gradient boosting algorithm to model the training set, and cross-validated them with k-folds = 4 and repeat 100 times respectively(20 times in GB_model, because code run too much time), and obtained their test Accuracy, Precision, Recall, F1 score and AUC to find better model, then compute there means and standard as follows table(
First four rows):

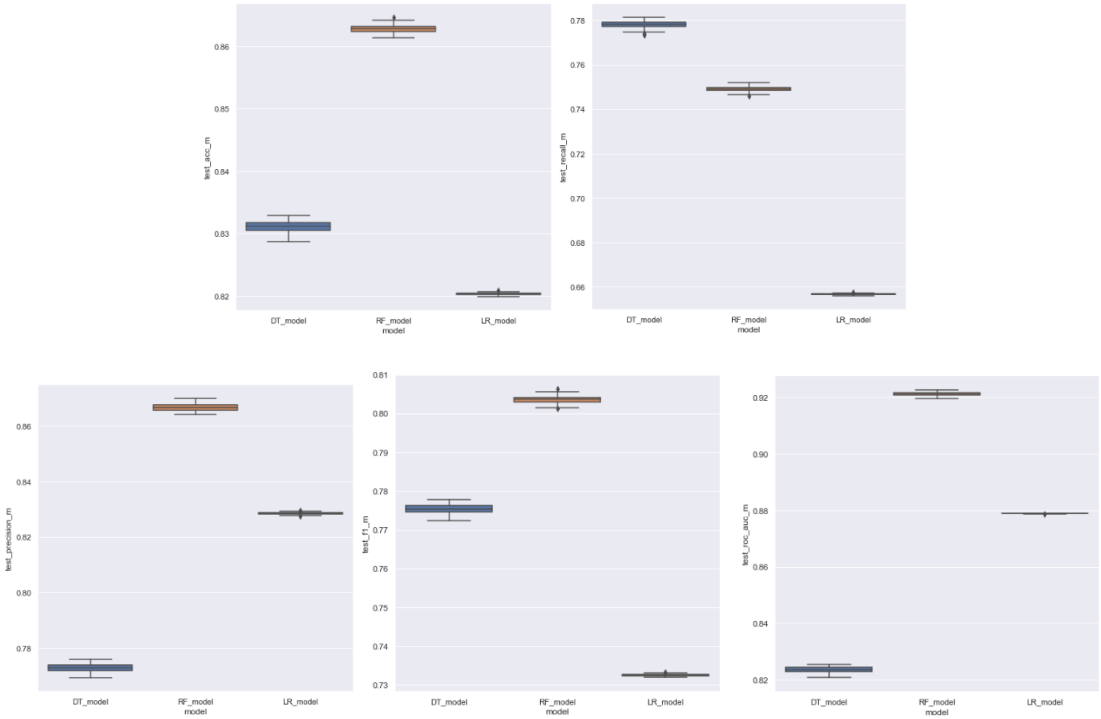
Model	**test_acc_m**	**test_acc_s**	**test_racall_m**	**test_recall_s**
DT_model	0.830414	0.001947	0.777300	0.003717
RF_model	0.861965	0.001595	0.748413	0.003866
LR_model	0.820304	0.001174	0.656593	0.002635
GB_model	0.822995	0.000923	0.631426	0.004482
DT_model	…	…	…	…

…(models)	…	…	…	…

Model	test_precision_m	test_precision_s	test_f1_m	test_f1_s
DT_model	0.771924	0.003251	0.774596	0.002721
RF_model	0.865199	0.002846	0.802570	0.002119
LR_model	0.828492	0.004119	0.732584	0.001929
GB_model	0.859098	0.004028	0.727850	0.001802
(models)	…	…	…	…

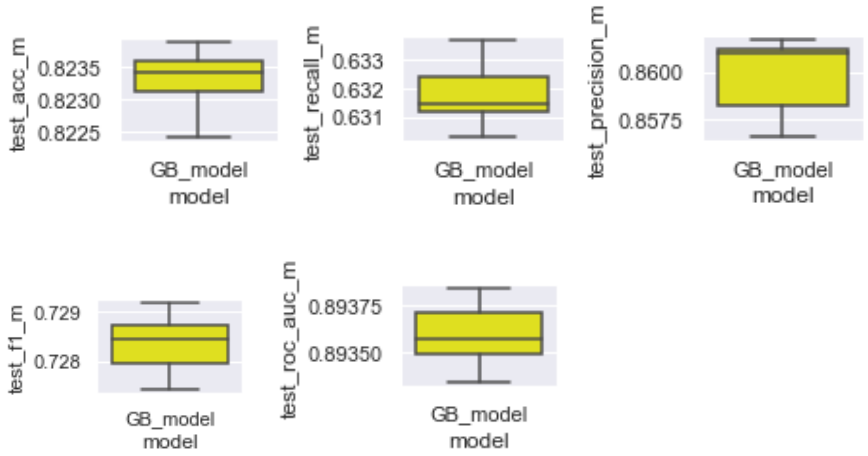
Model	test_roc_auc_m	test_roc_auc_s
DT_model	0.822763	0.002036
RF_model	0.921464	0.001818
LR_model	0.878874	0.000706
GB_model	0.893362	0.000761
(models)	…	…

After that, we spent box plots to compare the Accuracy, Precision, Recall, F1 score and AUC between the various models:



(Image order: Accuracy, Precision, Recall, F1 score and AUC)

(model order: DT, RF, LR)



(Because of GB_model the repeat time(20) is different to others(100), List its images separately.)

We can see that 'RF_model', which using random forest algorithm has largest Accuracy, Recall, F1 score and AUC.

Therefore, the random forest model works best, which is make sense, because random forest is easy to parallelize, and has great advantages in large data sets. Also random forest has ability to process high-dimensional data without feature selection.

Then, we adjusted some parameters of the random forest and obtained its accuracy rate as follow and it can be seen that the accuracy has been improved.:

```
RF model cross validation accuarcy score: 0.8733 +/- 0.0013 (std) min: 0.8724, max: 0.8756
```

And From the RF_model, we got some important characteristic variables, as shown in the following figure:

	feature
1	lead_time_bc
2	deposit_type
3	adr_bc
4	arrival_date_week_number
5	total_of_special_requests
6	previous_cancellations
7	stays_in_week_nights
8	market_segment

These variables have a high degree of coincidence with the important features obtained by Lasso before.

Summarize them and get our final feature variables: ***lead_time_bc, deposit_type, adr_bc, total_of_special_request, previouscancellation***

V. Conclusion

In this project, we used the theme of predicting the cancellation of hotel reservations and selected the data set from Kaggle: Hotel Booking Demand.

First, we performed EDA on the data, including data preprocessing and data visualization, obtained the relationship between some feature variables and target variables, and then divided our data set into two parts: the training set and the test set, and used several classification algorithms model the training set of data. According to the modeling and result analysis, we have obtained the Random Forest model is well fitted.

And we have explored several features that have an impact on guest cancellation, they are:

feature	meaning
lead_time_bc	Number of days that elapsed between the entering date of the booking into the PMS and the arrival date
deposit_type	Indication on if the customer made a deposit to guarantee the booking.
adr_bc	Average Daily Rate
total_of_special_request	Number of special requests made by the customer
previous_cancellation	Number of previous bookings that were cancelled by the customer prior to the current booking

We can say the results of this project are meaningful for the development of the hotel industry. it can better help hotels recognize why people cancel reservations and make improvements and can predict whether guests will cancel reservations to create higher surplus value.