Wei Luok Ngu / R09946023
Chia Ying Tsao / R09946013
Ching Ru Ho / R09946006

Final Project Report
Machine Learning Techniques
2020 Fall Semester

Instructor: Hsuan-Tien Lin
National Taiwan University
February 11, 2021

1 Dataset Overview

We have four csv files, which are `train`, `test`, `train_label` and `test_nolabel`. In `train` and `test`, each row refers to different bookings, and the columns refer to order's information and requests. The details of the dataset are as below:

(1) `train`: Contains 91531 rows and 33 columns.

(2) `test`: Contains 27859 rows and 29 columns.

(3) `train_label`: Label of daily revenue of 640 days (2015-07-01 until 2017-03-31).

(4) `test_nolabel`: Label of daily revenue of 153 days (2017-04-01 until 2017-08-31).

Noticed that `train` contains 4 columns, `is_canceled` , `adr`, `reservation_status` and `reservation_statue_date`, which are not in `test`. `is_canceled` and `adr` are the respond variables that we need to predict. `reservation_status` and `reservation_statue_date` are not contained in `test`, so we simply remove them. Furthermore, Label of daily revenue is taken from all **valid** bookings (i.e `is_canceled`=0) on that day, and it is calculated by

$$\lfloor \texttt{adr} \times (\texttt{stay\_in\_weedkend\_nights} + \texttt{stay\_in \_week\_nights})/10000 \rfloor \text{, where } \lfloor\cdot\rfloor \text{ denotes floor.}$$

2 Preprocessing

2.1 Missing data

Table 1 shows that there are 4 features containing missing data.

Features Name	NaN in `train` dataset	NaN in `test` dataset
`children`	4 (0.0044%)	0
`country`	468 (0.5113%)	20 (0.0718%)
`agent`	13217 (14.4399%)	3123 (11.2100%)
`company`	85917 (93.8666%)	26676 (95.7536%)

Table 1: Number of missing features and its proportion in each dataset.

The NaN in `children` are filled with 0, since 93.67% of the rows have 0 children. The NaN in `country` are filled with 'PRT', since it occurs most frequently. Also, we convert the `company` variable to binary, where 0 represents 'No company', which is the missing value. The value of agent changes to 1 if it's in the top 20 frequently occurs and 0 if it is missing, where 0 represents 'No agent'.

2.2 Outlier and Daily Revenue

ID 31980 in `train` dataset has `adr` = 5399.42 and `is_cancel` = 1, which is much higher than other orders (the highest `adr` is 1072 after ID 31980 is removed). Thus, we treat this as an outlier and remove it. Furthermore, we convert all categories features into dummy variables, and add some new features:

$$\texttt{previous\_cancel\_rate} = \frac{\texttt{previous\_cancellations}}{\texttt{previous\_cancellations + previous\_bookings\_not\_canceled}}$$

$$\texttt{nights} = \texttt{stays\_in\_weekend\_nights + stays\_in\_week\_nights}$$

$$\texttt{overbook} = 0 \text{ if } \texttt{(reserved\_room\_type = assigned\_room\_type)}, \text{ else } 1$$

3 EDA

To predict the daily revenue, the reasonable strategy is to predict whether a booking request is canceled or not, then predict `adr` of the fulfilled bookings to compute revenue. Thus, for the two stages, the data exploration will be the following 2 parts.

3.1 Target Variable: `is_canceled`

Figure 1 shows that about 35 percent of bookings were canceled, and the proportion is stable each year. The revenue is calculated by the fulfilled room reservation request.

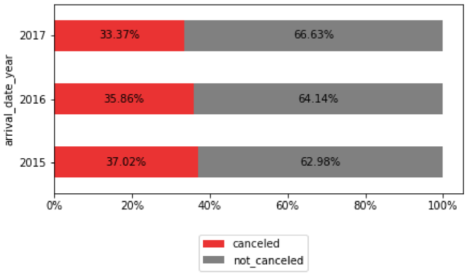

Figure 1: Booking cancellation ratio per year.

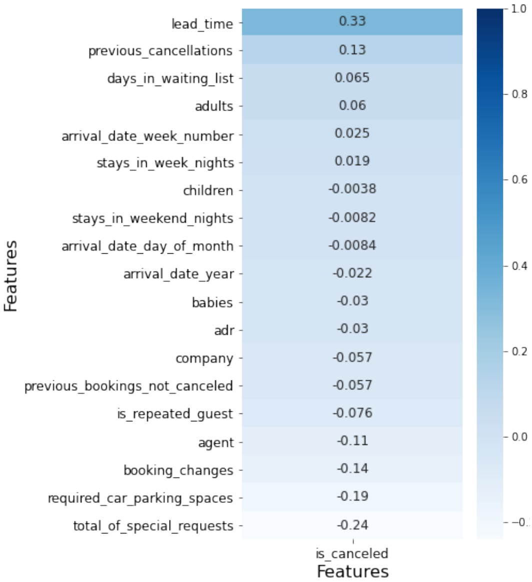

Figure 2: Correlation Coefficient of `is_canceled` and each feature

Next, we consider the relationship between each feature and `is_canceled`. As shown in Figure 2, `lead_time` highly correlates with cancellation. It makes sense since the earlier customer books the hotel, the more time to cancel it. Also, there is more possible that something out of your plan happens.

On the other hand, `total_of_special_requests` has negative correlation with cancellation. It seems that as the number of special requests increases, the customer takes it seriously and tends to keep the booking.

3.2 Target Variable: `adr`

Figure 3 shows the relationship between some of the features and the average revenue.

Most of the feature doesn't show strong correlation with revenue, such as `required_car_parking_space` in the plot. However, the feature like `adults` or `stay_nights` show positive correlation. Also, we could observe that revenue reach the peak during July and August.

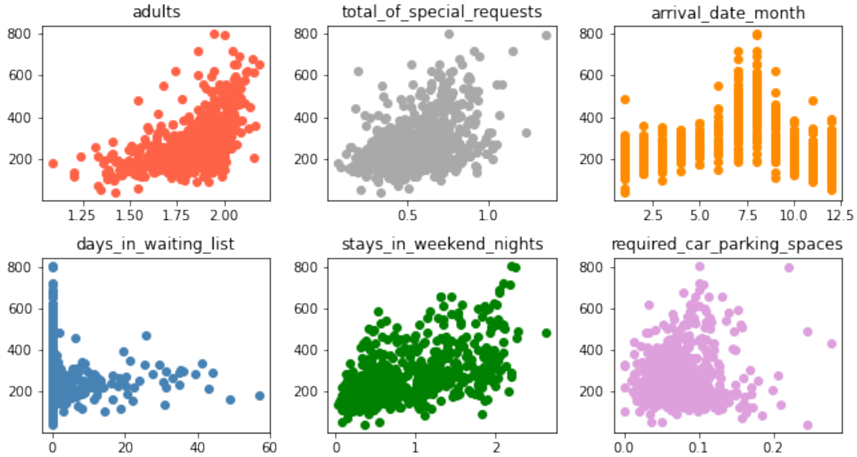

Figure 3: Relation between the average Revenue and each features

Furthermore, we plot the timeline of the average revenue as shown in Figure 4. Each year, the revenue is low at the beginning of the year and has a high revenue in summer vacation. The result suggests that we could try Time Series model such as ARIMA, which will be introduced later.

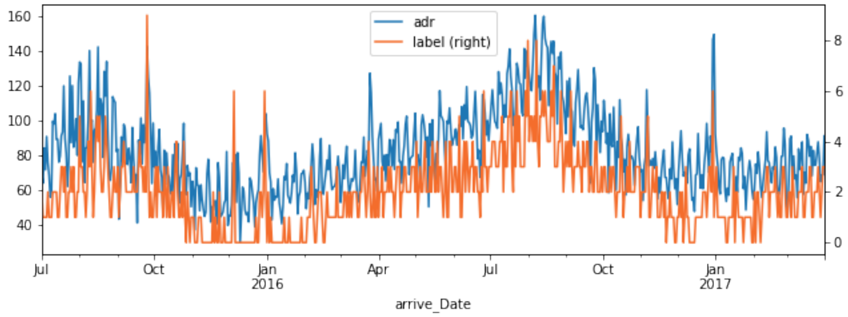

Figure 4: Trending of ADR and label by date

4 Experiment

We have two types of model, one for predicting `is_canceled` label, and one for predicting `adr`.

4.1 Prediction of `is_canceled`

We use some models that are taught in class, which are linear model, Logistic Regression (with regularization), non-linear models, Decision Tree Classifier, Random Forest Classifier, and Gradient Boosting Classifier. We pick some parameters and use 5-fold cross-validation to select the parameters that give the best accuracy, then use them to fit the models with full train data. For Random Forest Classifier, we use the accuracy of out-of-bagging instead of splitting the data and cross-validation.

By using these 4 models mentioned above to predict whether the order is canceled in our test data, the results and details are shown in Table 2. Noticed that the parameters not be mentioned in each model are set to default in Python sklearn package.

Obviously, Random Forest Classifier has the best train accuracy. However, we think this is overfitted, since the best accuracy of out-of-bagging is much lower, which means that this model is doing well on train data, but not so well on the unseen data.

Model	Parameter	Best 1-E_val	Best Parameter	Train Accuracy
Logistic Regression	regularizer = [0.001, 0.01, 0.05, 0.25, 0.5, 1, 10, 100, 1000]	0.8229	regularizer = 10	0.8244
Decision Tree Classifier	Criterion: gini, entropy max_depth = [1, 2, . . . , 15]	0.8238	1-E_val of all combinations are same, we take max_depth = 12 and criterion: entropy	0.8662
Random Forest Classifier	max_depth = [30, 40, 50]	0.88	max_depth = 50	0.9838
Gradient Boosting Classifier	max_depth = [1, 2, ..., 10]	0.8662	max_depth = 10	0.8997

Table 2: Details about each model uses for prediction of `is_canceled`

4.2 Prediction of `adr`

On the other hand, We use the linear model, Ridge Regression, the non-linear model, Neural Network, and try a new model, Regression-based Neural Network, which will be introduced later, to predict `adr` in our test data. By picking some parameters and using 5-fold cross-validation to select the parameters that give the lowest mean-squared error (MSE), we use them to fit the models with full train data. The results are shown in Table 3.

Model	Parameter	Lowest MSE	Best Parameter	Train MSE
Ridge Regression	regularizer = [0.001, 0.01, 0.05, 0.25, 0.5, 1, 10, 100,1000]	1104.1263	regularizer = 10	1097.6138
Neural Network Regressor	layer = [100, 200]	350.53	layer = 200	382.8109

Regression-based Neural Network	epochs = [30, 50]		epochs = 30,
			batch_size = 150,
			verbose = 1,
			validation_split = 0.2
				795.3298

Table 3: Details about each model uses for prediction of `adr`

4.3 Prediction Label of Daily Revenue

Next, we combine the 4 models that predict `is_canceled` and 2 models that predict `adr` to calculate the predicted daily revenue. The results are shown in Table 4.

The **Predicted Label MAE** is the mean absolute error of the predicted train result. We will use MAE to indicate the goodness of fitting of our models since the result of the scoreboard use MAE as evaluation method.

After getting 8 results of the test label, we apply Uniform Blending, a voting system on these 8 results, that is, the final label of each observation is the most frequent number among 8 models in that observation. The **Predicted Label Accuracy** is the training accuracy, which is the total numbers that are predicted correctly and then divided by the total number of labels.

Ignore the private score, we observe that, except Random Forest + Regression-based Neural Network (which will be introduced later), we have two models that have the lowest MAE in the public scoreboard, which are

Gradient Boosting + Neural Network, and Uniform Blending, both have same MAE, which is 0.5658.

Model	Predicted Label Accuracy	Predicted Label MAE	Public Score	Private Score
Logistic Regression + Ridge Regression	0.6609	0.3672	0.7895	0.9091
Decision Tree + Ridge Regression	0.6797	0.3438	0.5263	0.5974
Random Forest + Ridge Regression	0.7859	0.2234	0.75	0.6883
GB + Ridge Regression	0.7641	0.2469	0.5658	0.5844
Logistic Regression + Neural Network	0.6843	0.3329	0.6579	0.6233
Decision Tree + Neural Network	0.7281	0.2781	0.7237	0.6233
Random Forest + Neural Network	0.9078	0.0921	1.0132	0.9740
GB + NN	0.8140	0.1875	0.6579	0.5844
Uniform Blending			0.5658	0.5974
Random Forest + Regression-based Neural Network	0.75	0.27	0.3684	0.3246

Table 4: Details about each model uses for prediction of Label

4.4 Explanations about the Models

Ridge Regression is adding regularizer to linear regression, making it compute quickly and can be updated easily with new data, and it is relatively easy to understand and explain. Regularization can be used to prevent overfitting. Logistic Regression is similar, except that its range is restricted between 0 and 1, so it is used for classification.

However, linear models are unable to learn complex relationships and difficult to capture non-linear relationships, so we use some non-linear models, such as Decision Tree, it learns how to best split the dataset into separate branches, allowing it to learn non-linear relationships. However, Decision Tree are prone to overfitting, so we use Random Forest and Gradient Boosting, which are two algorithms that build many individual trees, pooling their prediction, it robust to noise and missing value, and it performs well then linear models 'out-of-the-box', but complex trees are hard to interpret.

Next, we introduce some pros and cons about our best performing model: Regression-based Neural Network.

Pros	**Cons**
- Can learn even very complex relationships	- Require a very large amount of data and long training time
- Hidden layers reduce need for feature engineering	- Prone to overfitting
(less need to understand underlying data)	- Model is a "black box", unexplainable

5 Other Approaches

5.1 Straight predict the labels via `is_canceled`

Besides the common way (predict `is_cancel` and predict `adr`), we also tried directly predicting the label after predicting cancellation. Both the models are classifier, and we tried using Decision Tree or Random Forest on the multi-classification problem (predict label). Total `adr` of each day is highly correlated with daily revenue, and we found that the regression model can not do well at the beginning, so we directly predict the label and simplify it to a classification problem. The result of this method gets MAE 0.67 of the public scoreboard.

5.2 ARIMA (Time Series Model based on adr)

ARIMA (Autoregressive Integrated Moving Average Model) is one of the popular Times Series models used in statistics and econometrics. If the data is stationary, or after the differencing process is stationary, we can be based on the idea that the information in the past values of the time series can alone be used to predict the future values.

Figure 4 seems to show that adr has some relationship with the time data. In our assumption, because hotels will decide price before consumers' booking, so we can assume that the price of hotel only corresponds to the time, such as low season and high reason.

However, according to our experiment on some Times Series models, we can have a good prediction performance on average adr by weekly or monthly data due to a long-term trending. However, if we want to use daily data to form a time series model, the performance is not good enough. The result of this method gets MAE 1.51 of the public scoreboard.

5.3 Regression-based Neural Network

The Regression-based Neural Network is implemented on the filtered dataset to predict adr. The input layer has 59 neurons since there are 59 features after preprocessing, and the output layer is only one layer, the activation function is linear. As for the hidden layer, there are 2670 neurons with ELU activation function. The training runs for 30 epochs and batch_size = 150.

6 Conclusion

The report proposes a methodology to predict the label of daily revenue. The methodology has three major steps, which are data preprocessing, EDA, and modeling. As the results above, the Random Forest + Regression-based Neural Network performs the best among all algorithms, which get 0.3684 on the public scoreboard and 0.3247 on the private scoreboard. In addition, for the models that are introduced in this course, decision tree + ridge regression performs best.

Last, future work should implement more methods to deal with this data. For instance, we can try not to change the data structure of the columns (for example, we change the company to binary class), and compare it before and after. Also, we can apply the grid search to choose the parameter instead of set and try the parameter manually.

7 Work Loads

Each member has the same contribution.

Wei Luok Ngu	Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Ridge Regression
Chia Ying Tsao	Data Preprocessing, Graphs, Regression-based Neural Network and Neural Network
Ching Ru Ho	Time Series Model, Gradient Boosting Classifier, Report Design

8 References and Properness of Citations

(1) The video and slides in this course cover 80% knowledge in this project.

(2) About Regression-based neural network : https://towardsdatascience.com/regression-based-neural-networks-with-tensorflow-v2-0-predicting-average-daily-rates-e20fffa7ac9a